



This is to certify that the

thesis entitled

THE DESIGN AND OPERATION
OF AN AUTOMATED, HIGH-EFFICIENCY
PHOTOIONIZATION MASS SPECTROMETER

presented by

Edward James Darland

has been accepted towards fulfillment
of the requirements for

Ph.D degree in Chemistry

Major professor

Date November 14, 1977

~~ANN 781 3-11~~



THE STUDY OF A
OF AN APPROXIMATE
PHOTOGRAPHICALLY

BY

A DISSERTATION

Submitted by
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Chemistry

1978

THE DESIGN AND OPERATION
OF AN AUTOMATED, HIGH-EFFICIENCY
PHOTOIONIZATION MASS SPECTROMETER

By

Edward James Darland

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Chemistry

1978

5103436
203

ABSTRACT

THE DESIGN AND OPERATION OF AN AUTOMATED, HIGH-EFFICIENCY PHOTOIONIZATION MASS SPECTROMETER

By

Edward James Darland

Photoionization mass spectrometry (PIMS) is a powerful technique for studying the energetics of atoms and small molecules and the kinetics of polyatomic ion decomposition and ion-molecule reactions. However, PIMS suffers from a number of severe experimental difficulties which often result in experiments that either provide data with poor precision (low signal-to-noise ratio) or are very long and tedious, even neglecting the time required to set up the experiment and reduce the data to a useable form. Maximizing the efficiency of operation of PIMS instruments is therefore very important if the full potential of PIMS is to be realized.

In this dissertation, operating procedures designed to achieve a very high overall operating efficiency (while maintaining high standards of accuracy and precision) are discussed, and a new, automated photoionization mass spectrometer designed to allow implementation of these procedures is described. A comparison of several types of data acquisition systems shows that a computer-based system is the simplest automated system which provides the flexibility necessary to implement fully the required operating procedures. The new instrument, incorporating a one-meter, near-normal incidence monochromator and a quadrupole mass filter, is controlled during the experiments by a minicomputer which is also used for data reduction and analysis.

The computer programs (software) required to implement these

operating procedures are described. The software is moderately complex, but a set of guidelines has been developed for facilitating computer-user interaction, and all software has been written following these guidelines. As a result, the complexity of the system is largely transparent to the user; the instrument is very easy to use, and this ease of use is a major factor in reducing the experiment set-up and data reduction time so that more time can be spent in actual data acquisition and interpretation.

The data acquisition software includes provisions for recording, at regular intervals during the experiment, any of several types of "correction measurements" that are necessary for eliminating the effects of scattered light, long term drifts, etc., from the final results. The data being acquired are partially corrected and plotted as they are acquired so that the quality of the data may be judged without waiting until after the end of the experiment. The software also includes a variable integration time option, which allows the integration time to be varied automatically on a point-by-point basis so that it is just long enough to acquire data with the desired signal-to-noise ratio. The procedure used, equivalent to "integrating to a constant background-corrected ion intensity", is shown to make more efficient use of data acquisition time than the constant integration time or "integration to constant photon intensity" methods which are more commonly used.

So that the instrument may be left safely unattended, the status of the vacuum system as well as the "general utilities" (high voltage power supplies, water flow, etc.) is continuously monitored by an extensive safety interlock system, which automatically takes appropriate action

Edward James Darland

in the event of any failure. The interlock system is described in detail, as are the computer-interface electronics. The latter include a unique pulse amplifier circuit designed for fast photon- or ion-counting applications. The pulse counter has a sensitivity of 250 μ V and a maximum count rate exceeding 90 MHz. It is entirely DC-coupled so that it exhibits no baseline shift at any count rate.

The new instrument has been used to study the mixed chlorofluoromethanes CFCl_3 , CF_2Cl_2 , and CF_3Cl . Photoionization efficiency curves have been obtained for the first time for CFCl_3^+ , which is produced only rarely, as well as for twelve other ions from these compounds. All three parent ions are therefore stable, but there is some evidence that the observed ionization potentials may be significantly above the adiabatic ionization potentials for all three molecules. Further work is recommended to resolve several apparant paradoxes in the present data.

It would be very difficult to thank or acknowledge all of the people who have had a role in the making of this instrument for me. However, among them I would like to thank my parents, for their growth and to the preservation of the instrument and the help of the Kaiser, for George, Mary and Bill, for their help and for the very special "thank-you".

I would also like to thank the following people, all of whom have helped in the development of the instrument:

To Susan

To Mom

To Dad

For whom no acknowledgment can suffice.

Bill Chicks, Warren, and the Kaiser, and the Kaiser's agent of Argonne National Laboratories, for their help and advice, making many of the modifications and answering innumerable questions on the practical aspects of the instrument. Their help and advice were indispensable in getting the instrument "off the ground".

George Lofel and Chris Kane--for their support, guidance, encouragement, advice, and patience during my years as a graduate student; for the things they taught me, and more of all for the things they allowed me to learn.

Just Oyster, Lee Elmslie, Dick Hents and Dick Warkentin, the staff of the Chemistry Department Machine Shop--for carefully maintaining the various

ACKNOWLEDGMENTS

It would be virtually impossible for me to acknowledge all of the people who have had a part in making the past six years so enjoyable for me. However, among those who have contributed most to my personal growth and to the preservation of my sanity (if any), I owe Mike and Joan Kelsey, Jon Powers, Marv and Sylvia Reimer, Frank Tully, and Pat Dehmer a very special "thank you".

I would also like to offer my heartfelt thanks to the following people, all of whom have made more direct contributions to the development of the instrument described in this dissertation:

Bill Chupka, Warren Jivery, Pat and Joe Dehmer, and Joe Berkowitz--for the information they shared unstintingly during several months I spent at Argonne National Laboratory; for showing me how their instruments worked, making numerous suggestions for improvements, and answering innumerable questions on the practical aspects of PIMS methodology. Their help and advice were indispensable in getting the MSU instrument "off the ground".

George Leroi and Chris Enke--for their support, guidance, encouragement, advice, and patience during my years as a graduate student; for the things they taught me, and most of all for the things they allowed me to learn.

Russ Geyer, Len Eisele, Dick Menke and Dick Watters, the staff of the MSU Chemistry Department Machine Shop--for carefully machining the vacuum

chambers, lamp, and all other non-trivial parts of the instrument, for taking the time to teach me many of the principles of mechanical design and drawing, and for building what I wanted even when I didn't draw it correctly.

Ron Haas, Dick Johnson, and Rick Sandford, the staff of the MSU Chemistry Department Electronic Shop--for carefully etching many printed circuit boards, for assembling much of the interlock system, and for keeping much of the commercial electronics gear in our lab in running order.

Marty Rabb, MSU Chemistry Department Electronics Designer--for designing and building the new lamp power supply, and for helpful discussions on a wide range of other electronic problems.

Stan Crouch--for his careful reading of many parts of this dissertation, and many helpful discussions on "photon counting and the futility of it all".

Frank Tully--for his inestimable aid in getting the vacuum system up and running, and for many hours spent discovering new and ever more interesting hardware and software bugs, characterizing them so thoroughly that they were relatively easy to fix. Also, many thanks to Frank and ...

Paula Kronebusch--for acquiring much of the earliest chlorofluoromethane data.

Gary Ray--for help in assembling some of the plumbing, for designing the current ion source, sodium salycilate detector assembly, and sample inlet system, and for help in a multitude of other thankless tasks.

Dave Rider--for keeping the instrument running during recent months while Gary and I were preoccupied with getting in that "last run" or in finishing the "final figure, except for . . . ", for his willingness to

dig in and find out what was wrong when something quit, and for his many fresh ideas.

Tom Atkinson and Rick Yost--for writing many useful programs for the PDP-11 system. Many of the schematic diagrams in this theses were prepared using Rick's LAYOUT, and final copies of all data plots were produced with Tom's MULPLT. Other subroutines written by Tom and Rick are used extensively in transferring data from the PDP8 to PDP11 computers as well as in PIMS11.

Finally, I would like to acknowledge the financial support provided by Michigan State University, the National Science Foundation, the Office of Naval Research, the Associated Midwest Universities of Argonne "Thesis Parts" Program, my parents, and my wife.

5.	Design Requirements	15
1.	Precision of Data	15
2.	Accuracy of Data	15
3.	Efficiency of Operation	15
6.	Merits of Previous Designs	16
1.	Basis of Comparison	16
2.	Manual Operation	16
3.	Data Loading	16
7.	The MDT Instrument	16
CHAPTER 11. THE MAIN SPECIFICATIONS		
4.	Introduction	17
5.	Signal Sources	17
1.	Overview	17
2.	Logic	17
3.	Power Supplies	17

C. Photon Optics	59
D. Photon Transport	63
E. Ion Sources	70
F. Ion Optics	73
G. Ion Transport	76
H. Vacuum System	page
List of Tables	xi
List of Figures	xiii
CHAPTER I. INTRODUCTION	1
A. The Thesis	1
B. Photoionization Mass Spectrometry	3
C. Advantages of PIMS	9
D. Fundamental Experimental Difficulties	12
E. Design Requirements	14
1. Precision of Data	15
2. Accuracy of Data	16
3. Efficiency of Operation	19
F. Merits of Previous Designs	22
1. Basis of Comparison	22
2. Manual Operation	23
3. Data Logging	26
G. The MSU Instrument	30
CHAPTER II. THE MASS SPECTROMETER	37
A. Introduction	37
B. Photon Sources	39
1. Overview	39
2. Lamps	44
3. Power Supplies	52

C.	Photon Optics	59
D.	Photon Transducers	63
E.	Ion Source	70
F.	Ion Optics	72
G.	Ion Transducers	76
H.	Vacuum System	80
CHAPTER III. THE INTERLOCK SYSTEM		90
A.	Introduction	90
B.	Capabilities and General Discussion	93
1.	Master Control Module	94
2.	Vacuum System Control Modules	97
3.	Sequencing	99
C.	Master Control Module	103
D.	Vacuum System Control Module	109
1.	Operation of a 115 VAC, Fully Automated System	112
2.	Wiring for Other Options	115
CHAPTER IV. CONTROL AND DATA ACQUISITION		119
A.	Introduction	119
B.	The Wavelength Drive	124
C.	Shutter	130
D.	Pressure Measurement	134
E.	DC Current Measurement	140
F.	The Pulse Counter	151
1.	Introduction	151
2.	The Pulse Counter	156
3.	Pulse Counter Performance	172
4.	Recommended Modifications	179

5.	Pulse Counting System Performance	181
6.	Summary	199
G.	Dual Counter	200
CHAPTER V. OPERATION OF THE INSTRUMENT		210
A.	Introduction	210
CHAPTER V	1. Stages of the Experiment	210
A	2. Progression of Stages	212
B	3. Organization of the Chapter	213
B.	Overview of the PIMS Software System	213
	1. Basic Considerations	213
	2. Specific Software Requirements	216
C	3. General Description	218
	4. System Conventions	221
C.	Typical Operating Procedures	225
D.	Interactive Setup Options	231
D	1. CLOSE	231
	2. CURRENT WAVELENGTH	232
	3. INITIALIZE	232
	4. MOVE	234
E	5. OPEN	235
	6. PRESCALE	235
	7. SCAN	235
	8. TEST	236
	9. TRANSFER	237
E.	Data Acquisition Options	243
APPENDIX	1. The RECORD Option	243
A	2. Variable Integration Time	248

3.	Periodic Correction Measurements	251
4.	Stray Light Correction Measurements	254
5.	Real-Time Output	255
6.	The TIMED ACQUISITION Option	256
F.	Data Reduction Options	260
CHAPTER VI. SOME PRELIMINARY RESULTS		272
A.	Introduction	272
B.	Results and General Discussion	275
1.	Relative Intensities	275
2.	Photoionization Efficiencies	277
3.	Formation of Positive Chlorine Ions	295
C.	CF_3Cl	298
1.	CF_3Cl^+	299
2.	CF_3^+	301
3.	CF_2Cl^+	304
D.	CF_2Cl_2	305
1.	CF_2Cl^+	305
2.	CFCl_2^+	305
3.	CF_2^+	308
E.	CFCl_3	309
1.	CFCl_3^+	310
2.	CFCl_2^+	312
3.	CCl_3^+	313
4.	Other Fragments	314
F.	Summary	314
APPENDIX A. ARTIFACTS		317
A.	Finite Slit Width	317

B.	Scattered Light	319
C.	Detector Misalignment	324
APPENDIX B.	BASIC PDP 8/I INTERFACING	329
APPENDIX C.	THE REAL-TIME CLOCK	336
A.	Introduction	336
B.	Basic Operation	338
1.	Decade Clock	340
2.	Programmable Clock	340
C.	Programming	341
1.	Instructions	342
2.	Programming Examples	346
D.	Circuit Description	348
1.	Control Register	348
2.	Command Decoder	350
3.	Preset Latch, Presetable Counter, Output Latch, and Drive	353
4.	Oscillator and Time Base Select	355
5.	Flag Circuits	355
APPENDIX D.	SIGNAL-TO-NOISE RATIO EXPRESSIONS	359
APPENDIX E.	SOFTWARE	365
A.	Introduction	365
B.	PIMS8 Programs and Subroutines	367
C.	PIMS11 Programs and Subroutines	451
D.	Variable Definitions	490
1.	PIMS8 Variables	490
2.	PIMS11 Variables	492
REFERENCES		497

LIST OF TABLES

table		page
I-1-	Requirements Imposed by Design Goals.	22
I-2-	Comparison of Four Design Approaches.	35
II-1-	Approximate Wavelength Limits of the Rare Gas Continua.	42
II-2-	Comparison of the Cober and ANL Power Supplies.	56
II-3-	Comparison of Common Vacuum Ultraviolet Photon Transducers.	64
III-1-	Summarized MCM Monitor and Control Functions.	96
III-2-	Summarized VCM Monitor and Control Functions.	100
III-3-	Terminal Connections for Main Power Options.	115
III-4-	Terminal Connections for Automatic Valve Options.	116
III-5-	Terminal Connections for Sharing a Mechanical Pump Between Two Systems.	117
IV-1-	Wavelength Reproducibility.	128
IV-2-	Summary of Other Wavelength Reproducibility Tests.	128
IV-3-	Wavelength Accuracy.	129
IV-4-	Relative Stability of the 8850 PC System for Different Voltages and Discriminator Settings.	185
IV-5-	Limits of Linearity for Different Discriminator Coefficients.	196
V-1-	Main PIMS8 Options.	220
V-2-	PIMS8 Input Prompt Conventions.	223
V-3-	PIMS8 Special Character Conventions.	224
V-4-	PIMS8 Data Storage Conventions.	226
V-5-	TRANSFER Options.	238

V-6-	PIMS11 Options.	261
V-7-	PIMS11 Data Types, Their Possible Corrections and BCE Calculation Methods.	268
V-8-	PIMS11 Calculations.	270
VI-1-	Relative Abundance of Parent and Fragment Ions from CF_3Cl , CF_2Cl_2 , and CFCl_3	276
VI-2-	Data Used in Thermodynamic Calculations.	296
VI-3-	Summary of CF_3Cl Appearance Potentials.	300
VI-4-	Summary of CF_2Cl_2 Appearance Potentials.	306
VI-5-	Summary of CFCl_3 Appearance Potentials.	311
VI-6-	Summary of Heats of Formation Found in This Work.	316
C-1-	Clock I/O Instructions.	342
C-2-	Control Register Bit Assignment.	345
D-1-	Error Involved in Using Simplified S/N Expressions	364
II-1-	The Ballistic Method	31
II-2-	The Hydrogen Lamp	37
II-3-	A D-Type Spark Gap Discharge Lamp	43
II-4-	The Hot Nitrogen Lamp	45
II-5-	Lamp Utilities	50
II-6-	DC Power Supply for the Hydrogen Lamp	52
II-7-	Spark Gap Power Supply for the Spark Gap Discharge Lamp	54
II-8-	Vacuum Tube Power Supply for the Hot Wire Discharge Lamp	56
II-9-	Atomic Nitrogen Discharge Lamp at 100 A	61
II-10-	The Helium Continuum and Scattered Light from the Different Discharge Transducers	63
II-11-	A Simple Ion Source	65
II-12-	Top View of Quadrupole Mass	66
II-13-	Side View of Quadrupole Mass	67

LIST OF FIGURES

figure		page
I-1.	PIMS Made Easy.	6
I-2.	A Manual Instrument.	24
I-3.	A Simplified Data-Logging Instrument.	26
I-4.	The MSU PIMS Instrument.	31
II-1.	The Basic Elements of a Mass Spectrometer.	37
II-2.	The Basic Elements of a Photoionization Mass Spectrometer.	38
II-3.	Schematic Diagram of the Potential Energy Curves of He ₂	41
II-4.	The Helium Continuum.	43
II-5.	The Hydrogen Many-Lined Spectrum.	45
II-6.	A II-Type Rare Gas Discharge Lamp.	46
II-7.	The MSU Hinterregger Lamp.	48
II-8.	Lamp Utilities.	50
II-9.	DC Power Supply for the Hydrogen Lamp.	52
II-10.	Spark Gap Power Supply for the Rare Gas Discharge Lamps.	53
II-11.	Vacuum Tube Power Supply for the Rare Gas Discharge Lamps.	58
II-12.	Atomic Nitrogen Emission Lines at 1134 Å.	62
II-13.	The Helium Continuum and Scattered Light Recorded with Different Photon Transducers.	68
II-14.	A Simple Ion Source.	70
II-15.	Top View of Quadrupole Mount.	74
II-16.	Side View of Quadrupole Mount.	75

II-17.	Effect of Increasing Mass Resolution on Ion Transmission.	77
II-18.	Daly-Type Ion Transducer.	79
II-19.	Location of Components Within the Vacuum Chambers. . .	81
II-20.	Top View of Instrument, Retracted.	83
II-21.	Top View of Instrument, Normal Position.	84
II-22.	First Differential Pumping System.	86
II-23.	Second Differential Pumping System.	87
II-24.	Monochromator Pumping System.	88
II-25.	Sample and Quadrupole Pumping Systems.	89
III-1.	Interlock System Organization.	93
III-2.	Hood Alarm Circuit (MCM).	104
III-3.	Diffusion Pump Water Flow Interlock (MCM).	106
III-4.	General Utilities Interlock (MCM).	107
III-5.	"Local" Mechanical Pump Circuit.	109
III-6.	Pump Control Circuit (VCM).	110
III-7.	Valve Control Circuit (VCM).	111
III-8.	Alternate Interconnection Method.	118
IV-1.	Stepping Motor Interface.	124
IV-2.	Eight Successive Scans of the Atomic Nitrogen Lines at 1134 Å.	127
IV-3.	Shutter Control Interface.	132
IV-4.	Auxiliary Shutter Control Circuits.	133
IV-5.	Sample Pressure Measurement System.	135
IV-6.	Valve Control Signal Generation.	136
IV-7.	Valve Timing and Control.	138
IV-8.	Valve Enable Signal Selection.	139
IV-9.	V-F Current Integration Method.	141

IV-10.	V-F Converter Module Circuit.	143
IV-11.	Linearity of the V-F Converter Module.	145
IV-12.	Effect of Temperature on the Analog Measurement System.	147
IV-13.	V-F Nonlinearities as a Percentage of Full Scale and of Reading.	150
IV-14.	Block Diagram of the Pulse Counter.	154
IV-15.	Anode Pulses from a 1P28 Photomultiplier.	159
IV-16.	The Effect of Prescaling on a Random Pulse Train.	161
IV-17.	Preamplifier/Discriminator Circuit.	162
IV-18.	Response of Amplifier to Input Pulses.	163
IV-19.	AC Coupling of Amplifier and Comparator.	165
IV-20.	Prescaler/Line Driver Circuit.	168
IV-21.	Auxiliary Voltage Generation.	170
IV-22.	The Pulse Counter.	173
IV-23.	Variation of Sensitivity with Frequency.	174
IV-24.	Sensitivity vs. Threshold Setting.	174
IV-25.	Photon Count Rate vs. Temperature.	178
IV-26.	Integral Pulse Height Distribution for the RCA 8850.	184
IV-27.	Integral Pulse Height Distribution for the RCA 1P28.	187
IV-28.	Integral Pulse Height Distribution for the Johnston MML.	189
IV-29.	Non-Linearity of the 8850 Photon Counting System.	193
IV-30.	Expanded Plot of Figure IV-29.	195
IV-31.	Block Diagram of the Dual Counter.	201
IV-32.	Control Circuit for 12/24 Bit Select of Dual Counter.	203
IV-33.	Dual Counter Flag and Bit Select Circuit.	204
IV-34.	Dual Counter Input Select Circuit.	205
IV-35.	Counter and Bus Driver Circuit.	207

IV-36.	Counter Enable Circuit.	208
IV-37.	Prescale Select Circuit.	209
V-1.	TRANSFER Output in PIMS8 Format.	241
V-2.	TRANSFER Output in ASCII Format.	242
V-3.	RECORD Data Acquisition Flowchart.	246
V-4.	Comparison of Variable and Constant Integration Times.	252
V-5.	Comparison of Photon and Ion Stripchart Recordings with PIMS8 Real-Time PIE Plot.	257
VI-1.	CF_3Cl^+ from CF_3Cl	278
VI-2.	CF_3^+ from CF_3Cl	279
VI-3.	CF_2Cl^+ from CF_3Cl	280
VI-4.	CF_2Cl^+ from CF_2Cl_2	281
VI-5.	CFCl_2^+ from CF_2Cl_2	282
VI-6.	CF_2^+ from CF_2Cl_2	283
VI-7.	CFCl_3^+ from CFCl_3	284
VI-8.	CFCl_2^+ from CFCl_3	285
VI-9.	CCl_3^+ from CFCl_3	286
VI-10.	CFCl^+ from CFCl_3	287
VI-11.	CCl_2^+ from CFCl_3	288
VI-12.	CF^+ from CFCl_3	289
VI-13.	CCl^+ from CFCl_3	290
A-1.	The Assumed Slit Function.	318
A-2.	Measured PIE for Two Overlapping Lines.	320
A-3.	Measured PIE for a Line Superimposed on a Continuum.	321
A-4.	Calculated Stray Light Effects.	323
A-5.	Experimentally Observed Stray Light Effects.	325
A-6.	Calculated PIE for Differing Ion and Photon Linewidths.	327
A-7.	Calculated PIE for Offset Ion and Photon Peaks.	328

B-1.	Edge Connector Contact Assignments.	333
B-2.	Generation of a Device Select Signal.	334
B-3.	Gating of DS and IOP Signals.	334
B-4.	Generation of Function Select Signals.	335
C-1.	Block Diagram of the Real-Time Clock.	339
C-2.	Control Register and LED Indicator Decoding Circuits.	349
C-3.	Latch/Drive Control Circuits (Command Decoder).	351
C-4.	Counter/Flag Control Circuits (Command Decoder).	352
C-5.	Latch/Counter/Latch/Driver Circuits (Programmable Clock).	354
C-6.	Oscillator and Time Base Select Circuits.	356
C-7.	MK5009 Reset Glitch.	357
C-8.	Decade Clock Flag Circuit.	358

CHAPTER I INTRODUCTION

A. The Thesis

The technique of photoionization mass spectrometry (PIMS) has had a wide variety of applications [1]. In recent years, it has been most widely used in three distinct areas: the study of unimolecular kinetics of polyatomic ions [2-4], the study of kinetics and thermodynamics of ion-molecule reactions [5,6], and the determination of ionization potentials of atoms and molecules and of appearance potentials of fragment ions [1,3]. Studies of this latter type have been especially valuable (and frequent) because the ionization and appearance potentials obtained are often of high accuracy (a few meV), and can be combined with other thermodynamic information to yield equally accurate values of such parameters as heats of formation, bond dissociation energies, electron affinities and so forth, not only for molecules but also for ions and radicals. In fact, PIMS has become recognized as one of the most accurate techniques available for such measurements [3].

Photoionization mass spectrometry has a number of remarkable advantages, but also suffers from a number of severe experimental difficulties. At its best, it is characterized by low signal levels and lengthy experiments. The many extremely valuable experiments that have been performed to date have been possible only because of the

incredible patience of the experimenters and the great amount of effort that has been devoted to increasing the efficiency of the instruments and their operation. As in many fields, a large number of useful experiments seems to remain just out of reach no matter how sensitive or efficient the technique becomes. Nevertheless, each increase in efficiency or sensitivity makes more experiments possible, many of which yield very useful and sometimes unexpected information. The information obtained has been sufficiently valuable to justify past efforts at improving the technique, and also to encourage continuing effort toward that same goal.

The Michigan State University photoionization mass spectrometer is, in many ways, similar to most modern PIMS instruments, but it is capable of attaining a significantly higher overall operating efficiency. This thesis is intended to be a complete description of the MSU instrument in which the unique innovations in hardware and operating procedure that make possible the improved efficiency are emphasized. Most of the work that has been performed with the MSU instrument to date has involved the measurement of ionization and appearance potentials, and most of the concepts discussed in later chapters of this thesis were developed specifically for such studies. However, many of these concepts are applicable to other types of PIMS studies, especially ion-molecule reactions. On an even broader scale, many of the guidelines concerning user-computer-instrument interaction described here could be profitably applied to virtually any sort of computer-interfaced experiment.

Succeeding sections of this chapter are devoted to a basic introduction to photoionization mass spectrometry and the concepts

underlying the overall design approach used in the MSU instrument. The basic elements of a photoionization mass spectrometer, the measurements that are made with it, and the information that can be obtained from such measurements are all described briefly in Section B. Section C contains a description of some of the advantages of PIMS and the most important of the experimental difficulties involved are discussed in Section D. Then, in Section E, the capabilities and characteristics that a photoionization mass spectrometer should have if it is to obtain accurate data efficiently are discussed and a list of design requirements is developed. In Section F, the relative merits of several earlier designs are discussed in terms of how well they meet these requirements. Finally, with this background, the general design of the MSU instrument is presented in Section G, along with a brief description of the contents of succeeding chapters.

An attempt has been made to organize the material presented in this thesis so that any of the chapters following this may be read independently. In fact, each chapter is organized insofar as possible so that, once the introductory section has been read, any of the major sections of that chapter can be read without reference to other sections. It is strongly recommended, however, that Chapter I be read in its entirety before any other portions of the thesis are read.

B. Photoionization Mass Spectrometry

No comprehensive reviews of photoionization mass spectrometry have been published since the one by Reid [1] in 1971, which is now rather dated. Nevertheless, that review, together with references [2, 3, 5, 7-13] provide an excellent introduction to the experimental

methodology of PIMS [1,5,7,8], the processes contributing to the observed signals [2,3,9,13], methods of data analysis [10-13], the applications of the technique [1-3,5] and its relationship to other techniques [3,14,15]. Since the introduction to PIMS presented here is very brief and more than a little abridged, the reader is urged to consult these references for more detailed information.

The first reports describing mass spectrometric analysis of ions produced by photon impact on gaseous samples were published in 1929 [16] and 1932 [17]. At that time, and for years afterwards, the technique was severely limited by the available vacuum and light source technology, and no further PIMS experiments were reported until 1956 [18]. The first really modern instrument (i.e. one incorporating a vacuum monochromator and thus allowing selection of photon energies greater than about 6 eV) was described shortly thereafter [19]. During the next decade or so, the experimental technology was refined and data of ever-increasing quality contributed to a better understanding of the processes that were being observed. This in turn aided the development of more appropriate data analysis methods, which increased the potential of the technique even further.

As awareness of the capabilities of PIMS grew, so did the number of experimental studies. Reid's review of the field [1] indicates that by 1970, more than 130 studies involving "conventional" PIMS had been published, roughly half of them during the previous three years. The power of the technique has attracted increasing numbers of investigators since 1970, in spite of the experimental difficulties that remain even with the best modern equipment.

In the absence of any commercial source for complete PIMS

systems, investigators have had to assemble their own. Thus, no two photoionization mass spectrometers are alike. With only a few exceptions, however, most of them comprise the same functional elements, differing only in the specific implementation of each element. Figure I-1 illustrates the elements present in a basic photoionization mass spectrometer. The radiation from a light source is dispersed so that photons with a known energy (wavelength) distribution can be passed through a sample region which contains one or more gases at a known pressure. If the photons have sufficient energy, their absorption may lead to the production of ions by one or more of a number of processes. The number of photons transmitted through the sample is measured, as is the number of ions of a particular mass (selected by the mass filter).

The ratio of the number of ions formed to the number of photons transmitted by the sample is defined as the photoionization efficiency (PIE). Under typical operating conditions (when the sample pressure is sufficiently low that the amount of absorption by the sample is small), the PIE is proportional to the partial photoionization cross section for the ion being observed [20]. The PIE of an ion is most frequently measured as a function of wavelength, the plot of PIE versus either wavelength or photon energy being referred to simply as the PIE curve. The accurate determination of such curves is one of the most common goals in PIMS because of the tremendous amount of information that can be gained from their analysis. In practice, absolute photon intensity measurements [7] are so difficult that they are rarely attempted during a PIMS experiment. Instead, only the relative variation in photon intensity is measured. The so-called relative PIE curves which result provide almost as much information as the absolute PIE curves.

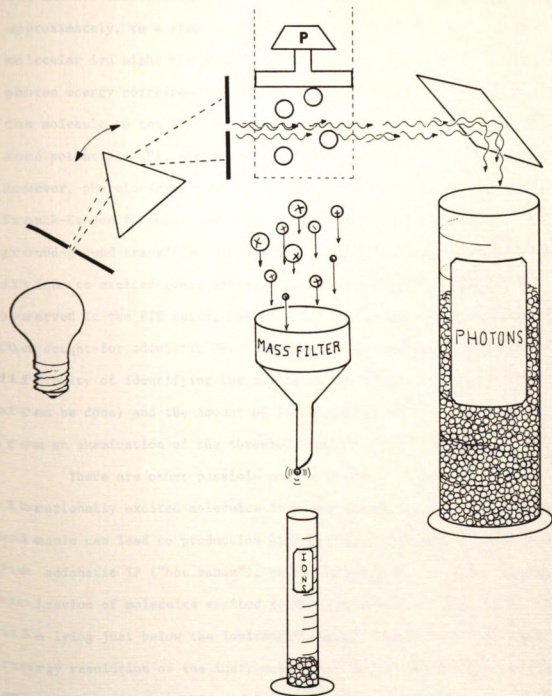


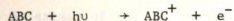
Figure I-1. PIMS Made Easy.

The determination of molecular ionization potentials is an example of one of the most common studies undertaken by PIMS. The threshold law for direct ionization by photon impact corresponds, at least approximately, to a step function [21,22]. The PIE curve for the molecular ion might therefore be expected to show a sudden onset at a photon energy corresponding to the transition from the ground state of the molecule to the ground state of the ion, i.e. the adiabatic appearance potential (IP), and this is in fact observed for some molecules. However, photoionizing transitions are vertical transitions and the Franck-Condon factors governing the transition may allow the ground-ground transition with only small probability relative to transitions to excited ionic states. Therefore, several steps may be observed in the PIE curve, one of which may or may not correspond to the sought-for adiabatic IP. The "extra" structure increases both the difficulty of identifying the adiabatic transition (although this can often be done) and the amount of information that can be extracted from an examination of the threshold region of the PIE curve.

There are other possible complications. Ionization of vibrationally excited molecules from the electronic ground state of the molecule can lead to production of significant numbers of ions below the adiabatic IP ("hot bands"), as can collisional or field induced ionization of molecules excited to Rydberg states of the neutral molecule lying just below the ionization energy. Because of the finite energy resolution of the instrument, what would otherwise be observed as sharp steps can be blurred by unresolved rotational or vibrational structure, and the PIE curve may be further complicated by the presence of autoionization structure above threshold. In spite of these

potential complications, extremely accurate values of adiabatic ionization potentials can frequently be measured [3] and even when complications are severe, accurate limits can often be set.

If the molecule ABC is ionized so that

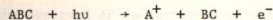


then

$$IP(ABC) = \Delta H_f(ABC^+) - \Delta H_f(ABC)$$

and if $\Delta H_f(ABC)$ is known, as is often the case, then $\Delta H_f(ABC^+)$ can be accurately determined.

Examination of PIE curves for fragment ions reveals that the onset of production of measureable numbers of such fragments, while rarely a perfect step function, is frequently rather well defined. The energy corresponding to such an onset is defined as the appearance potential (AP) for that fragment ion. The AP for a process such as



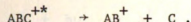
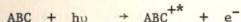
can also be used in a wide variety of thermodynamic cycles, such as

$$\begin{aligned} AP(A^+) &= \Delta H_f(A^+) + \Delta H_f(BC) - \Delta H_f(ABC) \\ &= \Delta H_f(A) + IP(A) + \Delta H_f(BC) - \Delta H_f(ABC) \\ &= IP(A) + D(A-BC). \end{aligned}$$

Equations such as these, in combination with other known information, can yield a wide variety of information about the thermodynamics of the molecules, ions and radicals under consideration.

The actual shapes of the fragment ion PIE curves near threshold depend not only on the relative positions and shapes of molecular and ionic potential surfaces, but also on the rate at which fragmentation occurs, the rates of competing processes, etc.[2,23]. One example is the so-called "kinetic shift" which may raise the measured AP for a

dissociative ionization process such as



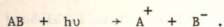
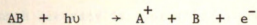
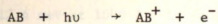
The rate at which the second step occurs may depend on the amount of energy in excess of the thermochemical threshold that is available in the parent ion. Since the transit time from source to detector in many mass spectrometers is on the order of a few microseconds, the rate constant for the second step must be larger than about $10^5 - 10^6 \text{ sec}^{-1}$ to produce measurable numbers of fragment ions. The excess energy required to produce this rate constant is the kinetic shift.

There are several other reasons why the measured appearance potential of a fragment ion may not correspond to the true thermochemical threshold for production of that ion. However, as pointed out by Chupka [2], dubious data can almost always be recognized, and even when the observed AP is suspected of being in error, the nature of the possible complications is such that the observed value is almost certain to be too high. Even in these cases, when PIMS cannot provide accurate thermochemical appearance potentials, it can still provide very precise upper limits.

C. Advantages of PIMS

The major advantage of PIMS over other photoionization techniques is that it usually provides unambiguous identification of the eventual products of an ionizing event. This is clearly an advantage over the measurement of ionization efficiencies by the total ion current method [24,25], which cannot distinguish between such

processes as these:



Photoelectron spectroscopy [14,26,27] (PES, another photoionization technique) is also an excellent technique for the measurement of ionization potentials. Although PES and PIMS often provide similar information, there are significant differences between the two. These differences could be regarded either as advantages or disadvantages, depending on the information being sought.¹ For instance, PES can provide precise information on how much internal energy is deposited in the ion by the ionization process, which PIMS cannot. But PES cannot provide any information on how the ion eventually disposes of that energy--whether it fragments or not, and into what pieces, with what probability. Another major difference is that photoelectron spectra are rarely complicated (or enriched, depending on point of view) by autoionizing processes, while autoionization structure is frequently the most intense of the features in a PIE curve.

A tremendous number of ionization efficiency measurements have been made during the past fifty years by the technique of electron impact mass spectrometry [3,28] (EIMS), over which PIMS has a number of very significant advantages. The most important is the relative ease with which the energy and energy bandwidth of the photon beam can be controlled. Wavelength accuracy and bandwidth of even the older PIMS

¹The two techniques are actually highly complementary, a fact which has prompted the development in recent years of combined photoion-photoelectron coincidence instruments [29-32].

instruments are typically better than 1 Å, which is equivalent to about 12.4 meV at 1000 Å. In fact, for more modern instruments, the wavelength accuracy is typically an order of magnitude better, and photon bandwidths as low as 0.035 Å (0.43 meV at 1000 Å) have been reported [33]. This precise control of ionizing energy is not only an advantage in ionization efficiency measurements, but also in the preparation of reactant ions in ion-molecule reaction studies. Reactant ions can often be prepared without simultaneously producing other, unwanted ions, and often with well-known distributions of internal energy [5].

In contrast, most EIMS measurements have used an electron beam with an energy spread of 1-2 eV. Furthermore, the actual energy of the electrons is influenced by stray fields and contact potentials so that the energy scale is not well characterized and must be calibrated with reference compounds. Since the contact potentials are likely to change with time, the calibration must be performed simultaneously with the experiment. Many of the procedures used to treat the data appear to have little theoretical justification [3,34], and the results of such measurements are typically a few tenths of an eV too high, but not predictably so [3]. In recent years, electron monochromators have been used to reduce the energy spread of the electron beam significantly (see the references in [3] and [34]), but even the best EIMS measurements are hard put to match the energy resolution and accuracy of even a very average PIMS instrument.

The only advantage of EIMS over PIMS is that much higher ion currents can be obtained, due to two factors. First, electron impact cross sections are larger than photoionization cross sections, by about

two orders of magnitude for 70 V electrons [34]. Second, the flux of ionizing particles is typically much higher in an electron beam than can be achieved with photons, perhaps by an order of magnitude even for undispersed radiation. However, much of the flux advantage is lost if an electron monochromator is used. Furthermore, the electron impact threshold law is not a step function, but rather an approximately linear function of excess energy (see the discussion in [3]). In contrast to the behavior of PIE curves, then, the EI ionization cross section will be several orders of magnitude smaller just above threshold than at higher energies, and it is not clear that EIMS has much of an intensity advantage in appearance potential measurements, provided that the two techniques are compared at comparable energy resolution. In addition, the photon beam introduces no space charge in the ionization region and causes no pyrolysis of the sample, which tend to be problems with electron beams and hot filaments.

D. Fundamental Experimental Difficulties

Although PIMS possesses numerous advantages, the price that must be paid to obtain those advantages is high, both in dollars and time. The photon energies needed to ionize and fragment most molecules correspond to wavelengths in the vacuum ultraviolet (VUV) (i.e. less than about 2000 Å). Most substances, including oxygen and nitrogen, absorb strongly in this region, and as a result the optical path in a photoionization mass spectrometer must be kept evacuated. This is made difficult by the fact that at wavelengths below the short wavelength cutoff of lithium fluoride (≈ 1050 Å, or 11.8 eV), there are no

satisfactory window materials which can be used to isolate lamp or sample gas from the rest of the instrument. The vacuum system must therefore be capable of maintaining an acceptably low pressure ($\approx 10^{-5}$ torr) in spite of intentional (and large!) leaks. Fortunately, the state of vacuum technology is such that this can be done, although the expense is significant. The book by Samson [7] provides an excellent discussion of the difficulties involved in working in the VUV spectral region and the techniques used to overcome them.

A more serious problem is the fact that, even with an excellent design and the best available equipment, ion signal intensities are often very low. Most experiments (with the possible exception of high-pressure ion-molecule reaction studies) must be performed with sample pressures of a few millitorr or less in order to avoid the collisional effects which can distort the PIE curves at higher pressure [35]. Thus, the ion signal cannot be increased by merely increasing the sample pressure. Instead, one must maximize the photon intensity and the ion collection efficiency of the mass analyzer. Unfortunately, the best available light sources in this spectral region tend to have limited intensities ($10^8 - 10^{10}$ photons $\text{sec}^{-1} \text{Å}^{-1}$, maximum) or high price tags (frequently more than \$10,000) or both. In addition, the best available optical materials have very low reflectivities (15–30% at 600 Å) so that every reflection to disperse or focus the light throws away a large fraction of whatever the light source is able to produce. Any decrease in the photon energy bandwidth to increase resolution also decreases the signal intensity, and further aggravates the problem.

Typically, when ion collection efficiencies approach 100%,

when the best available light source is used, and when the sample pressure and photon bandwidth have been increased as much as possible consistent with the goal of the experiment, one finds that many of the processes that might be of interest are still of extremely low intensity--perhaps only a few ions per second or worse. Use of ion counting techniques allows long enough data integration times that useable data can often be obtained even with such weak signals, but only at a cost of very long experiments. Data runs lasting several days have been reported [33], and they could conceivably last much longer. Given sufficient sample and patience, one could presumably obtain useful data at count rates of a few ions per minute, but at some point the time involved must become prohibitive.

E. Design Requirements

Each datum in the overall PIE curve must be measured with sufficient precision that all features of interest in the curve can be clearly discerned. In other words, the data must have a sufficiently high signal-to-noise ratio (S/N). If such features, although discernable, are distorted because of the non-ideal behavior of the instrumentation used in the measurement, then the high S/N may be of little use. Therefore, the primary goal in designing a photoionization mass spectrometer must be to create an instrument which can be used to acquire data with both precision and accuracy.

If precise and accurate data can be obtained, but only in experiments lasting for inordinate lengths of time, then the experiments will probably not be attempted at all. Thus, a secondary but highly

desirable goal is to design the instrument in such a way that the overall measurement process is as efficient as possible, so that the time necessary for a given series of experiments is minimized.

The triple goals of precision, accuracy and efficiency each impose certain requirements on the design of the instrument. These requirements are enumerated in the remainder of this section.

1. Precision of Data

A number of factors influence the S/N (and therefore the precision) of the measured photoionization efficiency. These are discussed in Chapter V, where it is shown that the limiting factor is almost always the S/N of the ion intensity measurement. In fact, in the usual case when ion counting techniques are employed, the S/N of a given PIE measurement is approximately equal to the square root of the total number of ions counted during the measurement. The precision of the measurement can therefore be increased either by increasing the number of ions formed or detected per unit time, or by increasing the measurement time, or both. Theoretically, any desired S/N could be achieved with even very low count rates, given sufficient time. Practically, however, instrumental instabilities usually impose some limit on the integration time. Thus it is important that the overall instrument design maximize the photon (and therefore ion) intensity, maximize the probability that an ion, once formed, will be detected, and also maximize the stability of the entire instrument, especially the data acquisition electronics. Furthermore, the experimental procedure used should ensure that every PIE measurement be at least long enough to attain the desired S/N.

2. Accuracy of Data

The accuracy with which the data are measured is also influenced by a variety of factors. Perhaps the most important of these is the accuracy of the data acquisition system(s). Ideally, the transfer function relating the recorded numbers to the photon and ion flux incident on the detectors should be linear. Although there may be some compromise involved (see Chapter IV), it is usually possible to achieve both adequate stability and linearity in these systems.

Even assuming ideal linearity in the data acquisition electronics, non-idealities of other sorts can adversely affect accuracy if not accounted for properly. For example, all of the most commonly used photon and ion detectors (see Chapter II, sections D and G) produce some background signal even in the absence of incident photons or ions, and these backgrounds are likely to change slowly with time. Unwanted backgrounds can be accounted for easily enough by occasionally closing a shutter between the light source and the sample region, measuring the background signal directly, and subtracting it from the other data. In fact, accuracy requires that such background measurements be made sufficiently often to "track" the change in background as a function of time, and the overall instrument design should allow this to be done.

The background-corrected photon and ion signals are still likely to drift with time for a number of reasons. For instance, the output of the light source and the grating reflectivity are certain to change with time, even at constant wavelength. However, changes in the photon or ion signals due to such causes will not affect the measurement of the PIE, since it is a ratio of ions to photons. However, if the photon detector efficiency deteriorates so that the measured photon

intensity decreases while the actual photon intensity does not, the accuracy of the measurement will suffer. Similarly, drifting sample pressure or a deteriorating ion multiplier, if not accounted for, can introduce error. Such drifts are unavoidable in real instruments, and although they are usually relatively slow, they can introduce artificial structure in those PIE curves which take a long time to measure.

The sample pressure can be monitored continuously, but about the only way to measure detector deterioration is to re-measure the PIE periodically at some "reference wavelength". Since the measured PIE at the reference wavelength is also affected by sample pressure, such reference measurements can be used to correct for pressure variation without having to make separate pressure measurements. In fact, reference measurements are sensitive to, and can be used to correct for, any sort of long term instabilities in the instrument. If the overall shape of the PIE curve is to be measured accurately, the instrument must be designed so that periodic reference measurements can be incorporated into its operating procedure. As with the background measurements, reference measurements must be made often enough to accurately sample the long term drifts that they are intended to correct.

Another source of inaccuracy is that the sensitivity of the photon detector may vary significantly with wavelength. This is an example of the kind of problem about which no information can be obtained during the experiment in which the PIE curve is measured. Instead, a separate experiment must be performed, comparing the response of the detector to that of a "standard" detector with a known wavelength response function. Post-experiment data analysis is then required to correct the PIE curve. A similar example is the variation

in ion detector sensitivity with ionic mass and composition. When it is important to know accurately the relative magnitudes of the PIE curves of different ions, correction factors must be measured in a separate experiment.

All real monochromators exhibit some degree of scattered light, which is another factor which must be taken into account. At any particular wavelength setting, the photons coming out of the exit slit will have not only the energies expected from the grating's dispersion and the slit widths, but also a (usually broad) distribution of other energies. The scattered light photons may cause ionization with a different probability than photons of the nominal energy to which the monochromator is set. The photon and ion detectors will measure such scattered photons and the ions they create along with the desired photons and ions. If the scattered light component of the measured ion and photon signals is not accounted for, it can grossly distort both the overall shape and the fine detail of the PIE curve. The possible artifacts, some of which have evidently been misunderstood in the past, are examined in detail in Appendix A.

Unfortunately, it is impossible to measure the actual scattered light distribution without employing a second monochromator, which is impractical in real PIMS instruments. One can, however, usually measure the photon intensity at one or two wavelengths that are not emitted by the particular photon source being used. The measured intensity at these wavelengths must be due entirely to scattered light, and these points can be used to estimate the amount of scattered light present at other wavelengths. Such estimated corrections will not usually be entirely accurate since at best they measure only "far scatter" and

cannot help estimate the "near scatter" to be expected at other wavelengths [36], but they may be good enough. Even if further manipulation is required to remove entirely the effects of scattered light (see Chapter V, section F), these measurements provide an excellent starting point, and the instrument should be designed to allow such scattered light measurements to be made.

3. Efficiency of Operation

Obtaining a high level of overall operating efficiency is especially important in photoionization mass spectrometry where the PIE curve for a weak fragment ion over a limited wavelength range can take literally days to acquire under the best of circumstances. As overall efficiency is increased, more experiments can be performed in a given time. Alternatively, better data can be obtained in that same time, so that processes which were too weak to be observed previously may be studied.

The instrument must produce as many photons of the desired energy as possible, and convey them to the sample region with an absolute minimum of loss. Any ions formed must have a high probability of being collected and focused into the mass analyzer, which should have a high transmission for ions of the desired mass. The photon and ion detectors must then be efficient in converting the incident particle flux into the electrical domain. Note that these are also requirements for good precision, as discussed above.

The extent to which the photon and ion signals are utilized can be as important as how efficiently they are produced. If intense signals are produced, but only a small fraction of the time can be spent

acquiring data, then most of the signal is wasted. Clearly, high efficiency demands that the percentage of time that the instrument is involved in actual data acquisition be maximized. The instrument should be designed to facilitate setting up the experiments so that the time spent doing so is minimized. Also, the data from one experiment often need to be reduced to a corrected PIE curve and plotted before the next experiment can be defined. Any time spent in doing so after the experiment is time that could be used performing the next experiment, and hence is wasted from the point of view of maximizing overall data acquisition time. In order to provide the user with the information necessary to make intelligent decisions about the type and sequence of succeeding experiments, without wasting potential experiment time, the ideal PIMS instrument would provide instantaneous display of fully corrected data. In practice, even partially corrected data may be sufficient, and the instrument should at least be designed to allow a partially corrected PIE curve to be displayed point by point as it is acquired (i.e., in "real time").

From the standpoint of total elapsed time, a PIE curve will be acquired most efficiently if each point is integrated for only a very short time. This conflicts with the requirement of long integration times for good precision, and some compromise must be reached. For any particular experiment, there will be some minimum acceptable signal-to-noise ratio. What the minimum is will depend on the system under study and the information that is desired, and must be defined by the investigator. Once defined, however, the goal of overall efficiency imposes the requirement that the integration time for each point be only long enough to reach the minimum desired S/N. Since the

ion intensity will vary with wavelength, the optimum integration time will also vary with wavelength. Best efficiency will be obtained if the instrument allows the integration time to be varied on a point by point basis so that all measured photoionization efficiencies are acquired with just the minimum acceptable signal-to-noise ratio.

Similarly, while accuracy requires that the correction measurements discussed above be made "often enough", efficiency requires that they not be made any more often than required for the minimum acceptable accuracy. Depending on the purpose of the experiment, this may mean that some or all of the correction measurements of which the instrument is capable should not be made at all. The instrument design should readily allow such changes in procedure. In fact, time spent acquiring data will be maximized if the instrument is designed to allow hardware or procedural modifications to be made quickly.

From the user's point of view, the overall efficiency of the study in progress will be markedly increased if the instrument does not require constant attention. If the instrument's operation can be automated, the user can spend time in other ways and will probably be more productive and certainly happier.

Finally, the instrument is not being fully utilized when shut down for repairs. It should therefore be designed to be as foolproof as possible, and ideally it should be able to take corrective or preventive action automatically in the event of an emergency or accident, especially if no operator is present.

A summary of the design requirements discussed in this section is presented in Table I-1.

TABLE I-1. Requirements Imposed by Design Goals

Goal	Requirements
Precision	High ion count rate Stable instrumentation Long integration times
Accuracy	Linear data acquisition electronics Post-experiment correction measurements (Photon detector sensitivity vs. wavelength) (Ion detector sensitivity vs. mass) During-experiment correction measurements (Background) (Pressure) ("Reference") (Scattered light)
Efficiency	High percentage of time doing data acquisition (Facilitate setup, use) (Real-time data reduction, display) Point-by-point optimization of integration time Flexibility for hardware or procedure changes Correct and automatic emergency action Constant operator attention unnecessary

F. Merits of Previous Designs

1. Basis of Comparison

Since no two photoionization mass spectrometers are alike, any attempt at classifying them is bound to lump together dissimilar instruments. This fact has not prevented classification attempts from being made, however, most of which have been based on the type of monochromator or mass analyzer used. While recognizing that differences in these important elements can have tremendous influence on the capabilities of the instrument, discussion of such differences is postponed to Chapter II. Here, however, an attempt is made to classify the various instruments in terms of the operating procedures used with them and their basic approach to data acquisition. At first glance, the

operating procedure might not seem to be a "part of the instrument", since it is, after all, only what the user decides to do with the instrument. However, it will be shown that the range of procedures from which the operator may choose is definitely limited by the hardware of the instrument, particularly the data acquisition system. Certain aspects of the experimental procedure can be designed into or out of an instrument, and therefore the operating procedures to be used are an important factor to consider in designing the instrument, and are a legitimate basis for comparison.

For the purpose of such a comparison, the various types of instruments discussed in the remainder of this chapter are assumed to have basically equal capabilities in other areas. Specifically, ion and photon signal intensities and instrument stability are assumed equal, as is the accuracy of the data acquisition electronics. The post-experiment correction measurements are assumed to be possible with all instruments, also.

2. Manual Operation

Nearly all of the early photoionization mass spectrometers and even some later ones are operated manually [19,37-42], and can be diagrammed as shown in Figure I-2. The user adjusts such parameters as the wavelength, mass, optical resolution and sample pressure via instrument controls which may be electronic or mechanical. Data acquisition consists of writing down photon and ion intensity data which are read on meters, digital counters or strip chart recorders. The operator can usually control the time constant or integration time of the data acquisition electronics to adjust the S/N of the data. There are also

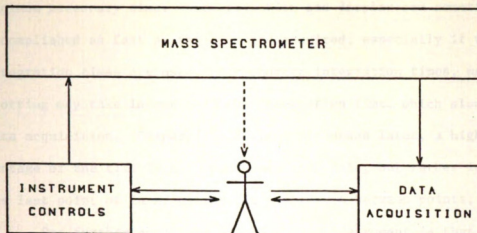


FIGURE I-2. A Manual Instrument.

other elements of data acquisition involved when the operator gains information by monitoring meters, knob positions, etc. on the instrument controls. The operator may also check the status of the mass spectrometer by reading vacuum gauges, checking positions of vacuum valve actuators, feeling or listening to the pumps, and so forth.

The manual approach has a number of advantages, and in fact meets most of the requirements outlined in the previous section. It is the quickest design to implement and, requiring an absolute minimum of data acquisition equipment, is the least expensive design from the standpoint of permanent equipment costs. It is also the most flexible design--changes in procedure are implemented literally as soon as the operator decides what to do. The user can make any of the necessary correction measurements, as often as necessary (including never).

Users of ion counting instruments will frequently allow the data to integrate until some relatively constant number of ions have been counted. As shown in Chapter V, this is very nearly the right procedure to use in order to acquire all points at a constant S/N in the

minimum necessary time. Data reduction and display can sometimes be accomplished as fast as the data are acquired, especially if very long integration times are used. For shorter integration times, manual plotting may take longer than the integration time, which slows down data acquisition. Compared to designs discussed later, a high percentage of the time is not spent acquiring data, but rather in plotting the last point or in adjusting the wavelength between points, etc.

One further advantage of the manual instrument is that if (correction: when) something goes wrong, the operator can take corrective action as soon as the problem is noticed. Unfortunately, the operator may not notice that there is a problem very quickly, and may have a slow response time or make an incorrect response even when the problem is noticed.

The major disadvantage of the manual approach, however, is that the presence of the operator is required continually. The instrument cannot be used continuously for more than several hours at a time, unless several people work shifts tending the instrument. Despite its initial low cost, the manual approach soon becomes expensive enough that one of the alternatives (to be discussed shortly) becomes economically competitive, if the operator is being paid something like a living wage.

A final drawback to manual data acquisition is not a necessary limitation of the approach, but is virtually always encountered in practice. As the operator's boredom increases, the "minimum acceptable S/N" seems to decline, as does the desired frequency of correction measurements. The S/N of manually acquired data seems to be always far less than the ultimate that the instrument could provide if only the

operator were more patient. Furthermore, data points are often taken at much wider wavelength intervals than optimum, so that some of the fine structure in the PIE curve may be distorted by undersampling or may be missed entirely.

3. Data Logging

In an attempt to overcome the limitations of manual operation, most recent designs have included provisions for some sort of automated data acquisition. One of the earliest approaches [43-45] is shown in Figure I-3. The basic equipment is the same as in the manual design, except that an electronic sequencer and some device to record digital data (e.g., a Teletype, paper tape punch, magnetic tape recorder, etc.) are added, and the wavelength is controlled with a stepping motor drive.

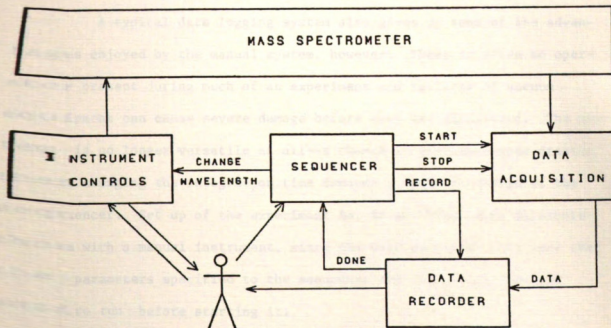


Figure I-3. A Simple Data Logging Instrument.

Setting up the experiment is done exactly as in the case of the manual instrument, with the user admitting the sample, adjusting wavelength resolution, mass setting and initial wavelength. In addition, an integration time and wavelength increment are set in the sequencer controls. When started, the sequencer integrates photon and ion signals for the preset time, causes the data to be stored, advances the monochromator the desired wavelength interval and begins the entire process over again. This type of operation with a simple sequencer and data recorder is frequently called "data logging".

The overwhelming advantage of this type of design is that the user need not be present. The instrument can run 24 hours a day, using long integration times and small wavelength intervals, without losing patience. The initial cost of the added equipment is soon repaid by savings in user's time and in the high quality of data that can be obtained.

A typical data logging system also gives up some of the advantages enjoyed by the manual system, however. There is often no operator present during much of an experiment and failures of vacuum equipment can cause severe damage before they are discovered. The system is no longer versatile at all--a change in procedure more drastic than changing the integration time demands a total re-design of the sequencer. Set up of the experiment is, if anything, more difficult than with a manual instrument, since the user must make very sure that the parameters specified to the sequencer are appropriate for the entire run before starting it.

The same integration time is used for every point in an experiment, no matter what the intensity may be. This means that the

tradeoff between integration time and signal-to-noise ratio must be made on a scan-by-scan basis, rather than the much more efficient point-by-point basis often used during manual operation. The correction measurements that should often be made during the experiment are usually not made with this sort of system, due to the extreme complexity that would be required in the sequencer design. If made at all, correction measurements must be made manually, and if they are required frequently, the advantage of not requiring the presence of an operator is eliminated. Hence, when such a data acquisition system is used, correction measurements are often made only at the beginning and end of a run, and do not necessarily give an accurate indication of drifts during long experiments.

Data reduction must be done after the experiment. If paper or magnetic tape is used to record the data, the post-experiment data reduction is easily performed by computer, but the user may have no way of telling how the experiment is progressing or whether the acquired data are actually of the desired quality until after such reduction is complete. It may take hours after the completion of a several-day experiment to discover that the entire run is wasted because of some factor overlooked during the setup or some problem which arose in the course of the experiment.

Another approach which has been used recently [46] attempts to provide at least a crude real-time PIE output, and also attempts to optimize data integration time on a point-by-point basis. In this "modified data logger" approach, the ion and photon signals are not integrated for a preset time, but rather until a preset number of photons have been counted. Thus in regions of low photon intensity (and

hence low ion intensity) the integration time will be longer. As shown in Chapter V, this approach is successful in optimizing the integration time, providing that the fluctuations in the ion intensity are caused by fluctuations in the incident photon intensity. This is often the case, but not always. If the PIE curve has sharp structure, or if the threshold region is being measured, then the ion intensity can vary radically in regions of relatively constant photon intensity, and this approach is no more valid than the constant integration time used in the simple data loggers.

The modified data logger has the further advantage that the number of ions recorded at each wavelength is normalized to a constant photon intensity, and therefore equivalent to the (uncorrected) PIE. If the data are printed out as they are acquired, it is possible to get at least some indication of the progress of the experiment without necessitating any further calculation.

In one design [47] which incorporates this modification, the sequencer varies the mass rather than the wavelength when the desired photon count has been reached. When all ions have been recorded, the user manually adjusts the wavelength and restarts the sequencer. This approach is claimed to provide "uniform precision with the maximum possible efficiency obtainable without complete computer control of the instrument" [47]. This claim can only be valid when (1) all fragment ions to be observed are created with equal or nearly equal probability at all wavelengths studied, and (2) the number of wavelengths at which data are desired is fewer than the number of ions to be studied. Except in very special cases, neither of these requirements is met. Different fragment ions from a given molecule typically have

intensities which differ by 2-3 orders of magnitude or more, even at those wavelengths where all ions are formed. The approach described in [47] will use the same integration time (at a given wavelength) for all ions, and the resulting PIEs will not be of comparable precision, although the precision for a given fragment may be roughly constant as the wavelength is varied. Furthermore, a typical PIE curve may contain data at hundreds or even thousands of wavelengths, though fewer than a dozen or so fragment ions will usually be of interest. The presence of the operator may be required often to change the wavelength so that this design, in spite of being "automated", should probably be regarded more as a slightly modified manual instrument than as an automated data logger. Either of the two data logger designs described previously will usually provide higher overall operating efficiency than this design, even though lacking "complete computer control".

The two basic data logger designs described earlier each introduce a number of problems, but the ability to collect data unattended over long periods of time can overcome the major disadvantage of the manual instruments. The fact that this one disadvantage is significant is demonstrated by the impressive quality of data that can be obtained with the automated instruments, despite their disadvantages.

G. The MSU Instrument

The photoionization mass spectrometer described in this thesis (Figure I-4) is also designed to permit unattended operation, but to simultaneously retain the advantages previously obtainable only by manual operation. It combines the best features of the designs

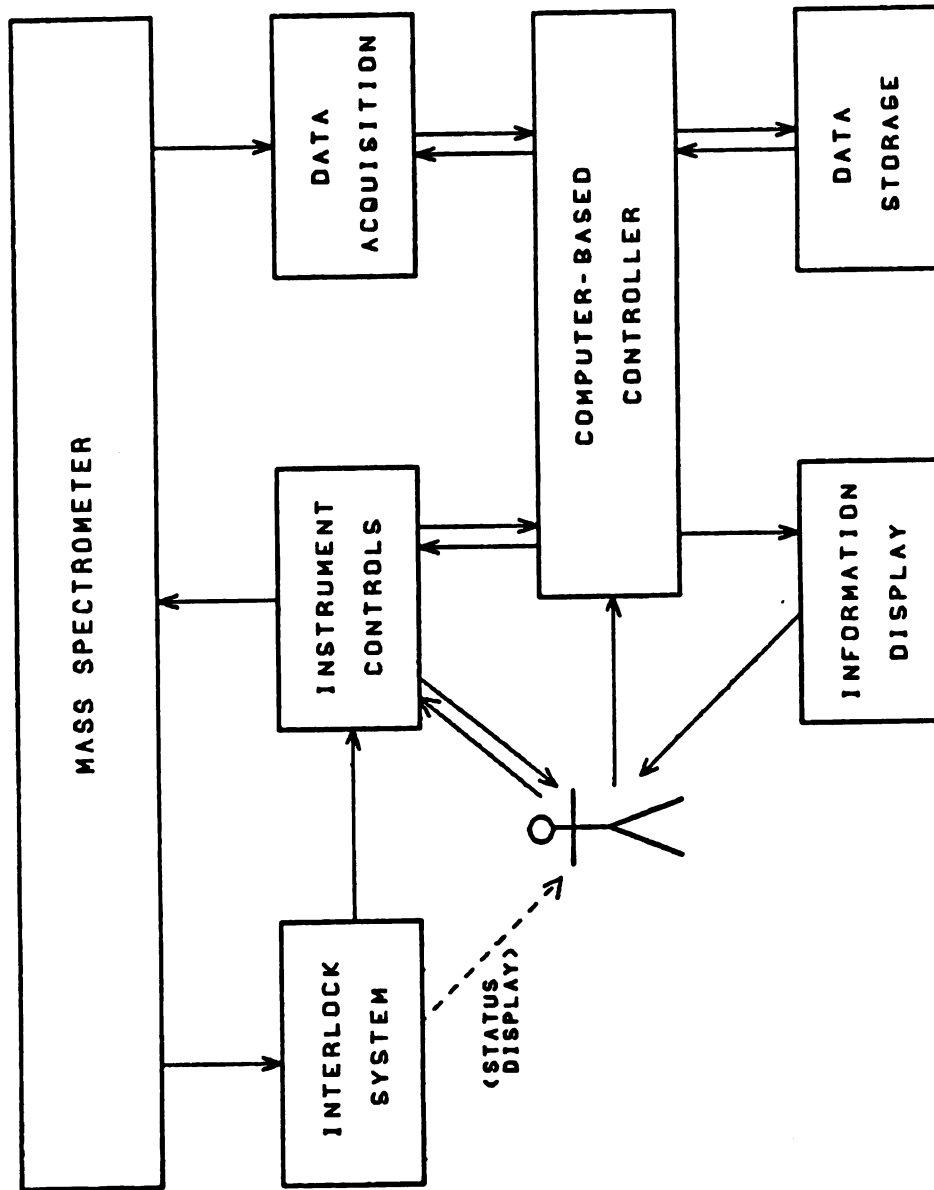


Figure I-4. The MSU PIMS Instrument.

discussed in the previous section, but also includes advantageous features not found in any previous design.

The basic elements of the instrument, described in detail in Chapter II, are fairly typical of those used in many other photoionization mass spectrometers, except that the vacuum system has been designed so that the components of the instrument within the vacuum chambers are easily accessible.

One of the major disadvantages of unattended operation is eliminated by incorporating an extensive interlock system (described in Chapter III) which continuously monitors the status of the instrument. If any undesirable condition is detected, it automatically and rapidly performs those actions necessary to prevent major damage to the instrument. Since it monitors the instrument's status continuously and has a very much faster reaction time than a human operator, the interlock system usually takes corrective action before the human operator has become aware of the existence of a problem. Another feature of the interlock system is that during start-up or shut-down of the instrument, it prevents components of the vacuum or electrical systems from being activated in an incorrect sequence, thus reducing accidents due to operator error. For these reasons, it is an extremely valuable part of the instrument even during manual operation.

The incorporation of a computer-based controller, rather than a simple sequencer, allows the other disadvantages of the data logging designs to be overcome. The computer presently controls a shutter in the photon beam, the wavelength drive, and the acquisition of photon and ion data. The interface and data acquisition circuits involved, including a novel high-speed, low cost circuit for photon or ion

counting, are described in Chapter IV.

The specific actions taken by the computer during an experiment are controlled by the programs (software) used, rather than by permanently wired electronic circuits. This is both the main disadvantage and the main advantage of computer-controlled instrumentation. The programs must be written and de-bugged before the instrument can be run. The software system is as important to the overall performance of the instrument as the hardware, and so designing the software becomes a task that must be performed in addition to designing the hardware of the instrument. However, the use of software to control the operations of the system results in a great deal of flexibility and versatility, since most changes in experimental procedure can be accomplished merely by running a different program, rather than by re-designing an electronic sequencer, and re-writing a program can usually be accomplished more quickly.

The computerized instrument is programmed to make the various necessary correction measurements that the simpler data logger instruments cannot perform automatically. The computer calculates and plots the PIE curve, at least approximately corrected for scattered light, background, etc., as each point is acquired and stored, so that the quality of the data can be judged at any time during the experiment. If it is not as high as necessary, the experiment is easily terminated and restarted with modified parameters without wasting further time. Also, the computer is programmed to integrate each point until a desired PIE (not photon) S/N is achieved, which results in nearly optimum efficiency of data acquisition.

Besides including all the advantages of other instruments, a computerized instrument can potentially provide even further advantages. For instance, any real instrument is certain to be incompletely automated (somebody has to at least find a sample!), so that certain tasks must be performed manually, especially during the initial stages of an experiment. The MSU instrument is designed to facilitate these manual tasks as far as possible, in order to reduce to a minimum the time spent in their performance. The computer is programmed to guard against common operator errors during the stage of the experiment in which data acquisition parameters are being specified. Post-data acquisition data reduction can be performed with the same computer so that the experiment can flow smoothly and quickly from setup to acquisition to analysis and back in a manner which can greatly increase the overall efficiency of the experiment.

Table I-2 summarizes the extent to which the basic designs (manual, data logger, modified data logger and the MSU computer-controlled design) meet the design requirements described earlier. It will be noted that the design reported here is (not surprisingly!) superior to the other designs in common use. It cannot be emphasized too strongly, however, that the major advantages attributed to the MSU design are not necessarily obtained as a result of adding a computer to an instrument. The potential advantages of computer-interfaced instrumentation are only potential advantages until realized by the software system to be used with the instrument. If the interactions between the instrument, computer and user are not carefully evaluated, it will not be possible to design an effective software system, and the result of merely "adding a computer" may be to create an instrument with

TABLE I-2. Comparison of Four Design Approaches^a

Requirement	Manual	Data Logger	Modified Logger	MSU
Correction measurements made during experiment	?	-	-	✓
Provides real-time display of corrected data	✓	-	*	✓
Pointwise optimization of integration time	✓	-	*	✓
Operating procedures easily changed (flexible)	✓	-	-	✓
Prompt, correct emergency action taken	✓	-	-	✓
Long integration times, close point spacing possible .	?	✓	✓	✓
Operator's constant attention not required	-	✓	✓	✓
Design facilitates setup and use	-	-	-	✓

^aKey:

- = does not meet requirement
- ✓ = meets requirement
- ? = can meet requirement, but usually does not in practice
- * = meets requirement under certain circumstances, or meets modified requirement

capabilities about the same as a data logger instrument, but which is more expensive and much more difficult to use.

Chapter V of this thesis discusses these concepts in more detail, both in relation to computerized instruments in general and photoionization mass spectrometers in particular. The specific operating procedures used with the MSU instrument and the important features of the PIMS software system are described, and data are presented which illustrate the specific advantages of the overall design approach.

CHAPTER II

THE MASS SPECTROMETER

A. Introduction

Any mass spectrometer can be described in terms of the basic elements shown in Figure II-1. The ion source comprises all components necessary to create ions from the sample. The ion optics guide the ions from the source to the detector and provide some way of discriminating against unwanted ions according to their mass-to-charge ratio. The ion detector, more accurately called a transducer, is responsible for converting the arrival rate of incident ions into an electrical signal proportional to that rate, which can then be processed by the data acquisition electronics. The maximum pressure at which each of these three components can function properly varies, but is always well

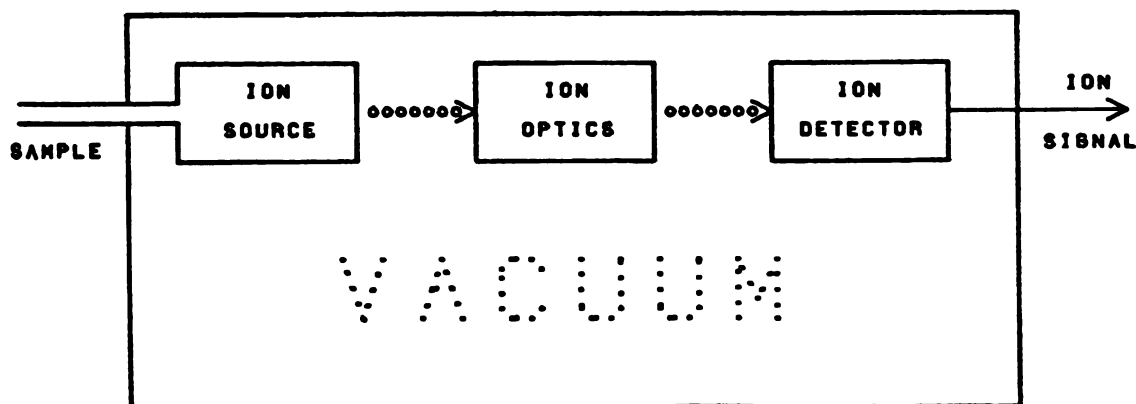


FIGURE II-1. The Basic Elements of a Mass Spectrometer.

below atmospheric pressure (e.g., about 10^{-5} torr for most ion optics). Therefore, a vacuum system capable of maintaining the necessary pressure must be included as the fourth basic element.

In the case of photoionization mass spectrometry, the ion source consists of a photon source, photon optics, an interaction region, and a photon transducer, as shown in Figure II-2. The photon source includes all the components necessary to create the photons, the photon optics guide the photons to the interaction region and discriminate against photons with unwanted energies, and the photon transducer converts the photon arrival rate to an electronic signal. Some photoionization mass spectrometers include no provision for photon energy

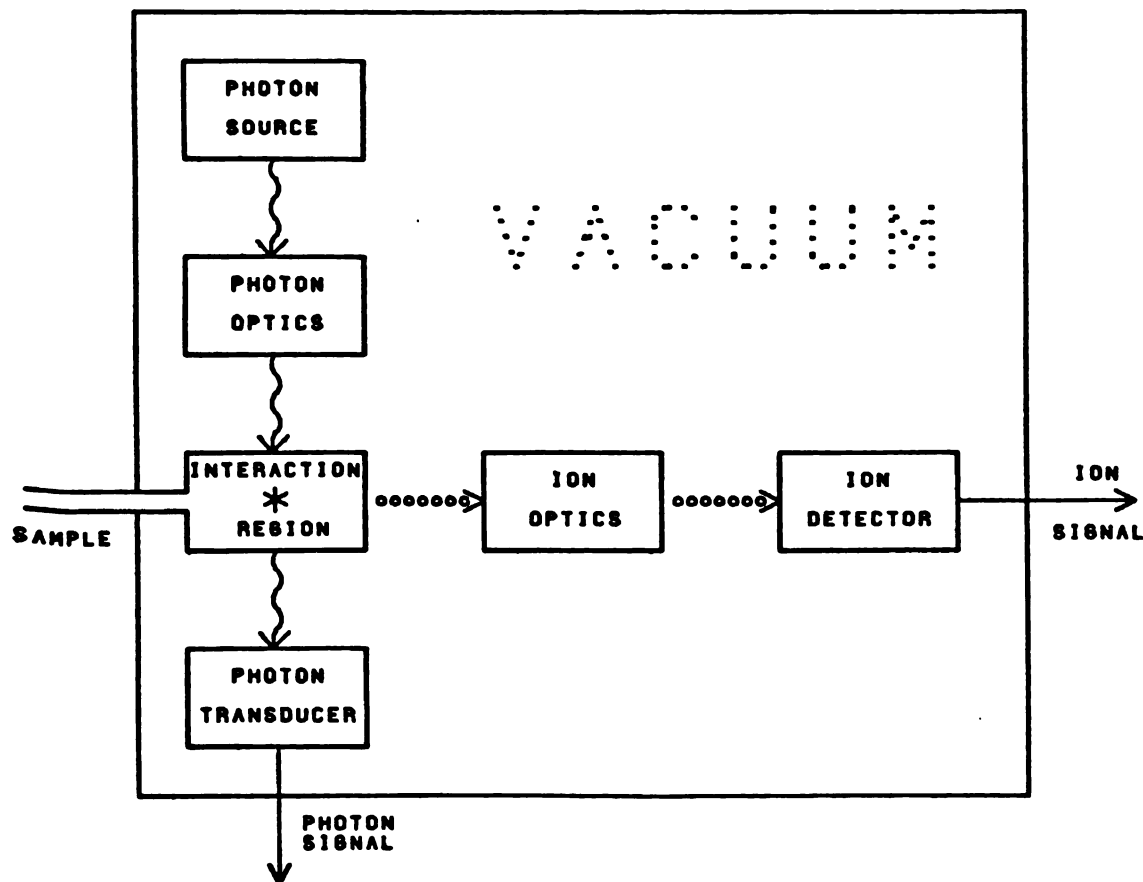


FIGURE II-2. The Basic Elements of a Photoionization Mass Spectrometer.

discrimination. This considerably simplifies the photon optics, but prevents the use of the instrument for determination of ionization efficiency curves, and such designs will not be considered further. The interaction region is simply some sort of sample molecule reservoir in which the ionization takes place. It is often referred to as the ion source, independent of photon source, optics, and transducer, a practice which is followed in the remainder of this thesis.

Each of the components in Figure II-2 can be implemented in a variety of ways. Each succeeding section of this chapter is devoted to one of the components shown in that figure, and includes both a brief discussion of the major alternatives and a description of the corresponding portion of the Michigan State University photoionization mass spectrometer.

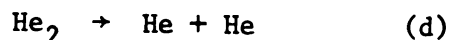
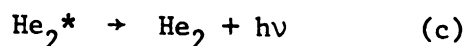
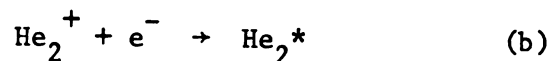
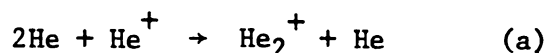
B. Photon Sources

1. Overview

The ideal photon source for PIMS would continuously emit intense radiation with a wide and continuous energy distribution, preferably covering at least the range of 300-2000 Å. Ideally, it should also be compact and inexpensive. References [7] and [8] give an excellent introduction to the many different devices which have been used as photon sources in the vacuum ultraviolet, most of which fail to meet any of these requirements. Only three types of radiation sources come close enough to meeting these requirements to be useful in PIMS: synchrotron radiation, the rare gas discharges, and the molecular hydrogen emission spectrum.

From the standpoint of wide, continuous energy distribution, the radiation emitted as a result of the centripetal acceleration of electrons in synchrotron rings is easily the best of the available sources. The energy distribution of a synchrotron source depends on the size of the ring and the electron energy, but might typically rise to a peak at about 100–200 Å (124–62 eV) after an onset at lower wavelength, and then slowly decrease towards longer wavelength, retaining perhaps 20% of the peak intensity at 2000 Å (6.2 eV). Although actually a pulsed source, the synchrotron repetition rate is so rapid that the pulsed nature can usually be ignored. Although far from compact, its biggest drawback is its expense; few synchrotrons exist in the world, and most people simply do not have access to such a light source. Therefore, although synchrotron radiation has been used for PIMS [38,39], most workers in the field are forced to use other photon sources.

In 1930, J. J. Hopfield discovered that a high-power pulsed discharge in helium produced an emission continuum ranging from roughly 600 Å to 1100 Å [48–50]. This "Hopfield continuum" results when excited He_2 molecules undergo a transition to the steeply repulsive portion of the unstable ground state of the molecule (see Figure II-3). The mechanism includes the following steps:



Since the recombination of He_2^+ ions and electrons (b) cannot occur efficiently under the conditions that exist during the discharge, the Hopfield continuum is emitted only as an afterglow, lasting typically

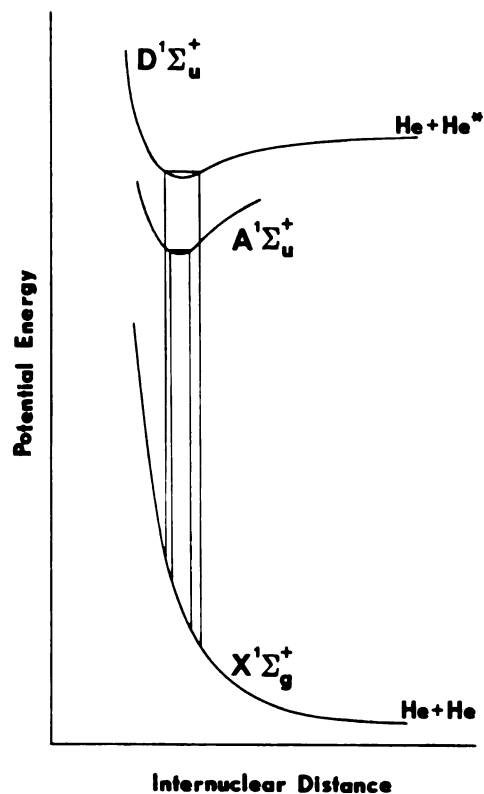


FIGURE II-3. Schematic Diagram of the Potential Energy Curves of He_2 .

ten to twenty microseconds after the discharge, depending on pressure. If the discharges occur frequently enough and are short compared to the time between discharges, the emission becomes essentially continuous.

Similar continua have been discovered for all of the rare gases [51-55]. Their approximate wavelength limits are summarized in Table II-1. The major advantage of these sources is that they can be produced

Table II-1. Approximate Wavelength Limits of Rare Gas Continua.

Gas	Approximate Range, Å
He	580 - 1100
Ne	740 - 1000
Ar	1067 - 1550
Kr	1250 - 1800
Xe	1480 - 2000

in an apparatus which is more conveniently used and is much less expensive than a synchrotron, although the expense and size of the instrumentation (described later) is still considerable. The continua are relatively smooth, except for a few sharp lines due to atomic emission from impurities in the gases.

Since absolute intensity measurements are rare, comparisons are difficult to make, but the rare gas continua are usually less intense than the radiation from a synchrotron source. Their intensity depends strongly on the power supply used to cause the discharge, however, and the better rare gas lamps may actually exceed the intensity of some of the synchrotron sources, especially at longer wavelengths. Although it is inconvenient to have to change gases in order to cover the entire wavelength range, the most important disadvantage of these continua as radiation sources for PIMS is that none extend below 580 Å (21.4 eV). Furthermore, although the Hopfield continuum extends to 1100 Å, it is extremely weak above about 1000 Å (see Figure II-4).

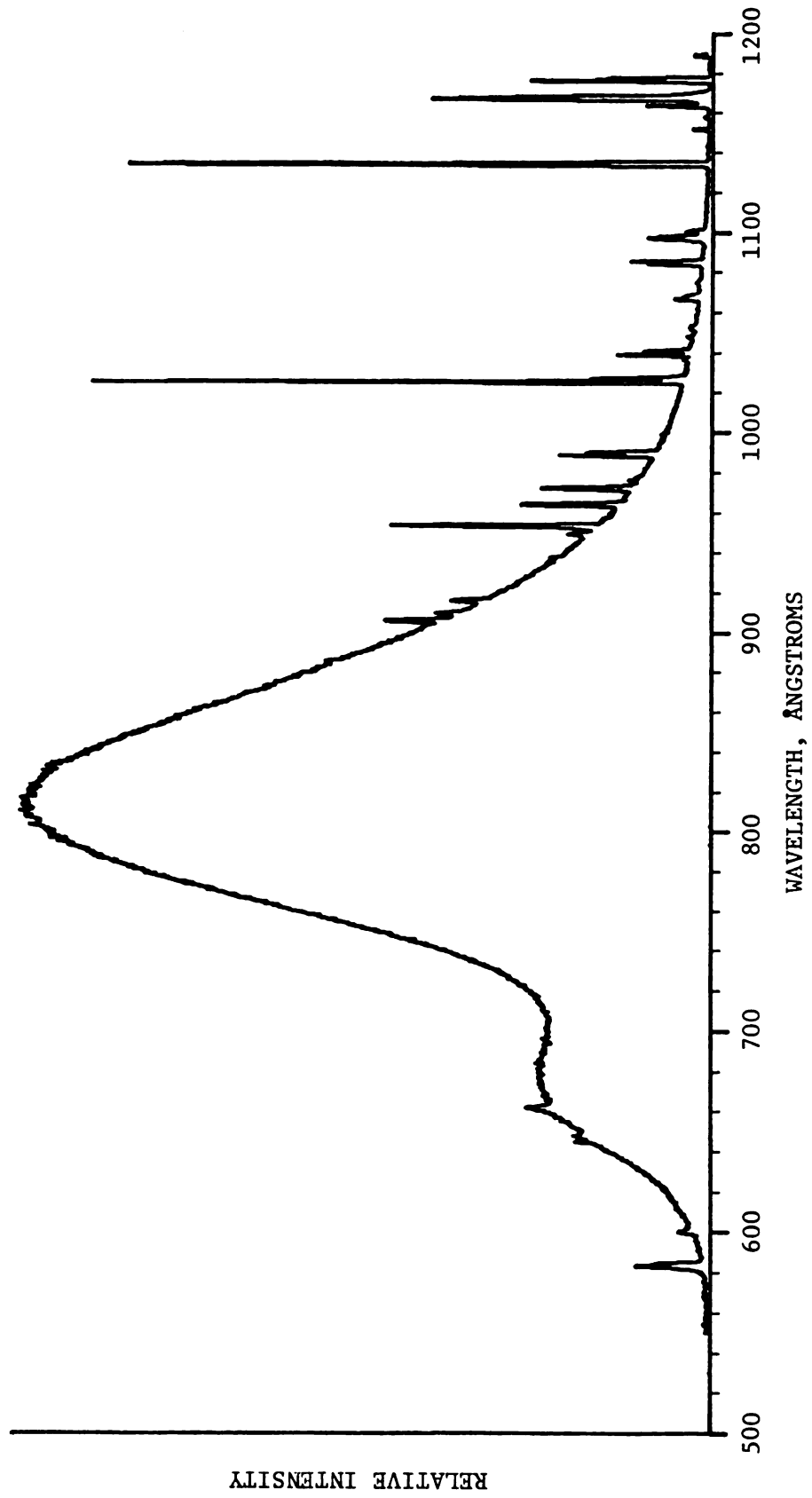


Figure II-4. The Helium Continuum.

In practice, there is a gap in the wavelength coverage between the onset of the argon continuum (1067 Å) and about 1000 Å, or about 11.6 to 12.4 eV.

This gap can be filled with the molecular hydrogen emission spectrum produced in a DC discharge through a few torr of hydrogen. The emission is not a true continuum, but consists of very closely spaced lines, as shown in Figure II-5. This pseudo-continuum, often called the "many-lined" hydrogen spectrum, can be used to obtain data at wavelength intervals on the order of an angstrom or so. While its non-continuous energy distribution is far from ideal, it at least produces significant intensity in a region which would otherwise be practically inaccessible. The rare gas continua and the many-lined hydrogen spectrum, taken together, can therefore provide useful intensity from 580 Å to 2000 Å (21.4 to 6.2 eV), which is a sufficiently broad range for an extremely wide variety of experiments.

2. Lamps

The rare gas continua have been produced in lamps following one of two basic designs. The Π lamp [56], shown in Figure II-6, is usually made entirely of quartz and is water cooled to prevent breakage from the thermal expansion that occurs as a result of the heat generated in the discharge. The electrodes, usually made of pure aluminum, cannot be cooled and therefore tend to sputter rather badly, and may even melt at high power levels and low pressures [6]. One of the main advantages of this lamp is that, with the electrodes in side arms, sputtered material is not likely to cause clogging of the monochromator entrance slit. The lamp is usually not shielded and therefore radiates

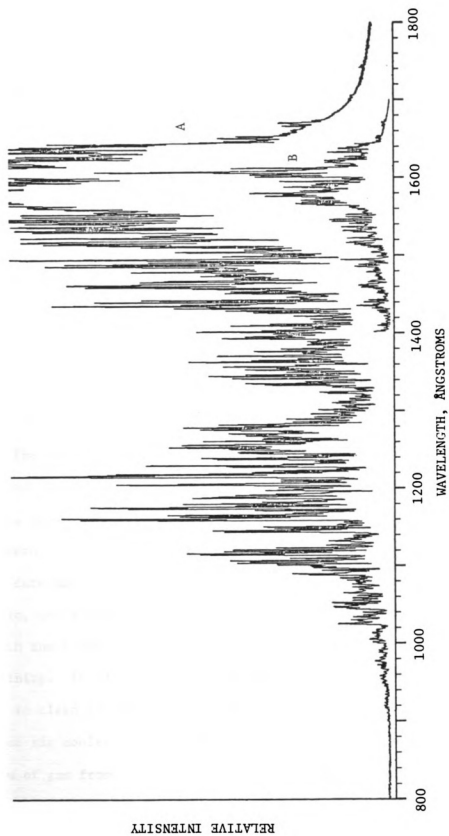


Figure II-5. The Hydrogen Many-Lined Spectrum. Curve B is reduced by about a factor of ten relative to curve A.

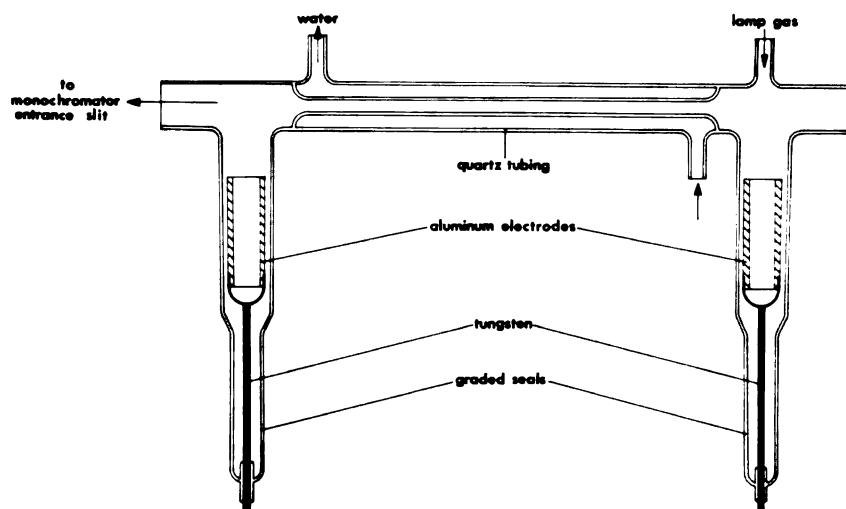


Figure II-6. A Π -Type Rare Gas Discharge Lamp.

electromagnetic noise which can severely affect the performance of sensitive data acquisition electronics.

The lamp used with the MSU instrument is a modified version of the second basic type of lamp, the so-called Hinterregger lamp [57]. The lamp, shown in Figure II-7, has been used in this laboratory to excite both the helium and argon continua, though virtually all work done to date has been with helium. The Hinterregger design is axially symmetric, which makes proper electromagnetic shielding much easier than with the Π lamp, and also reduces the shock hazard to people in the vicinity. It is easily disassembled to replace the quartz discharge tube or to clean the aluminum electrodes. The earliest Hinterregger lamps had air cooled cathodes, through which the lamp gas was admitted. The flow of gas from the cathode into the monochromator tended to carry sputtered material with it, leading to a rapid clogging of the entrance slit. The lamp shown here is based on a design first used in the

Figure II-7. The MSU Hinterregger Lamp.

- a) monochromator entrance slit
- b) plunger slit seal
- c) insulating disc
- d) one of three holes for gas inlet, pressure measurement and plunger control rod
- e) anode cooling water
- f) preslit
- g) O-ring compression seal
- h) quartz discharge tube
- i) aluminum anode
- j) shield and ground return
- k) one of three cathode support rods
- l) glass insulators
- m) aluminum cathode
- n) cathode cooling water
- o) backpumping port
- p) insulator (one of three)
- q) alignment screw (one of three)

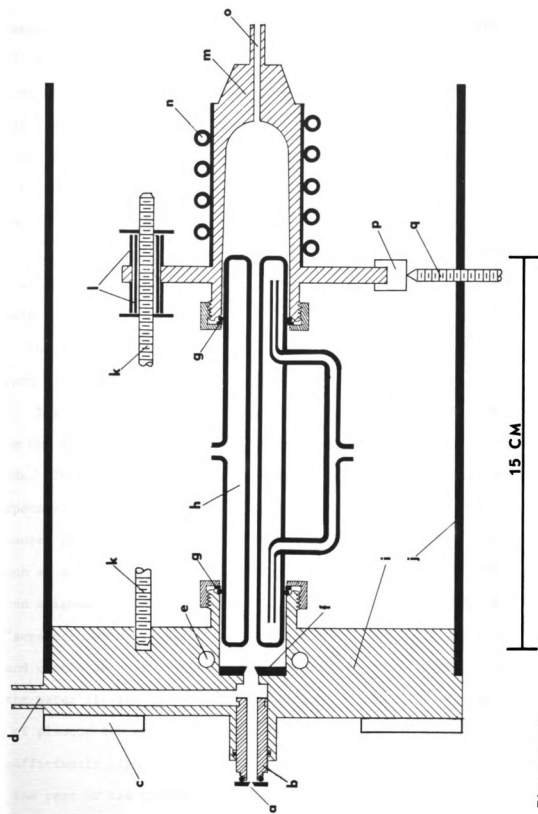


Figure II-7.

laboratories of W. A. Chupka and J. Berkowitz at Argonne National Laboratory (ANL), and incorporates a design modification first described by them [43] to reduce the slit clogging problem. The gas is admitted at the anode end of the lamp, and flows into the monochromator through the entrance slit. However, an auxiliary pump connected to the cathode also causes a gas flow from anode to cathode, which to a large extent prevents sputtered material from reaching the slits. Like most modern Hinterregger lamps, both cathode and anode are water cooled, as is the discharge tube, which allows operation at higher power without severe sputtering. The ANL design also incorporates a separate port at the anode for measuring the pressure of the gas in the lamp. This avoids errors that arise due to pressure gradients when the pressure is measured closer to the gas supply or backing pump.

The MSU lamp incorporates two minor modifications not present in the ANL design. One is the addition of alignment screws around the cathode. These were found to be necessary, as when longer lamps were incorporated to increase the output intensity, the weight of the cathode caused some droop in the main cathode support rods. This caused as much as a factor of two or three decrease in intensity compared with an aligned lamp. The second minor modification is the use of two "separate" water supplies, one for the anode and one for the cathode and discharge tube. The water is actually from the same supply, but the water flowing to and from the cathode passes through some fifty feet of plastic tubing (see Figure II-8). This long column of water has sufficiently high resistance to effectively insulate the cathode from the rest of the plumbing (at ground potential), and thus practically

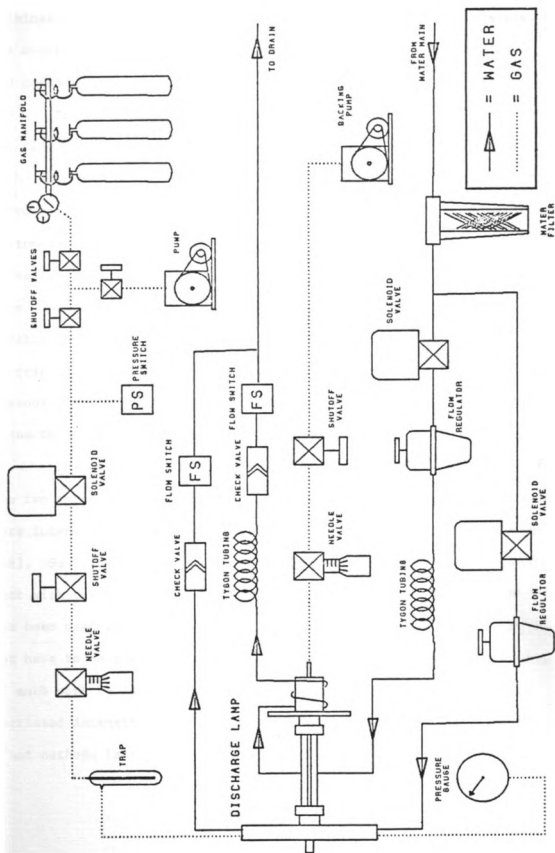


Figure II-8. Lamp Utilities.

eliminates the rather severe electrolysis of the water fittings that was observed when only a few feet of tubing were used between anode and cathode.

The lamp gas handling line is also shown in Figure II-8. The gas flow rate to the lamp and the amount of backpumping are controlled with micrometer-handled needle valves. A trap filled with molecular sieve can be cooled to liquid nitrogen temperature for purifying the helium used. This is found to reduce the intensity of the atomic emission lines superimposed on the continuum, but not to eliminate them. Since the reduced line emission is not accompanied by any measurable increase in the intensity of the continuum, and since the trap is fragile and inconvenient, it is often not used. The pressure switch, solenoid valves and water flow switches shown in Figure II-8 are all connected to the interlock system described in Chapter III.

The many-lined hydrogen spectrum can be produced in either of the two types of lamps described above, although an order of magnitude more intensity can be obtained by using a hot-filament cathode design [58]. So far, relatively little work has been done on the MSU instrument with the hydrogen source, and the Hinterregger lamp described here has been used for this purpose. This has the advantage that lamps do not have to be changed and realigned when changing sources. However, if much work is done in the future in the hydrogen lamp region, the increased intensity obtainable would certainly make construction of a hot cathode lamp for the purpose worthwhile.

3. Power Supplies

The power supply circuit for the hydrogen lamp (Figure II-9) is very simple, consisting of a 20 KV, 1 A, unregulated DC power supply [59], the lamp, and a ballast resistor of a few thousand ohms. The voltage supplied by the DC supply is far in excess of that necessary to power the hydrogen lamp, but it is also used to supply the power

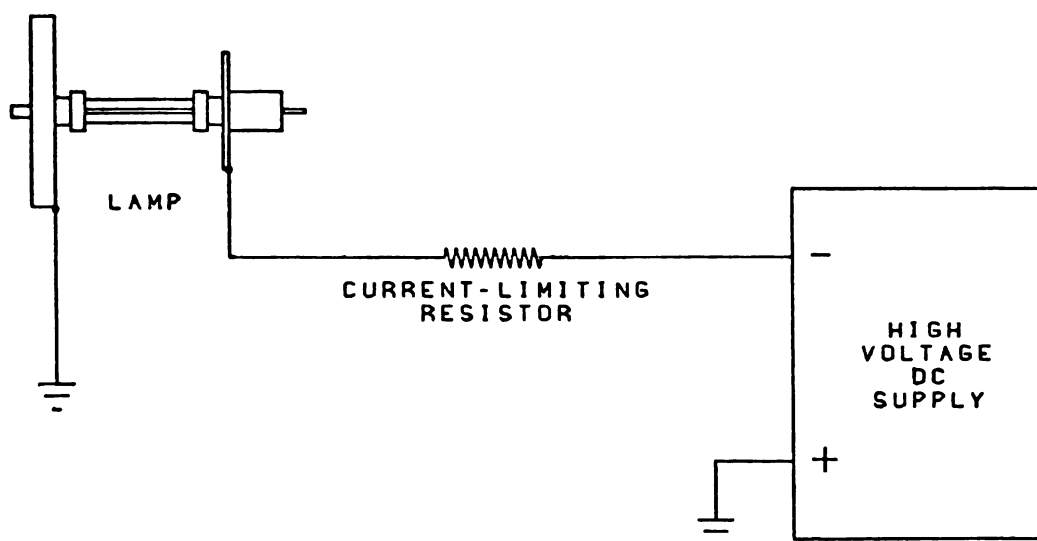


FIGURE II-9. DC Power Supply for the Hydrogen Lamp.

for the rare gas lamps, which require higher voltages. Typically, the voltage drop across the hydrogen lamp is only 1000 volts, with a 25 cm long x 4 mm id discharge tube at 2 torr and 300 mA.

The rare gas lamps must be pulsed, and so their power supplies are more complex than that for the hydrogen lamp. A variety of circuits have been used; one of the earliest is the simple spark gap circuit shown in Figure II-10. The DC supply charges the capacitor

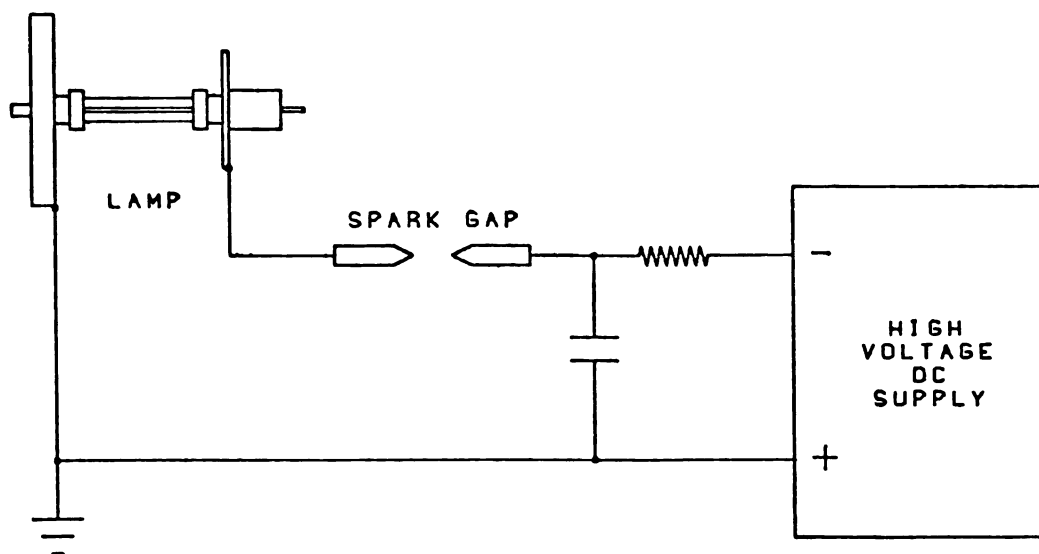


FIGURE II-10. Spark Gap Power Supply for the Rare Gas Lamps.

through the current limiting resistor until the breakdown potential of the gap (higher than that of the lamp) is reached. The capacitor then discharges through the lamp until the voltage left on it is less than the maintenance voltage of the spark gap and lamp, at which point the discharge ceases, the afterglow commences, and the cycle begins over again. The spark gap circuit is the simplest and cheapest circuit which can be used to produce the rare gas continua effectively, and these are its only advantages. Such circuits are extremely noisy (both electrically and acoustically), and have a limited repetition rate, which in turn limits the lamp intensity. Even "stabilized" spark gaps have somewhat unpredictable firing rates, resulting in about 4% random fluctuation in the average intensity of the lamp [60]. The spark gap electrodes also need to be adjusted and replaced rather frequently.

A power supply circuit first described by Huffman, et al. [60] overcomes many of these difficulties. In it, the spark gap is replaced with a thyatron circuit which can be triggered at a much more reproducible rate so that short-term fluctuations in lamp intensity are reduced to 1-2%. The noise and inconvenience of the spark gap are eliminated, and faster repetition rates (up to 20 KHz) become possible. Huffman, et al. report that under typical operating conditions (5000 Hz), the thyatron circuit provides four times the photon intensity of a simple spark gap circuit [60]. Typically, such a system would occupy two full-sized relay racks, one for the thyatron circuit and one for the high voltage DC supply. The cost of a commercially produced system would be in the range of about \$4000 for the DC supply and \$8000 - \$10,000 for the thyatron trigger circuit, although presumably either could be "home-made" for considerably less money.

One disadvantage with both the thyatron and spark gap circuits is that the discharge, once initiated, continues until the capacitor is more or less completely discharged. More recently, power supplies have been built which employ vacuum (rather than gas-filled) tubes in the final stage. These have the advantage that they can be turned off as fast as they can be turned on. The sharper pulse termination allows afterglow production to get underway sooner. Optimum repetition rates with such supplies tend to be in the range of 50-100 KHz, and much higher intensities can be obtained as a result.

The power supply used at Argonne National Laboratory is evidently the most powerful currently in use with rare gas lamps, but it has been described only briefly [33]. It is capable of generating 10 KV pulses

as short as 150 nsec (FWHM, 60 nsec rise time) at rates up to 100 KHz. Although the tubes in the final stage of the supply are capable of providing peak pulse currents of 120 A, in practice it is found that the lamp draws peak currents of only 20-30 A. No direct comparison has been made with the previously used thyatron circuit, but the ANL supply is estimated to provide at least an order of magnitude more intensity. It is only slightly more bulky than the thyatron supply, and costs about twice as much.

The power supply used for most of the work reported in this thesis is a Cober model 605P high power pulse generator [61]. It is much less powerful than the ANL power supply, but also smaller (15 x 20 x 22 inches) and less expensive (\$5500). It is capable of delivering 2200 volt pulses with a 1.5% duty factor at repetition rates up to 1 MHz. The maximum current output is 11 A, and the rise time is usually adjusted to match the fall time of 35 nsec. Dr. W. A. Chupka kindly provided an opportunity to compare the Cober directly with the ANL power supply. Both power supplies were used in turn to power a 10 cm helium lamp. Gas pressure, pulse width, and repetition rate were adjusted to find the maximum intensity that each supply was capable of providing; the results are indicated in Table II-2. The relatively low voltage available from the Cober prevents pressures higher than about 80 torr from being used, and limits the length of the discharge tube to about 15 cm (which can be used only at lower pressures). The relatively low current available from the Cober also limits the intensity obtained at a given length and pressure. The two effects combined to produce an ANL : Cober intensity ratio of 3.4 : 1 with a 10 cm helium lamp. The ANL supply has been used with 50

Table II-2. Comparison of the Cober and ANL Power Supplies.

Optimum Conditions (10 cm lamp)	Power Supply	
	Cober	ANL
Peak Voltage, KV	2.2	8-9
Peak Current, A	12	30
Pulse Width (FWHM), μ sec	0.1	0.15
Repetition Rate, KHz	130	100
Lamp pressure, torr	70	130
Relative Intensity	1.00	3.4

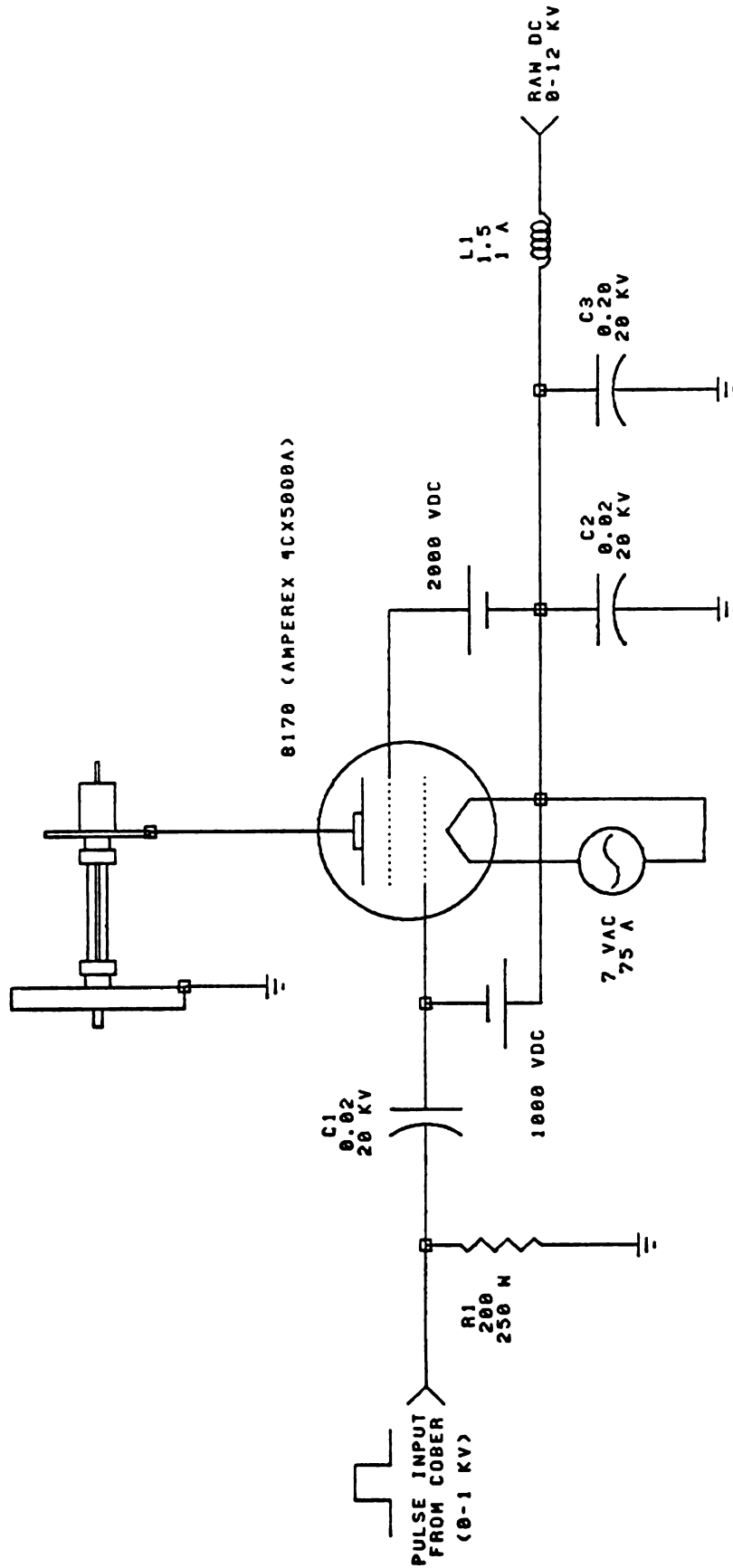
cm discharge tubes, and the photon intensity was found to increase linearly with the length of the discharge tube [33]. Therefore, the Cober is estimated to provide only about 1/17 of the intensity that can be obtained with the ANL power supply.

Fortunately, this is sufficient intensity for a large number of experiments. The Cober has been used intensively for three years, and has been found to be very dependable. It provides a very stable lamp output, and as long as a complete electrostatic shield is maintained around the lamp and the leads from the Cober to the lamp, no difficulty is experienced with radiated electrical noise. It's only drawback is that it provides significantly less than "state-of-the-art" intensity. On the other hand, the ANL supply is clearly (though intentionally) overdesigned. A power supply capable of generating 20-30

ampere pulses at 10 KV should be able to equal the performance of the ANL supply, but at a much lower cost.

Construction of a power supply designed to meet these specifications has been completed very recently in this laboratory [62]. Initial testing shows that some improvement is needed in parts of the control circuitry for the supply, but that the power portions of the circuit, shown in Figure II-11, do meet the design specifications. The circuit is a simple one-tube amplifier, driven by the Cober pulse generator. Capacitive coupling allows the Cober and lamp anode to remain at ground potential while the cathode of the tube is charged up to as much as -10 KV. Preliminary tests with a 25 cm long, 4 mm id discharge tube show maximum photon intensity is obtained with a helium pressure of about 140 torr, a 0.25 μ sec (FWHM) pulse width, and a repetition rate of about 60 KHz. The voltage pulses applied to the lamp are approximately trapezoidal with 70 nsec rise and fall times at 10 KV. Sharper edges can be obtained, but a slight impedance mismatch between the Cober and the load resistor presently used causes ringing if the rise time is much shorter than this. It is hoped that a different load resistor will allow a closer approach to proper termination so that sharper pulses and faster repetition rates can be used, which should yield even further increases in intensity.

Direct comparison of this power supply with the Cober is unfortunately rather difficult. The Cober has shown signs of aging in recent weeks, and is currently capable of lighting a discharge lamp only at reduced power, although it has sufficient capacity to drive the new power supply. Replacement parts for some of the components



NOTE: RESISTANCE IN OHMS, CAPACITANCE IN MICROFARADS, INDUCTANCE IN MILLIHENRIES
 R1 MUST BE NON-INDUCTIVE (E.G., DALE NH250-200).
 C1, C2 MUST ALSO BE LOW INDUCTANCE (E.G., PLASTICS CAPACITORS HC200-203).

Figure II-11. Vacuum Tube Power Supply for the Rare Gas Lamps.

suspected of causing the problem will not arrive in time for a direct comparison between the two power supplies to be included in this thesis. An estimate based on intensity measurements made several months ago is that the new power supply provides about seven times the intensity that the Cober did at its best. Use of longer discharge tubes should make an additional factor of two or three possible. A more exact comparison and further tests of the new power supply will have to be reported elsewhere once completed.

C. Photon Optics

Any optical elements of the instrument which are intended for use below $1050 \text{ \AA}$ must be reflective components, since no known material transmits well enough to be used as lenses or prisms. Unfortunately, even osmium - which seems to hold the current reflectivity record - only reflects about 30% of the radiation that hits it (normal incidence, $600 \text{ \AA}$ [63]). More commonly used coatings such as platinum, gold and aluminum are even worse [7], and thus the number of reflections must be kept to an absolute minimum. This can be accomplished by mounting the lamp and ion source directly on the entrance and exit slits of the monochromator and using a concave diffraction grating, which both diffracts and focuses the radiation with one reflection. The monochromator will therefore consist of one concave grating and two slits. Many different monochromator designs using these three elements exist [7], but most require that one of the slits be moved as the wavelength is scanned in order to maintain proper focus. The bulk of the equipment that must be located at the entrance and exit slits of a PIMS monochromator

virtually requires fixed slit positions, however, and only two types of monochromators have found wide use in the field.

One of these is the Seya-Namioka monochromator [64-66], in which the entrance and exit slits subtend an angle of $70^{\circ}30'$ at the grating. The wavelength is scanned by a simple rotation of the grating. The major advantages of the Seya-Namioka design are its simplicity and the fact that the entrance and exit slits are far apart, leaving plenty of room for coupling to the light source and the rest of the mass spectrometer. However, although reflectivity is better at this angle than at normal incidence, only relatively small gratings can be used, and optimum, stigmatic focusing is compromised for the simplicity of design. The image of the entrance slit at the exit slit is both enlarged vertically and curved, so that both resolution and intensity suffer.

These disadvantages are largely overcome by the near-normal incidence designs, in which the slits subtend a very small angle (ideally, but impractically, 0°), dramatically reducing astigmatism. The wavelength scanning mechanism in most near-normal incidence designs simultaneously rotates and translates the grating in such a way as to maintain correct focus. These differences, combined with the use of larger gratings, yield both increased resolution and intensity. (Resolution may be increased by a factor of five for instruments of comparable focal length and dispersion, and for comparable focal length and resolution, the intensity gain may be as much as a factor of ten [5].) Aside from the more complicated scanning mechanism, the main disadvantage of the near-normal incidence design is the close proximity of the entrance

and exit slits.

The McPherson 225 [67], a one-meter near-normal incidence monochromator, is probably the most widely used monochromator in photoionization mass spectrometry, and is also used in the MSU instrument. Our monochromator was modified by the manufacturer to allow the exit slit assembly to protrude four inches into the vacuum chamber of the mass spectrometer. The entrance and exit slits, subtending an angle of 15° at the grating, are separated by about 27 cm, which is sufficient, though not ample, room for both the lamp and the mass spectrometer chambers. Any one of four positions on a round slit table may be selected at the entrance slit position without breaking vacuum. Three positions hold fixed-width slits, and the fourth is a blank position used for sealing the entrance slit hole. The exit slit table is linear and one of four positions may be chosen with a rack-and-pinion gearing system which is in turn connected to a rotary vacuum feedthru on the mass spectrometer chamber via a chain-and-sprocket mechanism, so that any one of four exit slits may also be chosen without breaking vacuum. A small flap valve just behind the exit slit table serves both as an optical shutter and as a valve to seal the exit slit side of the monochromator. The monochromator can thus be kept evacuated even when the lamp is removed and the mass spectrometer chambers are at atmospheric pressure. The monochromator also features an extra-long wavelength drive screw shaft, which protrudes through a rotary seal in the side of the monochromator so that a stepping motor can be directly coupled to it.

Although other materials have better reflectivities below

1000 Å, aluminum is the best of the commonly available coatings above that limit [7]. The only grating used in this instrument to date is an aluminum coated grating with magnesium fluoride overcoating to prevent oxidation. It is ruled with 1200 lines/mm, blazed at 1200 Å. When equipped with this grating, the monochromator has a reciprocal linear dispersion of 8.3 Å/mm, and a range of 0-3000 Å. The best resolution obtainable with the smallest slits currently mounted (10 microns) in first order is approximately 0.12 Å FWHM, as illustrated in Figure II-12 for the atomic nitrogen emission triplet at 1134 Å.

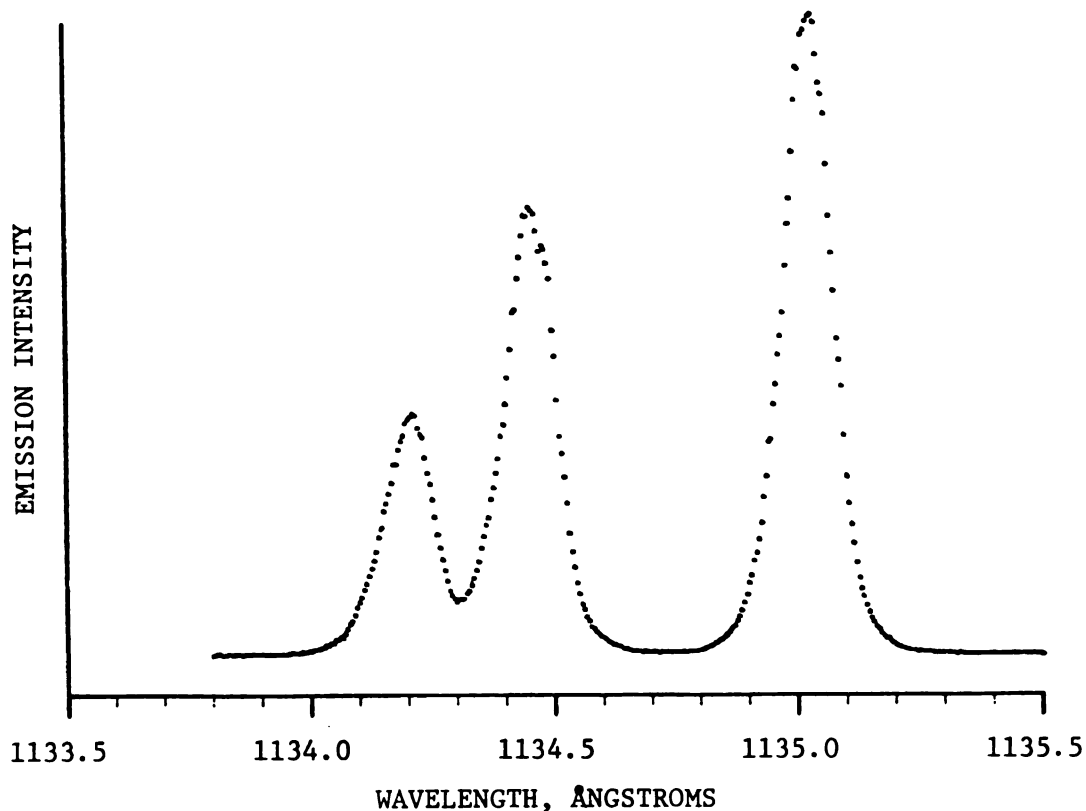


FIGURE II-12. Atomic Nitrogen Emission Lines at 1134 Å. These are three of the nitrogen impurity lines in the emission from the helium lamp.

D. Photon Transducers

There are four basic types of photon transducers which have been found suitable for use in the regions of the vacuum ultraviolet that are accessible to PIMS. Some of their important characteristics are summarized in Table II-3.

Solar blind photomultiplier tubes (PMTs) can be used down to $\approx 1050 \text{ \AA}$, if equipped with lithium fluoride windows. Such multipliers have high work function photocathode materials which are insensitive to visible wavelengths. They also exhibit much lower dark current than typical UV/visible photomultipliers, since thermal emission from the photocathode is very low. The low dark noise, combined with the high gain of the electron multiplier structure in the tube, means that such a transducer can be used to measure very low photon intensities, especially if photon counting techniques are used. The detection limit in Table II-3 was calculated assuming a quantum efficiency of 0.10, a gain of 10^6 , and DC current measurement techniques, with the dark current setting the detection limit. The sensitivity is entirely adequate for even the weakest photon intensities expected. The lithium fluoride window is subject to gradual deterioration when exposed to high energy photons, however, and the photomultiplier tube will show a gradual loss of sensitivity with time. The only other major disadvantage of this type of transducer is that it cannot be used at wavelengths below the lithium fluoride cutoff.

If, however, the window is removed entirely, the electron multiplier structure itself can be used as a VUV detector below about $1400 \text{ \AA}$.

TABLE II-3. Comparison of Common Vacuum Ultraviolet Photon Transducers

Characteristic	Transducer			
	Solar Blind ^a PMT	Electron Multiplier	Photodiode	Sodium ^b Salycilate
Dark Current (A)	3x10 ⁻¹²	≈10 ⁻¹²	0	3x10 ⁻¹⁰
Detection Limit (Photons/sec)	200	60	6x10 ⁷	10 ⁴
Wavelength Range (Å)	>1050	200-1400	200-1400	>200
Sensitivity vs. Wavelength	varies	varies	varies	constant
Rate of Aging	slow	medium ^c	slowest	fastest ^c

^aEMR 510G-08-13 with CsI photocathode and LiF window [68]

^bRCA 8850, K-Cs-Sb photocathode, cooled to ≈-20°C

^cDepends on cleanliness of vacuum and corrosiveness of sample

since photons in this region have more than enough energy to cause ejection of electrons from the metal surface of the first dynode. The ejected electrons are amplified by the remaining dynodes exactly as electrons from a photocathode would be. A wide variety of electron multiplier structures are commercially available for use in such fashion. Most common dynode materials have an even higher work function than the photocathode of the solar blind PMT, and thus a "bare" electron multiplier (or EMT) has an even lower dark current. The dynodes, which are all exposed to whatever ambient atmosphere exists in the vacuum chamber, will eventually acquire a thin coating of pump oil, and may be poisoned even sooner by reaction with certain sample vapors. This causes a loss in gain of the entire EMT, so that the sensitivity of the electron multiplier transducer will decrease at a rate which depends on the cleanliness of the vacuum and the reactivity of the dynode material to the sample being studied.

The simplest of the four transducers is the photodiode, consisting of a plate of clean, polished metal and an electrode which collects the photoelectrons emitted from the plate. It is essentially an electron multiplier with no stages of gain; because it has fewer surfaces to become poisoned or oil-coated, its sensitivity is much more stable than that of the electron multiplier. In addition, the photodiode can be made of a metal which is more resistant to chemical attack than commonly used electron multiplier dynode materials. Its major disadvantage is its lack of sensitivity. The detection limit shown for this transducer in Table II-3 is calculated assuming a photoelectron yield of 0.10 and a minimum detectable current of 10^{-12} amperes,

although use of a vibrating reed electrometer could reduce this limit by two orders of magnitude. Considering that a very intense photon source may be expected to provide 10^{10} photons $\text{sec}^{-1} \text{ \AA}^{-1}$ at the peak of the helium continuum, the sensitivity of this transducer is seen to be marginal for studies with narrow slits in the weaker portions of the continuum.

All three of the transducers discussed so far share one disadvantage: their photoelectron yield or quantum yield varies with the wavelength of the incident radiation, often quite dramatically. For instance, the photoelectric yield of copper-beryllium, one of the most commonly used dynode materials, peaks at about 700 $\text{\AA}$ and has lost 75% of its sensitivity by about 1000 $\text{\AA}$ [70]. The fourth type of transducer is commonly used to avoid, or at least correct for, this problem. It consists basically of a thin layer of sodium salicylate deposited on the outer surface of a typical UV/visible PMT. VUV photons striking the sodium salicylate layer cause it to fluoresce (the peak of the emission spectrum is about 4200 $\text{\AA}$) and the fluorescent emission is measured with the photomultiplier. The advantage of this transducer is that the fluorescent yield of sodium salicylate is at least approximately independent of excitation wavelength over a wide range [7]. Freshly prepared coatings have a constant efficiency from 200 $\text{\AA}$ to 1200 $\text{\AA}$, and exhibit a small rise above this wavelength. Unfortunately, sodium salicylate which is contaminated with pump oil or poisoned by reactions with any of a number of chemicals no longer has a constant response, but exhibits a drop in sensitivity between 1200 and 900 $\text{\AA}$. The ease with which a thin sodium salicylate layer can become

contaminated is one of the major disadvantages of this transducer, but a fresh layer of sodium salicylate can at least be used to calibrate the wavelength response of the other types of detectors discussed above. In addition to the ease with which it can be contaminated, the sodium salicylate transducer has two other disadvantages. The photomultiplier used typically has a much higher dark current than any of the other transducers mentioned, so that its overall sensitivity is not as high, although it is usually adequate. Also, the photomultiplier is sensitive to visible light, and care must be taken to prevent light leaks into the chamber.

The first photon transducer used with the MSU instrument was a simple photodiode. At the time it was used, however, the lamp intensity was so low that the photodiode could not be used to measure it. The photodiode was replaced with an electron multiplier and has never been tried again, although photon intensities are now at least three orders of magnitude greater than during the first attempts at lighting the lamp. We have generally used electron multiplier transducers since then. Currently in use is an EMR 510W-00-16 multiplier with 16 Cu-Be dynodes [68].

We have also used a sodium salicylate detector system occasionally, in wavelength regions where the response of the electron multiplier is very small, or in order to calibrate the electron multiplier. An RCA 8850 photomultiplier [69] in a thermoelectrically cooled housing [71] is optically coupled with a highly polished plexiglass rod to a quartz disc [72] upon which the sodium salicylate layer is deposited.

Figure II-13 shows the Hopfield continuum recorded with the

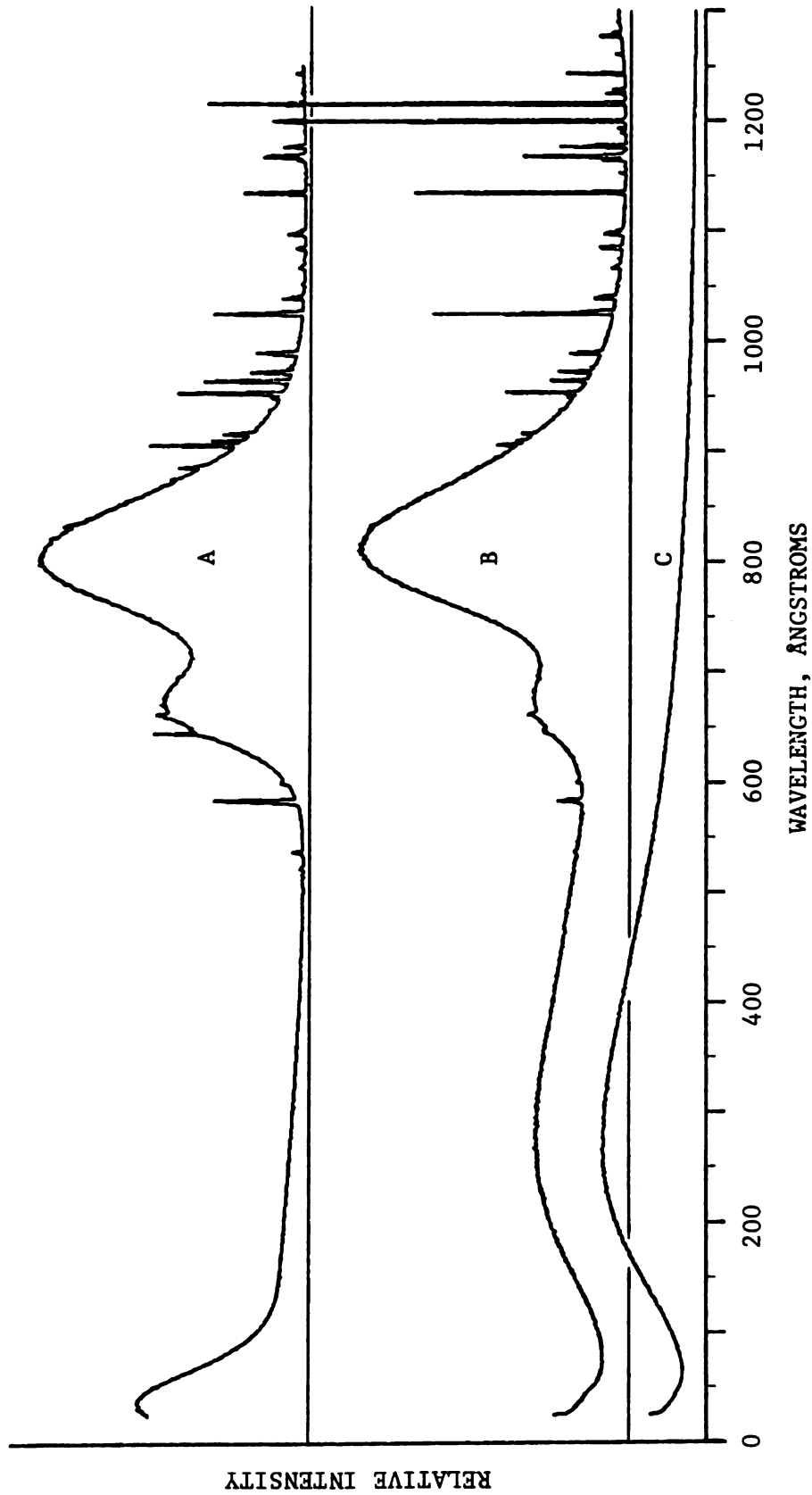


Figure II-13. The Helium Continuum and Scattered Light Recorded with Different Photon Transducers.
Curve A - copper-beryllium electron multiplier; curve B - sodium salicylate over an RCA 8850 photomultiplier; curve C - blank quartz filter over an RCA 8850.

electron multiplier (a) and sodium salycilate (b) transducers. The dropoff in sensitivity of the electron multiplier at wavelengths above 900 Å is demonstrated by the decreased intensity of both the continuum and the atomic lines in (a) relative to (b). Also of interest is the scattered light distribution, all of the light below 500 Å being attributed to scattered light since the source does not emit radiation shorter than about 550 Å. The helium lamp produces a great deal of near UV and visible radiation as well as the Hopfield continuum. The electron multiplier is very insensitive to radiation above 1500 Å, while the RCA 8850 is sensitive to wavelengths longer than about 2000 Å. It is this sensitivity to long-wavelength scattered light that causes the difference seen below 500 Å in Figure II-13. The fact that most of the radiation detected between 0 and 500 Å with the RCA multiplier is not vacuum ultraviolet radiation is supported by Figure II-13c, which shows the results obtained when a blank (instead of sodium salycilate coated) quartz disc is used as a 2000 Å high-pass filter in front of the RCA multiplier.

The sodium salycilate transducer detects so much scattered light when used with the Hopfield continuum that the effects of the scattered light are difficult to correct for (see Appendix A). This transducer is therefore used in this region only to calibrate the electron multiplier. The scattered light problem is much less bothersome when the hydrogen lamp is used, since it emits less visible and UV radiation, and since the many-lined spectrum doesn't begin until wavelengths greater than 800 Å, i.e., in a region where the scattered light distribution has fallen off considerably.

E. Ion Source

The most commonly used type of ion source in photoionization mass spectrometry [5] is simply a small box with holes for photon entrance and exit, and another hole for ion extraction. It incorporates a repeller plate to establish a field that pushes the ions towards the exit hole (see Figure II-14). The holes must be small enough that an adequate

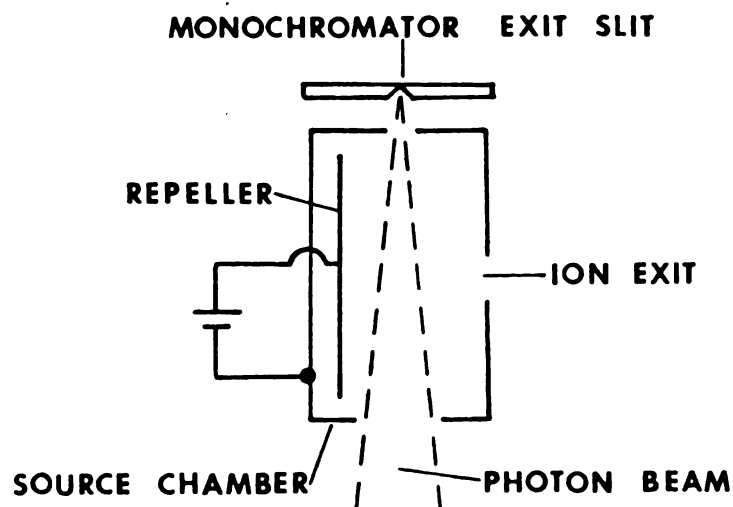


Figure II-14. A Simple Ion Source.

sample pressure can be maintained inside the source without placing an inordinate burden on the vacuum pumps which must keep the region outside the source at a lower pressure. If the photon exit hole is smaller than the cross section of the photon beam when it reaches the far wall, some provision must be made to collect the photoelectrons

emitted from the wall so that they are not accelerated into other regions of the instrument where they could potentially cause electron impact ionization.

The interpretation of appearance potentials is simplified if the sample gas can be cooled below room temperature to eliminate (or at least reduce) the effects caused by the internal energy spread of the molecules [10,12,73,74]. Several ion sources designed to accomplish this have been described in the literature [33,43,73]. All basically work by cooling the walls of the ion source and allowing the sample vapor to come into thermal equilibrium by making several contacts with the walls before actually entering the ionization region. Although much valuable work has been done this way, the approach is limited by the fact that many molecules condense before reaching very low temperatures, and the amount of cooling that can be accomplished for them is fairly small.

Sample cooling can also be accomplished using a radically different approach [38,75]. The ion source box is eliminated and instead photoions are created in a high intensity "supersonic" molecular beam [76]. In such a beam, most of the rotational energy of the molecules has been converted into translational energy during expansion from the nozzle, and rotational temperatures as low as a few degrees Kelvin can be obtained, even with easily condensed molecules. The main disadvantage of this approach is that the supersonic beam source introduces much greater complexity and requires much greater pumping capacity. The increased resolution obtained by the elimination of thermal broadening can be well worth the effort, however. A striking example of this

is the photoionization efficiency curve for NO^+ (from NO) near threshold - compare Figure 1 of Reference [77], obtained at 150 °K, to Figure 3 of Reference [75], obtained with a nozzle beam source.

A supersonic nozzle beam source has been designed for the MSU instrument, but is still under construction. The ion source currently in use is described in more detail elsewhere [78], but is basically a simple, ambient temperature box-type source. Intended for use with low sample pressures, it has fairly large exit holes (the ion exit hole is 0.25 inches dia., the photon exit hole is 0.140" x 0.350"). The sample is admitted via an all-metal vacuum manifold, which incorporates a variable leak to control the flow rate and therefore the sample pressure. A separate port on the ion source is connected to a capacitance manometer [79] which is used to measure the sample pressure.

F. Ion Optics

Most photoionization mass spectrometers have used time-of-flight, magnetic sector or quadrupole mass analyzers [5]. The pulsed operation of the time-of-flight mass spectrometer is advantageous in coincidence studies, but is otherwise a liability, both because the rare gas lamps do not emit pulses of light that are short enough and because discontinuous production of ions is an inefficient use of experimental time.

Magnetic sector [80] and quadrupole [81] instruments both allow continuous production and detection of ions, and can attain very high ion collection and transmission efficiencies. The magnetic sector instruments require that the ions be accelerated by a potential of several kilovolts before entering the sector region, and usually the

ion source is operated at several kilovolts above ground. The high voltages and extra lenses required to accomplish the acceleration make the sector instruments more complicated and less convenient than the quadrupole instruments, which work best at ions energies less than 100 volts. Another difference is that, because of the lower ion energy, the transit time from source to detector is usually longer in quadrupole instruments, by perhaps an order of magnitude. "Slow" unimolecular decompositions (rate constant less than $10^5 - 10^6 \text{ sec}^{-1}$) therefore have a greater probability of occurring in the quadrupole instruments before the ions are analyzed. This is an advantage in reducing the amount of "kinetic shift" [11] in appearance potential measurements. Quadrupoles are also more compact than sector instruments, needing no long acceleration region or bulky magnet. They also have a linear mass scale, and make it possible to choose the optimum balance between mass resolution and sensitivity by the simple adjustment of a front panel control.

The mass analyzer in the MSU instrument is an Extranuclear 324-9 quadrupole mass filter [82], with 1.9 cm dia. x 22 cm long rods. It is powered by an Extranuclear 311-type power supply with either a model E or model 13 high-Q head. The electrostatic aperture lens system supplied with the filter, modified by removal of the electron impact source, is used to focus ions from the ion source into the quadrupole entrance aperture. The quadrupole (as well as the lenses and ion transducer, which are connected to the quadrupole) is mounted on a support which is easily moved several inches in a direction parallel to the quadrupole axis, as shown in Figures II-15 and II-16. The push bar can be clamped (with set screws not shown in the figures) so as to

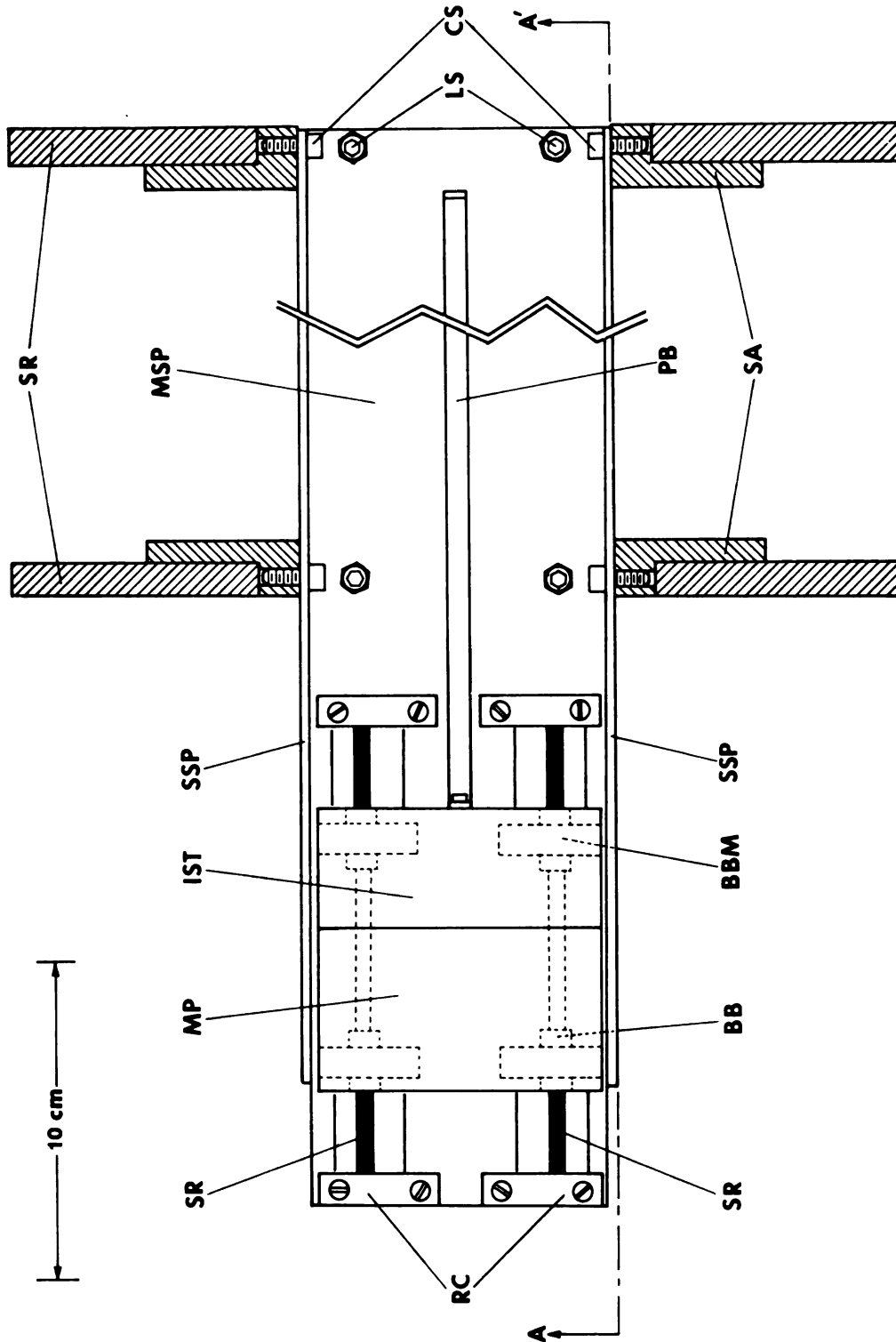


Figure II-15. Top View of Quadrupole Mount. The labels are defined in the caption to Figure II-16, which is a view along A-A'.

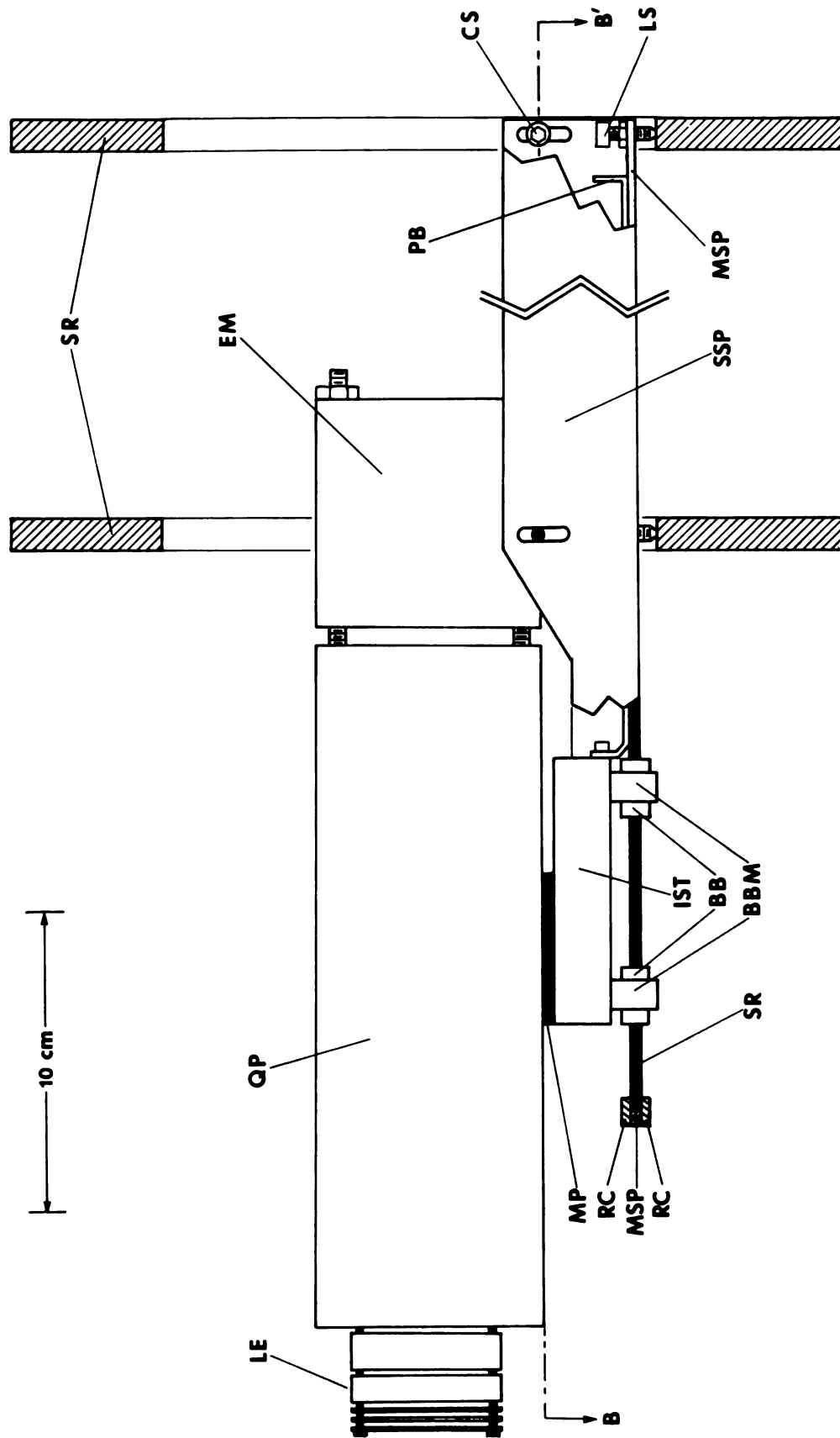


Figure II-16. Side View of Quadrupole Mount. BB-ball bushing; BBM-ball bushing mount; CS-clamping screw; EM-electron multiplier; IST-insulated support table; LE-lens; LS-locking screw; MP-mounting plate; MSP-main support plate; PB-push bar; QP-quadrupole; RC-rod clamp; SA-support adapter; SR-support ring; SR'-support rod; SSP-side support plate. Figure II-15 is a view along B-B'.

hold the quadrupole the appropriate distance from the photon beam for use with single or dual chamber ion sources of various depths.

This mass analyzer is reported to be capable of extremely high resolution [83], although this of course can be accomplished only at a cost of significantly reduced transmission. Fortunately, resolution of more than one AMU is virtually never needed in PIMS studies, so that relatively high ion transmission can be retained. Figure II-17 shows partial photoionization mass spectra of CF_2Cl^+ from CF_2Cl_2 , obtained at various resolutions. Single-AMU resolution may be realized for this ion at a cost of about 75% of the maximum transmission, and the importance of using only the minimum necessary mass resolution in order to avoid interference from other ions is graphically demonstrated.

G. Ion Transducers

An ion production rate of even 10^5 ions sec^{-1} is considered very high in PIMS. Since this corresponds to an ion current of only about 10^{-14} amperes, and since typical ion currents are two or three orders of magnitude less, it is absolutely necessary that the ion transducer employed incorporate some sort of gain mechanism. The most commonly used ion transducers are simply electron multipliers similar to those used as photon transducers. The probability of electron emission from the first dynode depends on the mass and composition of the ion, as well as the energy with which it strikes the dynode surface [84-86]. The accuracy with which an electron multiplier can be used to measure relative intensities of different ions is therefore limited, unless separate calibration measurements are made.

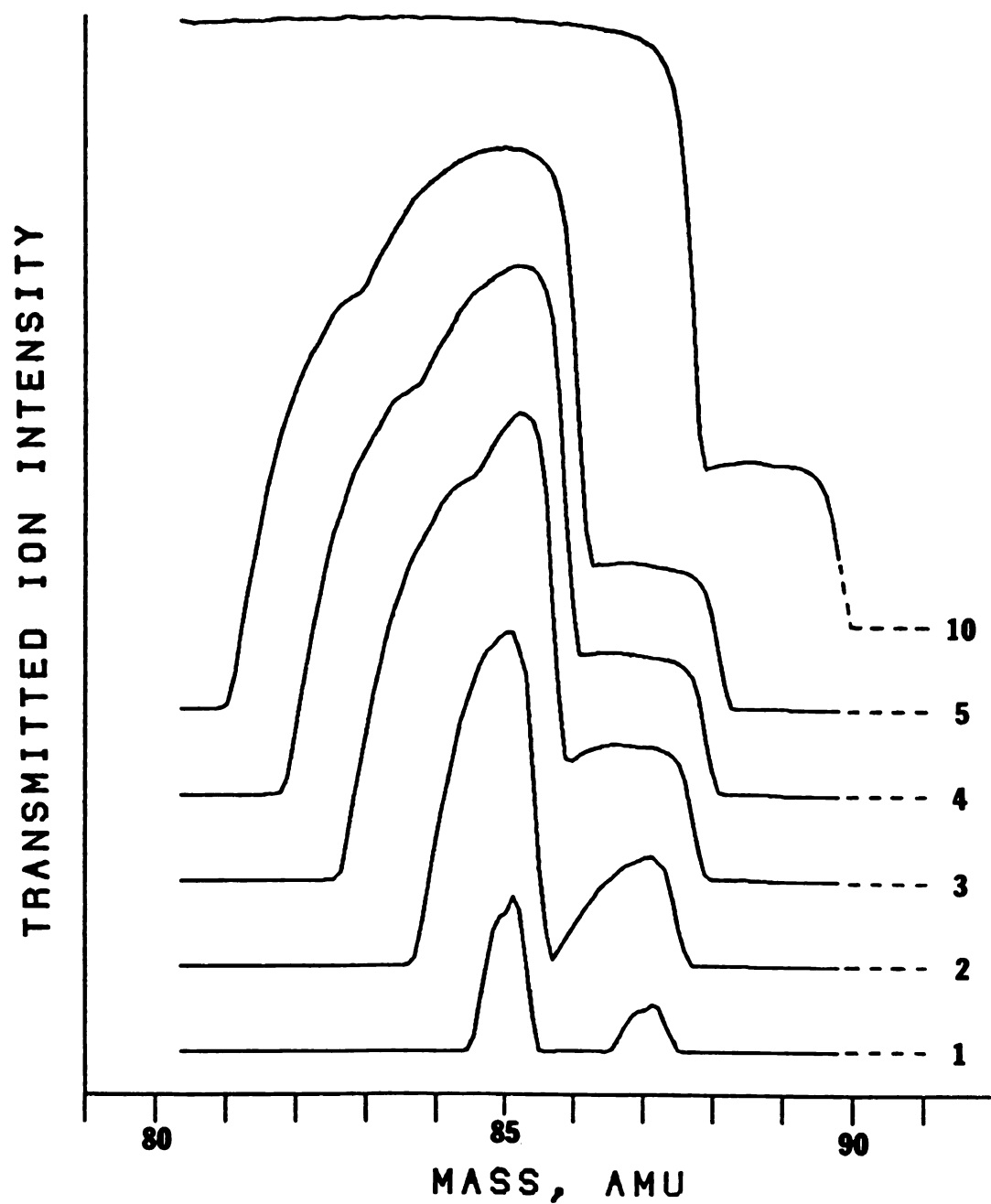


Figure II-17. Effect of Increasing Mass Resolution on Ion Transmission. These partial mass spectra show the mass region which includes the various isotopes of CF_2Cl^+ from CF_2Cl_2 . The spectra are plotted to the same scale, but are displaced for clarity. The number at the right of each curve gives the approximate mass filter resolution, in AMU.

Background count rates on the order of 0.1 to 1.0 counts per second are common, and very careful shielding can reduce these figures even further [87]. With such low background rates, ion intensities as low as 0.01 ions per second are measurable. The major problem with the use of electron multipliers is that their gain inevitably decreases with time, as mentioned in Section D.

Another type of transducer which has been used in PIMS is the Daly-type scintillation counter [88], shown in Figure II-18. The ions strike a metal (aluminum, stainless steel) converter electrode which is at a very high negative voltage (e.g., -25 KV). The very energetic impacts each release several electrons from the converter, which are then accelerated through the same high potential to a very thin metal film which is deposited over some scintillator material. The electrons pass through the film and each causes a light flash of perhaps several hundred photons within the scintillator. These light flashes are detected by an ordinary photomultiplier.

One of the main advantages of this system is that the photomultiplier is not exposed to the vacuum, and thus has more stable gain than the electron multiplier transducer. Also, the tremendous gain of the system (as much as a factor of 10^3 greater than the gain of the photomultiplier) results in much larger output pulses, so that less sensitive pulse counting electronics can be used, and pickup of environmental electrical noise is less of a problem. However, such a system is more complicated and difficult to use, especially because of the high voltages necessary. One such system [89] has attained a background count rate of 0.02 sec^{-1} , but reducing the count rate below 0.4 sec^{-1}

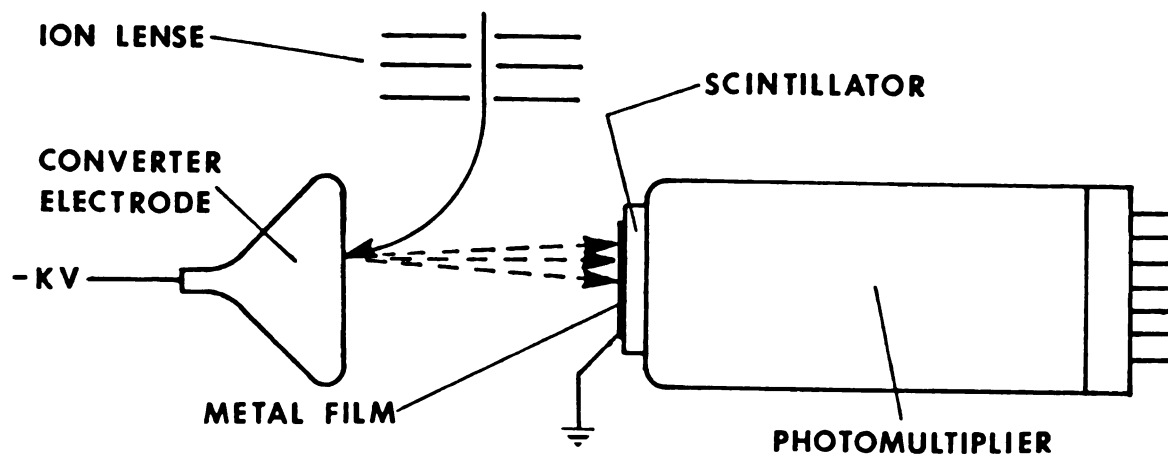


Figure II-18. Daly-Type Ion Transducer. Solid arrow: ion trajectory; dashed arrows: electron trajectories.

required repeated high temperature bakeout and ion bombardment of the converter electrode, as well as very careful optical, electrical and magnetic shielding of the photomultiplier and converter. This type of system has been used on only one photoionization mass spectrometer [90].

The electron multiplier used in the MSU instrument is a Johnston MM1, 20-stage focused mesh multiplier with Cu-Be dynodes [91]. This "ion multiplier" is mounted on the axis of the quadrupole, as shown in Figure II-15. Since the ion counting circuits used in the data acquisition system (see Chapter IV) are sensitive to frequencies well above the quadrupole frequency (2-3 MHz), the multiplier and its signal and high voltage leads must be shielded from the nearby quadrupole power leads. Evidently any rf radiation emanating from the quadrupole ion exit hole is relatively weak, and shielding mesh across this hole has not been required.

H. Vacuum System

Figure II-19 shows the location of the major components discussed above within the vacuum chambers of the instrument. In order to maintain the required pressure differential between the lamp and the monochromator (roughly $10^2 - 10^{-5}$ torr), two stages of differential pumping are used. The monochromator is equipped with a McPherson model 820 differential pumping assembly [67], the first stage of which is pumped with a 300 l/sec Roots blower pump. The second stage is pumped by a 300 l/sec, four-inch diffusion ejector pump, and the monochromator by an 1800 l/sec, six-inch diffusion pump. The sample and quadrupole chambers are also pumped by six-inch oil diffusion pumps, rated at 2400 l/sec and 1500 l/sec, respectively. A baffle around the quadrupole also provides some measure of differential pumping of the sample gas in these chambers.

The pumping speeds quoted above are for untrapped pumps. All four diffusion pumps are, however, equipped with freon-cooled baffles to reduce the amount of oil backstreaming into the chambers. Because of this, the actual pumping speed at the various chambers will be significantly lower. Nevertheless, these pumps are sufficient to maintain a pressure on the order of 10^{-5} torr in the monochromator with 80 torr of helium in the lamp (100 micron entrance slits). The pressure in the quadrupole region can be maintained below 2×10^{-5} torr for sample pressures below $1-2 \times 10^{-3}$ torr, even with the relatively open ion source currently in use.

Because of the noise and heat generated by the mechanical pumps used to back the four diffusion pumps, the mechanical pumps are located

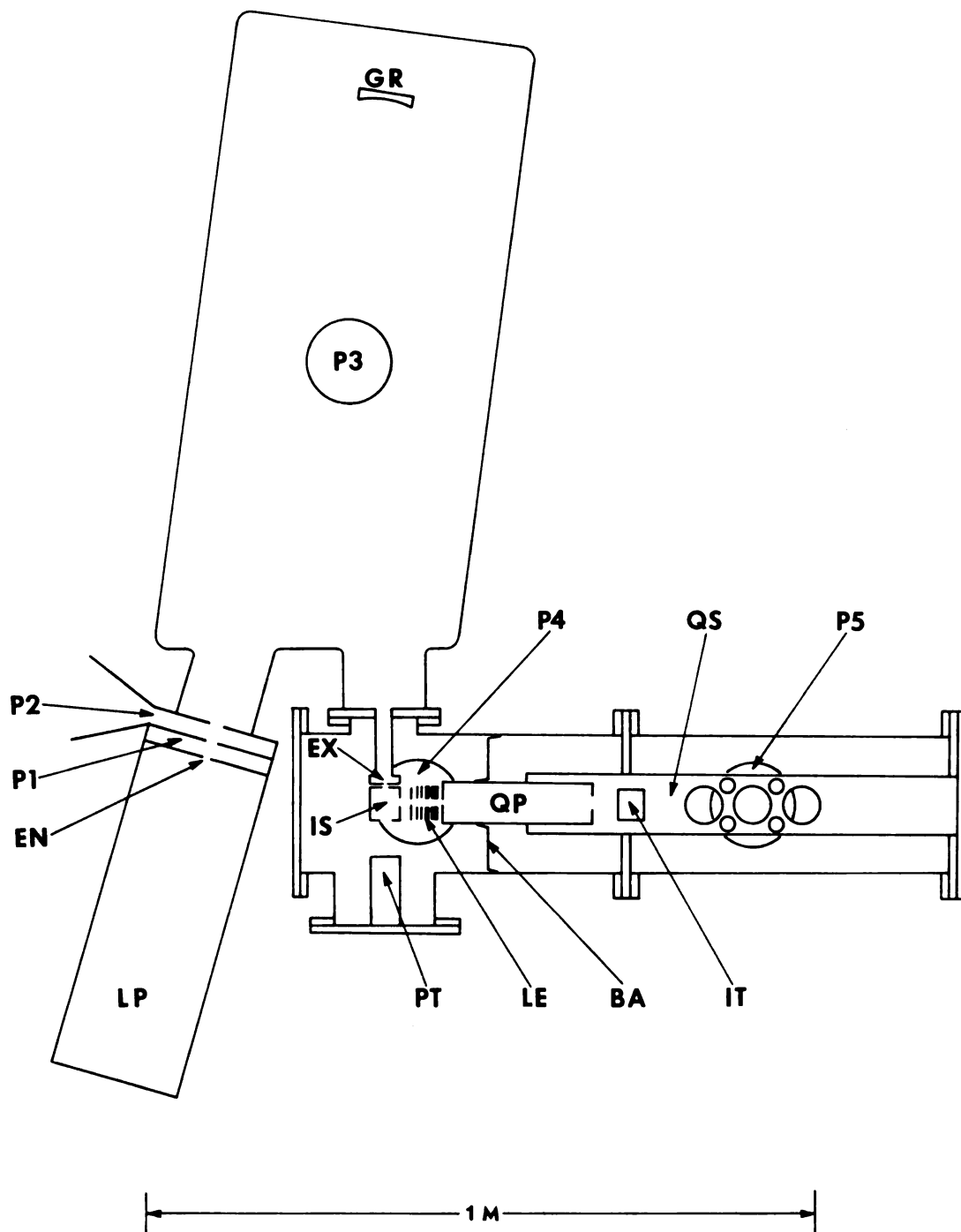


Figure II-19. Location of Components Within the Vacuum Chambers.
 BA-baffle; EN-entrance slit; EX-exit slit; GR-grating;
 IS-ion source; IT-ion transducer; LP-lamp; P1-first
 differential pumping port; P2-second differential
 pumping port; P3-monochromator pumping port; P4-sample
 chamber pumping port; P5-quadrupole chamber pumping
 port; PT-photon transducer; QP-quadrupole;
 QS-quadrupole support.

in a separate small room built into one corner of the laboratory, along with the Roots blower and the compressors for the refrigerated baffles. Cooling air enters the "pump room" via sound-absorbing ducts, so that the noise in the main part of the laboratory is significantly reduced. The exhaust of each of the mechanical pumps is connected to a condensate trap and to flexible ductwork which is in turn connected to a laboratory hood exhaust fan (on the roof of the building). These "minihoods" prevent the large amounts of oil vapor expelled by the pumps when pumping heavy gas loads from creating puddles of oil on the floor, and also prevent any toxic fumes from re-entering the laboratory.

The monochromator is mounted on a one inch thick aluminum plate which is supported by roller bearings so that it can be moved about a foot along the axis of the photon exit beam. Both the monochromator and booster diffusion pumps are attached in such a way that they move along with the monochromator, the connections to the backing pumps being via flexible metal hose. Each of the other two main vacuum chambers is mounted similarly, being movable along the ion beam axis, perpendicular to the photon exit beam. Figures II-20 and II-21 show the retracted and normal positions of the three main chambers. When in the retracted position, with the mass spectrometer chambers open to air, the monochromator can be kept evacuated, and with the gate valves closed, the sample and quadrupole diffusion pumps can remain on. The mobility of the chambers is extremely useful when making repairs or modifications to components in the vacuum chambers, which are fairly inaccessible in the normal position. The ability to keep the monochromator under vacuum and the other pumps hot greatly decreases the amount of time necessary

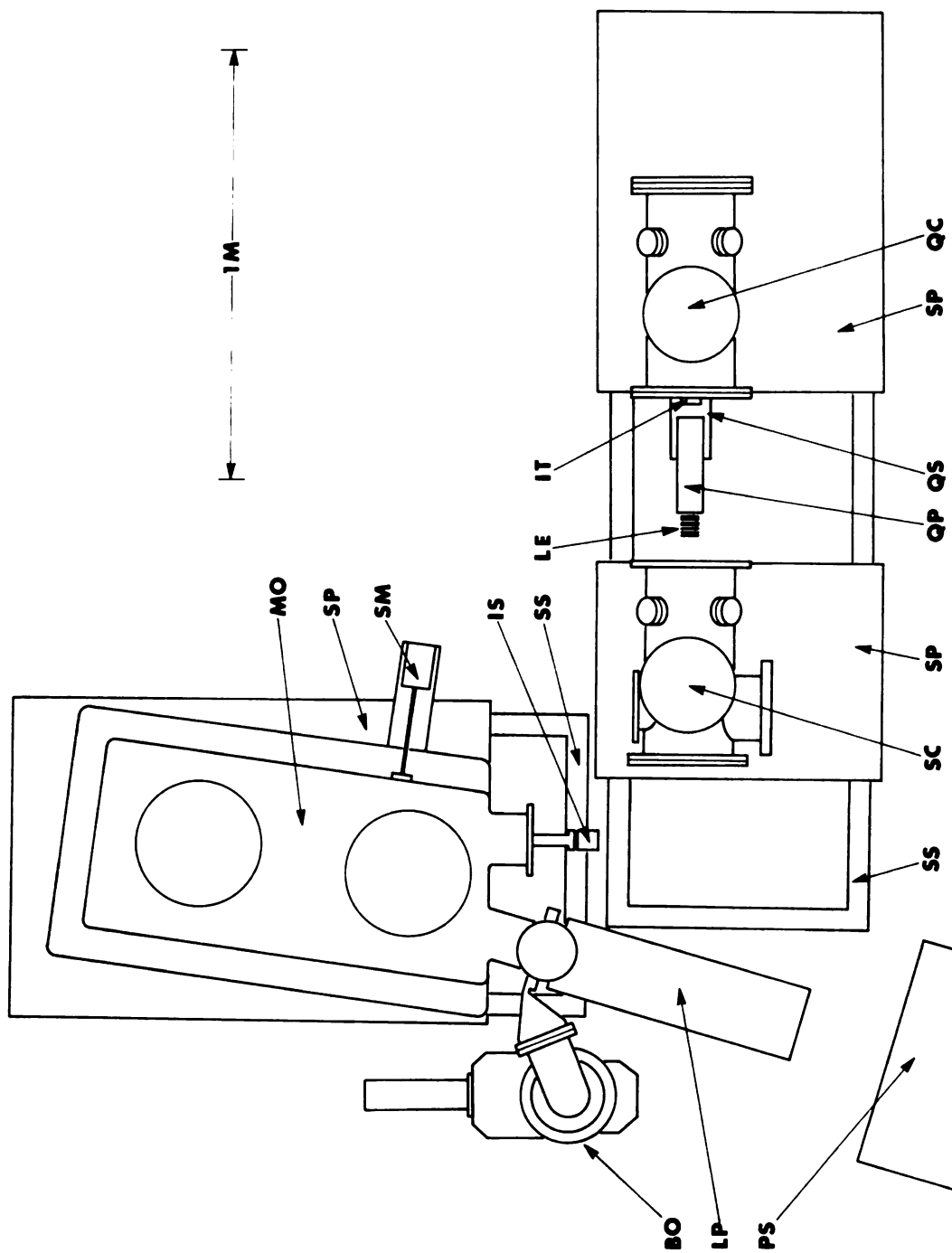


Figure II-20. Top View of Instrument, Retracted. BO-booster pump; IS-ion source; IT-ion transducer; LE-lens; LP-lamp; MO-monochromator; PS-lamp power supply; QC-quadrupole chamber; QS-quadrupole support; SC-sample chamber; SM-stepping motor; SP-support plate; SS-support stand.

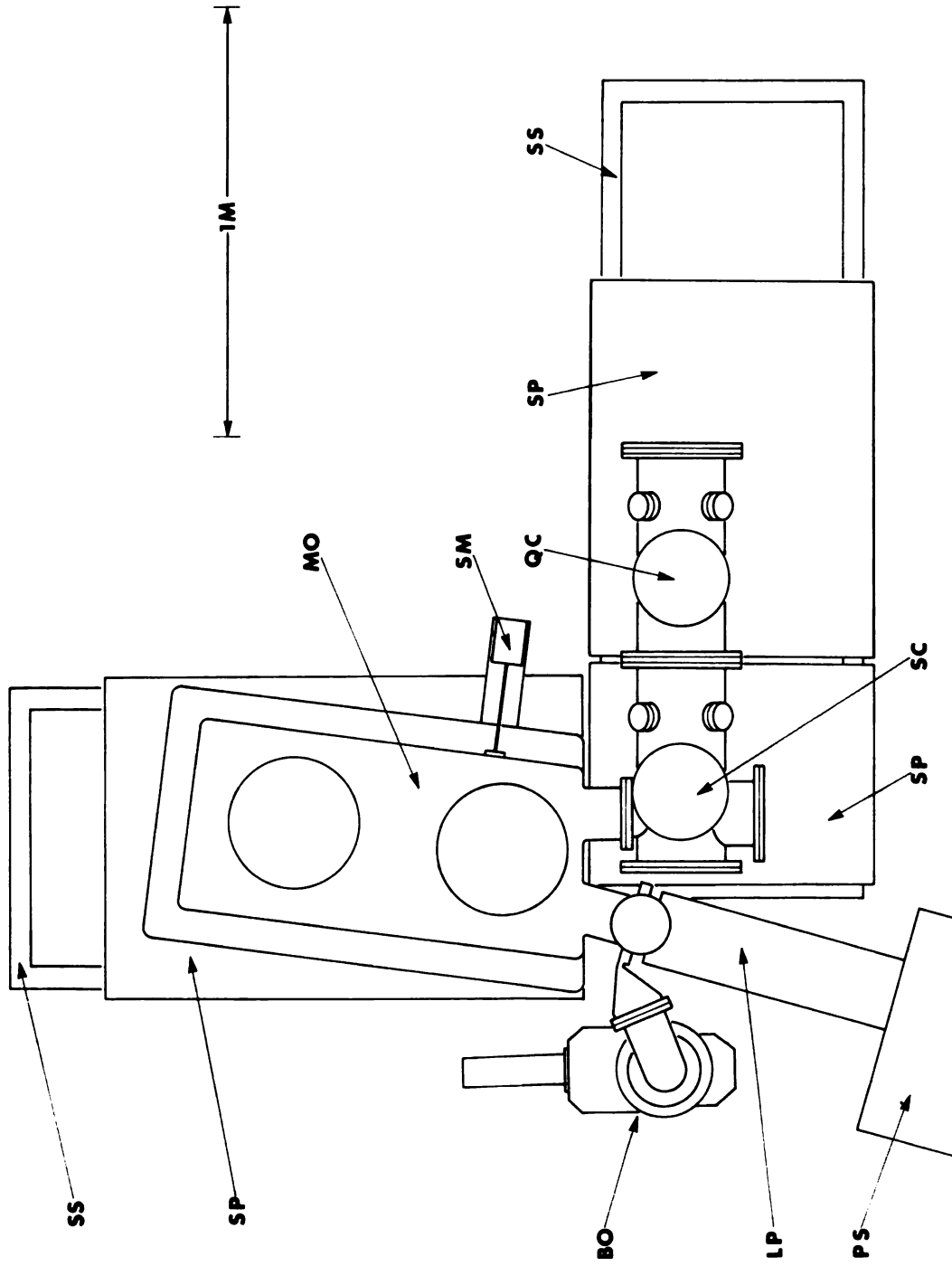


Figure II-21. Top View of Instrument, Normal Position. All labels are defined in the caption for Figure II-20.

to pump the system down to a working pressure after it has been "up to air", and also increases the life of the grating.

The vacuum plumbing is shown schematically in Figures II-22 through II-25. All valves are electropneumatically actuated (except the roughing and foreline valves of the monochromator and quadrupole systems) and can be closed automatically by the interlock system when necessary (see Chapter III). Note that water, compressed air and refrigerant lines have been omitted from these figures for clarity.

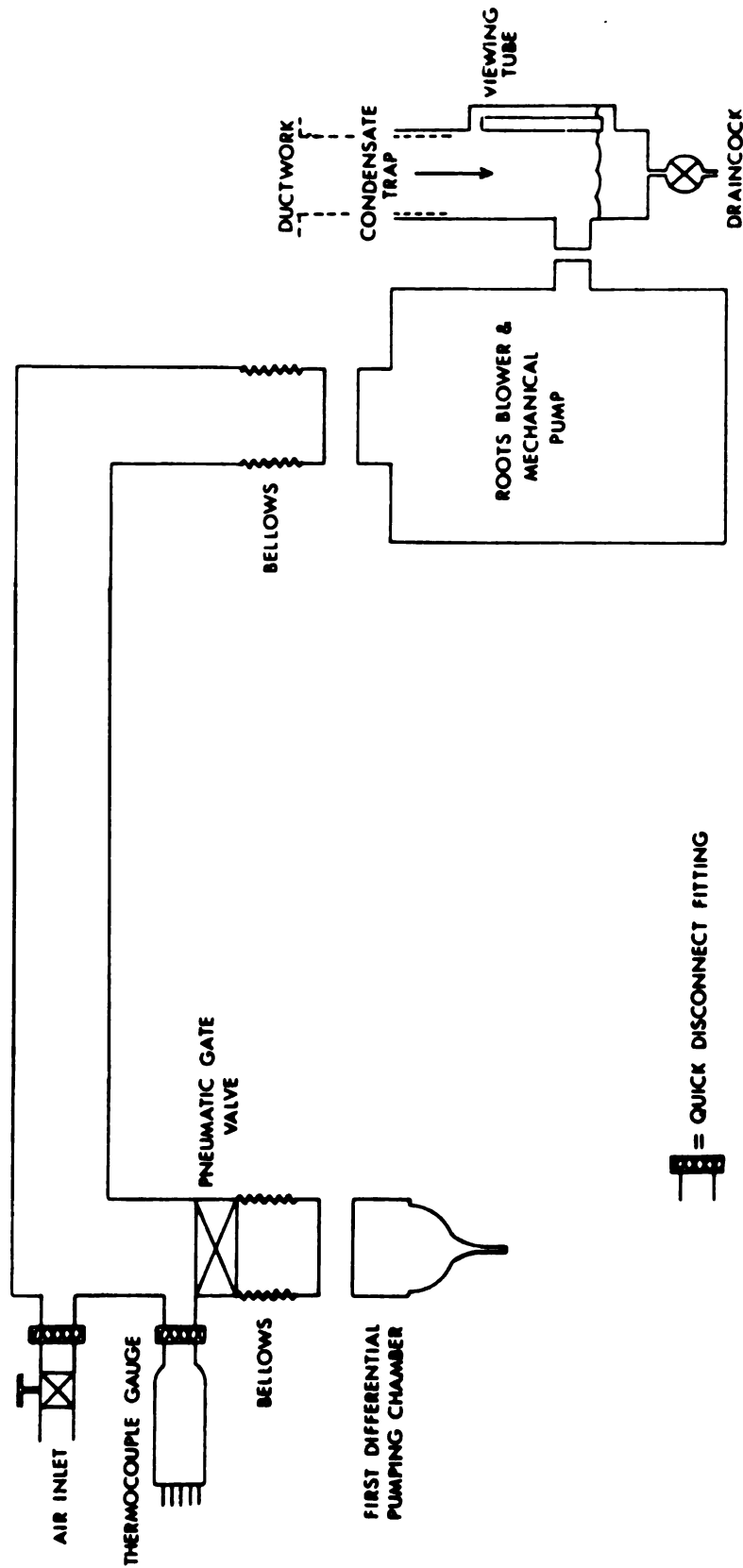


Figure II-22. First Differential Pumping System.

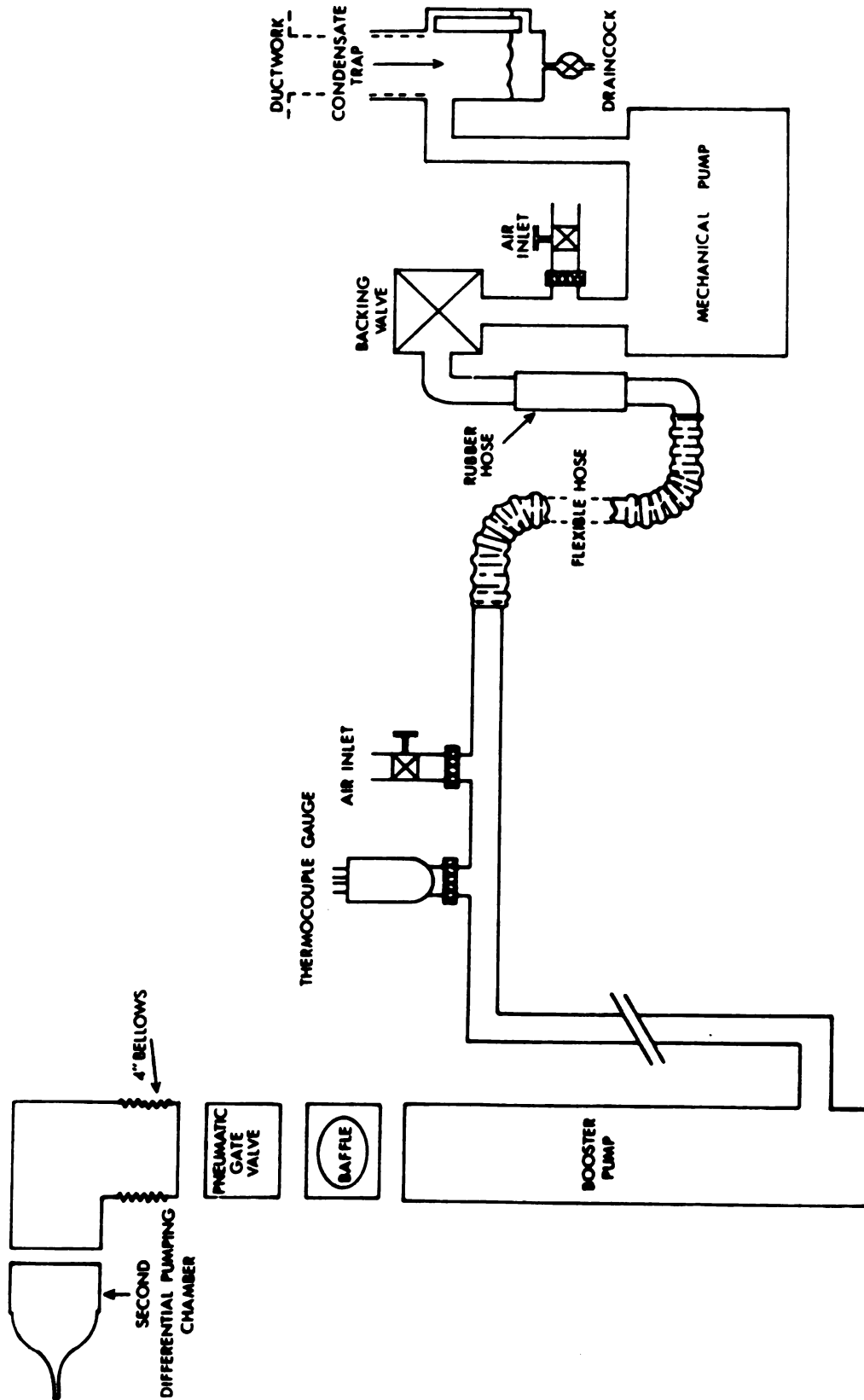


Figure II-23. Second Differential Pumping System.

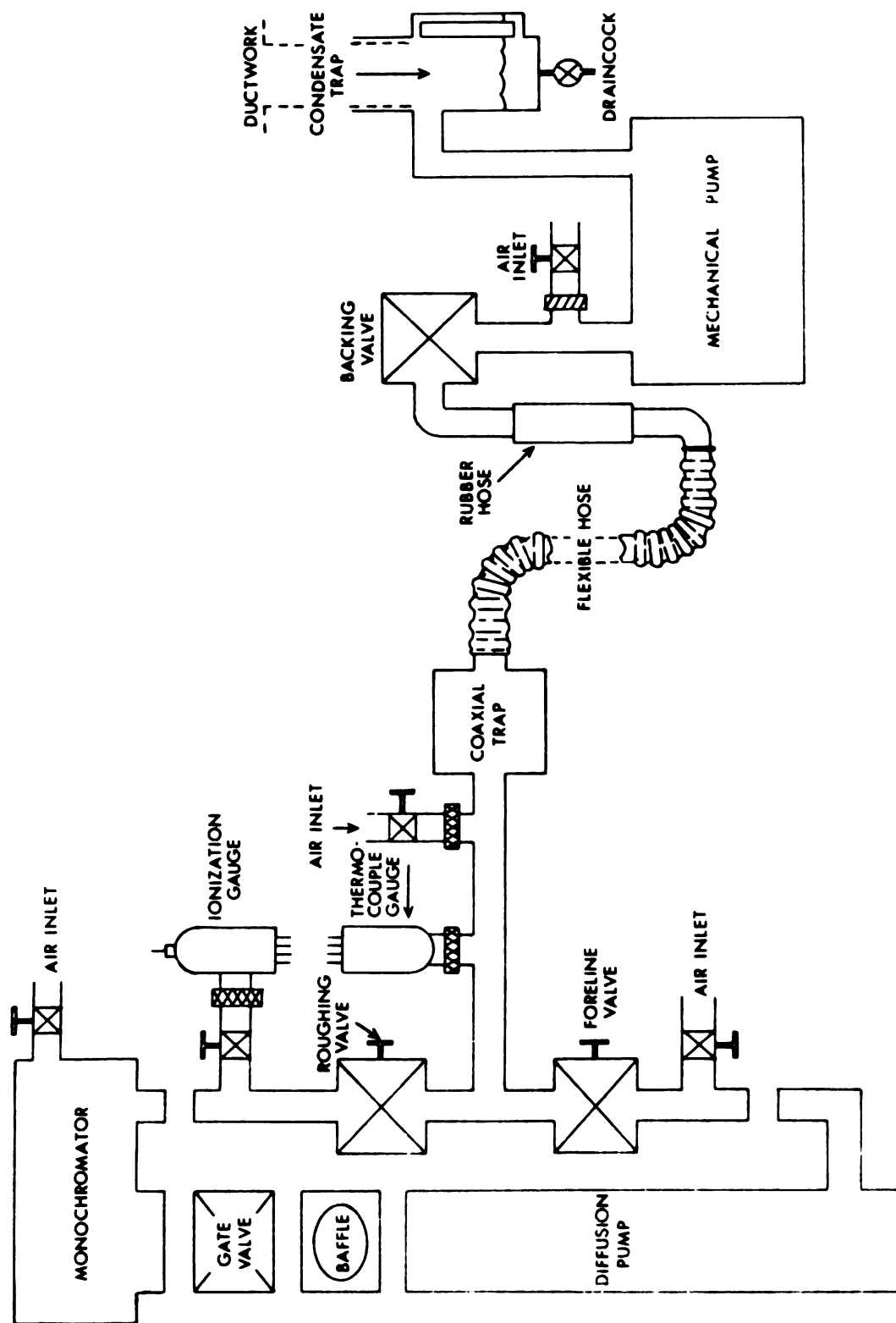


Figure II-24. Monochromator Pumping System.

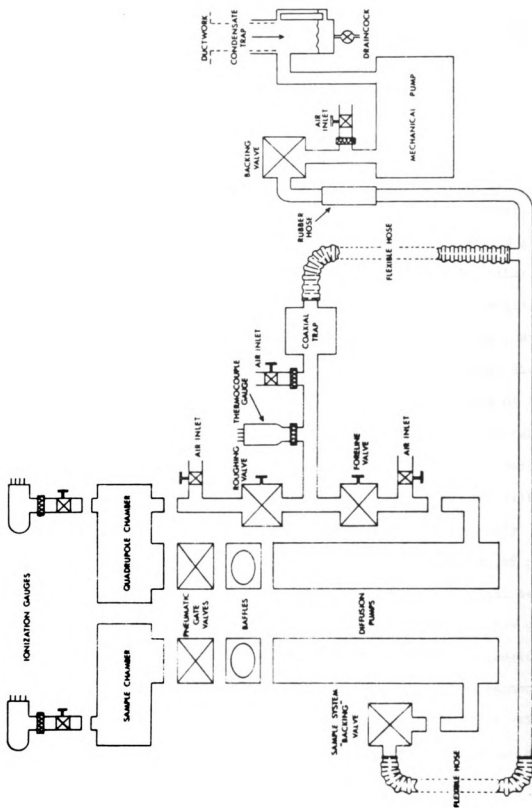


Figure II-25. Sample and Quadrupole Pumping Systems.

CHAPTER III

THE INTERLOCK SYSTEM

A. Introduction

The vacuum system of the photoionization mass spectrometer, described in Section H of Chapter II, is quite complex. Even though each component in the system may be quite dependable, the fact that there are so many components involved greatly increases the possibility that something will fail in any given time period. Even if the individual vacuum components were completely reliable, the necessary electrical and water utilities are not, so that occasional failures of the vacuum system are virtually inevitable. If proper action is not taken immediately when such failures occur, the expensive components inside the vacuum chambers (the grating, electron multipliers and mass filter) may be irreparably damaged. All of these components are extremely sensitive to the presence of even very thin contaminating films of pump oil. A vacuum accident may result in unacceptable amounts of oil being injected into the chambers. This oil will reduce the reflectivity of the grating, and even worse, may be polymerized by the high intensity VUV radiation striking the grating, resulting in a film which cannot be cleaned, thus destroying the usefulness of the grating. Pump oil also degrades the performance of the electron multipliers irreversibly. If the pressure in the chamber rises sufficiently while the power to the multipliers and quadrupole remains on, high voltage arcing may

occur which can ruin both types of components.

Although the risk of damage is most severe during an experiment when both the lamp and high voltage devices are on, damage can also result at other times. Unattended operation of the instrument thus poses considerable risk, since an accident may require the investment of significant amounts of time and money to return the instrument to working order. The risk can be avoided by leaving the gate valves closed and pumps off, which is of course impossible during an experiment, and which increases the amount of time necessary to start an experiment since the chambers must be first pumped down. The only reasonable alternative to taking such risks is to use some sort of automated safeguard system.

This chapter describes the interlock system which is used with the MSU photoionization mass spectrometer. It is designed as a "stand alone" system which is not computer operated; it can thus be used to protect the instrument between experiments as well as during them, allowing the instrument to remain safely under full vacuum conditions at all times.

Most of the devices controlled by the interlock system require either 115 VAC or 230 VAC power. At the time the system was designed and built (1973), solid state relays with logic level control inputs were still much more expensive than simple electromechanical relays. Although a much more flexible and sophisticated interlock system could be designed using standard TTL circuitry, the system described here uses RRL (relay-relay logic!) exclusively, providing considerable cost savings. If the interlock system were being designed today, the recent

decrease in price of dependable solid state relays would probably make an all-solid state design competitive in price. The parts cost for the entire interlock system was less than \$1400, only a little more than the cost of a new grating. This investment has been repaid many times over during the past four years, as none of the vacuum failures which have occurred have caused any damage at all to the protected devices.

The interlock system is not sophisticated enough to totally automate the pumpdown of the instrument. However, it does mandate the correct sequence of operations during startup, which greatly reduces operator errors, although it cannot eliminate them. When used to protect the unattended instrument, the action that it takes on detecting an error condition is very simple: any device which, if left on, could conceivably be harmed or cause harm to another device, is deactivated. The vacuum system then remains shut down until an operator decides that it is safe to start again.

Section B of this chapter is a discussion of the capabilities of the interlock system, which is built in several separate, though interconnected modules -- one master control module (MCM) and several basically identical vacuum system control modules (VCM), one for each pumping system described in Section H of Chapter II. The circuitry of the MCM is described in Section C, and that of the VCMs in Section D.

B. Capabilities and General Description

Figure III-1 shows the general organization of the interlock system. Each VCM is responsible for monitoring and controlling the

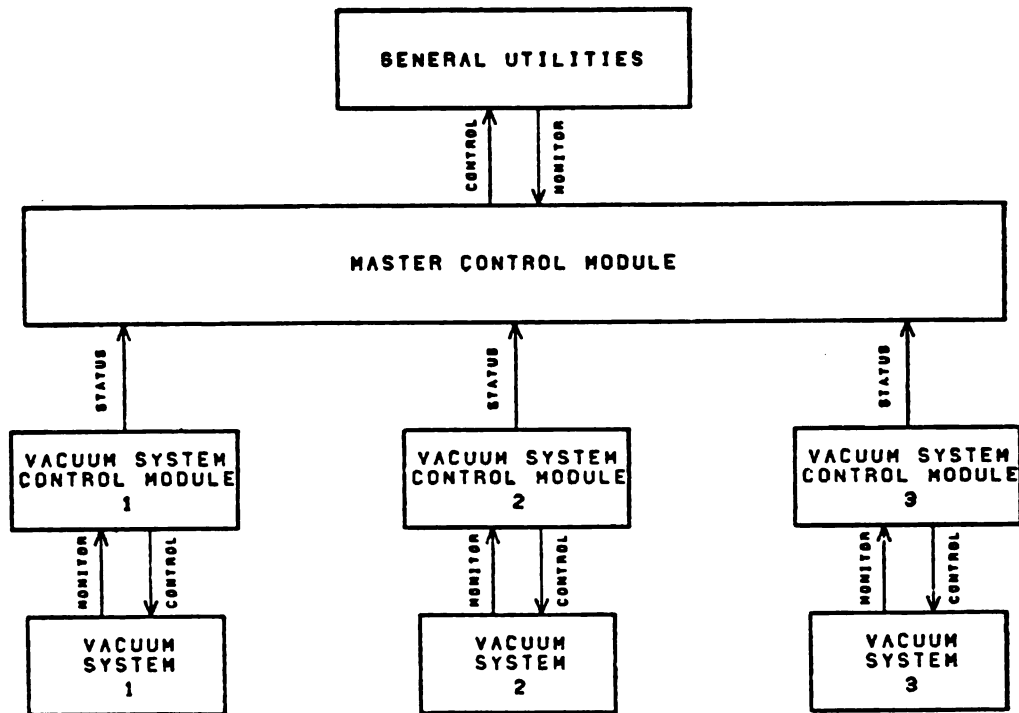


Figure III-1. Interlock System Organization. Although five VCMs are currently in use, only three are shown.

operation of the pumps and valves of a single vacuum system. Those devices which are not strictly a part of any of the individual vacuum systems (termed "general utilities") are monitored or controlled by the MCM. If a VCM detects an error condition, it not only shuts down that vacuum system, but also sends a status signal to the MCM which can then take appropriate action with the general utilities.

1. Master Control Module

The pressure in the vacuum chambers cannot be measured during an experiment, due to the high background signals that arise when the ion gauges are on. Whenever an error signal is received from a VCM, however, the MCM assumes that the pressure in the vacuum chamber may be too high for the electron multipliers or quadrupole to be run safely, and therefore turns them off. When these devices are off, the experiment cannot be continued and so there is no point in allowing the lamp to continue operating, or to allow further sample gas into the chambers. In fact, if the gate valves have been closed by the VCMs, there may be some danger in allowing sample to flow into the chamber continuously. Accordingly, the lamp power supply is turned off and lamp and sample gas solenoid valves are closed.

There are frequently occasions when it is safe and desirable to operate some of the general utilities without having all vacuum systems fully operational, especially during start-up or testing. To allow for such possibilities, the MCM is provided with a separate override switch for each VCM. When the switch is in the override position, an error signal from the corresponding VCM is ignored by the MCM, thus allowing for some operator discretion in the amount of automated control.

A simple flow switch in each water return line from the lamp (Figure II-8) is used to monitor the flow of cooling water to the lamp. If adequate flow does not exist, the lamp may be in danger of overheating, which may lead to breakage of the discharge tube. Water driven into the monochromator by atmospheric pressure could easily damage the grating, and would certainly require much effort to clean

up. Thus, when inadequate flow is detected by the MCM, the lamp power supply is immediately turned off. Since the low flow may in fact be caused by a broken discharge tube or water line, the solenoid valve in the water line is also closed to minimize flooding of the monochromator or laboratory. Again, there is no point in continuing the experiment with no light, and the other general utilities are deactivated.

The lamp power supply is also turned off by the MCM if the lamp pressure switch (Figure II-8) indicates that the lamp gas reservoir has fallen below a set pressure, since the grating can be permanently damaged if the power supply is allowed to remain on after the lamp gas has run out.²

The MCM can also monitor a blade-type airflow switch in the ductwork leading to the mechanical pump minihoods (see Section H, Chapter II). If the airflow falls below a preset limit for longer than a few seconds, and one or more of the mechanical pumps is on, an audible alarm is sounded to warn the operator that the pump exhaust fumes may be circulating back into the laboratory. So far, this feature has remained largely unused, since no noxious or toxic samples have yet been studied. A summary of the control and monitoring functions of the MCM is presented in Table III-1.

²An example of such an accident occurred during the term that I spent in the laboratory of W. A. Chupka and J. Berkowitz at Argonne National Laboratory. The helium supply for the one-meter PIMS instrument [43,92,93], which uses a thyratron type lamp power supply [60], ran out, but the discharge was observed to continue for several minutes, with changed color. The mechanism of the altered discharge is not understood, though it was found to be reproducible. However, the grating was ruined beyond the capability of a cleaning and recoating repair, presumably because of surface damage from sputtered lamp material. It is not known whether lamps powered by different types of high power supplies, or even if different lamps would behave in a similar fashion. We have been unenthusiastic about studying the matter further since the experiments, if successful, would be extremely expensive.

Table III-1. Summarized MCM Monitor and Control Functions.

Conditions Monitored
VCM status signals
Lamp cooling water flow
Lamp gas pressure
Minihad air flow
Devices Controlled
Lamp power supply
Lamp gas valve
Lamp water valves
Electron multiplier power supplies
Quadrupole power supply
Sample gas valves

2. Vacuum System Control Modules

Each of the pumping systems in the overall vacuum system include an electropneumatically operated gate valve, but the other main vacuum valves in the system (backing, roughing and foreline) may be either manual or automated, and in fact may not be present at all (see Figures II-22 through II-25). Some of the vacuum pumps require 115 VAC power, others require 230 VAC. The individual VCMs are nonetheless identical in basic design, being adapted to the specific requirements of the pumping system with which they are used by means of jumper wires on their back panels. The discussion here and in Section D will assume that all four main valves are present (e.g., the monochromator pumping system, Figure II-24) and are controlled via the VCM. With certain exceptions mentioned later, those units which use fewer automatic valves follow the same general guidelines, so that the discussion will be applicable to all units.

The main purpose of each VCM is to maintain a clean vacuum environment within the chamber, and to protect the chamber from the pumping system if the latter is not operating correctly. It does so by refusing to allow the gate or roughing valves of the system to be opened if anything is amiss. A secondary purpose is to protect the pumps themselves from damage due to improper operation, which is accomplished by simply turning them off.

In order to accomplish these goals, the VCM must be able to monitor the power to the mechanical and diffusion pumps to make sure they are on, and must be able to control the power to the diffusion pump so that it can be turned off if anything goes wrong. It must

of course have control of at least the gate valve, and preferably all of the valves, to ensure that they are opened only at appropriate times in the start-up sequence (and to ensure that they cannot be closed at improper times!). The details of the sequencing that is allowed by the interlock system are discussed in Part 3.

It is extremely important to monitor the foreline pressure, since if it is too high, the diffusion pump will exhibit an excessively high rate of backstreaming into the chamber, even though all other conditions for proper operation are met. Variations in foreline pressure once the experiment is started are usually small, unless some element in the vacuum system fails. Examples of things which could cause significant increases in foreline pressure include sudden and large leaks in the chambers, accidental closing of the backing valve, or failure of the mechanical pump. A sudden leak might be caused by the breakage of either the lamp discharge tube or some glass part in the sample inlet system. The VCM will prevent the backing valve from being closed via the control panel when the gate valve is open, but if the valve is unplugged or if air pressure is lost it will close nonetheless. The mechanical pump may fail for a variety of reasons, a common source being the breakage of the V-belts connecting the pump and motor. While such occurrences as these are hopefully rare, all have been known to happen, and the amount of damage that can be done is substantial enough to make precautions worthwhile.

The pneumatic gate valves need compressed air both to open and to close. If air pressure is lost while the valves are open, they will be "stranded" and the VCM will be unable to close them if the

need arises. Since the smaller backing valves only need air pressure to open, and are closed by a spring return, they will close automatically if air pressure fails, thus creating one of the situations, described above, which leads to an increase in foreline pressure. Therefore, the compressed air pressure is also monitored so that the gate valves can be closed after the pressure drops below a certain limit, but while there is still sufficient pressure to do so.

Since cooling water is also necessary for correct operation of the diffusion pumps, the flow of the water is also monitored. If it fails, the pumping system is shut down and a water solenoid valve is also closed to prevent possible flooding in case the failure was due to a ruptured water line.

In order to obtain the desired cleanliness in the vacuum chambers, the diffusion pumps must be baffled with a cooled baffle. The VCMs monitor the compressors for the refrigerant used to cool the baffles to make sure they are on before the gate valve is opened. If the compressors are turned off, the gate valves will close to prevent any material condensed on the baffles from being evaporated into the chambers.

Finally, some diffusion pumps have provision for a thermal switch used to indicate that the outer wall temperature of the pump is too high. This may be due to a loss of cooling water, but also to a low fluid level in the pump. The VCMs make provisions for monitoring such sensors. The functions of the VCMs are summarized in Table III-2.

3. Sequencing

The discussion here refers to a completely automated system. The requirements for a partially automated system are the same except

Table III-2. Summarized VCM Monitor and Control Functions.

Conditions Monitored
Mechanical pump power
Diffusion pump power
Position of automated valves
Foreline pressure
Diffusion pump cooling water flow
Diffusion pump wall temperature
Baffle refrigerant compressor status
Devices Controlled
Power to pumps
Power to valves
Coolant water solenoid valves
Status signal to MCM

that the operator must assume the responsibility for following the required procedures for the manual devices.

i. Mechanical backing pump:

To ensure the safety of persons working in the pump room far away from the interlock system control panel, a safety switch is located within easy reach of each mechanical pump. This switch will override all other switches and turn off the pump when it is thrown. Assuming that the safety switch has not been thrown, the mechanical pump can be started from the control panel at any time. However, it cannot be turned off while the diffusion pump is on, except with the safety switch or in case of power failure. If power fails, the mechanical pump will shut off and stay off until manually restarted.

ii. Backing Valve:

The backing valve can be opened only if the mechanical pump is on, and will automatically close if the mechanical pump goes off. It cannot be closed from the control panel while the gate valve is open, but can be at any other time, and will close automatically if power fails.

iii. Diffusion Pump:

The diffusion pump can be turned on and will remain on only if its wall temperature and foreline pressure are below safe values, the baffle cooling is on, and the backing pump is on. It can be turned off at any time the gate valve is closed, and is turned off automatically if power fails.

iv. Gate Valve

The gate valve can be opened only if the diffusion pump is on,

the roughing valve is closed and the backing and foreline valves are open. It can be closed at any time, and will close automatically if power fails or the diffusion pump is turned off.

v. Foreline Valve:

The foreline valve can be opened only if the mechanical pump is on and the roughing valve is closed. It cannot be closed while the gate valve is open, except in case of power failure, but may be closed at any other time.

vi. Roughing Valve:

This valve can be opened only if the backing valve is open and the foreline valve is closed (i.e., the gate valve is closed). It may be closed at any time and will close automatically on power failure.

vii. Status Signal to MCM:

The status signal will indicate that all is well only when the gate valve is open. Any condition that causes the gate valve to close (including closing it via the control panel) will result in a "fault" status signal until the gate valve is once again opened.

viii. Exceptions:

The only major exception to the guidelines outlined above is the first differential pumping system (the Roots blower). The VCM for this system is very much simpler than the others since the system includes only two pumps and one valve, the gate valve. This VCM is built on the same chassis as the second differential pumping system VCM for reasons of space economy.

The circuitry supplied with the Roots blower-backing pump assembly allows the backing pump to be turned on at any time, and

then turns the blower on automatically when the inlet pressure drops to a preset value. The blower will also turn off automatically if the pressure rises above this limit. Both pumps can be turned off by the operator at any time, and will shut off and remain off if power fails. The VCM for this system consists solely of a gate valve control, which allows the gate valve to be opened only when the blower is on. The gate valve can be closed at any time, and closes automatically if power fails or the blower is turned off.

C. Master Control Module

Figure III-2 shows the circuitry involved in the airflow monitor and hood alarm portion of the master control module. The air flow switches, which are closed when sufficient flow is present, are located in the ductwork above the mechanical pump minihoods. The mechanical pump switches are actually relay contacts in the individual VCMs which are connected as shown, via connectors on the master control unit rear panel. The alarm operates on 115 VAC power and is thus on unless relay K2 is energized. To allow for momentary fluctuation of the airflow switches, K2 is a time-delay-on-release type of relay. Its proper operation demands that it receive continuous 115 VAC power, since it otherwise could not keep from releasing immediately. K2 is triggered by relay K1 so that when K1 is energized, then K2 is also and the alarm is off. This will occur when all mechanical pumps are off, or when both airflow switches indicate sufficient flow. If these conditions are not met (i.e., airflow fails when any pump is on), then after the adjustable time delay of K2, the alarm will sound. If K2 is re-energized by K1

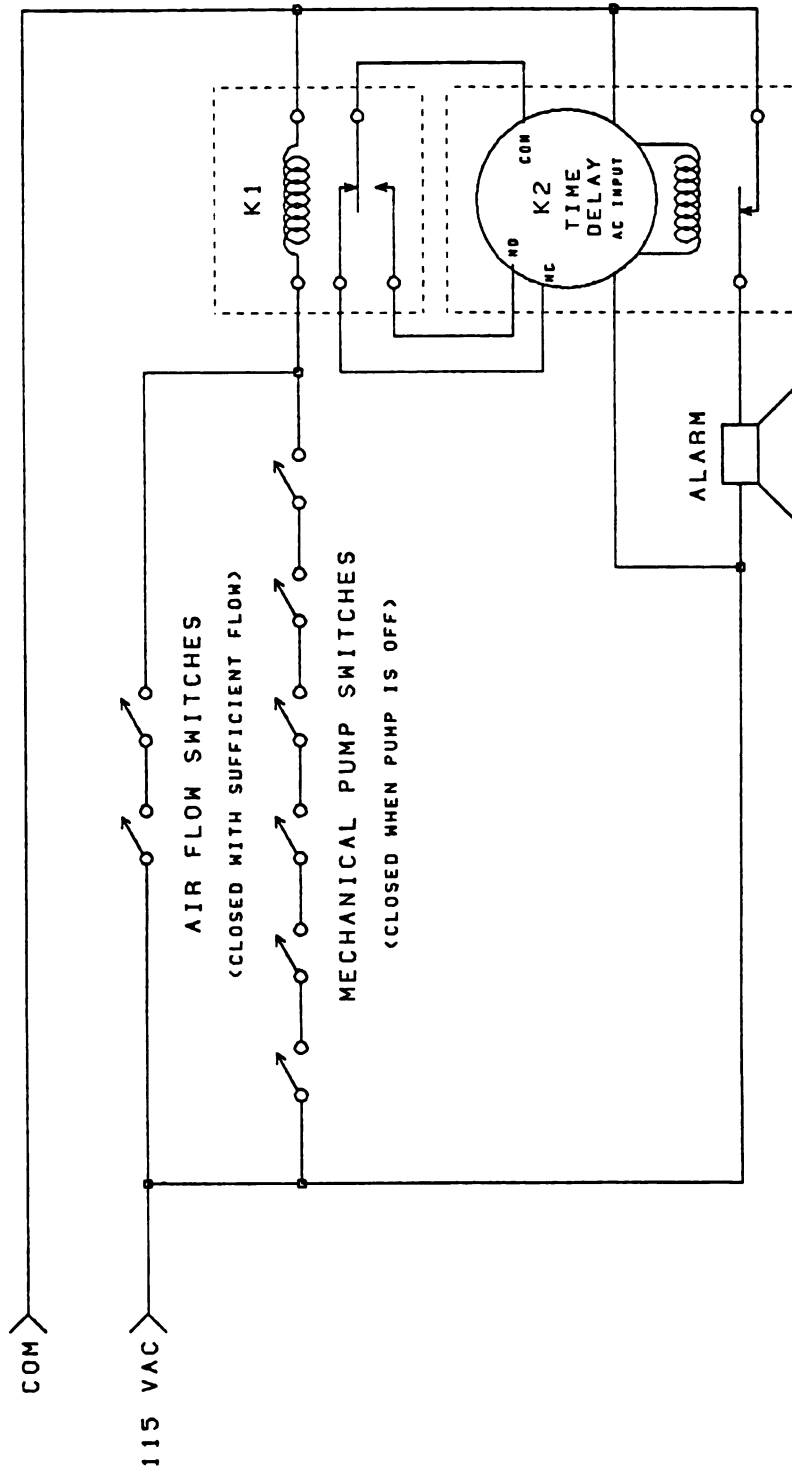


Figure III-2. Hood Alarm Circuit (MCM).

before the time delay expires, no action is taken.

Figure III-3 shows the circuitry which is used to control the diffusion pump water flow for each vacuum system. There is a similar circuit for each water-cooled pump. The water flow switch and water solenoid valve are located near the vacuum pump. When the override switch is closed, power is applied to the water solenoid valve, initiating water flow. Assuming the flow rate is sufficient, the flow switch will close, activating relay K3 and in turn time delay relay K4, and the override switch can be released. The second set of contacts on K4 is used to signal the appropriate VCM through one of its safety switch inputs (see Section D) that the water flow is sufficient. The time delay action of K4 ensures that air bubbles in the water line or other similar brief interruptions in water flow do not shut off the solenoid valve. If the flow interruption lasts longer than the delay of the relay (adjustable, 1-180 sec), then the water solenoid valve is shut, and the VCM is signalled by way of the open circuit to its safety switch input that water flow has ceased, and will take appropriate action. L1 is a small indicator light on the front panel which indicates that the flow switch is closed. These water flow circuits are perhaps more appropriately considered as part of the individual VCMs. However, the circuits are actually located in the MCM chassis, since the need for the time delay feature was not realized until after the VCMs were assembled.

The remainder of the circuitry in the main control module is shown in Figure III-4. It includes circuitry for monitoring and controlling the cooling water flow to the lamp cathode and anode, which is analogous to that for the diffusion pump cooling water discussed

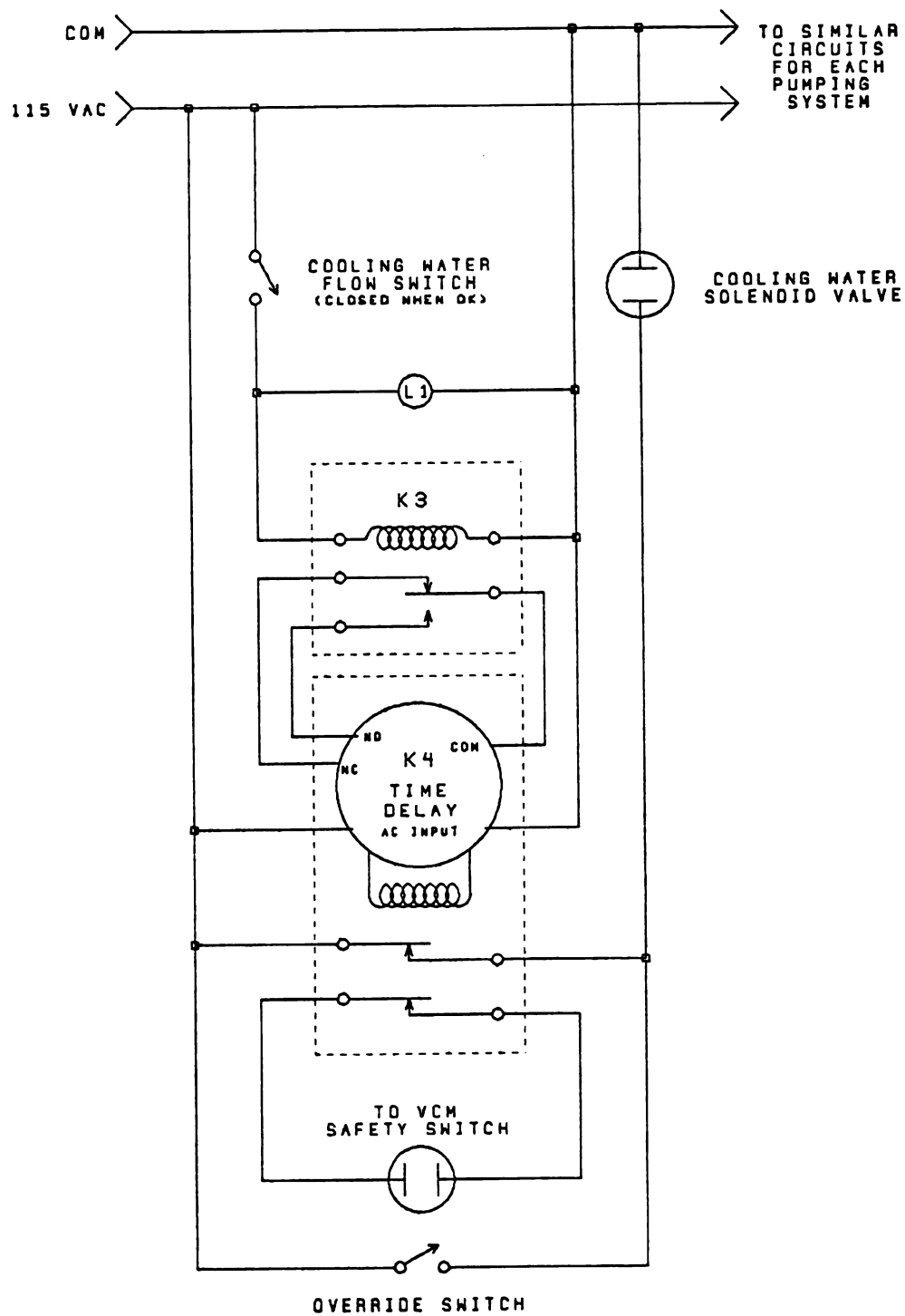


Figure III-3. Diffusion Pump Water Flow Interlock (MCM).

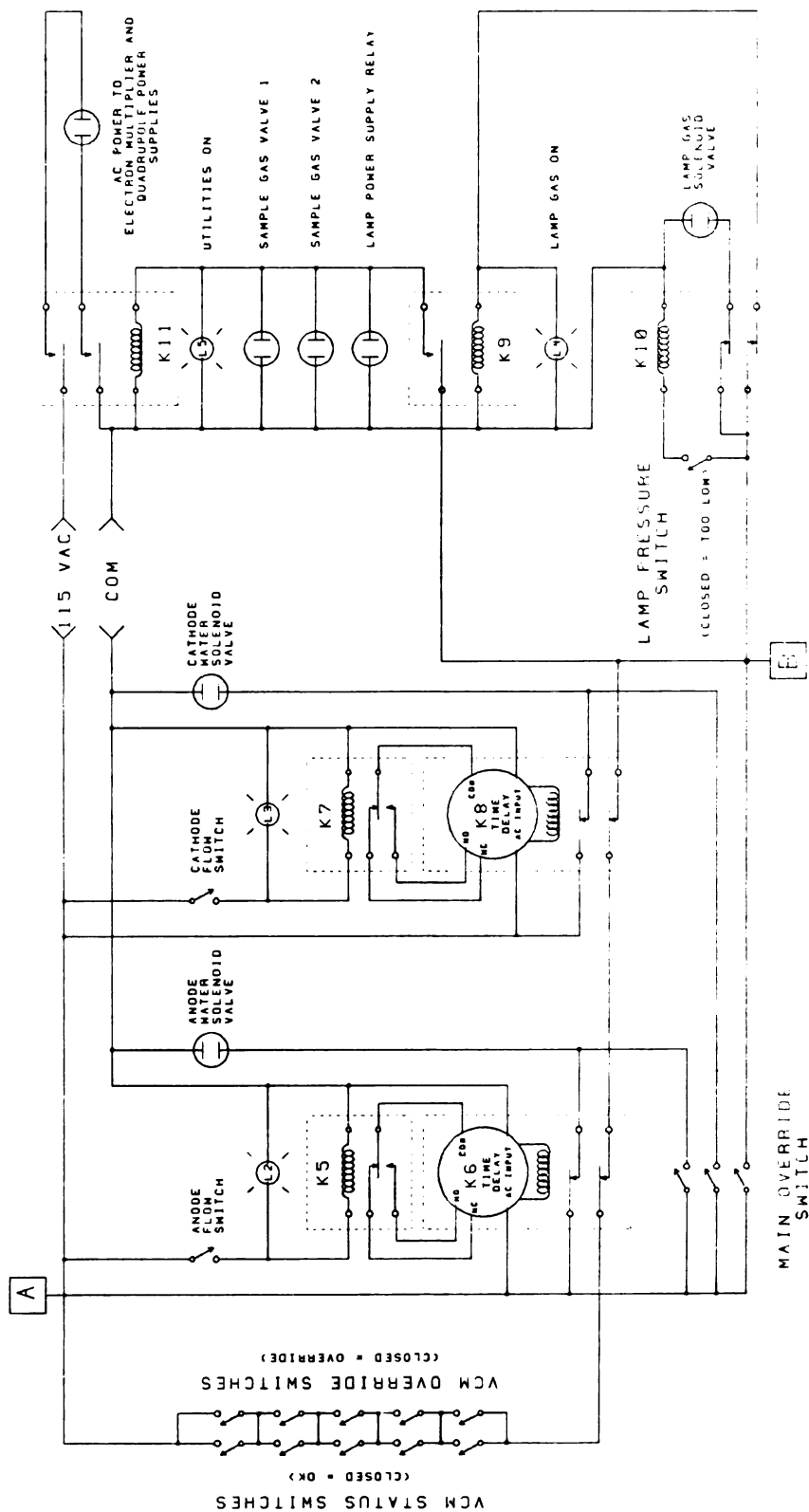


Figure III-4. General Utilities Interlock (MCM).

in the preceeding paragraph. The main difference is that a single override switch is used for both water circuits, and that the second set of contacts on relays K6 and K8 are used to control the power to the other "general utilities" -- the lamp power supply, sample gas solenoids, etc.

The VCM status switches are relay contacts located in the individual VCMs which are closed when the status of the individual vacuum systems is satisfactory (i.e., when the gate valves are open). Any VCM status switch can be bypassed by throwing the corresponding override switch. If such utilities as the lamp power supply are to be on, 115 VAC power must at least travel from point A to point B. This can be accomplished if the main control module override switch is thrown. Otherwise, the power must pass through each VCM status switch (or its override switch bypass) as well as the lamp water control relays. Thus, assuming no override switches are thrown, power will pass from A to B only if all gate valves are open and water flow to both lamp anode and cathode is satisfactory.

Having arrived at point B, the current will normally flow through the contacts of K10 to actuate K9, providing power for the sample gas valves and the relay which allows the lamp power supply to be on. When K9 is actuated, so is K11, which is a heavy-duty power relay which turns on the bus strip into which electron multiplier power supplies, quadrupole power supplies, etc. are plugged. Current also flows through the contacts of K10 to power the lamp gas solenoid valve.

If the lamp pressure switch should close, indicating a low pressure in the lamp gas reservoir, then K10 will be activated. This will

de-activate K9 and K11 and all of the devices which are powered through these two relays.

D. Vacuum System Control Module

The circuits for the vacuum system control modules are shown in Figures III-5 through III-7. The circuitry shown in Figure III-5 is all located in a small box near the mechanical backing pump in the pump

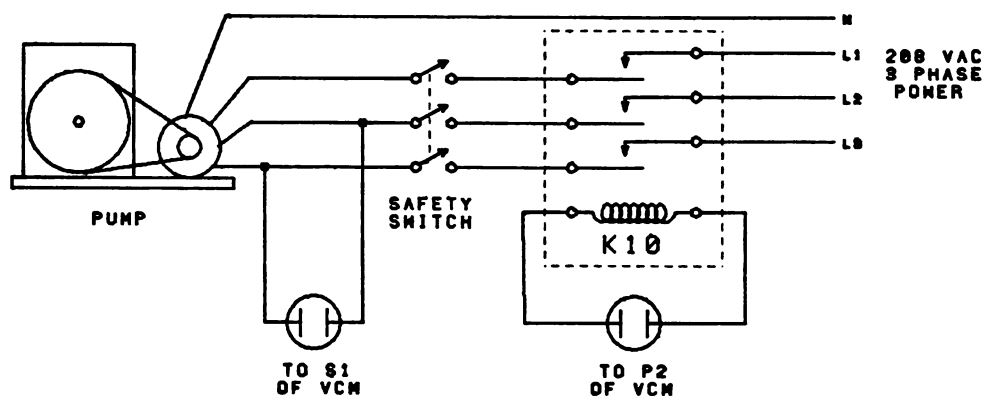


FIGURE III-5. "Local" Mechanical Pump Circuit.

room, while that in the other two schematics is all mounted in the VCM chassis near the instrument. The numbers in the small squares in these figures refer to terminals on the rear panel of the unit, which can be interconnected in a number of different ways to allow operation from different voltages or with a different number of automatic valves. They are shown connected for a complete (four-valve) system, and Part

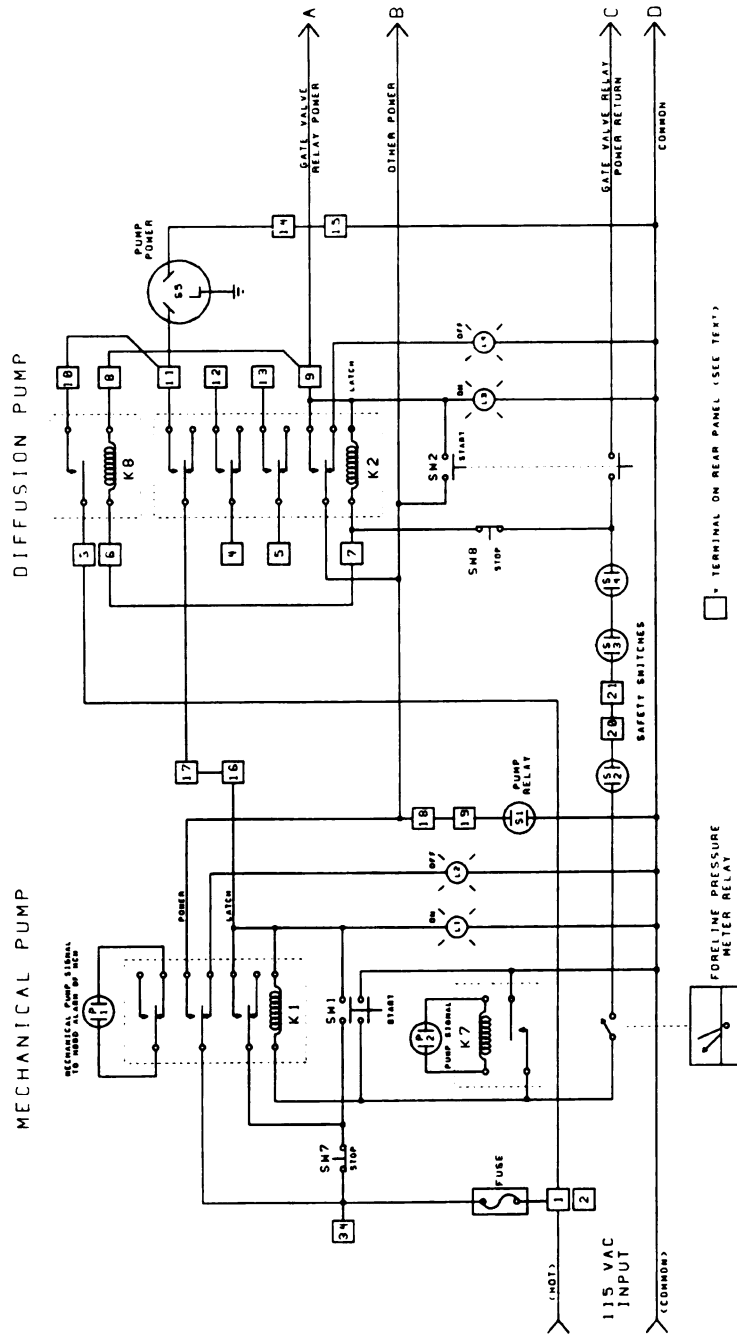


Figure III-6. Pump Control Circuit (VCM).

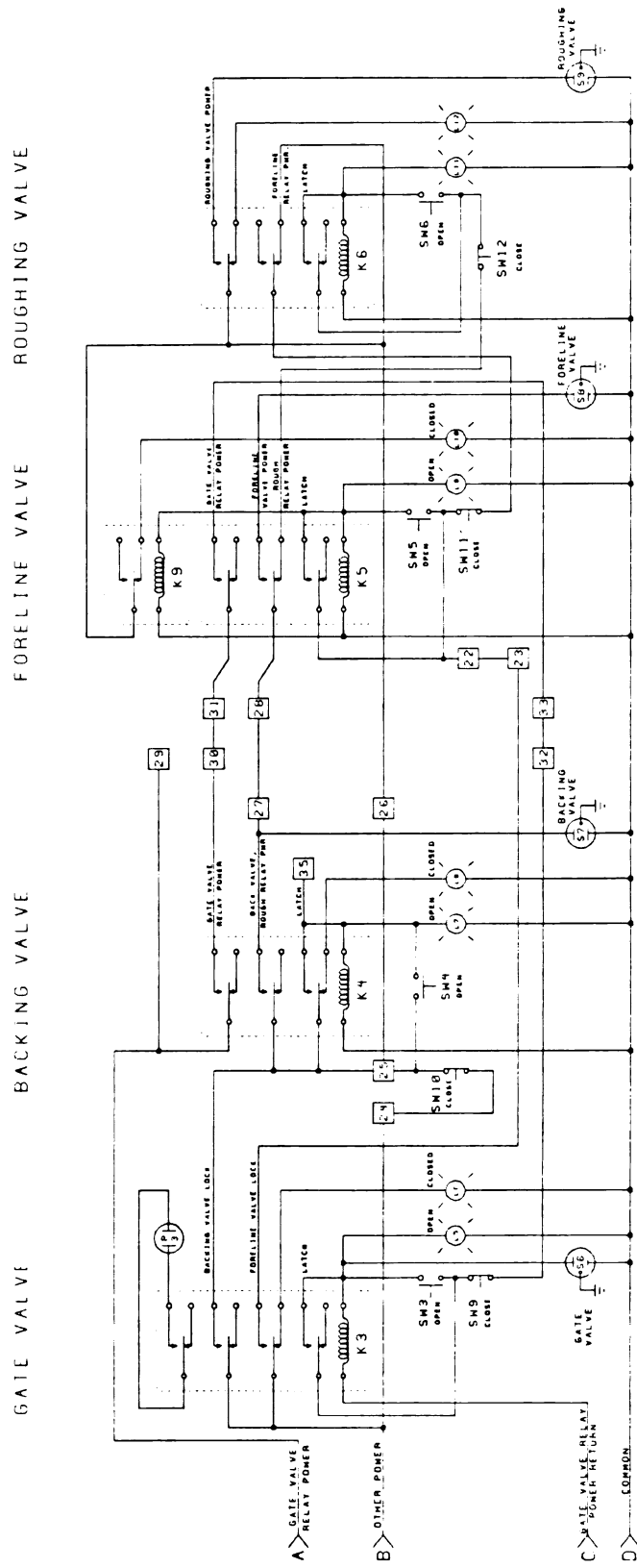


Figure III-7. Valve Control Circuit (VCM).

1 of this section discusses the operation of the VCM when it is so wired. Information on connecting the terminals for operation with different numbers of automatic valves, etc., is included in Part 2.

1. Operation of a 115 VAC, Fully Automated System

Relays K1 - K6 are the main control relays in the VCM; each is dedicated to a single pump or valve. Each of these relays is wired in a simple self-latching circuit. They are initially actuated by pressing a momentary action pushbutton switch (SW 1-6) which provides power to their coils (assuming certain other conditions are met). Once actuated, the coils obtain power via the first set of relay contacts, so that the relay remains actuated when the switch is released. If current flow through the coil is interrupted for long enough to de-energize the relay, it will remain off until the actuating switch is again pressed. One means of interrupting the current flow is with a second momentary switch (SW 7-12), although under certain conditions these switches may be shorted and thus made inoperable. Each relay is also wired to two indicator lamps on the front panel which indicate whether the corresponding valve (or pump) is open (on) or closed (off).

Relay K1 (Figure III-6) controls power to the mechanical pump via relay K10 (Figure III-5) which plugs into socket S1. Relay K7 is wired in parallel with the mechanical pump motor and is energized whenever the mechanical pump is on -- it controls the return current from K1. This system allows the mechanical pump to be operated from a different power supply than the VCM (for instance, on a 208 VAC three phase line as in the present example). If the mechanical pump or the VCM power fails, K7 or K1 will be de-energized, causing the other to also

be de-energized, and the entire system will be turned off until started again manually. Note that the safety switch (Figure III-5) will turn off the mechanical pump and de-energize both K1 and K7 regardless of their previous status.

SW1 provides current to energize K1 and also provides a bypass around K7 which is needed until the motor starts and K7 closes. All other control relays draw their power from B (Figure III-6) via contacts on K1. Thus if K1 is de-energized for any reason, the entire pumping system will shut down immediately. SW7 can be used to turn the mechanical pump off, except when the diffusion pump is on, in which case current to the relay coil is available through terminal 16 as well as through SW7, so that pushing SW7 has no effect. Thus, the mechanical pump cannot be accidentally turned off (from the control panel) while the diffusion pump is on.

Relay K2 (Figure III-6) controls power to the diffusion pump through its own contacts when a 230 VAC pump is used. However, it cannot handle the higher current required by our 115 VAC pumps. Relay K8 has a very high current handling capacity, and is added in parallel to K2 for this reason. The diffusion pump relays are energized by SW2 (providing the mechanical pump is on to provide a current path to SW2) and de-energized by SW8. The return current from these relays must pass through sockets S2-S4 as well as the switch contacts of a meter relay used to monitor the foreline pressure. Any sort of safety switch which is closed when the condition it monitors is within acceptable limits can be connected to S2-S4. One of these sockets is used for the water flow signal from the MCM, others may be used for diffusion pump

thermal switches or compressed air pressure switch. If any of these switches open, K2 and K8 will be immediately de-energized, turning off the diffusion pump.

The power for the gate valve must come through point A, and thus both the mechanical and diffusion pumps must be on for it to open. Note that this does not mean that the diffusion pump must be warmed up, only that it must be on. The operator is responsible for making sure that adequate warm-up time is allowed. K3 (Figure III-7) controls power to the gate valve. Since its return current passes through the same safety switches as the current from K2, the gate valve will shut whenever any of these switches are opened. SW9 will close the gate valve at any time. In order to open the gate valve, not only must SW3 be closed, but power must find a complete path from point A to SW3 through the contacts of K4 and K5. It will be able to do so only when both backing and foreline valves are open. When the gate valve is open, K3 provides another path for power to K4 and K5 so that the normal "close" switches for the backing and foreline valves have no effect.

The normally closed part of SW2 is in the circuit to prevent the gate valve from opening whenever the diffusion pump is started. If it were not included, power coming through SW2 and relay K2 can return to common not only through the safety switches, but also through the coil of K3 and the gate valve solenoid. This devious path was discovered only after the VCMs were completed, at which time the second half of SW2 was wired into the circuit as shown to prevent the problem. This solution works, but has the drawback that if the gate valve is open (and diffusion pump on) pushing the diffusion pump on button again, which

should have no effect, closes the gate valve! In retrospect, a better solution would be to draw power for the gate valve from the contact which provides the foreline valve lock.

Power for K6 comes via a contact on K5 which is closed when K5 is deenergized, and power for K5 comes from a contact on K6 which is closed only when K6 is deenergized. This ensures that the roughing and foreline valves cannot be open at the same time.

2. Wiring for other options

When the diffusion pump of the system requires 230 VAC power, the terminals must be wired differently than shown in Figure III-6 and III-7. Table III-3 lists the terminal connections for both options.

Table III-3. Terminal Connections for Main Power Options.

Diffusion Pump Voltage	115 VAC	230 VAC
Rear Panel	1-3	1-4
Terminal	6-7	2-5
Connections	8-9	11-12
	10-11	13-14
	14-15	16-17 ^a
	16-17 ^a	18-19 ^a
	18-19 ^a	20-21 ^a
	20-21 ^a	

^a Must be modified when a single mechanical pump is shared between two systems. See text for further details.

Table III-4 lists the terminal connections to be used when different numbers of automatic valves are used. In addition, the VCMs were designed so that two diffusion pumps could share a backing pump (as in the quadrupole/sample chamber pumping systems), with the two VCMs being simply interconnected via the terminal strips. Table III-5 gives the connections to be made for such a system. In this situation the sample system interlock utilizes the thermal and baffle switches corresponding to the sample system diffusion pump, but also involves the same foreline pressure switch as the quadrupole interlock system. The dual-role mechanical pump is latched on if either diffusion pump is on, providing another path for power through terminal 16 (though the power to the two diffusion pumps must be in phase). The power for the control relays in the sample interlock system comes through K2 of the quadrupole system, and both systems will shut down upon failure of the mechanical pump.

Table III-4. Terminal Connections for Automatic Valve Options.

Automatic Valves	Gate, Backing	Gate, Foreline, Roughing	Gate, Backing, Foreline Roughing
		22-23	22-23
Terminal	30-32	24-25-26-27-28	25-26
Connections		29-30-31	27-28
		32-33	30-31
			32-33

Table III-5. Terminal Connections for Sharing a Mechanical Pump Between Two Systems.

Sample System Terminal	Quadrupole System Terminal
17	16
18	18
21	20

Note that it is important that the 115 VAC power for the VCMs which are interconnected in this way be in phase. Unfortunately, the power used for our sample and quadrupole systems is not in phase, and a more complicated interconnection scheme has been used. It is shown in Figure III-8.

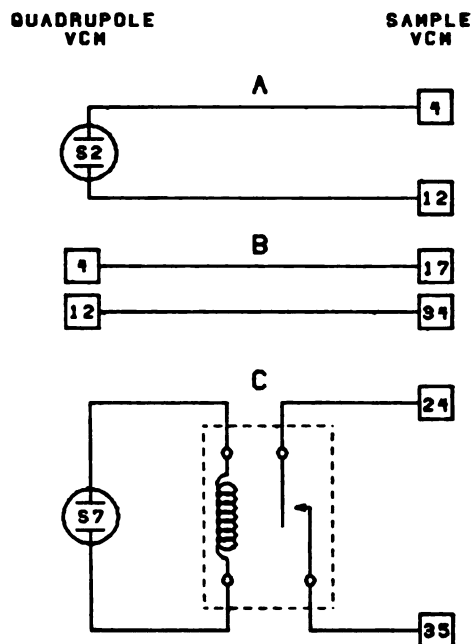


Figure III-8. Alternate Interconnection Method. The (shared) backing valve and mechanical pump are connected to the Sample VCM as usual. Connection group A ensures that the quadrupole diffusion pump will turn off if the sample diffusion pump does. Connection group B latches the mechanical pump on if the quadrupole diffusion pump is on. Connection group C latches the backing valve open if the quadrupole VCM backing valve relay is energized.

CHAPTER IV

CONTROL AND DATA ACQUISITION

A. Introduction

The instrument controller, in this case a minicomputer, must be capable of interacting with the instrument in several ways if efficient automated acquisition of accurate PIE curves is to be possible. The computer must be able to acquire integrated photon and ion intensity data and, if data acquisition is to be efficient, must be able to control the data integration time. The computer must also change the wavelength setting of the monochromator and, if periodic reference measurements or scattered light measurements are to be possible, the wavelength control must be bidirectional. In order to make background measurements, the computer must be able to prevent photons and photoions from reaching the corresponding transducers, perhaps by interposing a shutter in the photon beam before it enters the ion source.

These are the absolute minimum requirements, but there are other capabilities which may also be very desirable. For instance, when electron multipliers are used for both ion and photon detection, their background signals will be similar and very small. It is then appropriate to make photon and ion background correction measurements at similar and infrequent intervals. However, if a sodium salicylate photon transducer is used (see Chapter II, Section D), it will exhibit a much

higher and more unstable background than the ion electron multiplier, so that its background should be measured much more frequently. In the limit of a very weak signal, as much as half of the total photon measurement time should be devoted to measuring the photon background for optimum efficiency [94], but devoting such a fraction of the time to measurement of ion background would be very inefficient. It may therefore be desirable to include a second shutter between the ion source and photon detector which can be used to measure photon backgrounds without interrupting the integration of the ion signal.

The mass to be transmitted by the mass analyzer is usually set at the beginning of an experiment and left unchanged until the beginning of the next, so that the computer need have no control over it. However, as seen in Chapter II, Section F, it is important to use only the minimum necessary mass resolution in order to maximize the ion signal intensity. The setup of the experiment can be facilitated if the computer can be used to acquire partial mass spectra in the mass region of interest at various resolutions, and thus computer control of mass may be desirable.

It might also be useful for the computer to control the voltages applied to other elements of the ion optics. In general, the optimum focusing conditions will differ from ion to ion, depending on mass as well as the necessary resolution. If the computer were programmed to optimize these parameters, for instance by the use of some sort of SIMPLEX procedure [95], the amount of time the operator must spend in setting up each experiment could be greatly reduced. Furthermore, ion-molecule reaction studies in which the translational energy of the reactant ion is varied could then be easily automated.

Whenever a reference measurement is made, the time necessary to move to and from the reference wavelength as well as the actual reference data acquisition time are wasted as far as the acquisition of PIE data is concerned, and thus such reference measurements should be made as infrequently as possible. If the sample pressure fluctuates faster than any of the other system variables which are accounted for by such measurements, it would be more efficient to measure the pressure separately (and simultaneously with photon and ion data) so that reference measurements (to correct for the remaining variables) can be made less frequently. Simultaneous measurement of pressure, photon and ion data requires the use of a pressure transducer which does not itself create ions (e.g., the ionization gauge). The capacitance manometer is most often used in such cases, and has the additional advantage that its response is independent of the nature of the sample; furthermore it can be easily interfaced to the computer. The major disadvantage of the capacitance manometer is that, on its most sensitive pressure scales, it is extremely sensitive to temperature variations, which cause rather large zero drifts (as much as 1-3% of the reading per °C). This requires that the computer be able to check the true pressure zero frequently; in order to do so the computer must be able to control the opening and closing of the valves in a simple valving arrangement. If measurements are to be made as a function of pressure, then the computer must also be able to control the pressure, not merely measure it.

As of this writing, the MSU photoionization mass spectrometer includes only those items mentioned above as being absolutely necessary -- photon and ion data acquisition, bidirectional wavelength control, and

control of a shutter before the ion source. A valve control and pressure measurement interface has also been completed recently; however, the solenoid valves originally used were found to have unacceptably high leak rates. This capability is temporarily unused pending the arrival of new valves. The hardware necessary to implement the other capabilities discussed above is in various stages of design.

This chapter describes the hardware and electronics used to implement the computer control and data acquisition capabilities of the MSU PIMS instrument, and characterizes their performance. The computer is a Digital Equipment Corporation (DEC) PDP 8/I minicomputer [96], which uses a 12-bit word length and is equipped with 12K of memory. Data and program storage peripherals include a dual DEC-Tape drive and a dual floppy disc drive (Sykes 7200 series) [97]. Interaction between the user and computer takes place via a Tektronix 4010 graphics display terminal [98] or an ASR 35 Teletype [99]. Graphic output from the computer can be plotted not only on the Tektronix terminal, but also on a Heath EU-205 strip chart recorder [100] which is interfaced to the computer.

Nearly all of the electronic circuits described in this chapter are constructed on printed circuit boards that are mounted in a card cage in the computer rack. The card cage and basic computer interfacing techniques used are described in Appendix B. The real-time clock which is used by the computer for a wide variety of timing functions is described in Appendix C. Several conventions have been adopted for the schematic diagrams of all these circuits. Almost all of the logic circuits use standard 7400 series TTL integrated circuits (ICs). These are not specified except when the type of IC is not obvious from its symbol, in which

case the IC type number is shown in parentheses (e.g., a simple box used to represent a counter might be labelled with (74161)). All ICs use standard totem-pole outputs, except those marked with an asterisk (*), which have open collector outputs. Small numbers are sometimes shown just outside the IC symbol; these refer to pin numbers (for the dual in-line package, unless otherwise specified). The large numbers within the IC symbol are for reference only. Any resistor shown without a specified value is simply used as a pull-up resistor, and has a value between 1000-10,000 ohms (not critical).

The wavelength drive is described in Section B of this chapter, the shutter control in Section C, and Section D contains a description of the interface used to control the capacitance manometer valves. The remaining sections of the chapter are devoted to the electronics used in data acquisition. The two approaches which have been used most commonly to measure the output of electron multiplier devices are DC current measurement (DC) and pulse counting techniques (PC). The two approaches have different merits, and which approach is more advantageous is determined by the characteristics of the transducers, the photon and ion arrival rate and the limitations of the electronics used. Their relative merits have been amply discussed in the literature (see reference [101] and other literature cited in Section F), to which the reader is referred. Either type of measurement system can be used with the MSU instrument. The electronics used for DC measurements are described in Section E. The novel circuit developed for pulse counting measurements is described and evaluated in Section F, and in Section G the counter circuits which interface both DC and PC electronics to the computer are described.

B. The Wavelength Drive

The wavelength of the McPherson 225 monochromator is scanned by rotating the drive screw of its modified sine drive mechanism, the digital-to-shaft angle conversion being accomplished with a high resolution stepping motor (Responsyn HDM 150-1000-4) [102], which is directly coupled to the drive screw. The stepping motor has an angular resolution of 0.36° (1000 steps/revolution), which with a 1200 line/mm grating is equivalent to $0.025 \text{ \AA}/\text{step}$. At the maximum rated speed of 1500 steps/second, a region of $1000 \text{ \AA}$ can be scanned in less than 30 seconds. A set of 10:1 reduction gears can be inserted between the stepping motor and drive screw to obtain finer steps for high resolution work, although this reduces the maximum effective scan rate by a factor of ten.

The motor is driven by the computer via a commercial indexer (Responsyn PSD-4-3) which can also be controlled manually. The interface between the computer and the indexer is very simple, as shown in Figure IV-1. The computer sends the indexer either a "step forward"

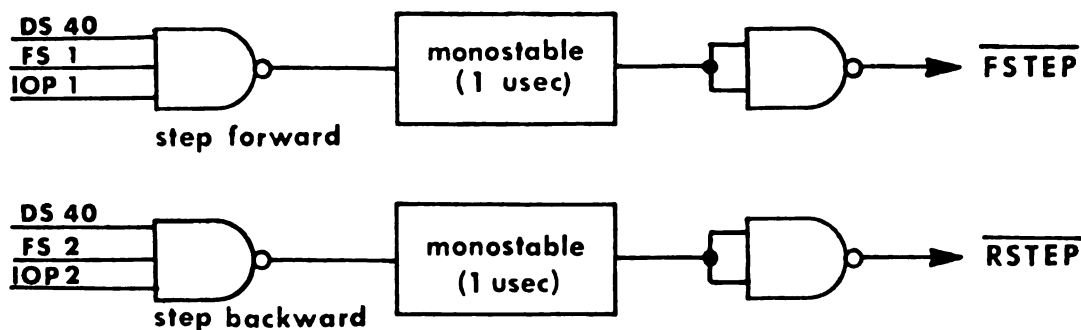


FIGURE IV-1. Stepping Motor Interface.

($\overline{\text{FSTEP}}$) or "step backward" ($\overline{\text{BSTEP}}$) command, the pulses being lengthened by the monostables in order to meet the input pulse width requirements of the indexer.

The wavelength drive screw could be damaged if the stepping motor attempts to drive it past its upper or lower limits. To prevent this possibility, the limit switches on the slave screw of the monochromator (located outside the vacuum) have been disconnected from the synchronous motor supplied with the monochromator (and not used by us). The stepping motor indexer has been modified to accept signals derived from these switches so that if the upper limit switch is actuated, or disconnected, the stepping motor will no longer respond to commands from either the computer or operator to move toward higher wavelength, although it will still move toward lower wavelength. The action of the lower limit switch is just the reverse.

No optical shaft encoder is used on the wavelength drive screw, nor is there any electrical feedback from the stepping motor to the computer, i.e., this is an open loop system. The computer keeps track of the wavelength by maintaining a step count in a software register which is incremented or decremented each time the computer issues a STEP instruction. This system is simpler than a closed loop system, but high wavelength accuracy depends on absolute reliability of the interface and stepping motor. Many stepping motors, however, tend to miss steps if they are started or stopped suddenly at full speed, and must instead be accelerated more gradually to full speed, and then decelerated before stopping. In addition, there may exist certain resonant frequencies in the system (motor plus load) which cause the motor to misstep

when run at the corresponding step rate. It is important that these problems be avoided if good accuracy is to be attained.

Figure IV-2 shows eight repetitive scans of the atomic nitrogen emission lines near 1134 Å, obtained with the reducing gears in place. Between each scan, the monochromator was scanned without acceleration at a rate of 1200 steps/second to a wavelength of 1034 Å and back, a total of 80,000 steps. Any missteps should show up as an apparent shift in wavelength of the emission peaks. Table IV-1 summarizes the apparent peak locations of the scans shown in Figure IV-2, and Table IV-2 summarizes results from two similar series of scans recorded on different days. In order to ascertain whether the small apparent wavelength shifts were due to lost steps, the reducing gears were removed and several further scans of the same lines were recorded (again with an 80,000 step round trip to lower wavelength in between scans). If the observed deviations were in fact due to lost steps, they should be ten times as large without the reducing gears. In fact, the results (also summarized in Table IV-2) are different only by a factor of 2-3, most of which can be attributed to the difficulty of precisely measuring peak positions by eye when only a few points could be obtained over the width of the peak. Thus the observed wavelength shifts are not attributed to lost steps, but instead to the limit of reproducibility of the wavelength drive system of the monochromator. This limit of reproducibility is assumed to be about 0.03 Å.

Since there is no evidence for lost steps, there is no need to accelerate the stepping motor to its maximum scan rate. Although the programs used (see Chapter V) include provisions for such acceleration,

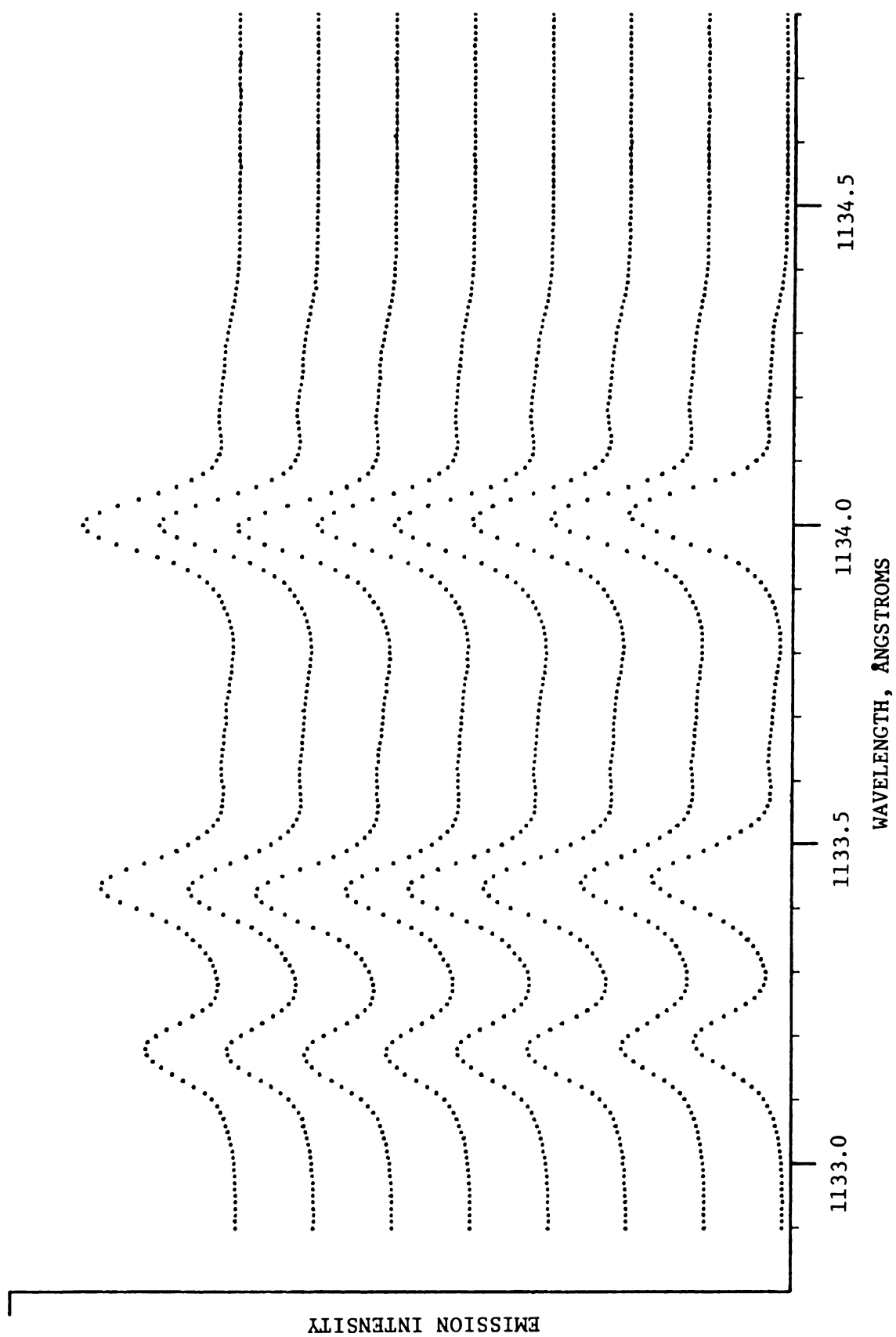


Figure IV-2. Eight Successive Scans of the Atomic Nitrogen Lines at 1134 Å.

Table IV-1. Wavelength Reproducibility.

Scan	Measured Wavelength (Å)		
	Peak 1	Peak 2	Peak 3
1	1134.152	1134.407	1134.979
2	1134.152	1134.407	1134.976
3	1134.146	1134.395	1134.976
4	1134.149	1134.407	1134.973
5	1134.149	1134.410	1134.976
6	1134.157	1134.407	1134.981
7	1134.160	1134.419	1134.984
8	1134.163	1134.418	1134.996
mean	1134.154	1134.409	1134.980
st. dev.	.0059	.0075	.0073
range	.017	.024	.023

Table IV-2. Summary of Other Wavelength Reproducibility Tests.

No. of Scans		Peak 1	Peak 2	Peak 3
8 ^a	st. dev.	0.013	0.012	0.014
	range	0.042	0.035	0.035
8 ^a	st. dev.	0.0054	0.0062	0.0063
	range	0.017	0.022	0.020
7 ^b	st. dev.	0.020	0.026	0.026
	range	0.043	0.059	0.075

^a10:1 reduction gears used.^bNo reduction gears used.

Table IV-3. Wavelength Accuracy.

Known λ	Measured λ	Difference	Known λ	Measured λ	Difference
948.6855	948.606	.080	1097.2372	1097.161	.076
949.7430	949.662	.081	1098.0951	1098.016	.079
950.1121	950.033	.079	1098.2599	1098.178	.082
950.7327	950.669	.064	1100.3597	1100.289	.071
950.8846	950.825	.060	1134.1653	1134.147	.018
952.3178	952.252	.066	1134.4149	1134.396	.019
953.4152	953.386	.029	1134.9803	1134.973	.007
953.6549	953.640	.015	1163.8836	1163.840	.044
953.9699	953.948	.022	1164.3246	1164.295	.030
963.9903	964.051	-.061	1167.4485	1167.394	.055
964.6256	964.691	-.065	1168.3344	1168.298	.036
965.0413	965.116	-.075	1168.5358	1168.479	.057
971.7381	971.774	-.036	1176.5098	1176.378	.132
972.5368	972.551	-.014	1177.6948	1177.571	.124
973.2342	973.235	-.001	1199.5496	1199.420	.130
973.8852	973.890	-.005	1200.2233	1200.109	.114
976.4481	976.481	-.033	1200.7098	1200.583	.127
977.9594	978.006	-.047	1215.6701	1215.581	.089
988.7734	988.875	-.102	1243.178	1243.112	.066
990.8010	990.927	-.126	1243.307	1243.234	.073
1027.4307	1027.439	-.008	1302.1685	1301.991	.178
1028.1571	1028.172	-.015	1304.8576	1304.699	.159
1039.2304	1039.323	-.093	1306.0286	1305.901	.128
1040.9425	1041.018	-.076	1310.5403	1310.446	.094
1041.6876	1041.742	-.054	1319.675	1319.541	.134

Standard deviation of differences: 0.073 Å.
 range of differences: 0.304 Å.

they are rarely used.

It should also be noted that when measurements such as those described above are made immediately after re-lubrication of the drive screw (when the lubricating oil is relatively "thick" on the threads), the first few scans show a monotonic shift toward lower wavelength of about about 0.03 Å per scan. This is in the direction that would correspond to a thinning of the oil between the drive screw and nut threads. After five or six scans, no further changes are observed larger than those reported in Tables IV-1 and IV-2. When one considers that 0.01 Å corresponds to a linear travel of only 5×10^{-5} cm along the wavelength drive screw, a reproducibility limit of 0.03 Å is not too surprising.

In order to test the accuracy of the wavelength drive system, several lamp emission spectra were recorded at high resolution over the wavelength region from 940 to 1320 Å, with a "dirty" helium lamp as the light source. This region contains many atomic emission lines whose wavelengths are well known [103]. Table IV-3 compares the known and measured wavelengths of 50 of these lines. The observed deviations fall in a range of 0.15 Å over the entire region, which is typical of all scans recorded. This is indicative of the (maximum) wavelength error to be expected when no atomic lines exist nearby for calibration. When such lines are nearby, the wavelength accuracy can be considerably higher.

C. Shutter

The small flap valve just behind the exit slit assembly of the monochromator is used not only as a vacuum valve, but also as the optical shutter. Its control knob has been removed and replaced with an extension

shaft capped with a small lever. The lever is attached to a pneumatic piston which extends and thereby closes the shutter whenever compressed air is admitted to it via a small 3-way solenoid valve. When this valve is de-energized, an internal spring in the piston opens the shutter. The position of the piston is adjusted so that when it is fully extended, the flap valve is closed far enough to block all radiation from reaching the exit slit, but not far enough to engage the detent which keeps it closed tightly. The pin connecting the piston to the lever is easily removed, allowing the flap valve to be closed all the way manually when it is necessary to use it as a vacuum seal.

The solenoid-actuated air valve operates on 115 VAC power, which is controlled by a small solid state relay (Teledyne 675-4). The relay is in turn controlled by a 5 volt signal from the shutter interface near the computer. The interface circuits are shown in Figures IV-3 and IV-4.

Whether the shutter is open or closed is determined by the state of JK flip-flop 4. Its output is buffered by the 75450 which is an AND "peripheral driver", capable of supplying sufficient current to operate reliably the solid state relay which is located some 50 feet away, near the shutter. The output of the flip-flop is also used to activate a light emitting diode (LED) on the front panel when the shutter is open. The INIT pulse, which is received whenever the computer is started, is used to open the shutter, which is thereafter controlled by the signals transmitted by the and-or-invert (AOI) gates 3 and 7. These AOI gates are included to allow either manual or computer control of the shutter. The computer generates requests to open or close the shutter via NAND

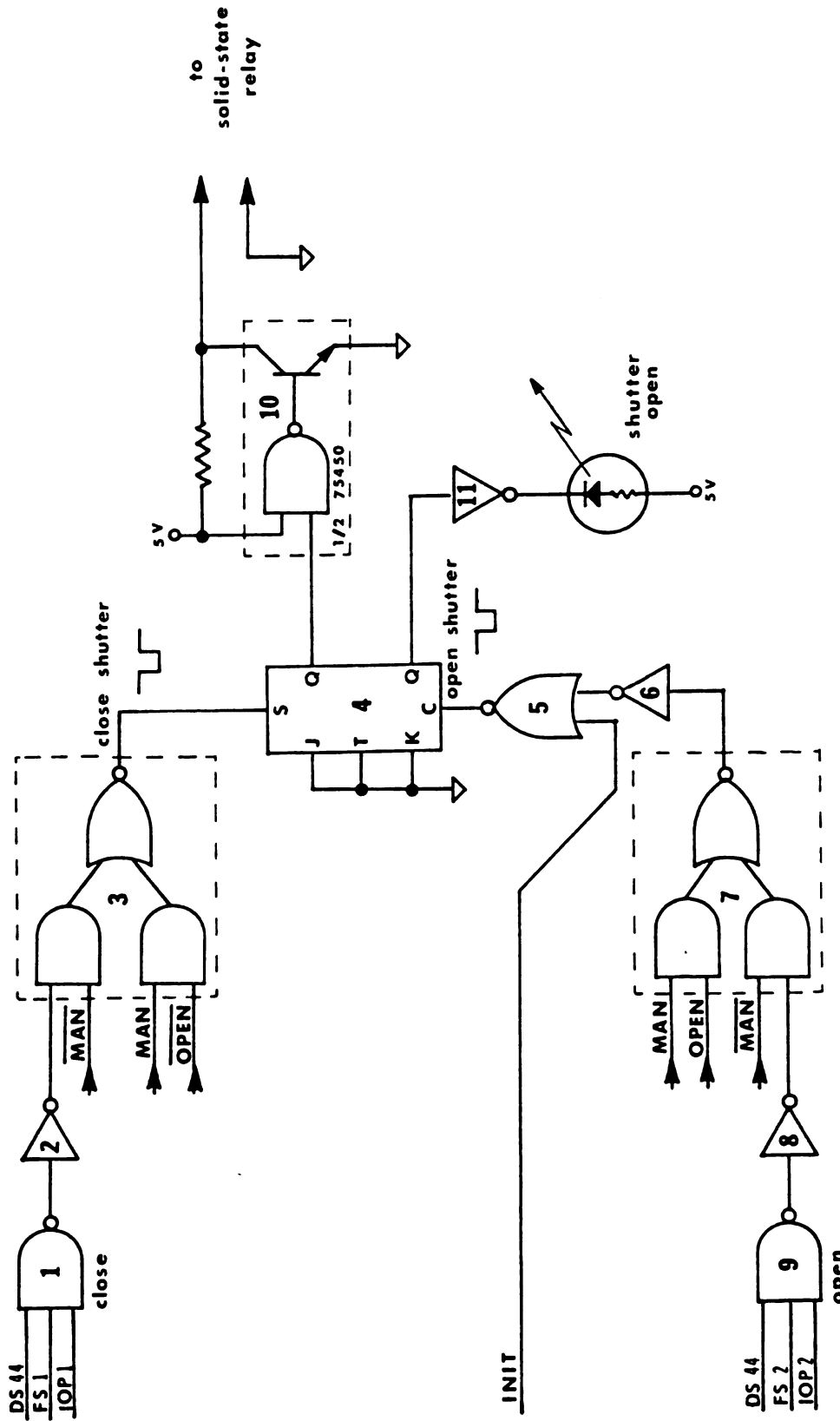


Figure IV-3. Shutter Control Interface.

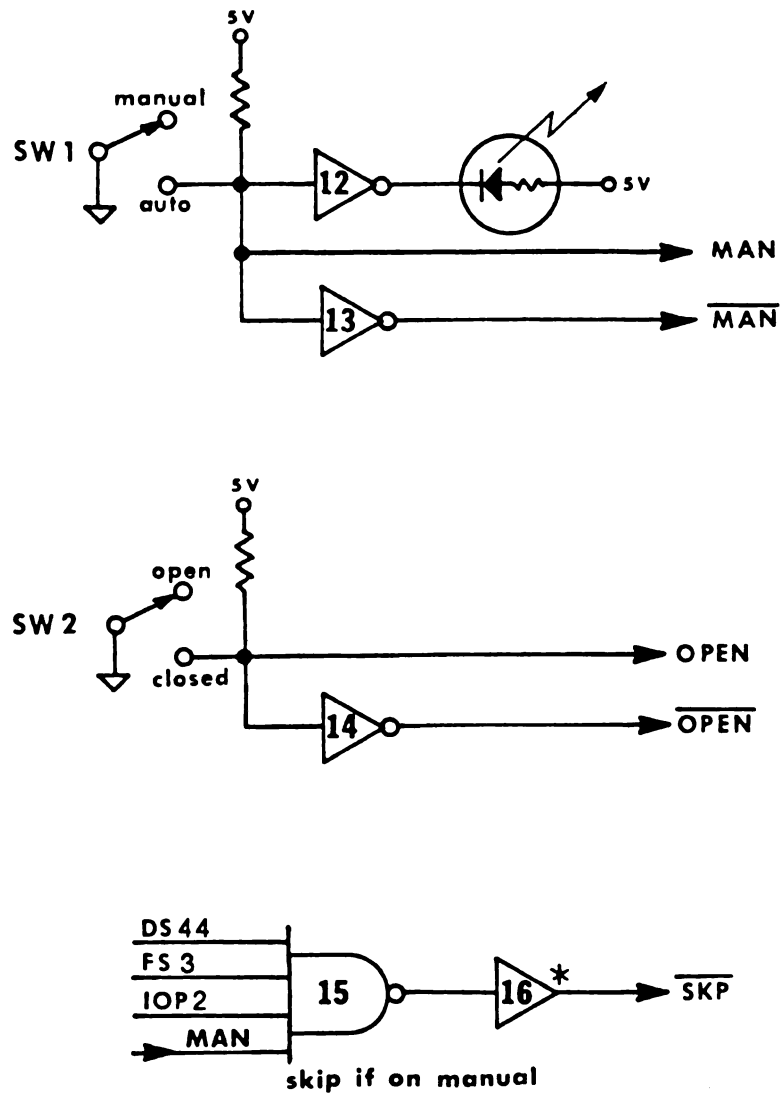


Figure IV-4. Auxiliary Shutter Control Circuits.

gates 1 and 9 and the user can generate similar requests via front panel switch SW2. The AOI gates will transmit only one of these two sets of signals, with the choice determined by front panel switch SW1.

The inclusion of manual shutter control creates a potential problem. If the user inadvertently leaves SW1 in the MANUAL position after starting an experiment, the computer will be unable to make the desired background correction measurements. Accordingly, the shutter interface includes gates 15 and 16, which implement a "skip if on manual" instruction that the computer can use to check whether it has control or not. The computer is programmed (see Chapter V) to give the user a warning message and to refuse to start an experiment if the shutter interface is not in the AUTO mode.

D. Pressure Measurement

Pressure measurement is accomplished with a capacitance manometer consisting of a Datametrix type 1014A control unit and a 571D-IT-1B2-H5 capacitance manometer "head" [104]. The head is a two port, differential pressure transducer which measures the difference in pressures between the two ports. An "absolute" pressure can be measured if one side of the head is connected to a pressure which is much lower than the pressure being measured. In the MSU instrument, the quadrupole chamber pressure serves as this reference. With an eye toward future double-chamber ion-molecule reaction studies, the pressure measurement interface was designed to allow the single head to be used to measure the pressure in either of two separate chambers. The valving arrangement is shown in Figure IV-5.

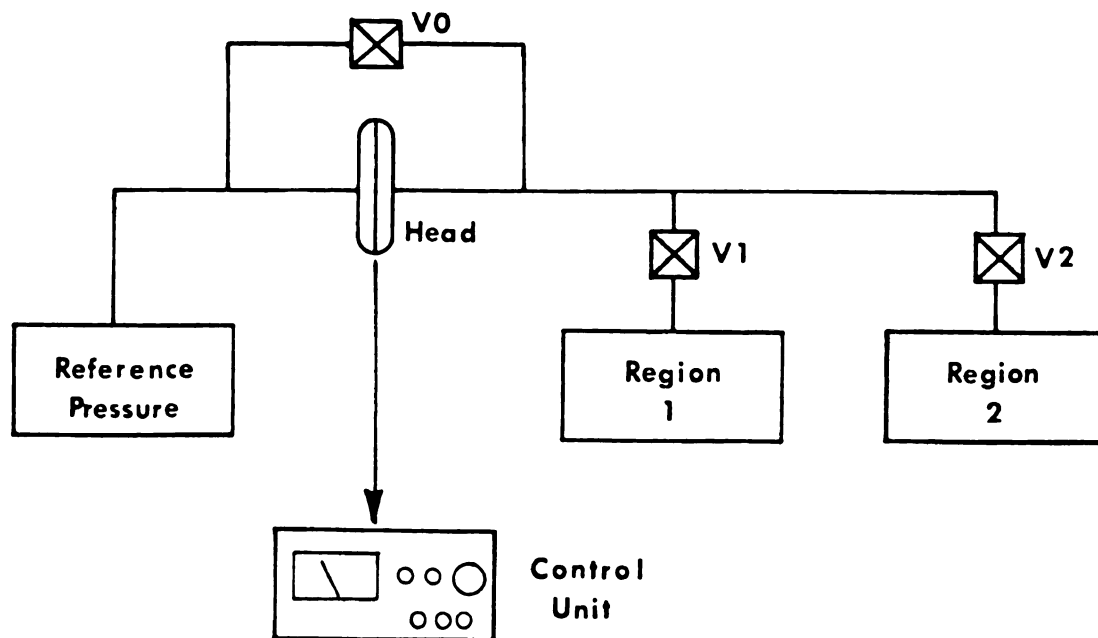


FIGURE IV-5. Sample Pressure Measurement System.

When only valve V0 is open, the pressures on either side of the head equilibrate, which allows the zero output level from the control unit to be checked. When only V1 is open, the pressure in region 1 is measured; when only V2 is open the pressure in region 2 is measured. The actual acquisition of pressure data is accomplished by measuring the voltage output of the control unit with a V/F converter and counter arrangement identical to that described in Sections E and G of this chapter. This section describes only the circuits which control the valves shown in Figure IV-5.

The valve control interface, like the shutter control, is operated either manually or under computer control, depending on the position of SW2 (Figure IV-6). Similarly, the computer can check (via gates 15 and 16) to see whether it has control of the interface.

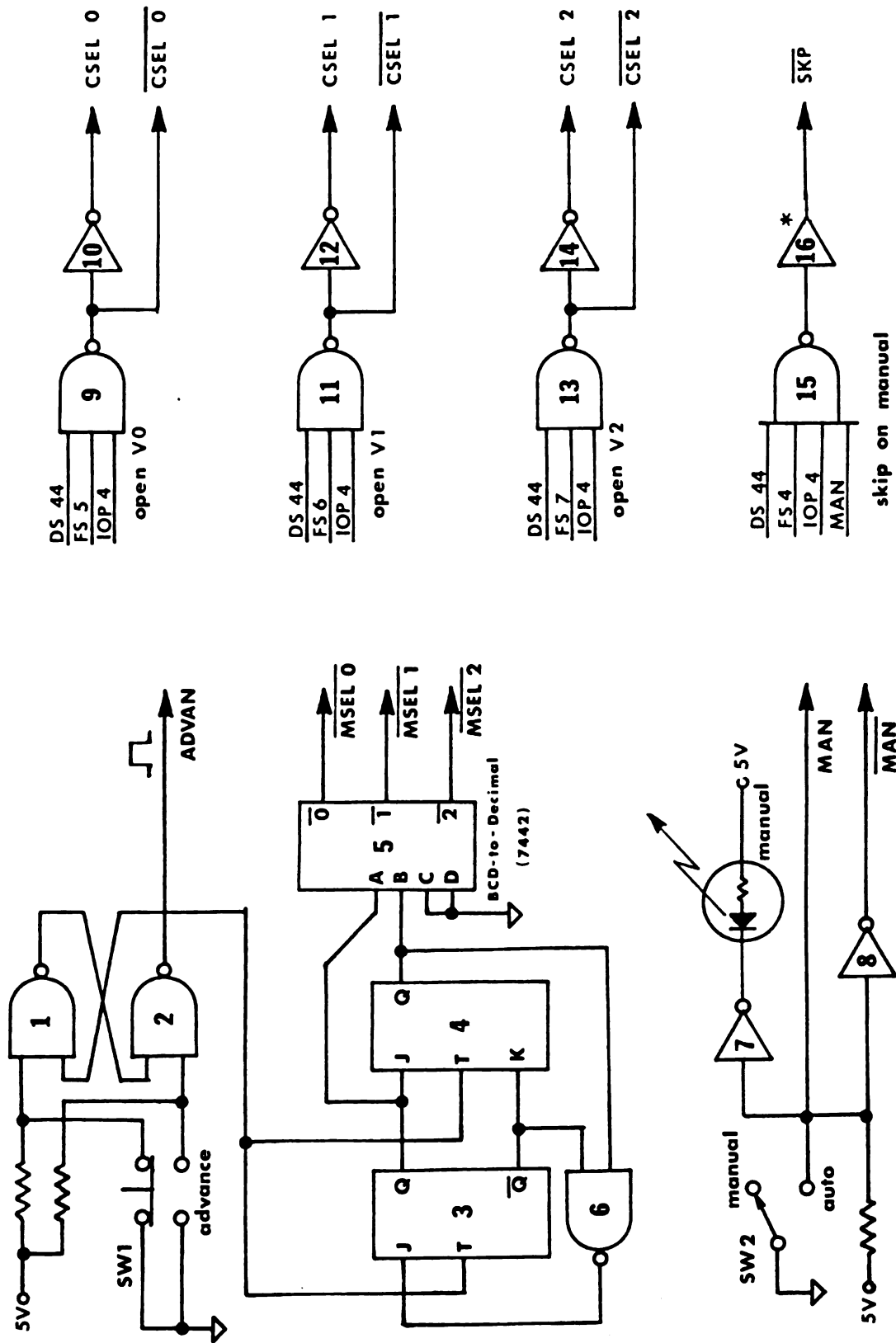


Figure IV-6. Valve Control Signal Generation.

It is both unnecessary and undesirable to have more than one of the three valves open at any given time, and the interface is designed to prevent this. When the interface is requested to open a given valve, it first closes all valves, waiting about a half a second to give any open valve time to close, and then opens the appropriate valve. As with the shutter, the valves are powered by 115 VAC, which is controlled by solid state relays. The solid state relays are driven by gates 25, 27 and 29 shown in Figure IV-7. These gates as well as the front panel indicator LEDs are controlled by JK flip-flops 21-23.

Assume that SW2 is in the manual position. When the advance pushbutton (SW1) is pushed, the ADVANCE signal ("debounced" by gates 1 and 2) will be passed by AOI gate 18, creating the CLOSE VALVES pulse which closes all valves. That same signal also triggers monostable 19, which 0.5 seconds later causes monostable 20 to emit a short OPEN VALVES pulse which goes to the toggle input of all three valve control flip-flops. Any flip-flop which has a logical 1 at its J input (ENABLE n , $n=0,1,2$) will be set by this pulse, opening the corresponding valve.

Flip-flops 3 and 4 (Figure IV-6) are wired as a modulo-3 binary counter which advances one every time an advance signal is received. The binary output is decoded by the BCD-to-decimal decoder 5 so that only one of the $\overline{\text{MSEL}}_n$ signals will be low. AOI gates 36-38 (Figure IV-8) insure that these signals control the ENABLE n lines during manual operation. Thus only one of the ENABLE n lines will be high, and the OPEN VALVES pulse will cause only one valve to open. Each time the advance pushbutton is pushed, a different valve will be opened.

It is possible to open valves out of sequence manually if the

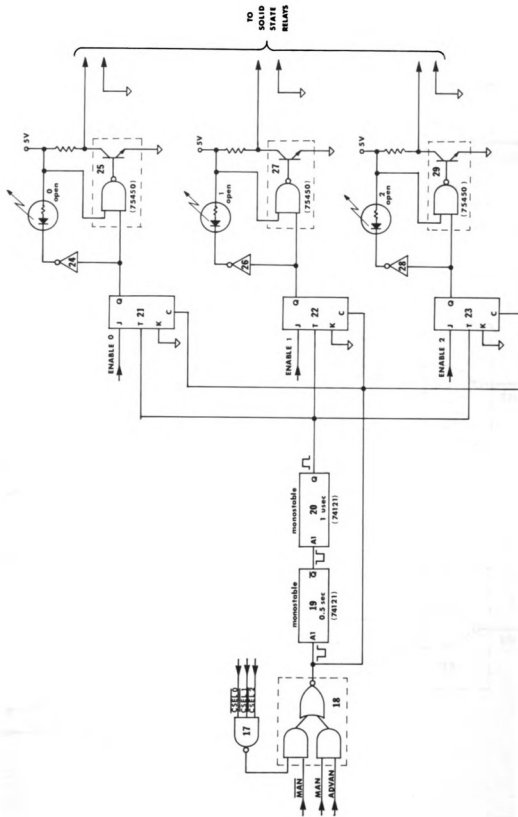


Figure IV-7. Valve Timing and Control.

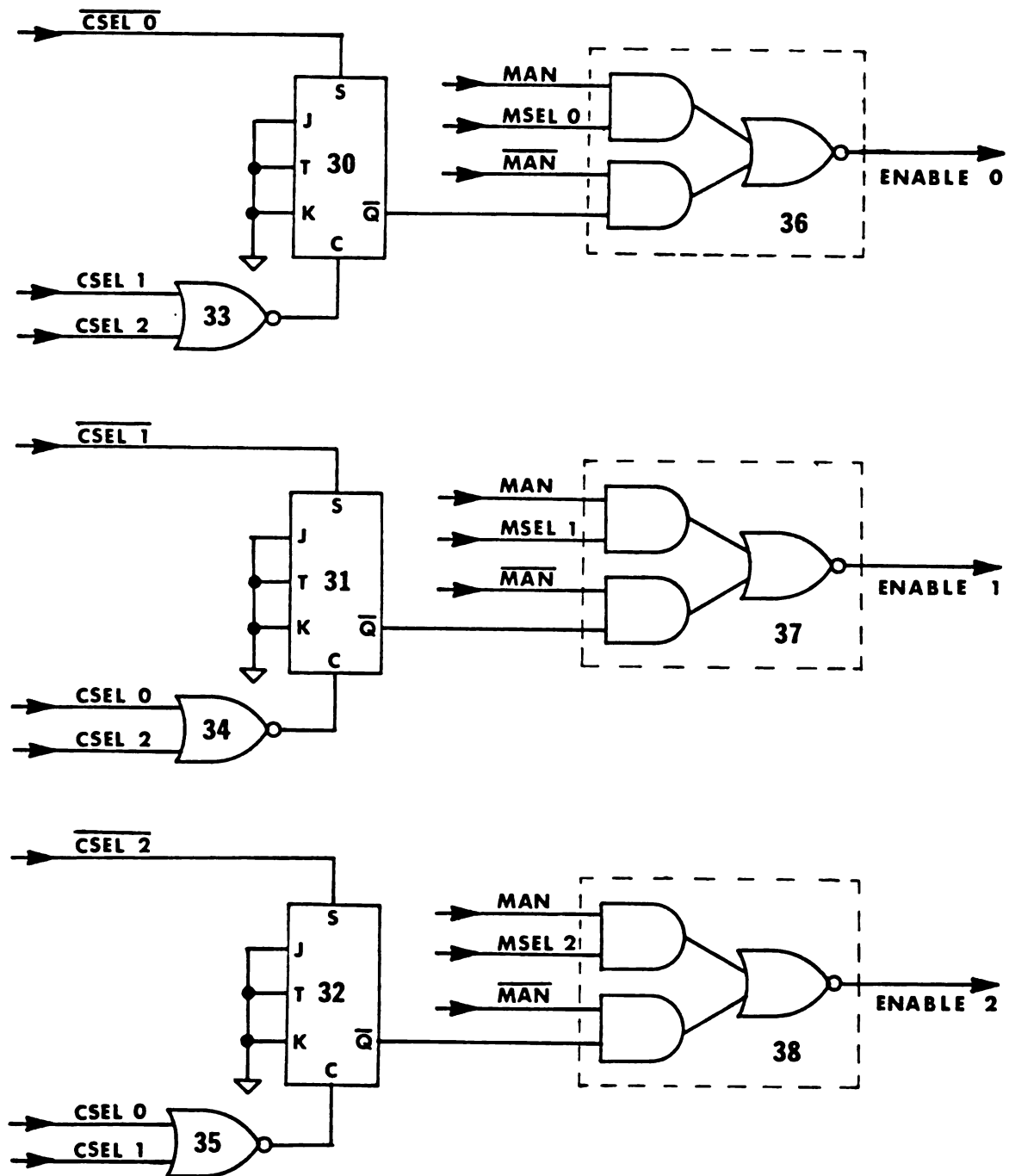


Figure IV-8. Valve Enable Signal Selection.

advance pushbutton is pressed an additional time during the delay between the CLOSE VALVES and OPEN VALVES signals. The first advance signal will begin the sequence, but the second will merely advance the binary counter. Since it is the counter value at the end of the delay time that determines which valve is opened, one valve can easily be skipped. It is also possible to close all valves by holding the advance pushbutton for longer than the delay time. The CLOSE VALVES signal will then remain active after the OPEN VALVES pulse, and all valves will be left in their closed state when the pushbutton is released.

Computer-controlled operation is similar, except that the computer has "random access" to the valves and is not forced to open them in any particular sequence. When the computer is to open a particular valve, its instruction is decoded by gates 9-14 of Figure IV-6, which generate the CSEL_n and $\overline{\text{CSEL}}_n$ signals. In AUTO mode, any of these will trigger the CLOSE VALVES and OPEN VALVES signals via AOI 18 (Figure IV-7). These signals also control the operation of flip-flops 30, 32 and 34 (Figure IV-8), which are connected so that only one will be set while the other two are cleared. During AUTO operation, the $\overline{Q}$ outputs of these flip-flops control the OPEN ENABLE lines so that once again, only one valve will be opened.

E. DC Current Measurement

Typically, electron multiplier current measurement systems are built around high quality current-to-voltage converters. The output voltage from such a converter can be measured with an analog device such as a meter or chart recorder, or it can be converted to a digital

signal and displayed or recorded. When, as in PIMS, the current must be integrated over some period of time, the integration can be performed in either the analog or digital domains, although the latter is more accurate for integration times longer than a few seconds.

One of the simplest methods of achieving such digital current integration is shown in Figure IV-9. The voltage output of the I-to-V

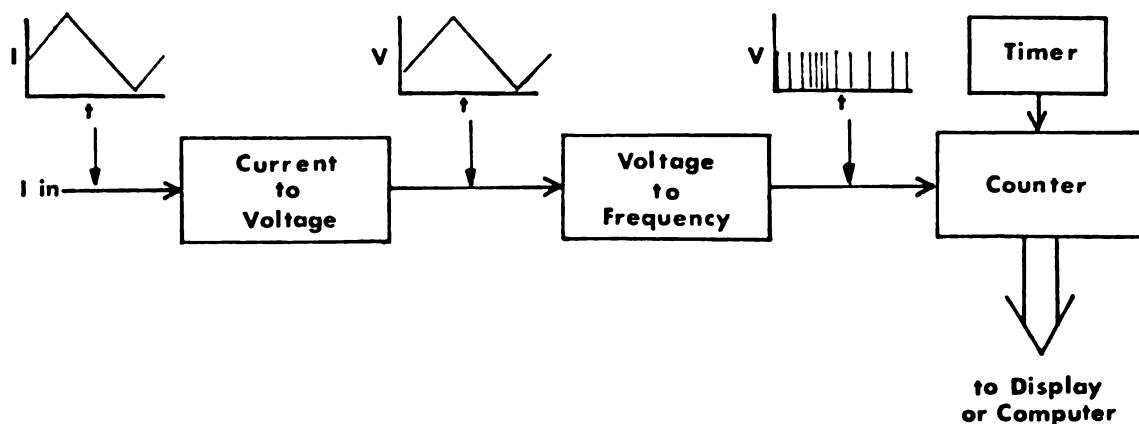


FIGURE IV-9. V-F Current Integration Method.

converter is converted to a frequency which is proportional to that voltage. The pulses from the V-F converter are counted with a digital counter, which at the end of the measurement interval contains a count proportional to the integral of the current over that same period. This is the system used in most automated photoionization mass spectrometers, including the MSU instrument.

Two different current-to-voltage converters have been used. One, a Kiethley Instruments Model 417 high-speed picoammeter [105] has a 3V full scale output, which is switch-selectable, and may correspond to a

current between 1.0×10^{-13} amps to 3.0×10^{-5} amps. The other, a Kiethley 427 current amplifier, has a full scale output of 10V and a gain selectable between 10^4 and 10^{11} volts/ampere. The digital counter and its interface to the computer are described in Section G of this chapter. The remainder of this section is devoted to the V-F converter unit.

Two very similar V-F units have been built. The circuit used is shown in Figure IV-10. It is based on the Datel VFV series converter [106]; one unit incorporates a VFV10K V-F converter with a 10 KHz full scale output, the other uses the VFV100K converter with a 100 KHz full scale output. Both converters have a 0-10 Volt input range, and specified maximum non-linearities of 0.005% of full scale (VFV10K) or 0.05% (VFV100K). The input to the converter is buffered with an AD521K [107] differential input instrumentation amplifier with switch-selectable gain. In addition, an operational amplifier inverter in the VFV converter can be switched into the circuit so that inputs of either polarity may be used. Full scale output from the V-to-F converters can thus be obtained for inputs of 1, 2, 3, 5, 10 or 20 volts, with either positive or negative polarity. This versatility is required because these units are used to interface a wide variety of voltage sources to the computer when not in use with the PIMS instrument.

The output pulses from the V-F converter are fed to a 75450 AND driver integrated circuit, which is connected as a differential voltage mode line driver [109]. When coupled with the line receiver circuit described in Section G, this circuit is capable of providing communication over very long lines (5000 ft.) with excellent noise immunity [14]. Thus the V-to-F units may be located close to the voltage source whose output

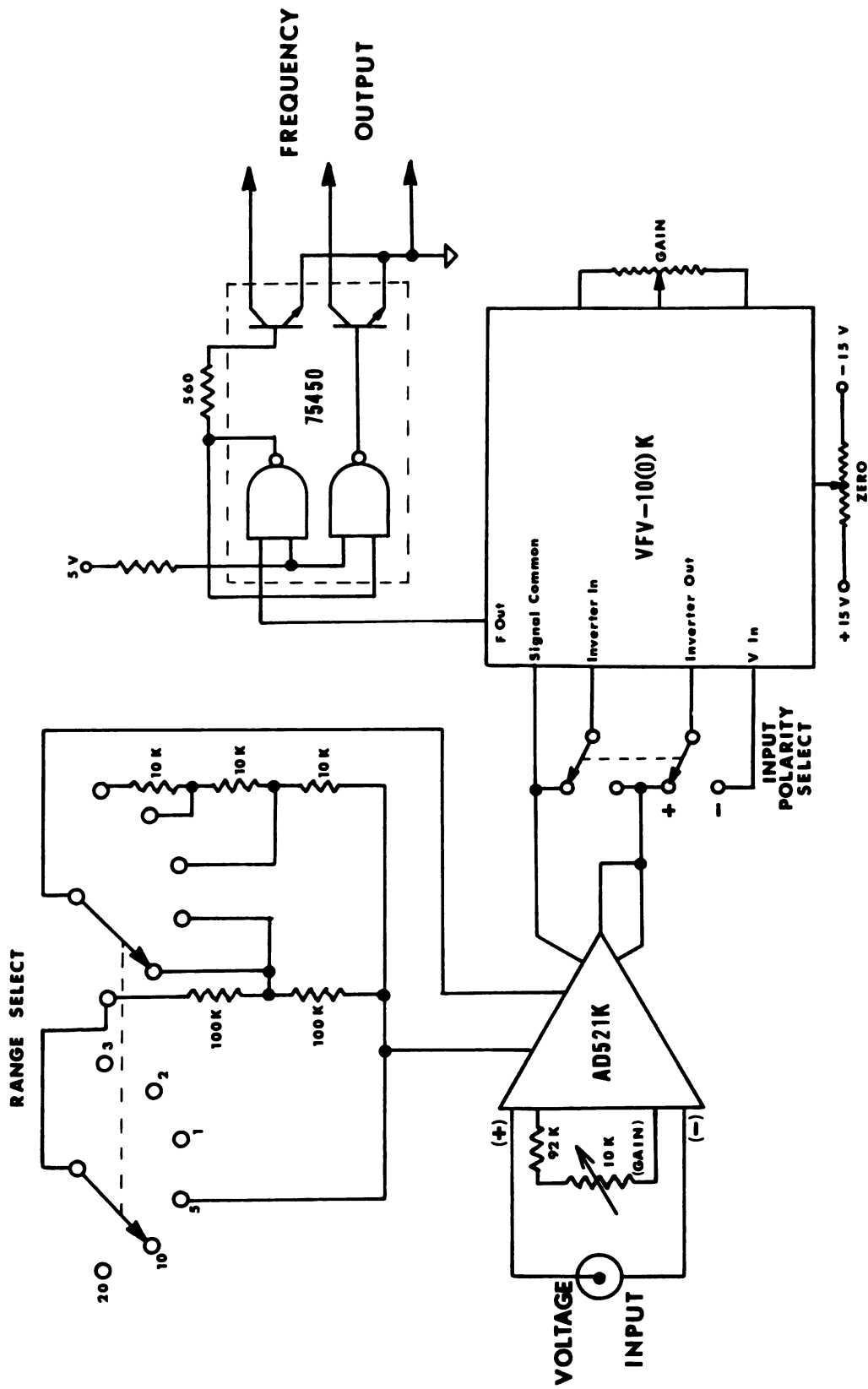


Figure IV-10. V-F Converter Module Circuit.

is being measured.

The linearity of the instrumentation amplifier-converter units was tested by integrating the output pulse over precise time periods ranging from 10 to 100 seconds at various input voltages. The input voltages were measured with an ESI 300 Portametric bridge [108]. Figure IV-11a shows the results from one such study, with the 10 KHz converter on the -10V range. The plot shows deviation from the expected count rate given by the equation

$$\text{ECR} = (-1000.00)V + 0.0 \quad \text{IV-1}$$

expressed as a percentage of full scale (10,000) as a function of input voltage. A least squares fit of a straight line to the same data gave a "best fit" line of

$$\text{count rate} = (-999.164)V + 2.239 \quad \text{IV-2}$$

A plot of deviation from this equation as a function of input voltage is shown in Figure IV-11b. A second set of data was obtained after adjusting the zero and gain potentiometers of the instrumentation amplifier and converter more carefully. The deviations from Equation (IV-1) for this data are shown in Figure IV-11c. (Note the ordinate scale change for the curve.)

A number of sets of data such as those plotted in Figure IV-11 were obtained with different input polarity and range settings on the converter units. The detailed shape of such deviation curve differed

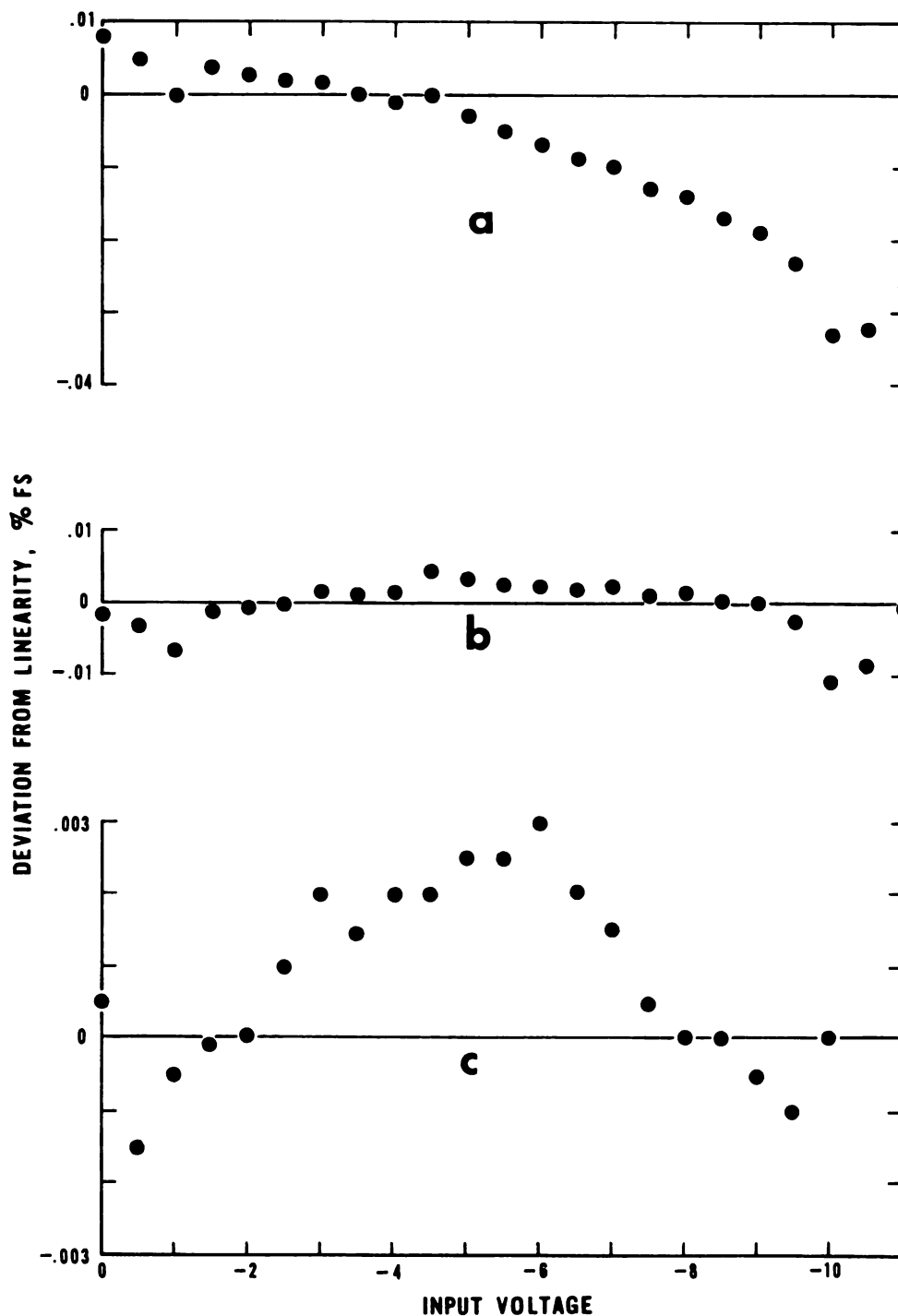


Figure IV-11. Linearity of the V-F Converter Module. The module was on the -10V range for all measurements shown here. Plots a and b were obtained before final adjustment; deviations are from equations IV-1 and IV-2, respectively. Plot c (note scale change) shows deviations from equation IV-1 after more careful adjustment.

from setting to setting, but it was always possible to adjust the maximum deviations to less than 0.005% for the 10 KHz unit and 0.05% for the 100 KHz unit.

The stability of the current measuring system as a whole seems to be limited by its sensitivity to temperature. The output of the Keithley 417, for instance, changes by about 0.16%/°C near full scale [110]. The V-F converter units appear to be about equally sensitive to temperature. Unfortunately the temperature drift of both the Keithley and the converter are in the same direction, so that the two add rather than cancel. The temperature in our laboratory is fairly constant during the day, but at night it tends to oscillate. The oscillations are approximately sinusoidal having a 2-4°C peak-to-peak amplitude and a period of 1-2 hours. The effect on the current measurement system is easily noticed when long integration times are used. As an example, Figure IV-12 shows the recorded ion intensity as a function of time for collisionally ionized CO_2^+ ions just below the true CO_2 ionization threshold. The actual behavior of the collisional ionization, determined in other measurements, is shown as a solid line. The fluctuations in the point-wise plot reflect the temperature variations discussed above. The effects are especially noticeable here because the ordinate full scale in the figure is only about 3% of the converter full scale output. Such effects can be corrected for by making appropriate reference measurements, and these data demonstrate dramatically the need for doing so. The data also suggest that significant improvements could be made in the overall system performance by placing the current-to-voltage and voltage-to-frequency converters in a more stable temperature environment.

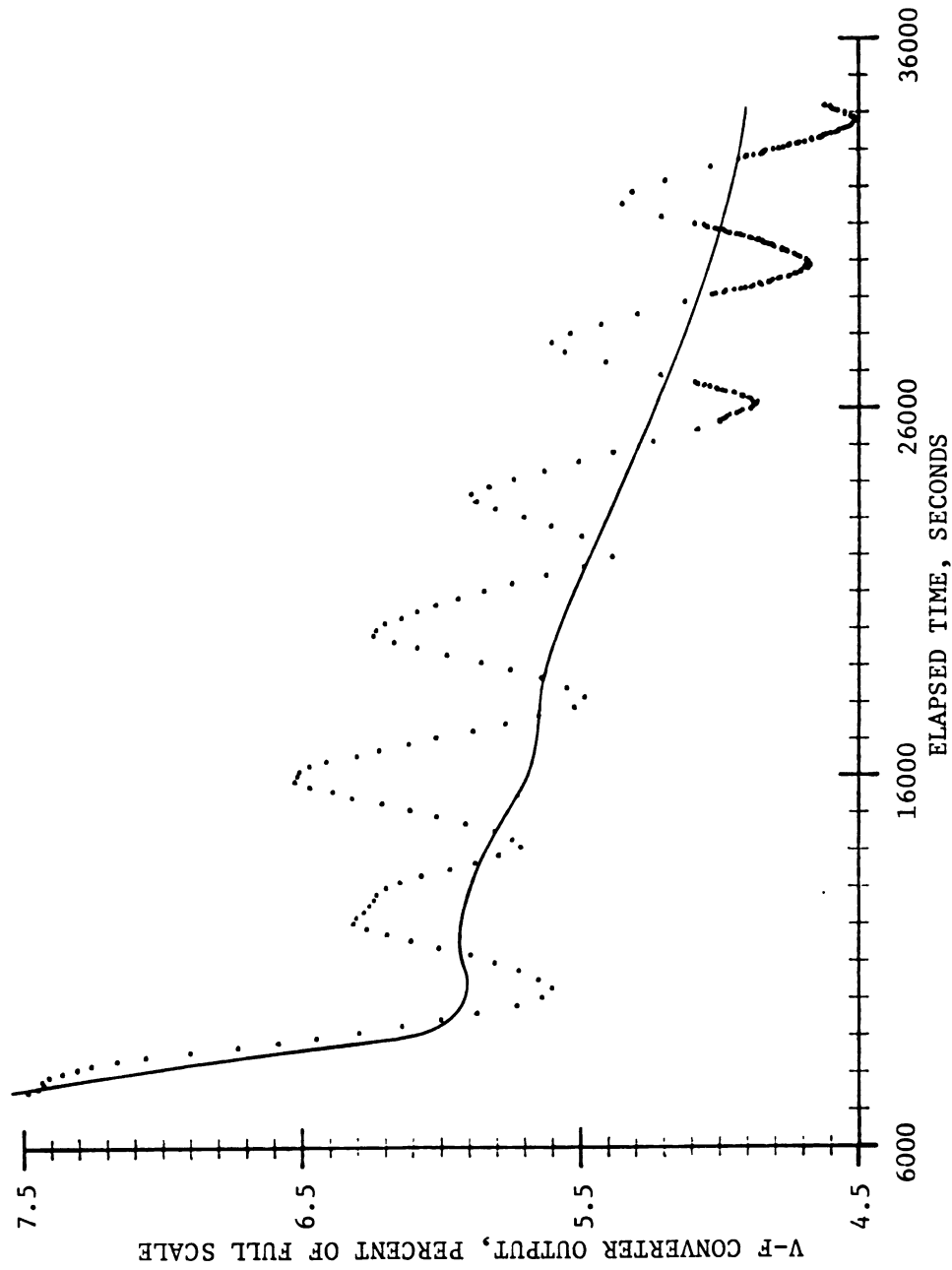


Figure IV-12. Effect of Temperature on the Analog Measurement System. The apparent oscillations in the ion intensity (points) are caused by room temperature oscillations of several degrees. The solid line shows the expected behavior of the ion intensity curve, determined in other measurements.

The V-F systems described here have been used to measure both photon and ion intensities, although the ion counter described in the next section has been used almost exclusively to measure ion intensities since its (recent) completion. All cables between the high voltage power supplies, electron multipliers, and measurement system electronics are coaxial cables in which the outer conductor serves as both shield and common. This has made it difficult to eliminate ground loops, and pick up of electrical noise is a problem on the more sensitive scales. Our laboratory is next to the main mechanical room of the building, and most of the power lines for our half of the building pass through its walls. It is an environment particularly rich in 60 Hz electrical noise. Noise pickup is severe enough that the system cannot be used to measure currents smaller than about 1×10^{-11} amperes, and even that with difficulty. Fortunately, at such low average currents, pulse counting techniques which are practically immune to the 60 Hz noise are entirely adequate.

The effect of the 60 Hz noise is reduced on the less sensitive scales. It usually causes no problem, since the AC component of the voltage waveform seen by the V-F converter is averaged to zero during the integration process, providing that the entire waveform is within the range of the converter. Since the V-F converter input is unipolar, if the lower part of a noisy waveform is allowed to cross zero, the averaging will not be performed resulting in a very poor signal-to-noise ratio and a loss of accuracy. The user must make sure that enough offset is added to the electrometer output to keep the noise signal always above zero.

Perhaps the most serious limitation of the present system is that

the range of the current-to-voltage converter must be set manually. Since it cannot be adjusted during an automated (operator-less) run, it must be set to a range which will keep all signals during the run on scale. If, as is often the case, one feature in the photon or ion intensity curve is much stronger than the rest, then the remainder of the data may have to be acquired using only the lower 10 percent of the electrometer range. While the resolution of the V-F converters, especially with long integration times, is more than adequate even if only 10% of their range is being used, their accuracy is not as good. Many of the errors involved in the measurement (non-linearities, zero drifts, etc.) are typically relatively constant and small when expressed as a percentage of full scale. However, if expressed as a percentage of the measured value, the errors become much larger for measurements near the bottom end of the range. As an example, consider deviations from linearity of the V-F converter. Figure IV-13 shows deviation measurements similar to those described earlier, plotted both as a percentage of full scale and as a percentage of the measured value. The disadvantage of obtaining most of the data at the lower end of the range is obvious; if the experiment were performed manually, the range of the I-V converter would be changed as often as necessary to maintain the lowest possible error. This cannot be done with the present system without losing the advantages of automated operation. However, a new current measurement system incorporating an I-to-V converter with remotely-selectable range is in the process of being designed.

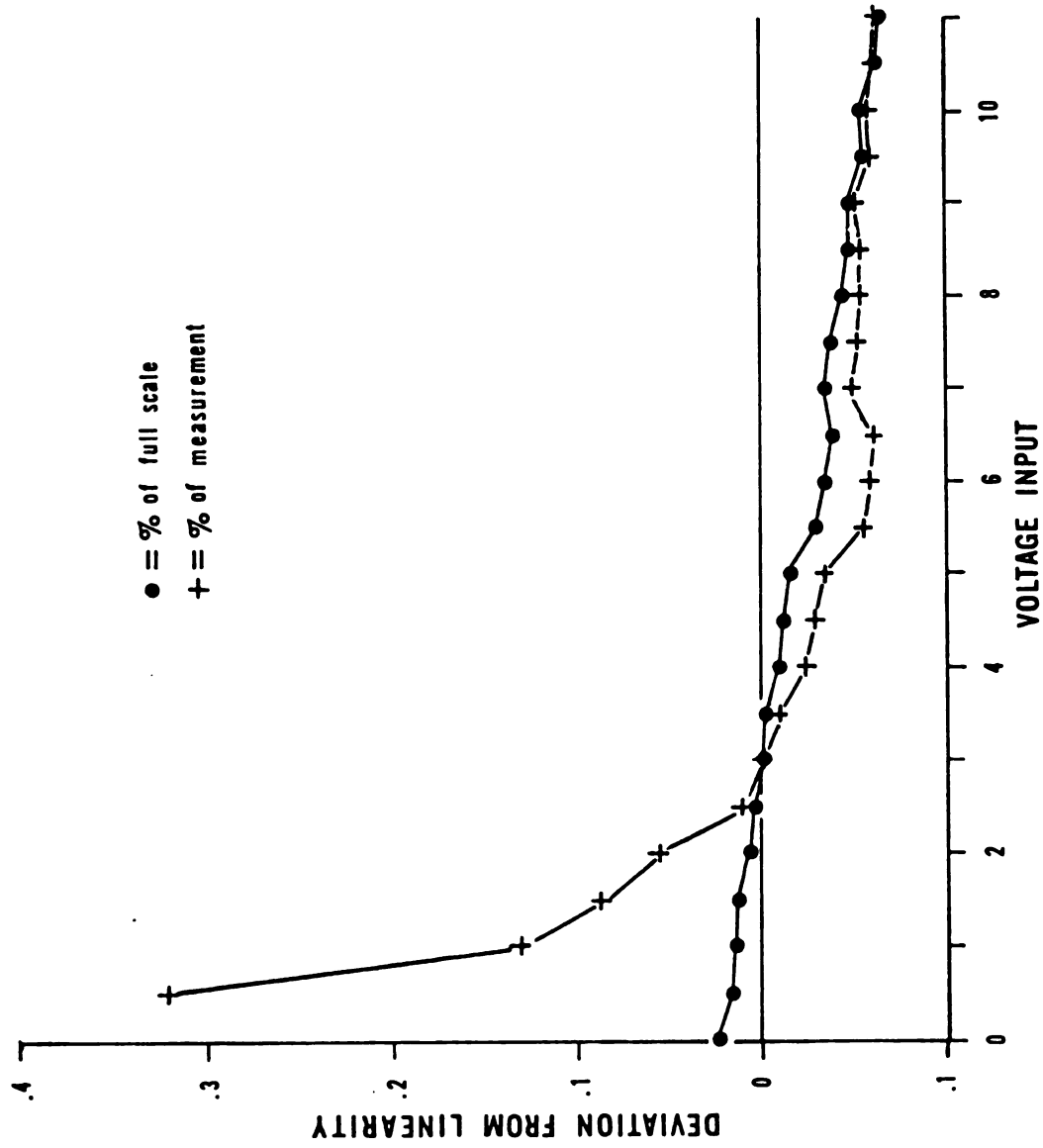


Figure IV-13. V-F Non-linearity as a Percentage of Full Scale and of Reading. V-F range, +10 V; assumed transfer function is Counts = 1000 V + 0.0.

F. The Pulse Counter

1. Introduction

Electron multipliers are widely used as particle flux transducers. Whether incorporated into a photomultiplier for measuring photon flux, or used "bare" for detecting electron, ion, or VUV photons, the output of the multiplier is usually measured with either pulse counting (PC) or direct current (DC) techniques. When the particle flux is extremely low, such as in PIMS experiments where only a few ions per second are generated, pulse counting techniques must be used [89]. At the other extreme, when the flux is very high, pulse overlap counting loss precludes use of pulse counting, and DC measurement techniques must be used [101]. Examples of this latter situation include measurement of photon intensity in PIMS, or in analytical absorption spectroscopy with bright light sources. In between these two extremes is a range where either technique may be used. The relative merits of PC and DC within this range depend on many factors, and the reader is referred to the paper by Ingle and Crouch [101], which examines these factors in detail. The exact limits of the range in which either PC or DC may be used are quite variable, since they depend on the characteristics of the PC and DC measurement systems³ as well as the accuracy and precision required of the measurement. For instance, the upper limit of PC utility depends on the speed of the overall pulse counting system, as well as the magnitude of pulse overlap

³ When making an evaluation of these limits, it is important to use the characteristics of the entire PC or DC measurement system, since the capabilities of each often depend as much on the transducer and power supplies as on the actual PC or DC electronics.

loss that can be tolerated.

In spite of the fact that DC measurement systems which are adequate for high flux measurements are readily available, there has been a continued interest in developing pulse counting systems which can operate at ever higher count rates, for several practical reasons. For instance, pulse counting systems often have significant signal-to-noise ratio and stability advantages over DC systems at equivalent fluxes [101]. Furthermore, PC is an inherently digital technique; PC measurements require fewer data domain conversions than DC measurements, and are thus less susceptible to the inevitable errors which accompany such conversions [101,111]. The development of faster PC systems not only extends these advantages to measurements made at higher fluxes, but also has important consequences in terms of cost and convenience. In order to retain the advantages mentioned above, one should use PC whenever possible, and DC only when necessary. If the measurements to be made span a range of fluxes which extends above the upper limit for the PC system, this may require switching back and forth between measurement systems rather frequently, which is often extremely inconvenient or even impossible (e.g., during an unattended experiment). Furthermore, high-quality DC current measurement instrumentation is expensive, and although many pulse counter designs have been reported in the literature [112] or are available commercially [113]; those designs which excel in speed, sensitivity and stability also tend to be fairly expensive. It is therefore desirable to use a PC system which is as fast as possible, so as to increase the probability that all necessary measurements (or at least an entire series of measurements) can be made without resorting to DC

techniques.

Unfortunately, no matter how fast the pulse counting electronics are, pulse overlap within the electron multiplier itself will eventually cause unacceptable counting losses.⁴ Although dead-time compensation (DTC) [114] or post-experiment count loss formula corrections (CLFC) [115,116] can be used to extend the linearity of the system well into the region of pulse overlap, neither can be employed without compromising other potential advantages of the PC technique [101,115]. Optimum use of PC requires that the user be able to evaluate the trade-offs involved. Even if DTC or CLFC are employed, there may well exist some upper limit beyond which the use of DC measurement techniques becomes mandatory, and economic considerations then make it desirable that the pulse counter employed be as inexpensive as possible, while retaining high speed, sensitivity, and stability.

A block diagram of the pulse counter developed for use with the MSU PIMS instrument is shown in Figure IV-14. The digital counter (which is also used with the V-F converter modules described in Section E of this chapter) is described in Section G. The real-time clock used by the computer for timing purposes is described in Appendix C. The remainder of the circuitry (amplifier, discriminator, prescaler and line driver) is described and characterized in this section. For the sake of brevity, these four units will henceforth be termed the "pulse counter", although it should be realized that some sort of digital counter and timer are necessary to make a complete pulse counter.

⁴See Part 2 for a fuller discussion of this point.

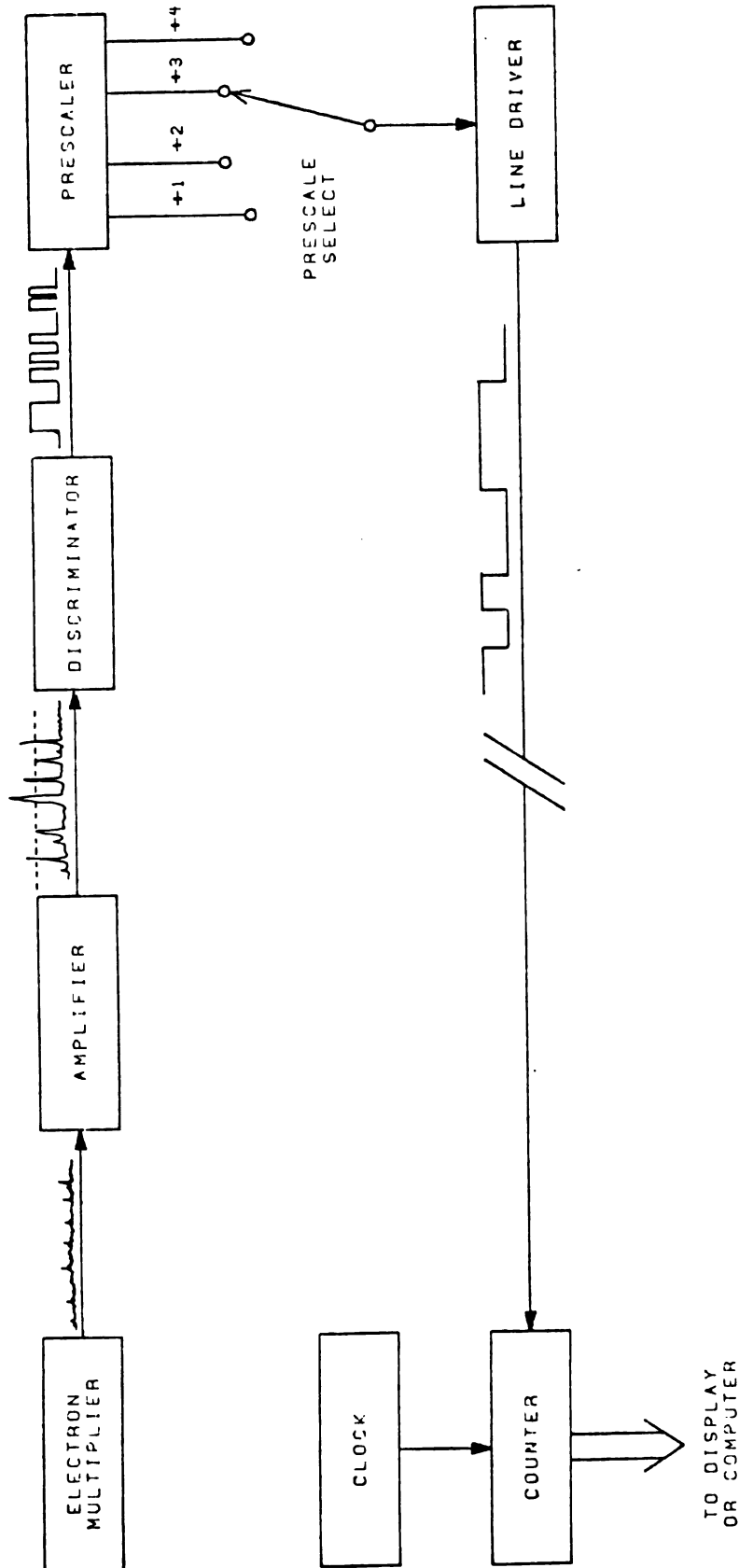


Figure IV-14. Block Diagram of the Pulse Counter.

The pulse counter described here has performance characteristics approaching those of state-of-the-art pulse counters, but it can be built for a small fraction of the cost of a commercial pulse counting system. It makes pulse counting an affordable alternative to DC measurement techniques. The pulse counter has high sensitivity as well as easily adjusted gain and discriminator levels, which are very stable as a function of time and temperature, and independent of pulse rate. These properties make it possible to perform an accurate characterization of the entire pulse counting system relatively quickly, and under the exact conditions to be used in the actual measurements. This provides the user with the information necessary to determine the optimum measurement system parameters for any given experiment. For instance, the user can easily obtain quantitative information on the benefits and disadvantages of using DTC for a given application, thus considerably simplifying the decision about whether to use DTC (and if so, how much). Furthermore, the discriminator level is voltage programmed and the amount of pre-scaling can be selected remotely. These features create the possibility of real-time computer control of the measurement system parameters, which could be automatically optimized to meet changing experimental conditions.

A pulse counting system consisting of the pulse counter described here and the RCA 8850 photomultiplier [69] used as part of the sodium salicylate photon transducer, described in Chapter II, has been thoroughly characterized. This system has attained maximum counting rates of greater than 35×10^6 photons sec^{-1} . With optimum dead time compensation, the system is linear to within $\pm 0.5\%$ up to a counting rate of 12 MHz, which

corresponds to a DC current of about 3×10^{-6} amperes. The results of these studies also indicate that, at least for this photomultiplier and for counters of similar sensitivity, the stability of the pulse counting electronics is more important than often assumed, but that the loss of stability inherent in the use of DTC is not necessarily as serious as often assumed. Similar, though less thorough, characterizations have been performed on systems employing an RCA 1P28 photomultiplier [69] and a Johnston MML electron multiplier [91]. Results for these systems were usually very similar to those for the 8850 system, and will be alluded to only briefly.

The circuitry of the pulse counter is described in Part 2 of this section, and its performance is characterized in Part 3. Part 4 contains several recommendations for modifications which should improve the performance of the pulse counter even further. The experiments used to characterize the 8850 system are described in Part 5, and their results and implications are discussed. Finally, this rather lengthy section is summarized in Part 6.

2. The Pulse Counter

1. Design Requirements:

The pulse counter described here was intended for use not only with the PIMS instrument, but also with a wide variety of computer-interfaced spectrometric instrumentation in other laboratories as well as our own. The primary design goals were high speed and sufficient sensitivity for use with common photomultiplier tubes, such as the 1P28.

The speed of the counter is important since pulse overlap in the

counter is usually the limiting factor in determining the maximum count rate at which accurate data can be obtained. State-of-the-art counters are capable of counting periodic pulses at up to about 100 MHz, i.e., they can resolve pulses separated by as little as 10 nsec. Unfortunately, the photon arrival rate is random, and a significant fraction of the pulses are separated by less than 10 nsec, even at much lower average rates than 100 MHz. For example, a 100 MHz counter with its discriminator set to pass all the pulses is expected to miss about 1% of the pulses at an average count rate of only 1 MHz.

Faster pulse counting electronics could help the pulse overlap problem only up to a point. Even relatively fast electron multipliers operated at moderate voltages have anode pulse rise times of about 1.5 - 3 nsec [117]. Fall times depend on the impedance and capacitance of the connection between the anode and the input to the pulse amplifier, but will certainly be longer than the rise times, and the FWHM of the pulse observed at the counter input will thus be on the order of 2-3 nsec. The width of the pulse near its base (where the discriminator is usually set to obtain best stability) might be expected to be several times the FWHM, or roughly 10 nsec. Thus, a 100 MHz pulse counter with a 10 nsec dead-time approaches the practical limit set by pulse overlap within the electron multiplier itself, and count rate improvements of more than a factor of 2 or so are probably not possible with the great majority of currently available photomultipliers. Therefore, 100 MHz was adopted as a speed goal which would be adequate if met, and which was felt to be quite ambitious enough.

The sensitivity of the counter is important for two reasons.

Pulse heights from a typical electron multiplier have a random and broad distribution. If the counter is not sensitive enough to count the smallest pulses, then some of the signal is wasted, increasing the measurement time or lowering the signal-to-noise ratio of the measurement. Furthermore, the stability often assumed for pulse counting systems is based on the assumption that there is more than enough sensitivity to count all the anode pulses. If this is true, then neither changes in the discriminator setting of the counter nor variations in the pulse height distribution due to power supply fluctuations or changes in the electron multiplier will alter the count rate, and the system will exhibit very good stability.

Ease of use and restricted space within the photomultiplier housings dictated that the pulse counter not be mounted inside the housing, but rather in a separate enclosure connected to the electron multiplier or photomultiplier with a short length of common 50 ohm coaxial cable. To minimize reflections, an anode load resistor of 50 ohms must be used. The amplitude of the voltage pulses generated across this resistor by the anode current pulses determines the sensitivity that the rest of the circuit must have. A Tektronix 7623A storage oscilloscope [98] was used to observe the voltage pulses developed across a 50 ohm load by anode current pulses from the 1P28, 8850 and MM1 multipliers. Of the three multipliers, the 1P28 produced the smallest pulses when run at normal operating voltages (<1000 V), the results shown in Figure IV-15 being typical. These measurements indicated that, although many pulses were as large as 5-10 mV, many were smaller than 1 mV. Therefore, a design goal of sub-millivolt sensitivity was adopted.



FIGURE IV-15. Anode Pulses from a 1P28 Photomultiplier. 50Ω load.
Vertical: 5 mV/div. Horizontal: 5 nsec/div. PMT voltage:
900 V.

To minimize broadening of the anode pulse by excess cable capacitance, the pulse counter must be located near the multiplier. Often, however, the digital counter is located some distance further away (about 50 feet in the case of the PIMS instrument/computer). To increase the versatility of the counter, it must therefore include circuitry capable of transmitting the signal to the digital counter over long distances (e.g., 100 feet or more) with good noise immunity. Commonly available integrated-circuit line drivers are incapable of responding to pulses as short as those expected at the output of the discriminator (ca. 10 nsec) [109], so that some sort of signal conditioning is required between the discriminator and line driver. A common solution to this problem is simply to insert a monostable "pulse stretcher" between the two, to make certain

that the pulses at the line driver input are slow enough for it and the succeeding circuitry to handle. However, this approach limits the overall speed of the system to the maximum frequency at which the monostable can be re-triggered.

Instead, the pulse counter described here uses a fast binary counter between the discriminator and line driver to prescale the discriminator output, as shown in Figure IV-14. The inclusion of such prescaling has two effects on the pulse train observed at the line driver input, which can be seen qualitatively in Figure IV-16. After division by eight, the average pulse rate is reduced by a factor of eight, so that succeeding circuitry does not have to be so fast. Perhaps even more important is the fact that the prescaling exerts more of an influence on the high-frequency end of the pulse arrival time distribution than on the low frequency end. This has the effect of reducing the "randomness" of the pulse train--the maximum instantaneous pulse rate is much closer to the mean rate after prescaling than before, so that the line driver and succeeding circuitry need to be able to respond to pulse rates only a little higher than the average rate.

While prescaling does not limit the maximum count rate, it does limit the maximum attainable count resolution. At high count rates, when prescaling is most needed, the loss in resolution should be negligible, but at low rates, it may be important. Fortunately, at low count rates, less prescaling is needed and, if the amount of prescaling can be changed, the count resolution can be improved when necessary.

ii. Preamplifier/Discriminator Circuit:

The preamplifier-discriminator portion of the pulse counter is

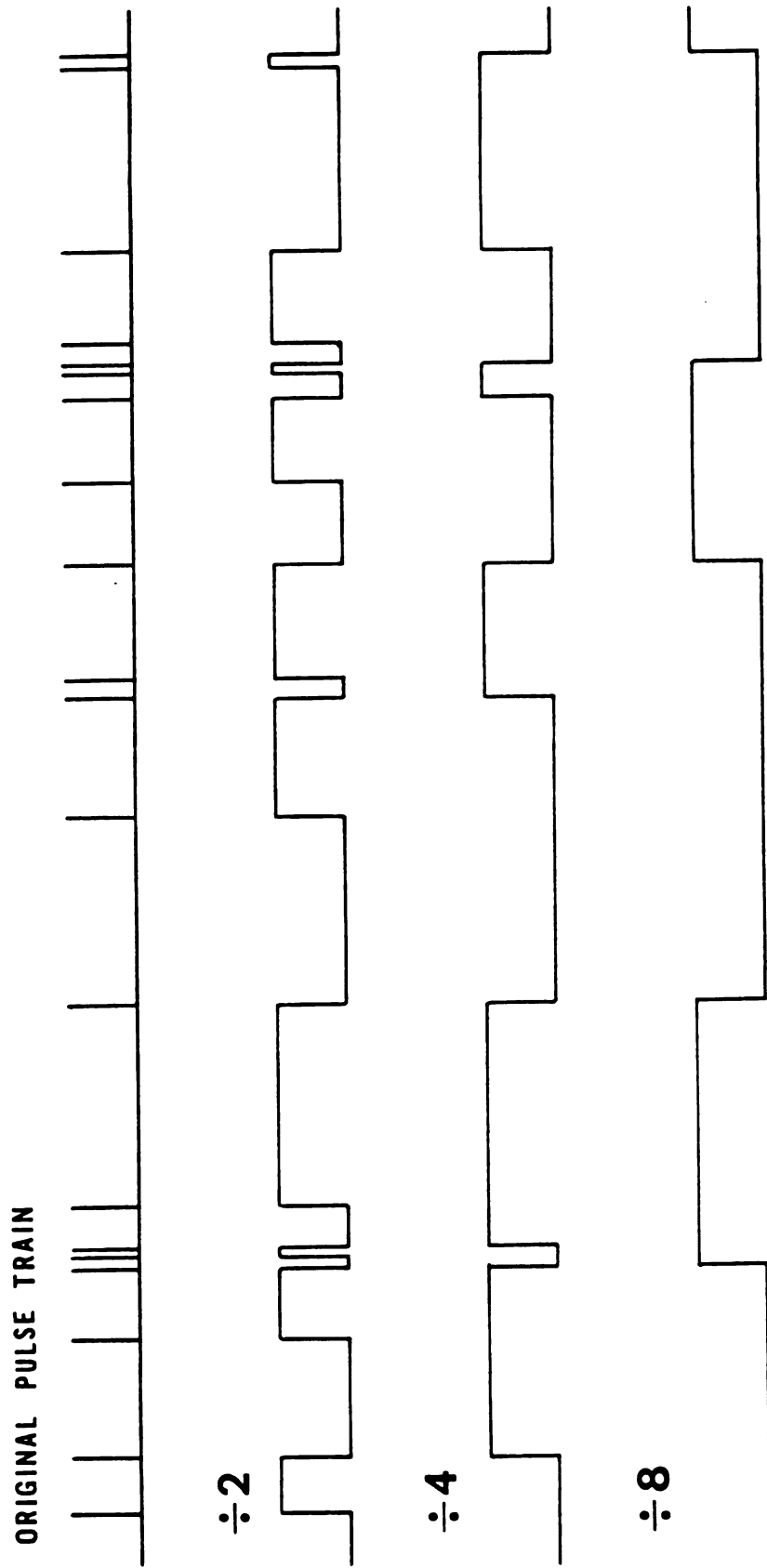


Figure IV-16. The Effect of Prescaling on a Random Pulse Train.

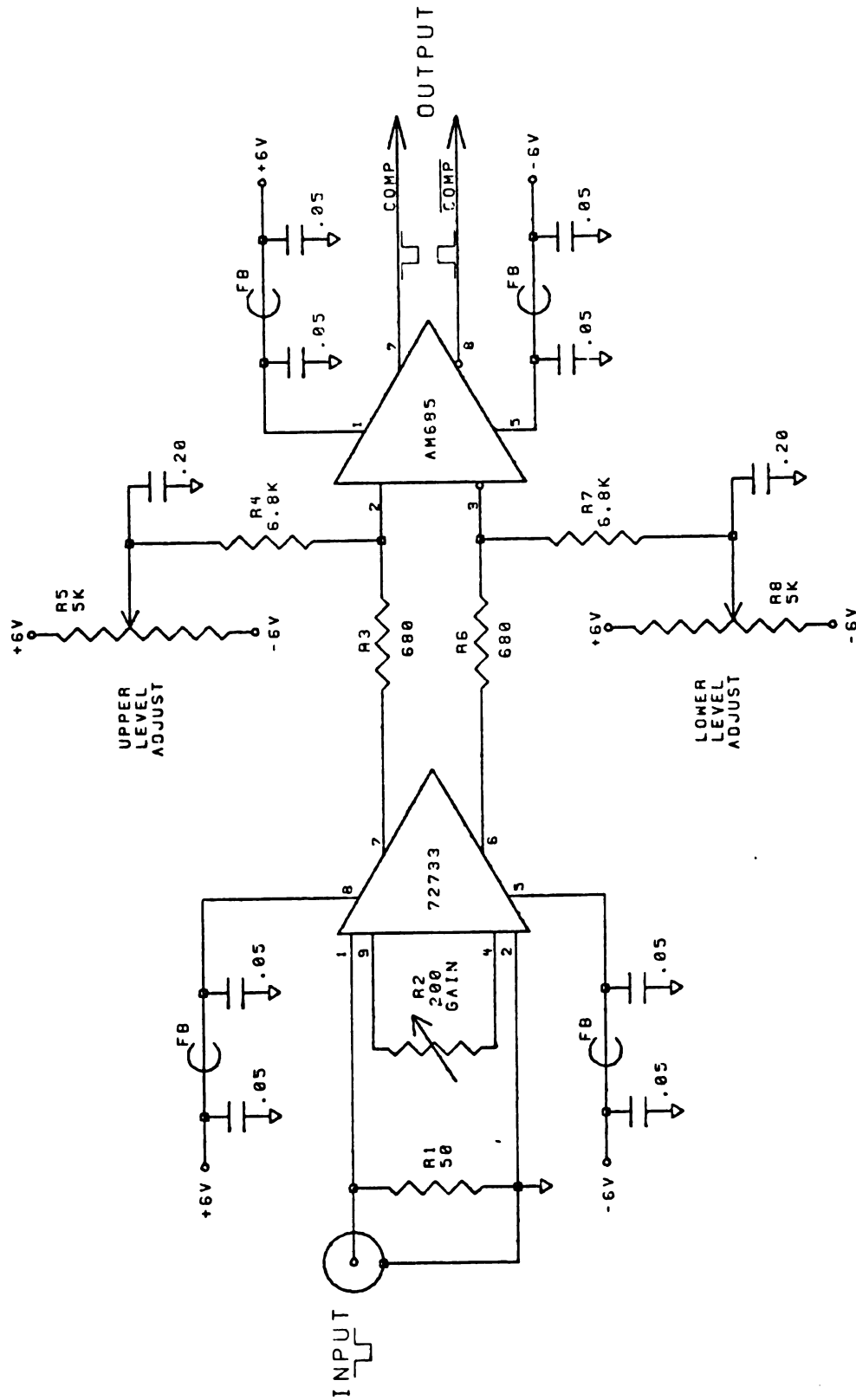


Figure IV-17. Preamplifier / Discriminator Circuit. All resistor values in ohms, all capacitor values in microfarads. FB = ferrite bead.

very simple, being based on only two integrated circuits, as shown in Figure IV-17. A high-speed comparator (Advanced Micro Devices type AM 685 [118]) with differential inputs and emitter-coupled logic (ECL) outputs is used as the discriminator. The preamplifier is a Texas Instruments SN72733 adjustable gain, high-speed differential video amplifier⁵ [119]. When the counter is in its quiescent state (both amplifier inputs at essentially zero volts), the amplifier outputs have a common mode voltage, V_{Acm} , which is typically about 2.9 volts, and are separated by an offset voltage, V_{Ao} , which depends on the gain of the amplifier, but is typically 0.6 V at a gain of 400. A negative pulse at the input of the amplifier causes the outputs to move closer together, and for sufficiently large input pulses, the outputs will cross (see Figure IV-18).

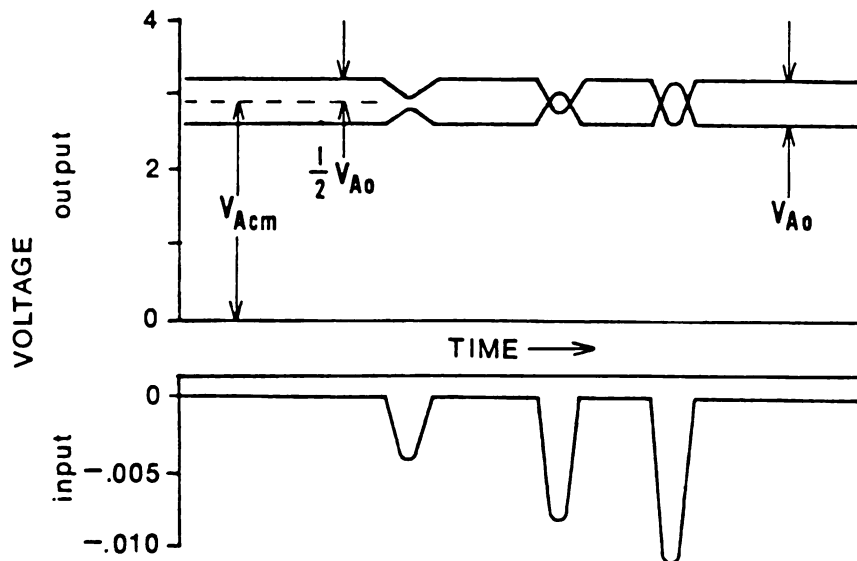


FIGURE IV-18. Response of Amplifier to Input Pulses.

⁵Also manufactured by Motorola [120], and perhaps others.

A quiescent comparator input common mode voltage, V_{Ccm} , and an input offset voltage, V_{Co} , can be defined which are analogous to V_{Acm} and V_{Ao} . The comparator is triggered whenever its inputs cross, i.e., when the input pulse to the comparator is larger than V_{Co} :

If the amplifier outputs are connected directly to the comparator inputs, then $V_{Acm} = V_{Ccm}$ and $V_{Ao} = V_{Co}$. In this case, the threshold of the counter is given simply by the minimum pulse amplitude necessary to make the amplifier outputs cross, or V_{Ao}/G , where G is the gain of the amplifier. Though conceptually simple, such a design is undesirable for several reasons. An amplifier gain of more than 600 is necessary to obtain sub-millivolt sensitivity, and the maximum gain of the amplifier is only 400. The only way of adjusting the threshold of the counter is to change the gain of the amplifier, which also changes its bandwidth and temperature characteristics. Finally, V_{Acm} is near the maximum allowed comparator input voltage of +4V; input pulses only moderately larger than threshold could easily cause one amplifier output to exceed the limit, resulting in poor overload characteristics.

All of these problems can be avoided by using capacitive (or AC) coupling between the amplifier and comparator, as shown in Figure IV-19, which allows V_{Ccm} and V_{Co} to be adjusted independently of V_{Acm} and V_{Ao} . However, AC coupled circuits exhibit baseline shift which changes the effective threshold level at higher count rates [121]. The various diode clamp techniques which are commonly used to limit baseline shift do not completely eliminate it [122], and design of baseline correction circuits suitable for use at 100 MHz is a problem which is far from trivial.

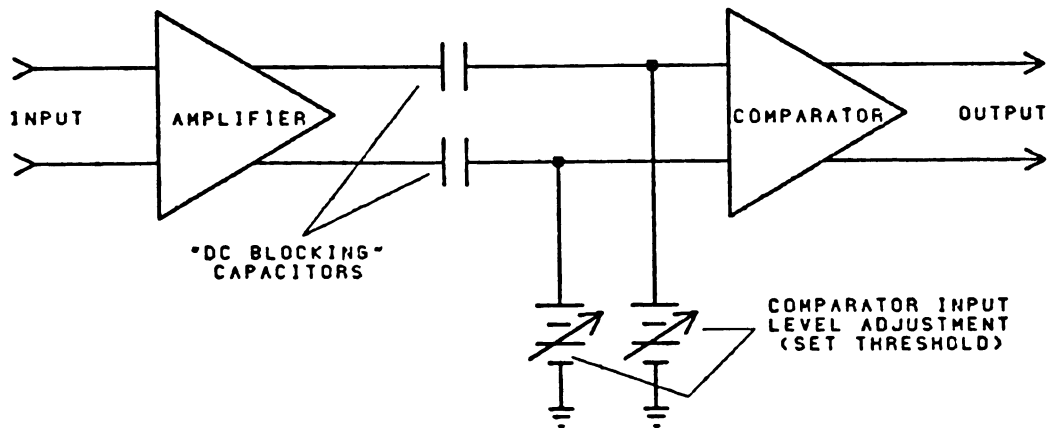


FIGURE IV-19. AC Coupling of Amplifier and Comparator.

This pulse counter, therefore, retains DC coupling, but through two simple resistor networks as seen in Figure IV-17. Resistors R3 and R4 (and similarly R6 and R7) act as simple voltage dividers, transmitting a large fraction of the amplifier output pulse to the comparator input. However, R5 and R8 allow the voltage at one end of each divider to be varied independently over a range of +6V to -6V, which in turn allows independent control of the quiescent values of the comparator input voltages. With the component values shown, V_{Ccm} can be reduced to approximately 1V, which yields much improved overload characteristics. V_{Co} can be adjusted continuously within a range of approximately ± 300 mV. A typical value might be 60 mV, resulting in a tenfold gain in sensitivity compared with the directly coupled amplifier and comparator. Both V_{Ccm} and V_{Co} can be varied independently of the gain of the amplifier, which can be set so as to obtain the best compromise between bandwidth and temperature stability (see Part 3). Furthermore, there is no

baseline shift at any count rate.

The comparator is essentially a high-gain, high-speed amplifier, and as such is particularly prone to oscillation. Oscillations will occur when the differential input voltage remains in some critical range (CR) near zero for longer than the propagation delay of the comparator [123]. The minimum slew rate that the input signal must have in order to avoid oscillation near the comparator triggering point is simply the width of the CR divided by the propagation delay of the comparator. Fortunately, the slew rate of the amplified anode pulses is much higher than the minimum necessary slew rate, so that hysteresis (which can only limit sensitivity and maximum speed) is unnecessary.⁶

The tendency to oscillate does limit the overall sensitivity of the counter, however, since the CR limits the minimum useful value of V_{Co} . The width of the CR (which may be as small as a few mV or as large as several hundred mV) depends on the (fixed) gain of the comparator, the extent to which the comparator's output is coupled back to its input via stray capacitance, and the source impedance seen by the comparator at its inputs [123]. Careful construction techniques can help minimize the stray capacitance between comparator output and input. The use of a comparator with ECL outputs is also advantageous, since the voltage swing for an ECL logic transition is only about one-fifth of that for a TTL logic transition, and thus there is less signal to couple back to

⁶The pulse counter was in fact constructed so that various amounts of comparator hysteresis could be added, to study their effect. Hysteresis was found necessary for preventing oscillation only with input frequencies less than 1 MHz. Since anode pulses with FWHMs less than 50 nsec have no frequency components below about 20 MHz, hysteresis is unnecessary for electron multiplier pulse counting application, no matter how low the input count rate.

the inputs. The width of the CR is also found to be smallest when the comparator is driven by a balanced source with an output impedance of about a thousand ohms or a little larger.⁷ The differential outputs of the video amplifier used in this design provide a balanced source for the comparator, although they have a low output impedance (about 20 ohms [124]). The coupling resistors R3 and R6 serve not only to provide easily adjustable V_{Co} and V_{Ccm} , but also to increase the source impedance seen by the comparator inputs.

The inclusion of R3 and R6 has one major disadvantage. Every pulse from the amplifier must charge up the input capacitance of the comparator through these resistors and, if too large a resistance is used, the resulting RC time constant becomes the limiting factor in the frequency response of the counter. The value of 680 ohms was chosen as a compromise between high frequency response (>90 MHz) and small critical range (≈ 30 mV), or high sensitivity. Different values for these resistors could improve either of these characteristics at the expense of the other.

iii. Prescaler/Linedriver Circuit:

The prescaling and line driver circuits of the pulse counter are shown in Figure IV-20. The signal from the comparator is converted into a TTL level signal ($\div 1$) which is used to drive the light emitting diode (LED) that indicates the state of the comparator (it is on when the comparator is in its quiescent state). Since it is possible to make V_{Co} negative, which results in a "permanently triggered" comparator, the STATE LED is particularly useful when adjusting the threshold level

⁷ This applies to the metal can package. The DIP package may have its smallest CR at different input impedances.

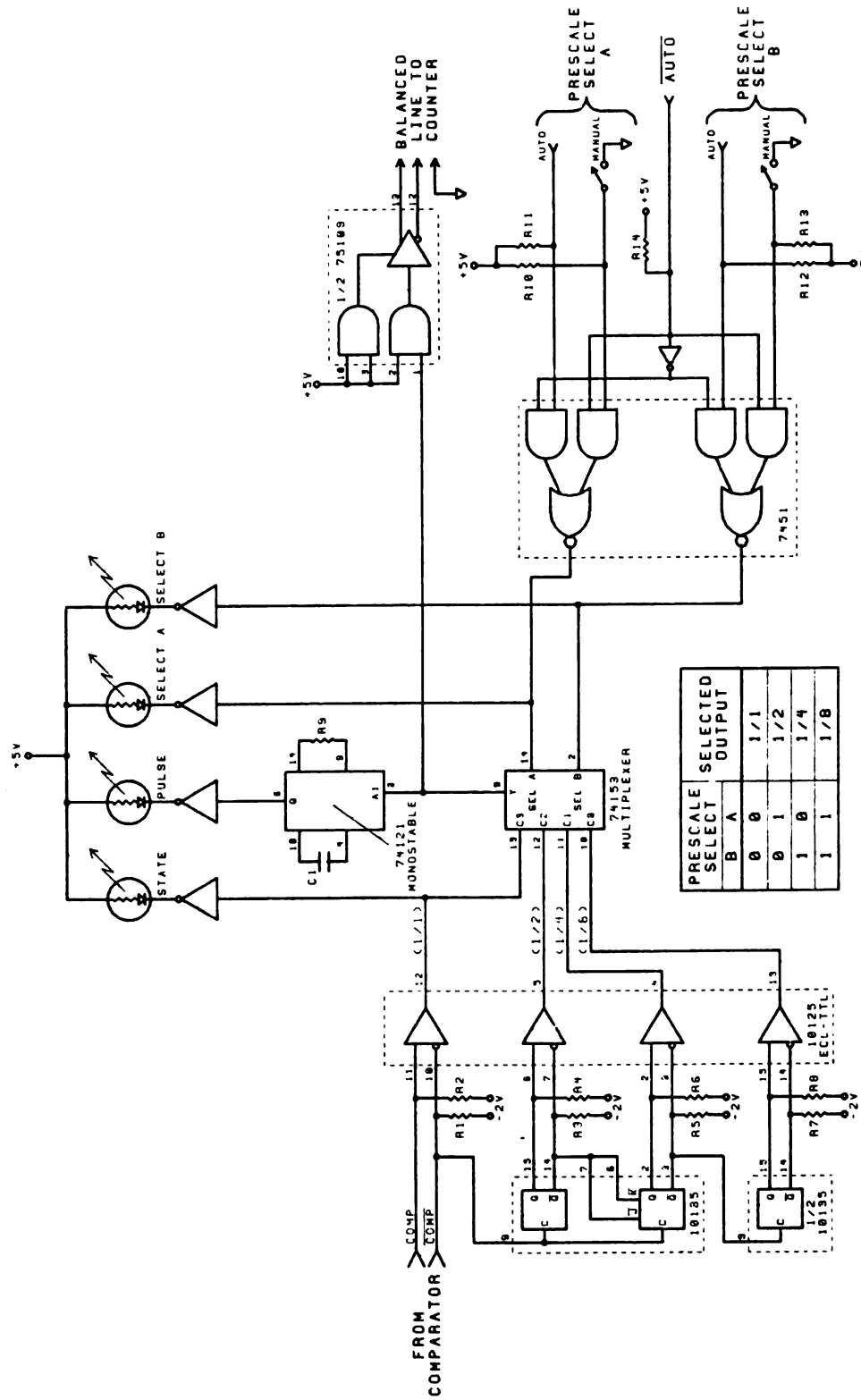


Figure IV-20. Prescaler / Line Driver Circuit. Pulldown resistors (R1-R8) are 50 Ω , pullup resistors (R10-R14) are 3K Ω . Values of R9 and C1 are not critical (see text).

of the pulse counter.

The output of the comparator also goes to three fast ECL flip-flops connected as a binary counter, which divide the pulse stream from the comparator by 2, 4 and 8. The output from each stage of the counter is converted into a TTL level signal. A 74153 multiplexer passes one of these signals to the 75109 differential current-mode line driver, which transmits the pulse train to the digital counter described in Section G. The signal selected by the multiplexer also triggers a monostable which flashes the PULSE LED. R9 and C1 are not critical, being adjusted so that the pulse width from the monostable is just wide enough (a few msec) to create a visible flash from the LED. The PULSE LED is used to indicate visually whether pulses are being detected by the counter, and can even be used to judge roughly how many, at low count rates.

The amount of prescaling used (i.e., which signal is transmitted by the multiplexer) can be determined either manually or under computer control. When the $\overline{\text{AUTO}}$ line is open or at +5V, the manual prescale select switches, SW1 and SW2, control the multiplexer. When the $\overline{\text{AUTO}}$ line is grounded by circuitry in the dual 24-bit counter (see Section G), these switches are disabled, and the multiplexer is controlled instead from the computer via the remote prescale select lines. Two LEDs are used to indicate the status of the multiplexer control lines.

The entire circuit of the photon counter is designed so that only two main supply voltages (+6V and -6V) are needed. The auxiliary voltages needed for the TTL and ECL integrated circuits are derived on the circuit board using the simple diode circuits shown in Figure IV-21.

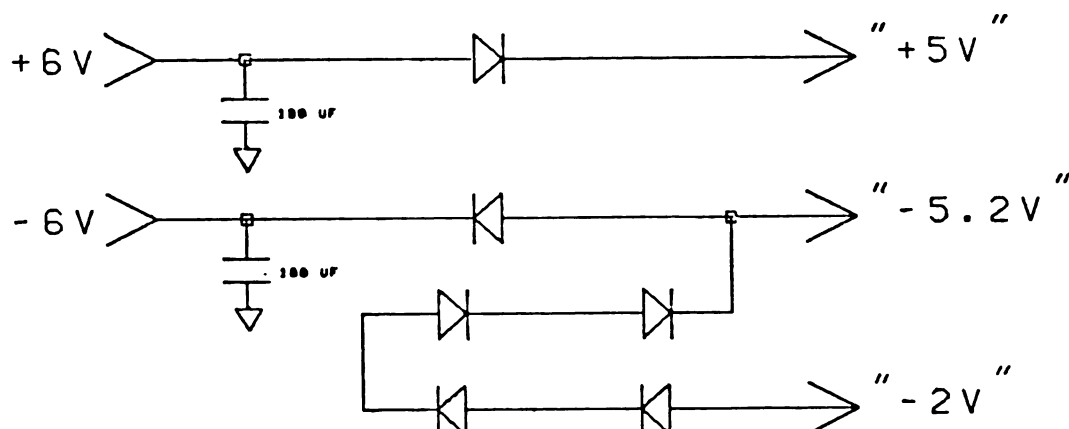


FIGURE IV-21. Auxiliary Voltage Generation. All diodes are 1N4001.

iv. Construction

When working with high frequency, low level signals, the construction methods used are as important as the rest of the circuit design, and some of the details of construction found to be necessary with the pulse counter are summarized here.

The entire circuit (except for the power supply) is built on a single, 4" x 9", double-sided printed circuit board, on which as complete a ground plane as possible is maintained. Since the amplifier section of the circuit is very sensitive to high frequency signals, extra precautions are necessary to shield the amplifier and comparator from both internal and external noise sources. For instance, fast logic edges (such as normal TTL transitions) are very rich in frequencies above 30 MHz, which can be radiated very easily. It was found necessary to place a metal shield across the circuit board between the amplifier-comparator section and the ECL and TTL logic of the prescaler section to prevent noise radiated

in the latter section from being picked up at the amplifier input. Better results were obtained when the input and output connectors (as well as the PULSE and STATE LEDs) were soldered directly to the circuit board instead of being connected with short wires, since such wires radiate and pick up unwanted noise more easily than does foil on a circuit board with a good ground plane. The main circuit board is shielded by internal partitions from the AC voltages present in both the power supply and temperature control circuits. The DC power to the circuit board is passed through the partitions via small feed-through capacitors [125] which are used to bypass high frequency noise. Ferrite bead and capacitor filters (see Figure IV-17) are used to filter any remaining high frequency noise from the power inputs to both the amplifier and the comparator.

Not only must the pulse counter be mounted in an enclosure which shields it from external radiated noise, but extreme care must be taken to ensure that no noise "sneaks in" on the 115 VAC power line. Many of the early problems with the pulse counter were due to the presence of about 50 mV of high frequency (>80 MHz) noise which has been found to be superimposed on the 115 VAC power lines of our laboratory at all times. In addition, there are occasionally very large voltage spikes (>300 V) whenever electrical equipment such as electric drills, heat guns, etc., are operated nearby. These noise sources resulted in many spurious counts, and were not removed by normal capacitive filtering in the power supply. The pulse counter now includes both a radio frequency interference (RFI) filter [126] and transient suppressors [127] at the power entrance to the enclosure, which successfully eliminate this noise.

Preliminary experiments indicated that the thermal characteristics

of the circuit might be important. The pulse counter therefore is built in an enclosure which is large enough to include a small circulating fan [128] and proportional temperature control circuitry so that the thermal behavior of the counter can be easily studied. The temperature control circuit, which is not described here, was purchased (for \$5) from Poly Paks [129]. Several 1000 ohm, 10 watt resistors in parallel serve as the heat source.

Because the input and output connectors on the front panel are soldered directly to the circuit board, the board is rigidly connected to the panel. Since it was anticipated that numerous modifications might have to be made to the first counter before optimum component values were found, an enclosure was designed which allows the main circuit board and front panel to be easily removed, for quick access to both sides of the circuit board. The completed pulse counter is shown in Figure IV-22. Its overall dimensions are 11 x 3.5 x 7.5 inches. The total cost for all parts and materials in the pulse counter unit shown in Figure IV-22 was less than \$130.

3. Pulse Counter Performance

1. Sensitivity:

Figure IV-23 shows how the threshold of the counter varies as a function of the frequency of a sinewave input signal when the amplifier gain and comparator input offset voltage remain fixed. The data of this figure cannot be used to determine how the counter will behave when the input rate of very short pulses is varied, but they do indicate that the sensitivity of the counter is lower for pulses which have frequency

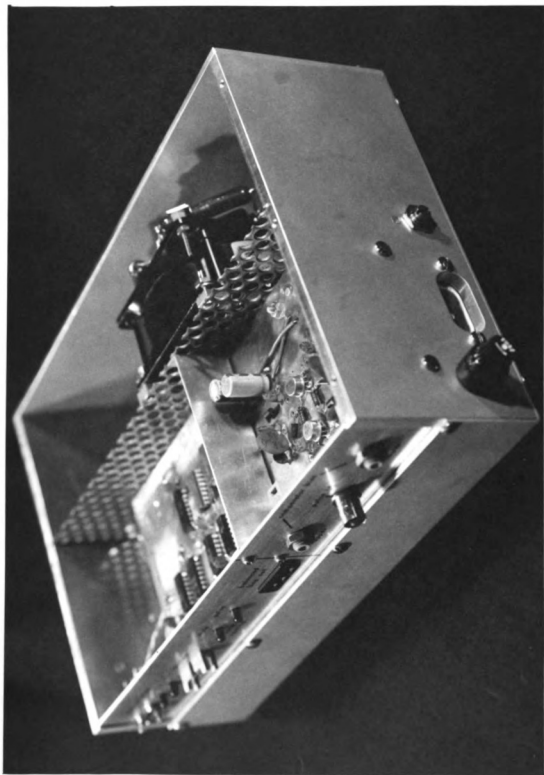


Figure IV-22. The Pulse Counter.

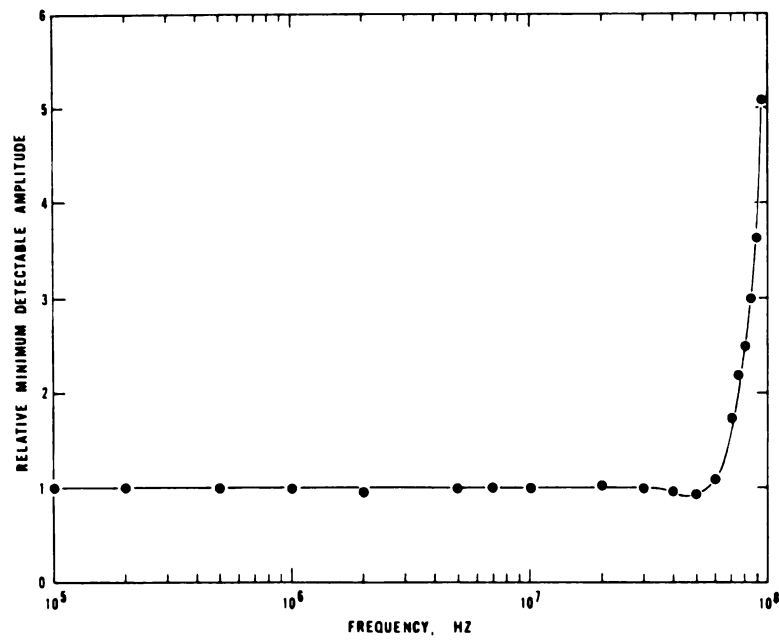


Figure IV-23. Variation of Sensitivity with Frequency.

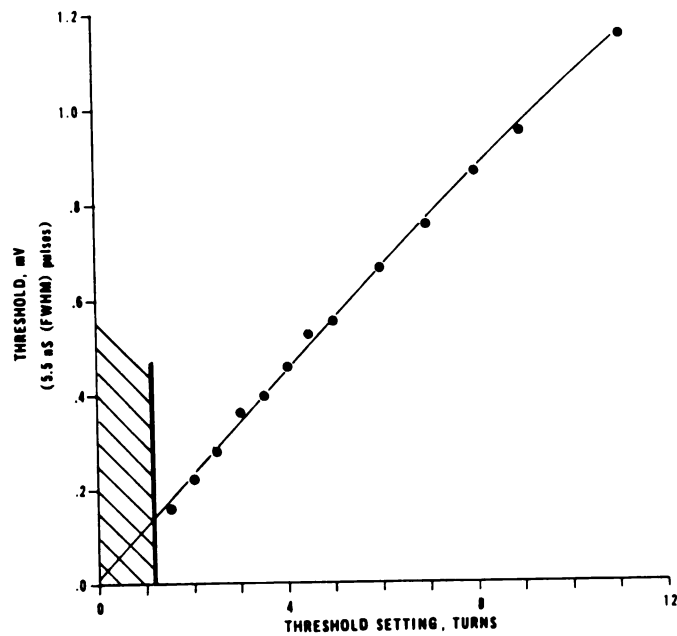


Figure IV-24. Sensitivity vs. Threshold Setting.

components greater than about 50 MHz, as expected from the gain vs frequency curves for the amplifier [124]. Therefore, it is necessary to calibrate the sensitivity of the counter using pulses with about the same width as the actual anode pulses from the multiplier to be used. As shown in Figure IV-15, the FWHM of the anode pulses from the 1P28 are about 5.5 - 6 nsec, and the pulses from the 8850 and MM1 were similar at moderate voltages. The narrowest pulses obtainable from a Tektronix PG501 pulse generator [98] had a FWHM of just 5.5 nsec, and an amplitude which was variable over a range of about 0 - 5 volts. These pulses were connected to the counter via a calibrated 66 db attenuator, which reduced their amplitude to something less than 2.5 mV. Since such small pulse amplitudes could not be accurately measured with the oscilloscope used (Tektronix Model 7904, with a 7A12 plug-in, 5 mV/cm maximum sensitivity [98]), the amplitude of the pulses at the counter input was calculated from the unattenuated amplitude of the pulse generator output, measured simultaneously with the pulse counter threshold.

The input pulse amplitude necessary to cause reproducible triggering of the pulse counter was measured as a function of comparator threshold setting (recorded arbitrarily as the number of turns above the lower limit of R3, Figure IV-17), providing the results shown in Figure IV-24. The lower limit of sensitivity, set by the onset of comparator oscillation (also shown in Figure IV-24), is about 130 μ V. In practice, threshold settings lower than about 250 μ V are not used, so that numerous small noise pulses that are picked up by the cable connecting the counter to the multiplier will not be counted, and this is a more conservative estimate of the usable sensitivity of the counter. Measurement of V_{Co}

at the same threshold settings as in Figure IV-24 indicates that the effective gain of the amplifier for such narrow pulses is about 250, although the low frequency gain is nearly 400.

ii. Speed:

Pulse counter speeds are specified in several ways. One method is to specify the maximum frequency of a periodic waveform to which the pulse counter will respond. The waveform is often not specified, but if it is a series of very narrow pulses, then this method is equivalent to specifying the dead time or pulse pair resolution of the counter. The dead time can also be evaluated by measuring the width of the comparator output pulse for very narrow input pulses (assuming that the amplifier and/or comparator limit the speed of the counter).

The sensitivity of the counter for 5.5 nsec pulses showed no measurable change up to a pulse repetition rate of 57 MHz, which is the limit of the pulse generator used. The output pulse of the comparator when the amplifier was driven by such pulses was about 11 nsec wide. The maximum frequency at which the counter would trigger reliably with a sinewave input was about 93 MHz, which is consistent with a pulse pair resolution of about 11 nsec. Although this is slightly short of the design goal of 100 MHz, there are indications that minor modifications to the circuit would allow operation at frequencies up to 105 MHz (see Part 4).

iii. Stability:

Some of the results of preliminary experiments using this pulse counter circuit indicated that while the sensitivity was apparently

sufficient, the importance of good stability in the pulse counter should not be neglected (see Part 5). An examination of the sources of instability in the 1P28 and 8850 photon counting systems revealed that by far the worst such source was the variation in effective threshold with temperature. Several factors contribute to thermal drift of the effective threshold, but changes in the gain of the amplifier and in V_{Ao} were found to be most important. While it is possible to measure the temperature coefficients of amplifier gain and offset voltage, such measurements are difficult because the changes are relatively small. A more sensitive (and more meaningful) method of measuring the change in effective threshold is simply to monitor the count rate at a constant light intensity as the temperature of the counter is varied.

Representative data from such experiments are shown in Figure IV-25. Although the temperature of the counter can be controlled rather well, it takes several minutes to come to equilibrium after a sudden change of a degree or two, and thus each run shown in Figure IV-25 took more than an hour. The most stable light source available at the time of these measurements showed a fluctuation of about 1% in intensity with a period which varied from a few minutes to a half hour or so. The magnitude of these fluctuations is indicated by the double-headed arrow in the figure. The fluctuations in curves b and c at temperatures below 40°C are of about this magnitude, and were not reproducible; they are therefore attributed to light source intensity variation. This variation prevented an entirely adequate measurement of the temperature behavior of the counter. Nonetheless, the data are sufficiently good to show that the stability of the counter is much greater when the amplifier is set to

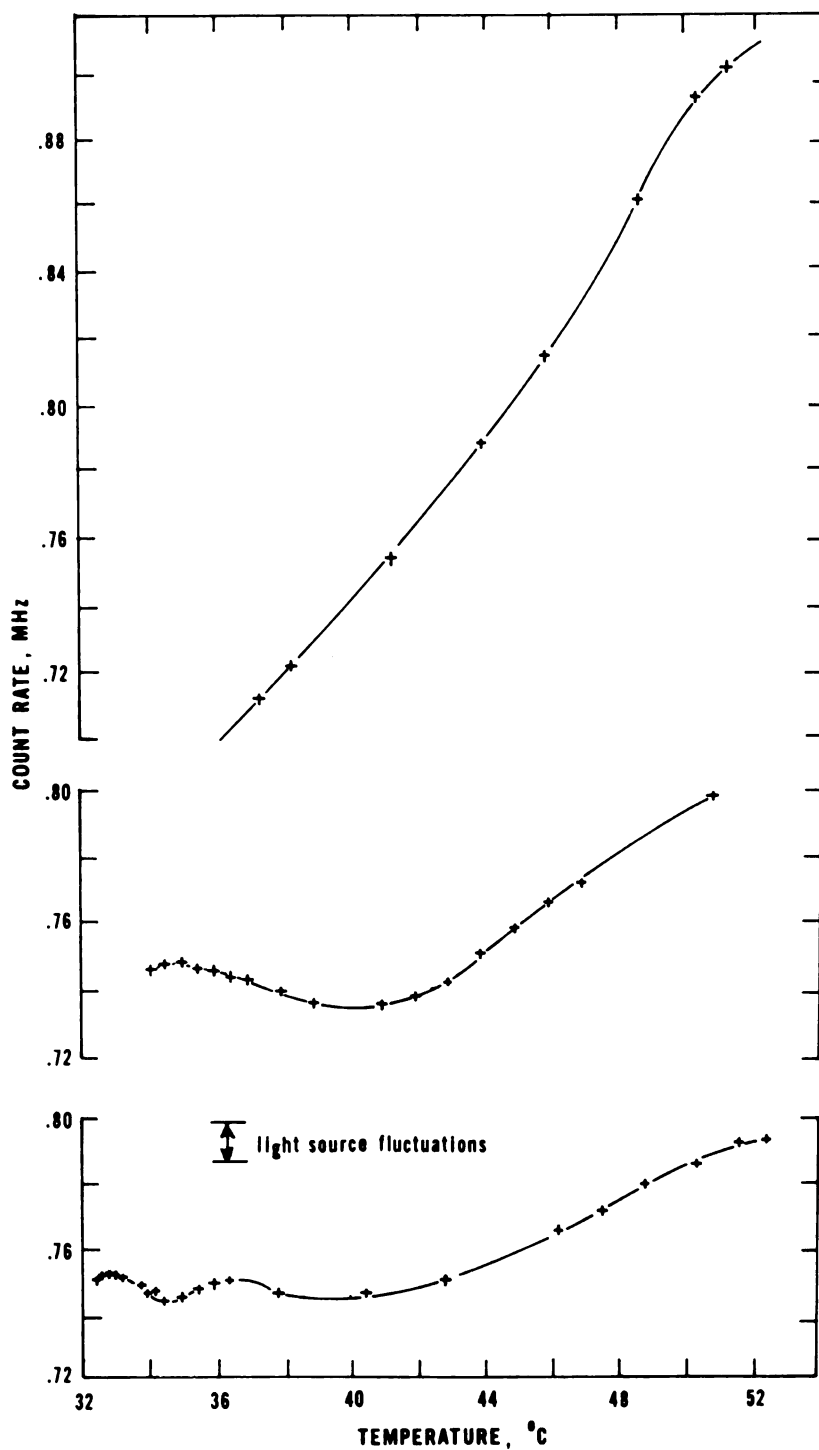


Figure IV-25. Photon Count Rate vs. Temperature. Curve a was obtained with the amplifier set to a gain of ≈ 100 ; curves b and c were obtained at a gain of nearly 400.

its full gain. The gain vs. temperature data supplied by the manufacturer [124] shows much more stable behavior at a lower gain. However, these data apply only to the gain of the amplifier, not to its output offset voltage, which is a function of both gain and temperature. The effects of temperature on gain and on V_{Ao} seem to cancel at full gain and temperatures below 40°C. Over the range of 32 - 40°C, the total variation in count rate attributable to temperature instability is at most 1%, and probably much less. Furthermore, the data of Figure IV-25 were obtained at a discriminator setting corresponding to optimum dead-time compensation, and even greater stability can be attained at other settings (see Part 5).

The temperature controller used with the pulse counter maintains the temperature to within $\pm 0.25^\circ\text{C}$ for days after an hour's equilibration time. The present data are insufficient to show whether such control is actually necessary for good stability. However, temperature control circuits are sufficiently inexpensive to be included in future models of the counter. With such temperature control, the instability of the counter will almost certainly be negligible compared to light source instabilities or changes in photomultiplier cathode sensitivity with temperature [117].

4. Recommended Modifications

There are several minor changes which could be made in the present design of the counter to yield improved performance. The sensitivity can be increased by merely increasing the load resistor (R_L , Figure IV-17) and input cable impedance. Since the impedance of the cable must match the resistor, and since 100 ohm is the highest-impedance coaxial cable

which is routinely available, it would seem that the sensitivity can only be increased by a factor of 2 in this way, so long as the amplifier is housed separately from the electron multiplier.

The prototype circuit board used in the present pulse counter includes small lead sockets so that many of the components on the board can be easily changed without frequent unsoldering. These sockets, being large compared to the component leads, add considerably to the stray capacitance on the board. Their elimination will probably allow a further reduction of the critical range for oscillation. It will also reduce the input capacitance of the comparator input, and the "slow down" effect of the amplifier-comparator coupling resistors. Inclusion of small, variable capacitors across R3 and R6 would also allow the frequency response of the coupling circuit to be increased.

The connection between the comparator and the prescaler is currently a 50 ohm transmission line. The comparator is capable of driving a 50 ohm line, but it can drive a 100 ohm line at a faster rate. (An amplifier-comparator circuit identical with that described here except that the comparator outputs were not connected to the ECL flip-flops, but only to a short 100 ohm line, has operated at frequencies up to 105 MHz.) It would therefore be wise to reduce the width of these foil connections on the circuit board to make them 100 ohm lines and to use 100 ohm terminating resistors.

Another way of obtaining faster frequency response would be to use two cascaded gain-of-ten or -twenty amplifiers instead of a single gain of 400 amplifier, since the bandwidth of the amplifiers is considerably better at the lower gains. This approach was tried early in the

project, and rejected because of extreme sensitivity to stray noise pick-up and oscillation. However, it is probably worth trying again, now that considerably more experience has been gained in proper shielding and construction techniques as well as the control of oscillation.⁸ A possible disadvantage is that the temperature behavior of the resulting circuit could be much worse.

5. Pulse Counting System Performance

The accuracy with which a pulse counting measurement can be made is limited at high count rates primarily by pulse overlap counting loss (i.e., by deviations of the count rate vs. particle flux curve from linearity) [101]. The percentage of pulses lost at a given count rate depends not only on the speed of the counter, but also on the pulse height distribution of the electron multiplier and on the discriminator coefficient, A_d (defined as the fraction of the pulses passed by the discriminator). Hayes, et al. [116] have shown that the count loss is controlled by an "effective deadtime" which can be far different from the pulse pair resolution as measured in Part 3. As the discriminator coefficient is decreased from 1, the effective deadtime becomes smaller, eventually passing through zero and becoming negative. This is, in fact, the basis for the DTC method of Ash and Piepmeier [114]. When the effective deadtime is very small (when A_d is about 0.4-0.5 [101,114]), the pulse counting system is linear to much higher particle fluxes than when $A_d = 1$. Since the linearity of the system depends so heavily on A_d , the speed measurements discussed in Part 3, although useful as a basis for comparison

⁸In fact, one commercial pulse counter [130] appears to use cascaded gain-of-ten amplifiers for its highest sensitivity range.

between counters, are not terribly useful predictors of the linearity of the entire pulse counting system, which must be evaluated for the system, not the counter, and under conditions that will exist during the actual measurements.

A complete characterization of a pulse counting system requires at least two types of measurements. The general shape of the anode pulse height distribution must be known, and the linearity of the system must be confirmed, at least at the discriminator setting to be used. The pulse height distribution will show where the discriminator must be set for the best stability and for the best discrimination against dark counts. Linearity measurements, especially if made at several discriminator settings, will show the setting required for best linearity. Results from both types of measurements can then be combined to indicate the extent to which the conditions required for optimum linearity compromise the stability and sensitivity of the measurement system. Thus, after performing these two types of measurements, the user will have sufficient information to decide on the optimum parameters for the measurements that are anticipated. Pulse height distributions have been obtained for the 8850, 1P28, and MM1 systems, and linearity studies have been performed on the 8850 and 1P28 systems. For these measurements, the 8850 was mounted in a PPI model 57 thermoelectrically cooled housing (but was kept at room temperature) [71] and powered by a Hewlett-Packard 6110A power supply [136]. The 1P28 was mounted in a Heath EU-701-30 photomultiplier module [100] and used the integral power supply of the module. The MM1 was used as the ion transducer in the PIMS instrument, and was powered with a Power Designs 1543A power supply [137].

i. Pulse Height Distributions:

Figure IV-26 shows integral pulse height distributions measured for the RCA 8850 pulse counting system at a number of photomultiplier voltages. These curves were obtained by measuring the dark count rate and the light count rate at a constant light intensity (dark counts subtracted) as a function of discriminator setting, a plot of the results being proportional to the integral pulse height distribution.

The "conventional wisdom" in pulse counting is that the best stability is obtained when all the pulses are counted (i.e., when $A_d = 1$), since the system is then being operated on the flat portion of the "counting plateau" and small changes in discriminator coefficient due to changes in the gain of the multiplier or in the counting electronics will not influence the count rate. At the same time, counting all the pulses ensures the best possible signal-to-noise ratio in a given integration time.

One revealing feature of the pulse height distributions shown in Figure IV-26 is that the plateau region is not flat, although it has a very shallow slope at higher PMT voltages. Similar behavior for an RCA 8853 (a variant of the 8850) has been interpreted as showing that there are very small pulses which are not counted at even the lowest discriminator settings [131]. The data of reference [131] were obtained with much higher PMT voltages, but with a much less sensitive pulse counter, which had a fixed, 100 mV threshold. Since the "plateaus" observed in Figure IV-26 extrapolate to the same count rate at all voltages, it is assumed that the extrapolated count rate is "all the pulses". Therefore, the pulse counter described here does have sufficient sensitivity to count all the

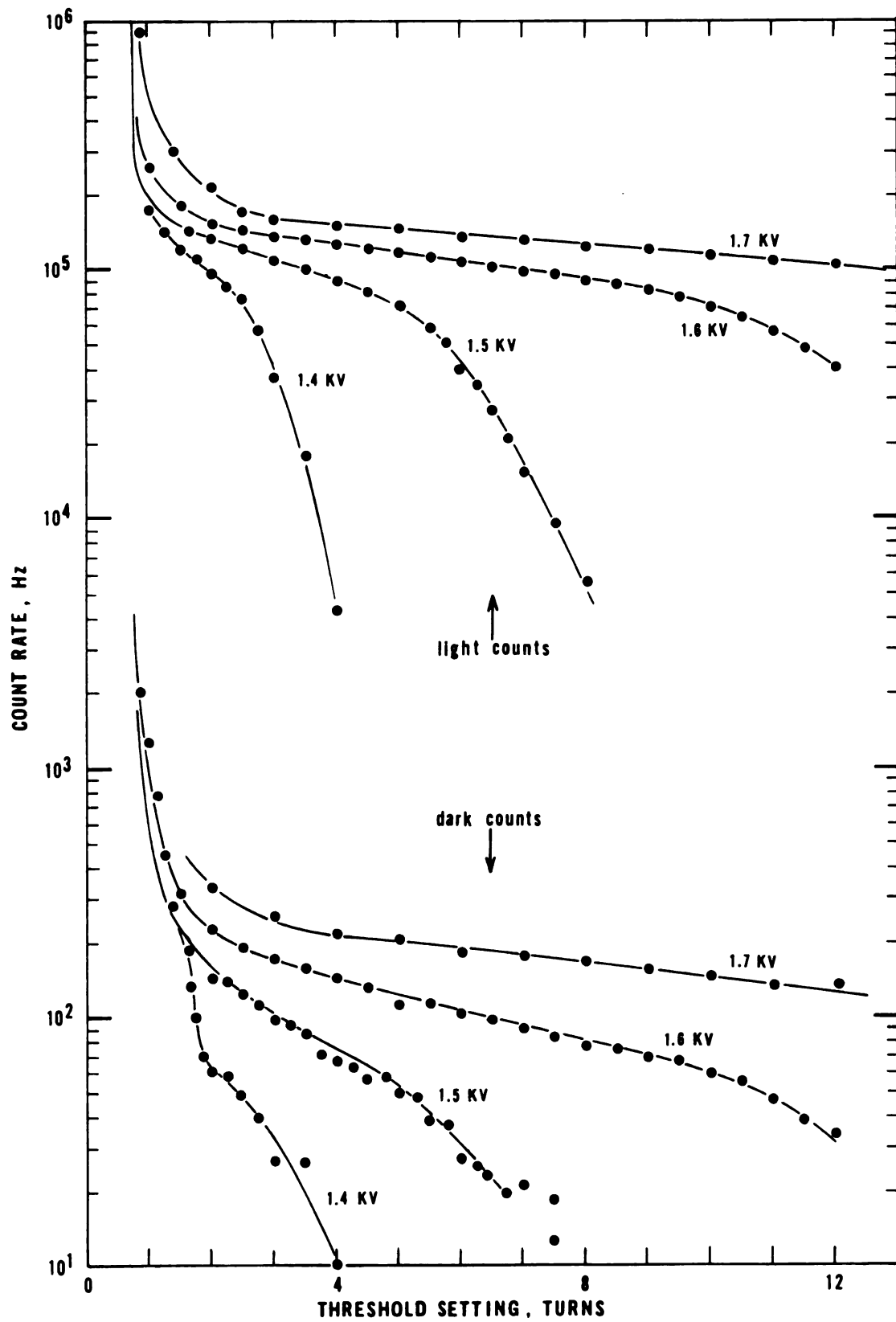


Figure IV-26. Integral Pulse Height Distributions for the RCA 8850. The PMT voltages are indicated.

pulses, at least for PMT voltages above 1400 V. The failure to observe a perfectly flat plateau must then be attributed to a differential pulse height distribution for this particular photomultiplier which does not exhibit a sharp, well isolated peak, but rather a broad, indistinct peak.

Whatever the reason for this type of pulse height distribution, these data show that absolute stability should not be assumed. They also allow quantitative measurement of the relative stability of the system under different conditions. Such relative stabilities (defined here as percent change in count rate per unit change in counter sensitivity) are recorded for several discriminator settings and PMT voltages in Table IV-4. These measurements, obtained from an expanded plot of the data in

Table IV-4. Relative Stability of the 8850 PC System for Different Voltages and Discriminator Coefficients.^a

Voltage	Ad					
	0.9	0.8	0.7	0.6	0.5	0.4
1500	3.2	2.3	2.3	2.2	2.8	4.3
1600	1.6	1.05	1.000 ^b	1.00	1.3	2.3
1700	0.65	0.56	0.48			

^a All values normalized to entry (b), equivalent to 0.081% change in count rate per microvolt change in sensitivity.

Figure IV-26, show that for this system, contrary to the conventional assumption, the best stability is not obtained at $A_d = 1$, but rather at $A_d \approx 0.6$. Furthermore, the stability at $A_d = 0.5$ (shown later to be the setting for "optimum" DTC) is only about 30% worse than the best stability

obtainable at a given voltage.

Thus, the stability of the pulse counting system at $A_d = 1$ (or anywhere else) should not be assumed. Fortunately, it requires only a few minutes to acquire a sufficiently accurate pulse height distribution to check this assumption. When, as in this case, the assumption is not valid, the pulse height distribution then provides much of the data necessary for the user to decide on the optimum conditions for a particular measurement. If better stability is needed, a higher PMT voltage may be used, although (as shown by the data in Figure IV-26), the ratio of light counts/dark counts is worse at higher voltages, and whether this is acceptable will depend on the application.

One point that the user should keep in mind when comparing the stability of the PC system to that of a DC system is that even when the PC system is not completely stable, the type of instability may be quite different from that of the DC system. As pointed out in Section E, a zero drift in the current amplifier of a DC system will cause an error which is usually a small percentage of full scale. However, if the measurements are being made near the lower end of the scale, then that same error may be very large relative to the reading. On the other hand, a zero drift in the amplifier of a PC system will cause the system to count a different fraction of the pulses, and thus the error will be a constant percentage of the reading, not of full scale.

In contrast to the 8850 system, the pulse height distributions for the 1P28 system (see Figure IV-27) show no noticeable plateau at any voltage. Thus, the stability for this system is much worse than for the 8850 system. In addition, the dark count rate for the 1P28 is much higher than for the 8850, making it less useful for low light level applications.

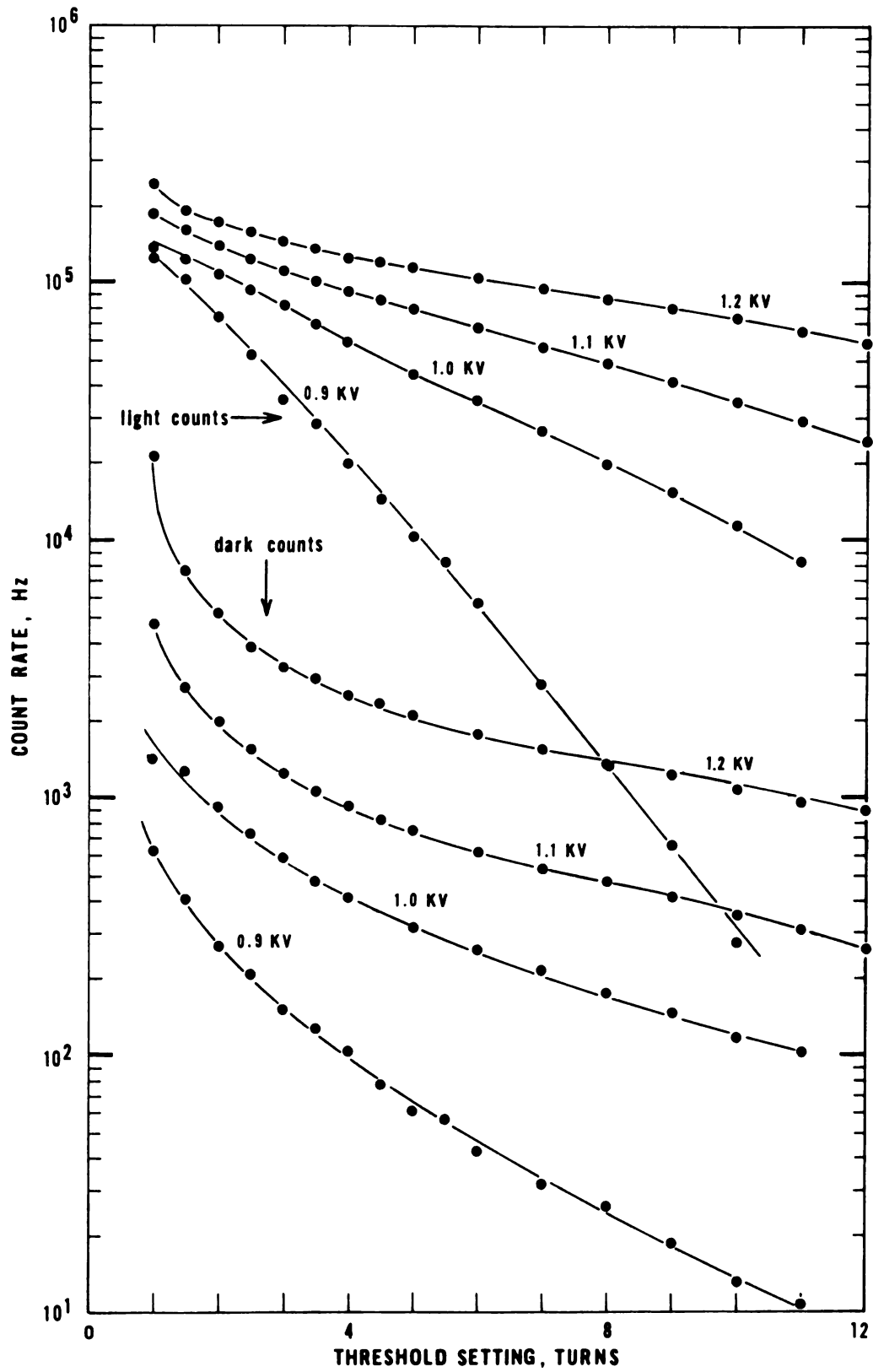


Figure IV-27. Integral Pulse Height Distributions for the RCA 1P28. The PMT voltages are indicated.

It is not at all certain that these results are typical of those to be expected from 1P28 type photomultipliers, since the 1P28 used for these experiments is quite old and has probably seen much abuse in the past. In addition, the power supply of the photomultiplier housing has been found to be quite "noisy", and contributes a significant fraction of the dark counts at lower voltages. While not bearing directly on the suitability of 1P28 photomultipliers in general for photon counting applications, these results again confirm the necessity of testing the entire system instead of assuming that a low discriminator setting will provide stable operation on a flat counting plateau.

The behavior of the pulse height distributions for the MM1 system (Figure IV-28) is intermediate between that of the 8850 and 1P28 systems. As the voltage on the MM1 is increased, not only is the gain of the electron multiplier increased, but so is the energy of the ions striking the first dynode. Thus, changing the voltage not only alters the slope of the pulse height distribution, but also its relative shape--there is no plateau at lower voltages, but at higher voltages, a plateau is quite evident. Also, as expected for a bare electron multiplier (see Chapter II) the dark count rate for the MM1 is much lower than for either of the photomultipliers studied.

ii. Linearity:

The linearity studies reported here were performed using a method similar to that described by Heroux [132]. Basically, the apparent transmittance of a neutral density filter is measured with the pulse counting apparatus as a function of light intensity. If

$$I_0 = \text{photon flux on the PMT without the filter}$$

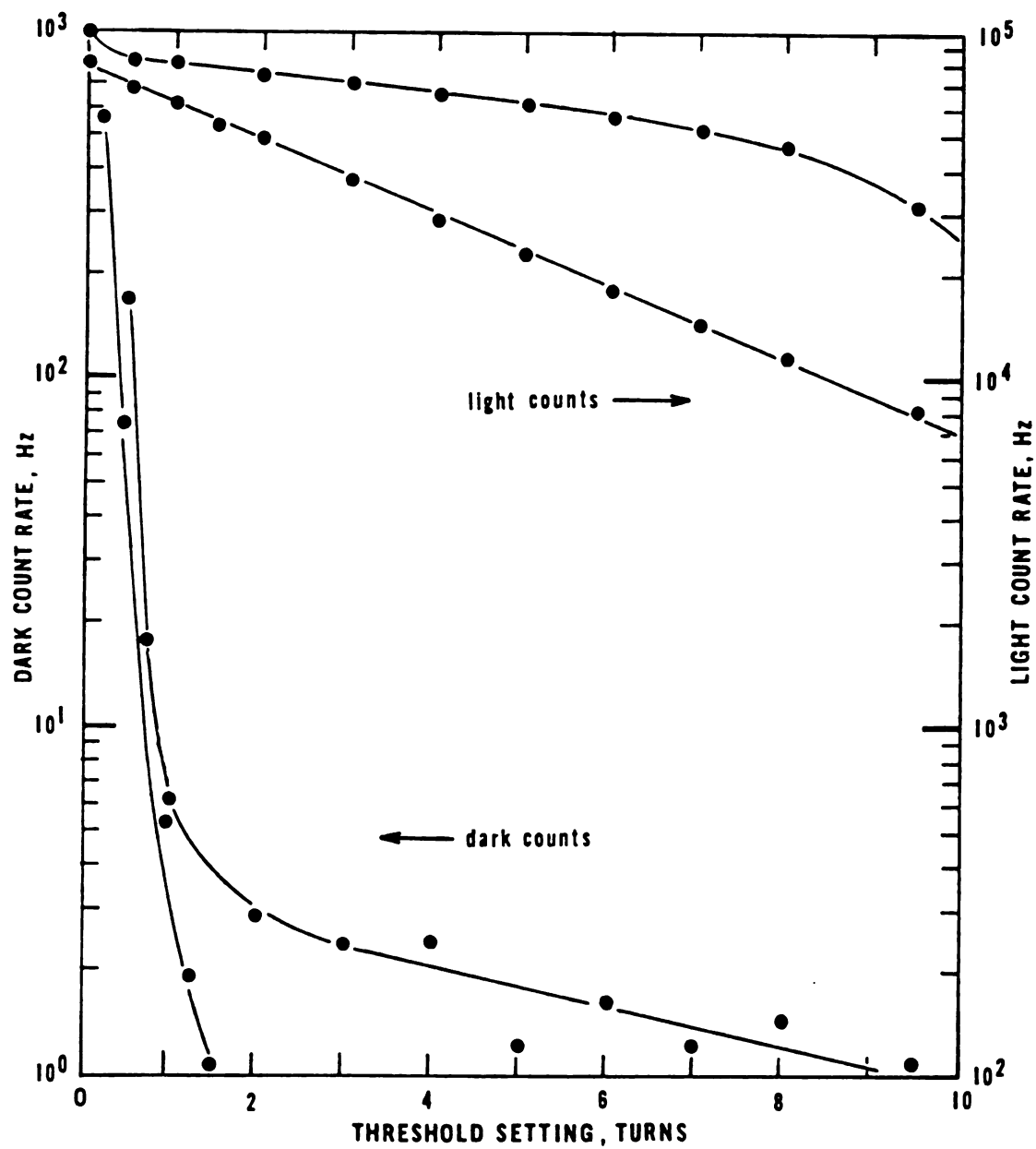


Figure IV-28. Integral Pulse Height Distributions for the Johnston MM1. EMT voltages are indicated.

I = photon flux on the PMT with the filter

T = the actual transmittance of the filter, I/I_0

N_0 = number of counts observed in time t without the filter

N = number of counts observed in time t with the filter

T' = the apparent transmittance of the filter, N/N_0

it can be seen that while T is independent of light intensity, T' will be independent of light intensity (and equal to T) only to the extent that pulse overlap counting losses do not occur in the measurement of N and N_0 . When such losses do occur, they will affect N more than N_0 , and T' will deviate from T by an amount which is indicative of the amount of dead time loss or pulse overlap gain in the measurement of N_0 .

This method assumes that pulse overlap loss in the measurement of N is small. Providing that T is small, this will be a good assumption. This is an easier method of testing the linearity of the pulse counting system than comparing the count rate against a measured DC photocurrent, since it avoids the difficulty of ascertaining the accuracy and linearity of an analog DC measurement system over many orders of magnitude.

The apparatus used for the linearity studies included a Heath EU-701-50 light source module (the tungsten light was used), a Heath model EU-700 monochromator and a Heath model EU-721-11 alternating cell module [100]. The latter was modified to hold a neutral density filter which was in the light beam whenever the sample cell carriage was in the sample position, but out of the beam in the reference position. The carriage position could be controlled by the computer. The shutter

of the sample cell compartment was connected to a small solenoid so that it also could be computer controlled.

The original goal was to measure the count rate at which 0.1% of the counts were lost due to pulse overlap. In order to do so, the transmittance of the filter (about 0.013) must be known to an accuracy considerably better than 0.1% of its value, i.e., to within something less than 0.000013. Since none of the analog spectrometers available were capable of this accuracy, the transmittance of the filter was measured with the photon counting system at very low light levels (about 10,000 photons/sec), where pulse overlap in the measurement of N_0 is certain to be negligible. However, the value of N then had to be measured at count rates of only 130 photons/sec, which is comparable to the dark count rate for the 8850. Precise measurement of such low light intensities, especially in the presence of 1% light source instability, demanded the use of a synchronous measurement procedure similar to that of Arrechi, et al. [133]. Rather than measure N for sufficient time to accumulate a statistically significant number of counts and then measure N_0 , both N and N_0 (as well as the dark count, D) were measured in rotation for periods of time (a few seconds) much shorter than the period of the light source drift. The short integrations were continued until the desired total integration time had been reached (from 15 seconds to 10 hours, depending on the light intensity).

This procedure, fortunately automated and controlled by the computer, was found to eliminate problems caused by light source drift, so that in principle the very low intensity signal of 130 photon/sec could be measured with an accuracy exceeding 0.1%. The required time

for such a measurement, however, is excessive (days) and I found my patience exhausted after about 24 hours of integration at low light levels. That experiment yielded a value of $T = 0.01304 \pm 0.00026$, an accuracy of only $\pm 0.2\%$, which is not sufficient to measure the count rate required for 0.1% counting loss.

Figure IV-29 shows the results of the linearity studies on the 8850 system at several discriminator settings. This figure is essentially a plot of the deviation from linearity vs light intensity, which shows pulse overlap losses (and gains) much more clearly than the conventional plot of count rate vs light intensity. The ordinate is the percent deviation from the correct count rate. It is calculated as

$$\text{percent deviation} = (T'/T - 1)(100)$$

so that -50 corresponds to loss of half the pulses due to pulse overlap. The corrected count rate used as a measure of light intensity for the abscissa of Figure IV-29 is essentially the count rate that would be obtained for N_0 if there were no pulse overlap gain or loss, and the discriminator had been set to pass all pulses. Its calculation requires knowledge of the discriminator coefficient, A_d , which is measured from the pulse height distribution of Figure IV-26. The corrected count rates were then determined from N and the known value of T as

$$\text{corrected count rate} = N/[T(A_d)].$$

It can be seen from Figure IV-29 that at very high count rates, the assumption that pulse overlap losses or gain do not affect N is not strictly valid. However, the assumption is expected to introduce only a few percent

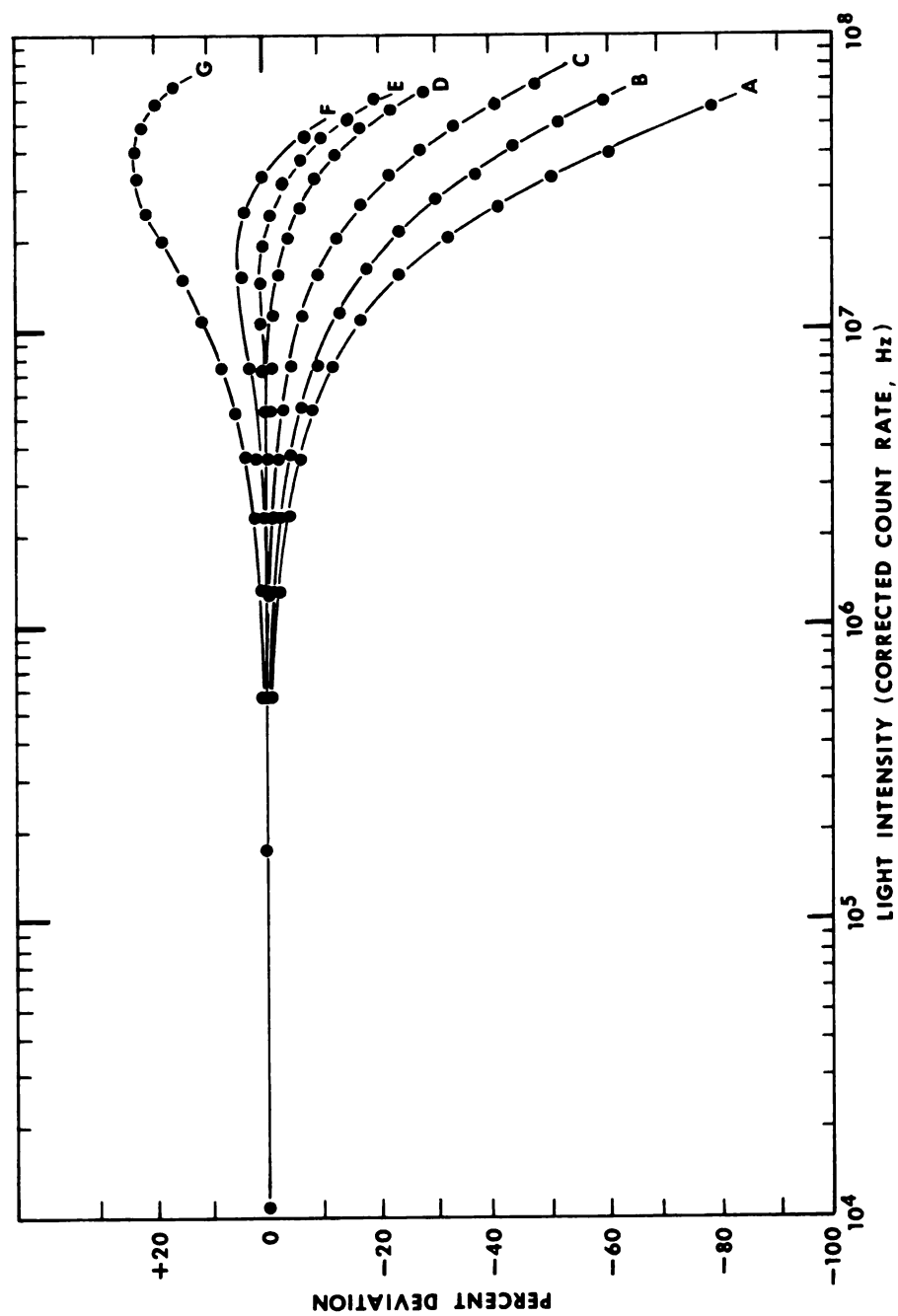


Figure IV-29. Non-linearity of the 8850 Photon Counting System. The discriminator coefficients for the different curves shown in this figure are listed in Table IV-5.

error in the corrected count rate which is calculated. If greater accuracy were required, the count loss vs corrected count rate plot could be used to correct the measured values of N , the corrected values then being used to calculate a new plot, but this was not judged to be necessary for the present purposes. Figure IV-30 shows an expanded scale plot of a portion of Figure IV-29. The count rates at which various amounts of pulse overlap loss occurred for different discriminator settings are summarized in Table IV-5. The linearity studies on the 1P28 system are not reported here, since they were less complete, and show extremely similar behavior.

If only pulse counting instrumentation is available, or if it is inconvenient to switch between PC and DC techniques, it may be useful to extend the upper range of the PC system as far as possible. Other than by obtaining a faster pulse counter, this may be done in either of two ways: by employing some sort of dead time compensation to make the system linear to higher count rates, or by employing appropriate count loss correction formulas to correct the measured data for the non-linearity of the system. The latter approach has been treated in some detail by Hayes, et al. [115,116]. The basic approach is to evaluate the effective deadtime of the system, ρ' , under the conditions it will be used (e.g., by analysis of linearity measurements), and then to use the value of ρ' obtained in an appropriate equation to correct the measured data. The uncertainty in ρ' ($\sigma_{\rho'}$) is found to limit the precision of the measurements at high count rates, so that beyond some point DC techniques must still be used [115].

One dead time compensation technique requires setting a second discriminator to detect large, presumably overlapped, pulses, which are

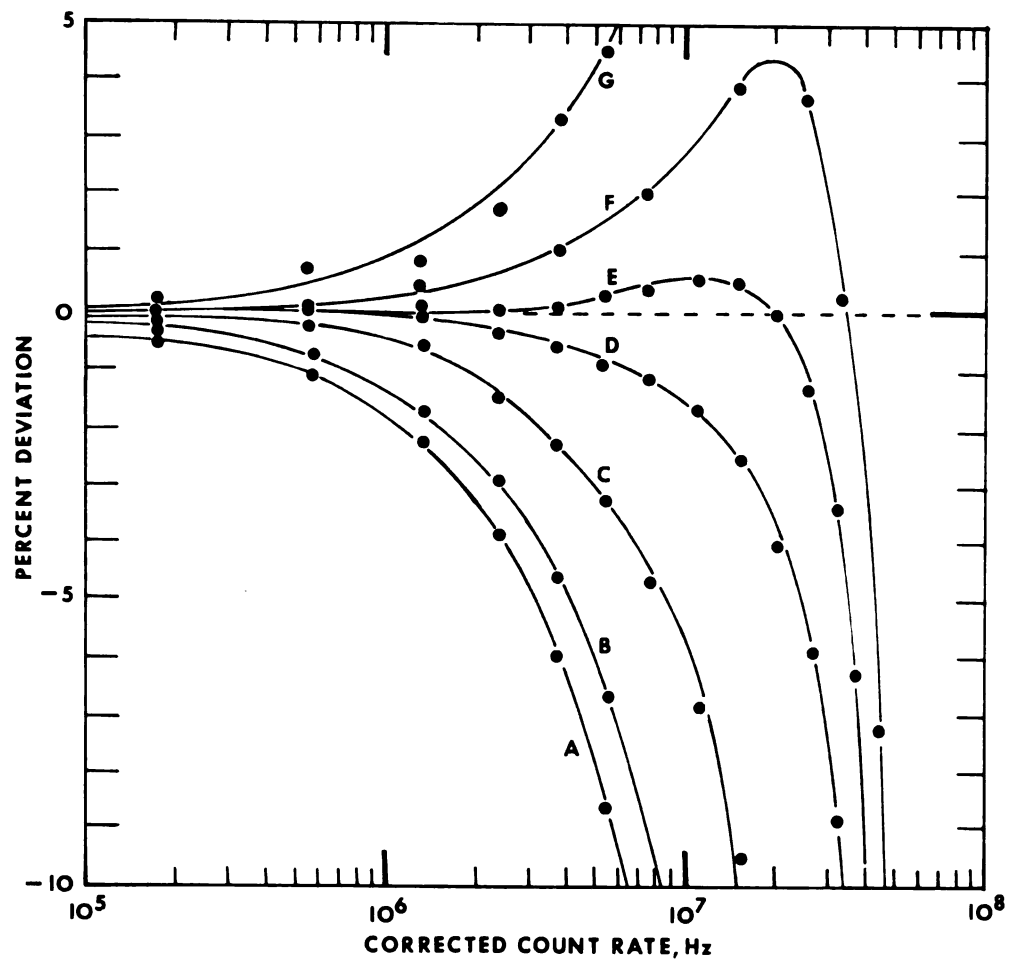


Figure IV-30. Expanded Plot of Figure IV-29.

Table IV-5. Limits of Linearity for Different Discriminator Coefficients.

Threshold Setting (turns)	Discriminator Coefficient, Ad	Curve ^a	Upper Limit of Corrected Count Rate for Indicated Degree of Linearity (MHz)				
			±0.5%	±1%	±2%	±5%	±10%
1.0	∞1	A	0.20	0.50	1.1	3.3	≈6.0
2.0	0.79	B	0.40	0.72	1.4	4.2	8.0
3.0	0.66	C	1.1	1.8	3.3	9.0	14.0
4.0	0.55	D	3.7	6.6	13.0	25.0	33.0
4.25	0.50	E	24.0	26.0	30.0	36.0	45.0
4.50	0.48	F	1.8	3.3	7.5	42.0	48.0
5.0	0.42	G	≈0.70	1.2	2.3	5.8	≈10.0

^aRefers to labels in Figures IV-29 and IV-30.

then counted twice. This approach adds considerably to the complexity of the instrumentation, and is typically found only with (expensive) commercial pulse counters. A second DTC technique, first proposed by Ash and Piepmeier [114], involves setting a single discriminator so that the effective deadtime is approximately zero. This approach is more generally applicable, since it involves only a single discriminator, and is highly effective. For example, the data of Figure IV-30 and Table IV-5 show that at $A_d = 0.5$, the linear range for $\pm 0.5\%$ accuracy extends to a corrected count rate of 24 MHz (measured count rate = 12 MHz), which is more than two orders of magnitude higher than with $A_d = 1$. This corresponds to a measured photocurrent of about 3×10^{-6} amperes. While this current is still below the upper limit of linearity for DC measurements (which can be extended to even higher light intensities by lowering the photomultiplier voltage), it is nonetheless a significant gain in the upper range of the pulse counting system.

The extended linearity, of course, comes at a price. The signal-to-noise ratio of the measurement will be lower for a given integration time, since half of the pulses are discarded to begin with [134]. This can be made up for only by integrating for longer periods of time, but this may not be possible since the stability of the system is worse under optimum DTC conditions, as shown in Table IV-4. In this case, the loss of stability is only about 30%, as discussed above, but for systems exhibiting a flatter plateau at higher discriminator coefficients, the loss of stability could be much worse. If the measurements are to be made at low count rates, DTC or CLFC are not needed, and the user can adjust the counter for best stability or signal-to-noise ratio (or possibly both,

with another photomultiplier!). If the measurements are made at higher count rates, data such as these presented here allow the user to choose the discriminator setting providing the best tradeoff between linearity, stability and signal-to-noise ratio. It is possible that, since high count-rate measurements can be made much more quickly than low count rate measurements (for a given precision), the loss in stability and signal-to-noise ratio may be insignificant. Furthermore, the voltage programmability of the discriminator setting of the pulse counter could be exploited to allow the computer to select a low discriminator setting (high A_d) for low count rate measurements, and automatically switch to a higher setting when the count rate becomes sufficiently high that DTC is needed.

An additional disadvantage of the DTC technique (and of the CLFC technique) is that it requires that linearity studies be made in order to decide on the optimum discriminator setting. The time required to do so, however, is not as great as it might first appear. Once the transmittance of the neutral density filter has been measured, only measurements at high light fluxes are required, and since they can be made quickly with good precision, finding the optimum A_d will typically require less than an hour for a new system.

Hayes, et al. [116] make an additional criticism of DTC. They find that the variance in the effective dead time, σ_ρ , is greater for smaller ρ' . Since, when using their equations, σ_ρ is the limiting factor in the precision with which high count rate measurements can be made, they suggest that DTC (which requires $\rho' \approx 0$) should be avoided, and that A_d should be set near 1, where the deadtime can be measured more precisely.

This criticism seems to be unfair, since it is based on the assumption that CLFC are being applied. If DTC provides adequate linearity, then there is no need to apply any correction formula, the effective dead time need not be known, and the variance of the effective dead time can have no effect on the precision of the measurement.

6. Summary

Since its completion some four months ago, the pulse counter has been used routinely for measuring the ion intensity in the PIMS instrument and it has proved to be extremely dependable and easy to use. The total parts cost for this pulse counter is under \$130. Its speed (>90 MHz, or 11 nsec pulse pair resolution) is within 10% of the best commercially available pulse counting systems (100 MHz, 10 nsec pulse pair resolution). Its minimum threshold (as low as 130 μ V with 50 ohm input) is only about a factor of three more than that of the most sensitive commercial pulse counter of which I am aware (50 μ V) [130], although thresholds less than 250 μ V have not been found to be useful without extremely careful shielding of the multiplier and connecting cables.

Variation in count rate due to temperature instabilities in the pulse counter is less (and probably far less) than 1% over the temperature range 32 - 40 $^{\circ}$ C, and the temperature is maintained to within $\pm 0.25^{\circ}$ C. Since the usual assumption is that pulse counters will be operated on a counting plateau where such instabilities are unimportant, most manufacturer's literature and published descriptions of pulse counters do not discuss the stability of those counters. Thus, it is difficult to compare the stability of this pulse counter to others. However, the data presented show that a flat counting plateau may not exist, either due to lack of

sensitivity of the counter, or to a broad pulse height distribution from the electron multiplier. Therefore, the stability of the PC system should not be assumed.

Finally, this section has shown how relatively simple measurements can be made with this pulse counter to characterize an entire PC system, providing the user with all the information necessary to decide on optimum operating conditions.

G. Dual Counter

This section describes the counter and interface which is used by the computer to integrate the photon and ion signals, shown in block diagram form in Figure IV-31. The heart of the circuit consists of two 24-bit counters which can be started, stopped and cleared simultaneously by the computer. The inputs to the counters can come from any source which yields pulses compatible with one of the line receiver circuits in the INPUT SELECT section. Typically, the input comes from a source of TTL pulses, one of the V-F converter units described in Section E or one of the pulse counter/prescaler units described in the preceeding section. At the end of the measurement period, the numbers in the counters are transferred to the computer via its accumulator register in 12 bit words. The computer can use the flags to check for counter overflow. The counter described here replaces an earlier, dual 12-bit counter. It is capable of being operated as a 12-bit counter to maintain compatibility with the software employed by other users of the computer, although it is always used as a 24-bit counter by the PIMS software. The interface also includes circuitry used by the computer to control the amount of prescaling

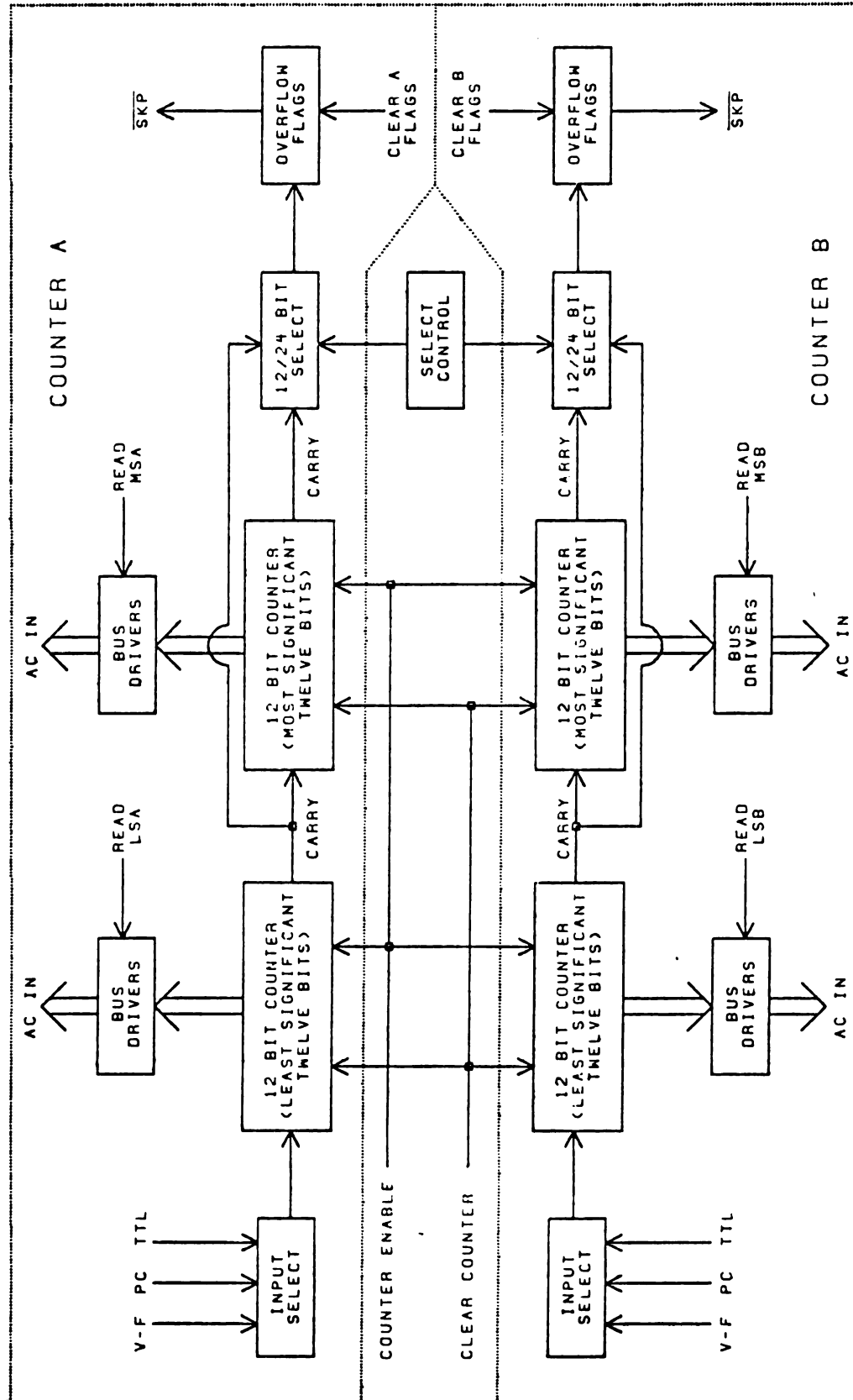


Figure IV-31. Block Diagram of the Dual Counter.

employed by any pulse amp/prescaler units in use, although this is not indicated in Figure IV-31.

The circuit which determines whether the counter will behave as a 12 bit or 24 bit counter is shown in Figure IV-32. When SW1 is in the AUTO position, the computer can control the state of flip-flop 11, via gates 3 and 4 and AOI gates 5 and 6. The 12BIT and 24BIT signals are used by the flag circuits to determine which bit is used to trigger the counter overflow flags. Either mode can also be selected manually via SW1, and as with the shutter and valve control interfaces, the computer can check (gates 7,8) to see if it has control of the operating mode.

The flag circuit for one of the two counters is shown in Figure IV-33. It is triggered by the falling edge of either the twelfth or twenty-fourth bit of the counter (A12 or A24) depending on the status of the 12BIT and 24BIT signals. The first falling edge (counter overflow) after the flags have been cleared will set flip-flop 16, the A OVERFLOW flag, lighting an indicator LED on the front panel. The state of this flag can be tested by the computer with a SKIP ON A OVERFLOW instruction, implemented with gates 21 and 22. If a second overflow occurs before the OVERFLOW flag is cleared, flip-flop 16 will be cleared, and flip-flop 17 (the A ERROR flag) will be set. The OVERFLOW flag will continue to be alternately set and cleared by succeeding counter overflows, but the ERROR flag will remain set until cleared by the computer, indicating that the OVERFLOW flag has been set at least once previously. The flags can be cleared at any time by the computer (via gate 14) and are automatically cleared by the INIT pulse whenever the computer is started.

The INPUT SELECT circuit for one of the two counters is shown in Figure IV-34. It consists basically of the line receivers necessary

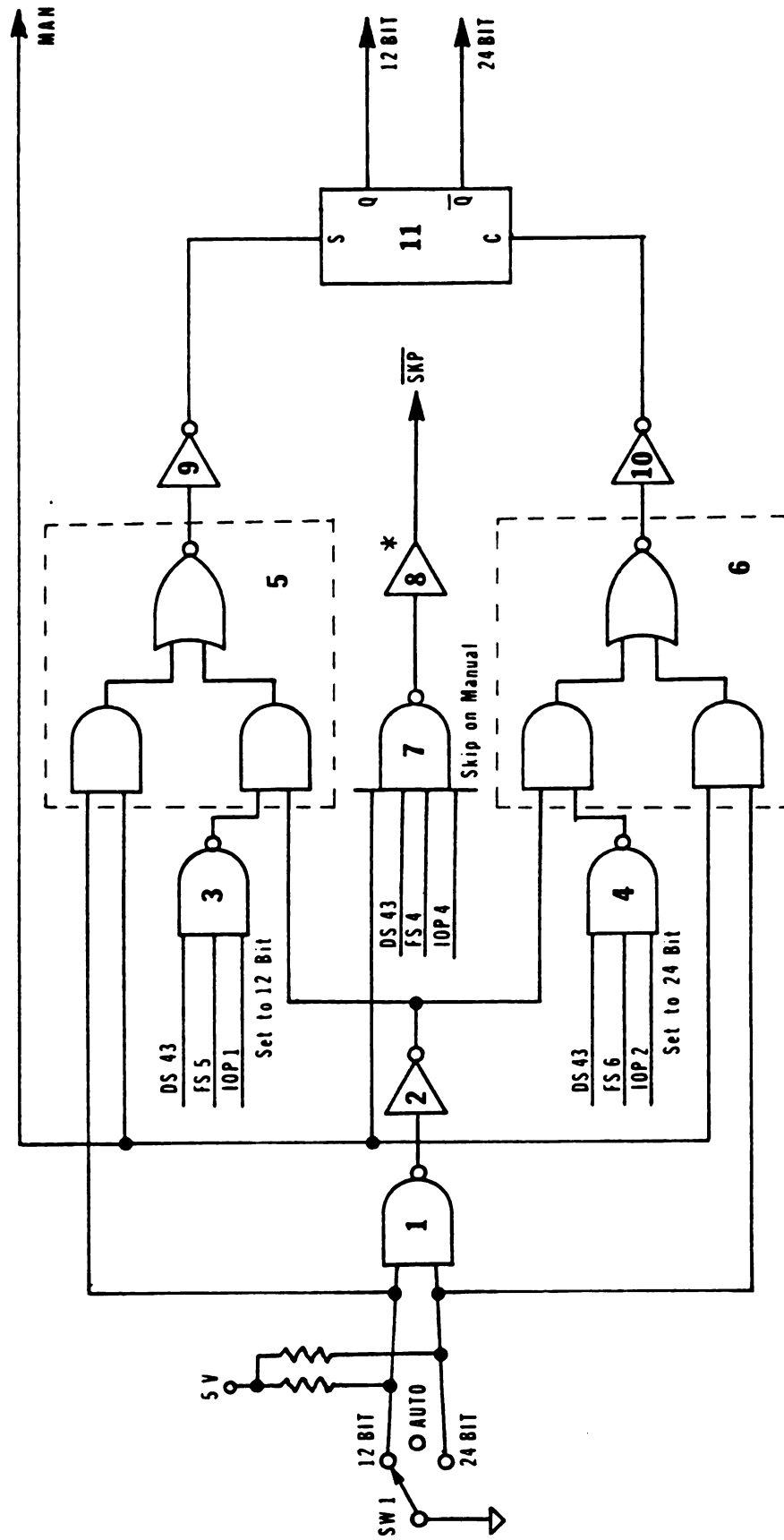


Figure IV-32. Control Circuit for 12/24 Bit Select of Dual Counter.

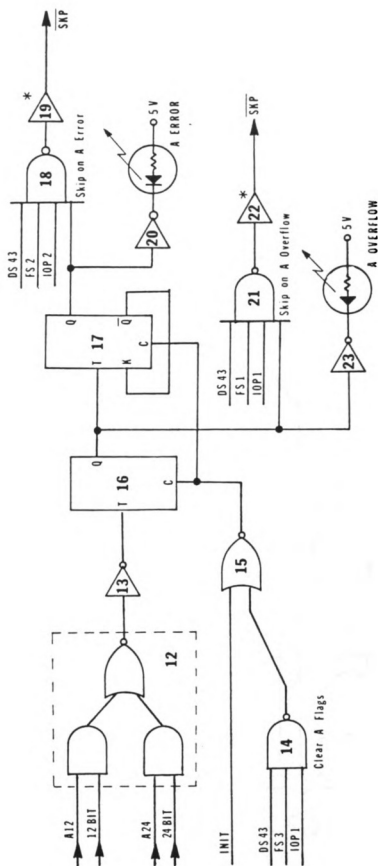


Figure IV-33. Dual Counter Flag and Bit Select Circuit. Only the circuitry for the "A" half of the counter is shown. That for the "B" half is similar, only using different device selects.

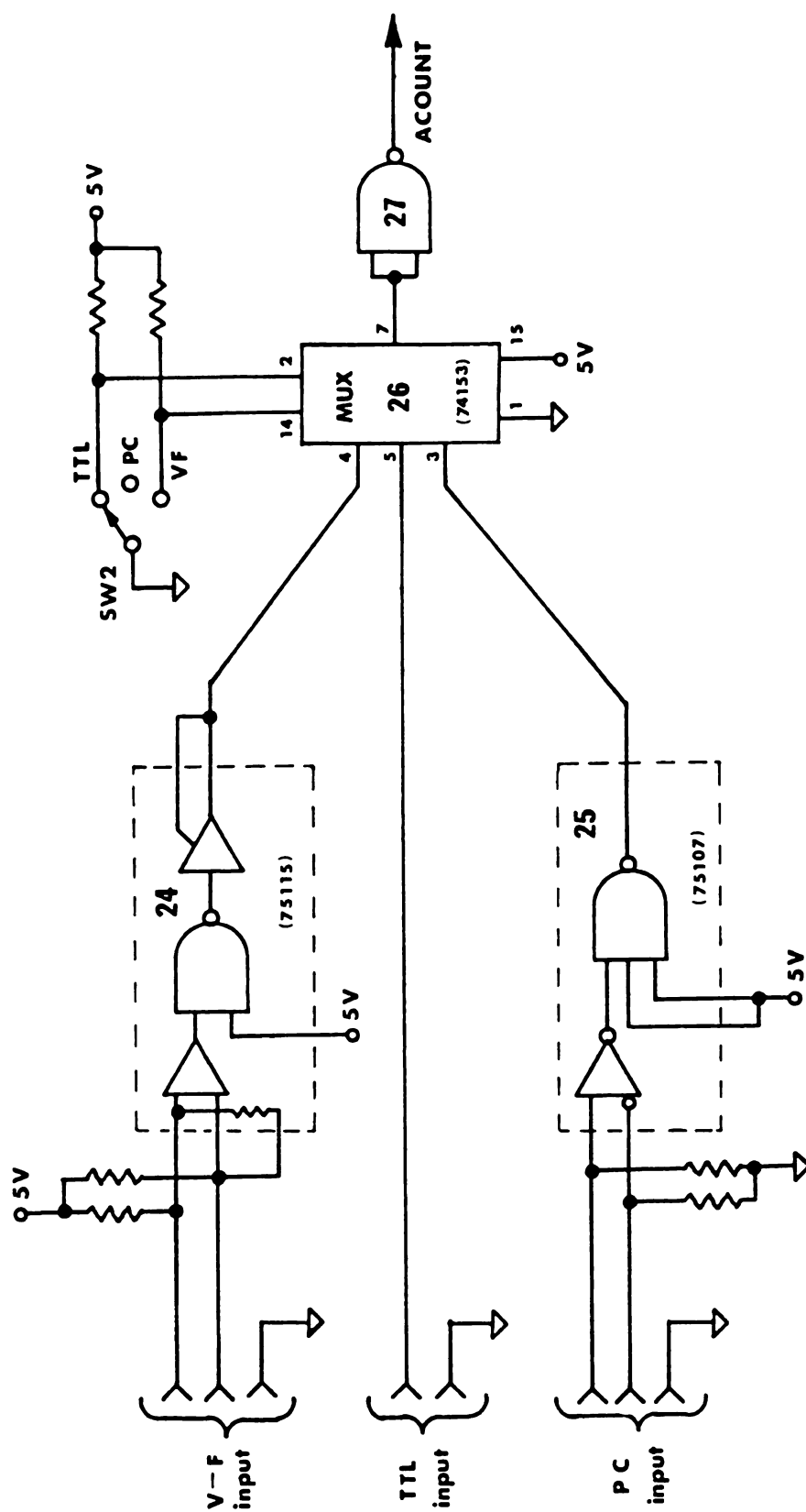


Figure IV-34. Dual Counter Input Select Circuit. Only the A Input select is shown, that for the B Input being identical. Line termination resistor values are one-half the characteristic line impedance. Pullup resistors on SW2 are not critical.

to convert the signals from the V-to-F converter and the pulse amplifier/prescaling circuit into TTL, and a multiplexer which permits one of the possible signals to pass to the counter input. Which of the three signal sources is selected is determined by the position of front panel switch SW2.

One of the 24-bit counters and its associated circuitry is shown in Figure IV-35. The counter itself comprises six 74161 synchronous 4-bit binary counter integrated circuits, which were originally connected as recommended in reference [135]. However, that design was found to be too slow to keep up with very closely spaced pulses from the pulse counting units. Rather than substitute an asynchronous counter IC, which would require modification of a great deal of the circuit, the counter was simply re-wired as shown. Each four-bit IC is still synchronous, but the six ICs in the counter are connected in an asynchronous fashion. This design has proved to be completely satisfactory, and capable of accurately counting periodic pulses at 30 MHz. The bus drivers, also shown in Figure IV-35, are simply open collector NAND gates. The READ MSA, READ LSA and CLEAR COUNTER signals are generated by gates 41, 39 and 55, being buffered by gates 42, 40 and 56, which are 7437 type NAND buffer ICs with sufficient fanout to drive the number of loads required for these signals. The COUNTER ENABLE signal is generated as shown in Figure IV-36, and is also buffered by a 7437 type NAND gate. One of the prescale select circuits is shown in Figure IV-37. When SW1 (Figure IV-32) is in the auto mode, the $\overline{\text{AUTO}}$ signal disables the manual prescale select switches on the pulse counter unit (see Section F) and prescaling is then controlled by the two bit latch (78) which can be set by the computer. Control reverts to the

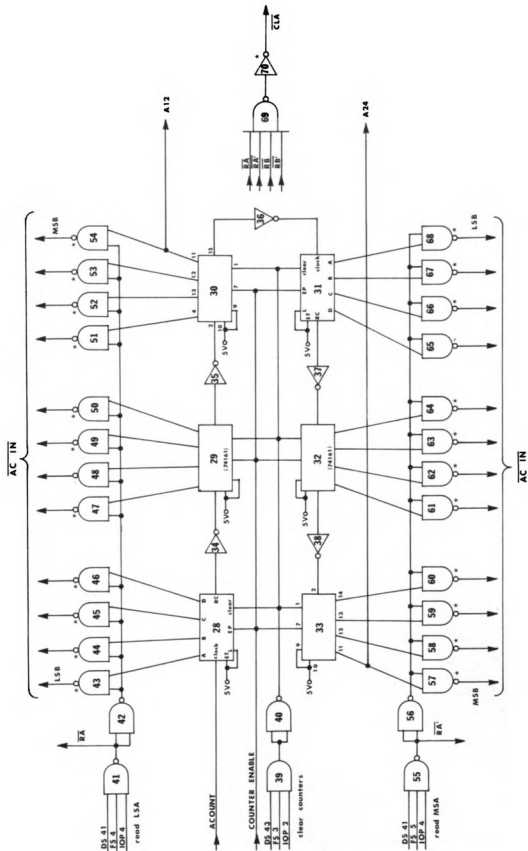


Figure IV-35. Counter and Bus Driver Circuit. Shown only for the "A" side of the dual counter, the circuit for the "B" side is identical, except for the device select codes used.

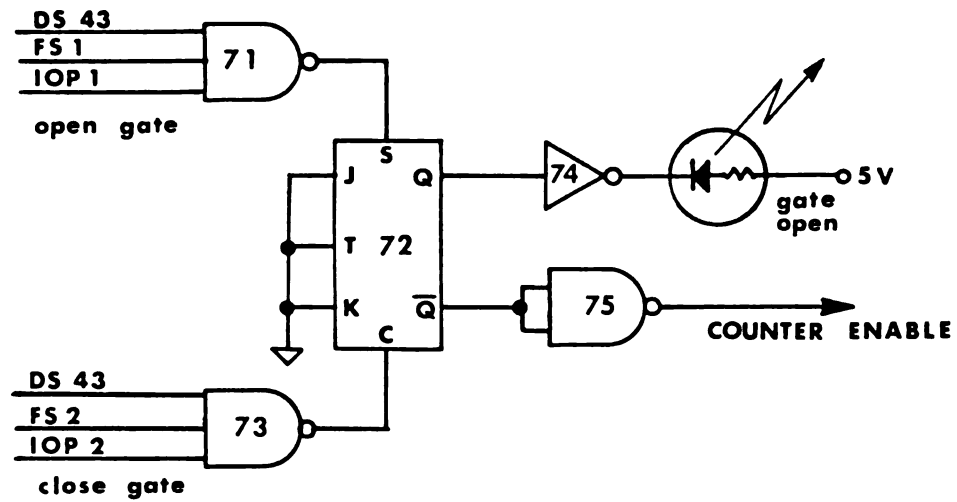


Figure IV-36. COUNTER ENABLE Circuit.

manual prescale select switches in the pulse counter when SW1 is in either the 12BIT or 24BIT position.

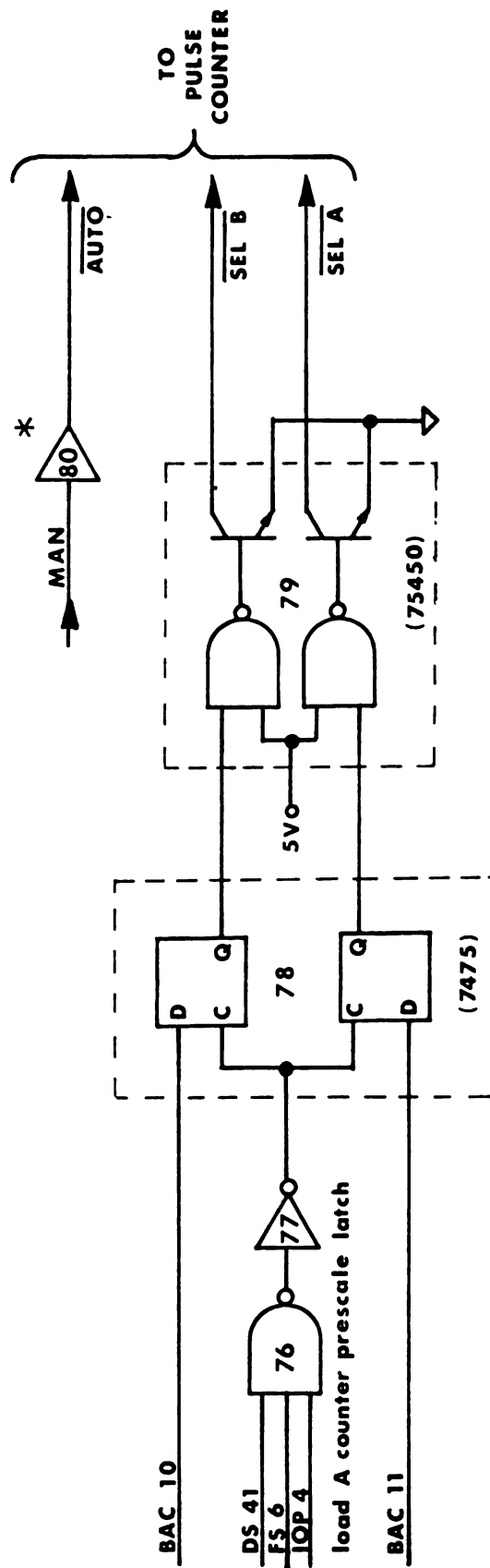


Figure IV-37. Prescale Select Circuit. The circuit for the "B" counter prescale select is identical, except for the device select code used.

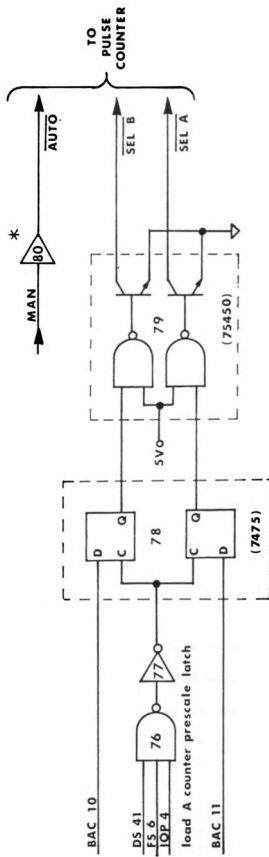


Figure IV-37. Prescale Select Circuit. The circuit for the "B" counter prescale select is identical, except for the device select code used.

CHAPTER V

OPERATION OF THE INSTRUMENT

A. Introduction

The primary goal of the user of the photoionization mass spectrometer is to operate the instrument in such a way as to obtain an accurate photoionization efficiency curve. Secondary goals are to minimize the duration of the experiment and the amount of the user's time spent controlling its progress. This chapter describes the operating procedures developed for the MSU instrument in attempting to accomplish these goals.

1. Stages of the Experiment

Each experiment performed with the PIMS instrument passes through four stages, which are characterized by distinctly different degrees of interaction between the computer, the user, and the instrument.

1. Stage 1--Manual Setup:

At first, the interaction is entirely between the operator and the instrument, as those tasks are performed over which the computer has no control, and about which it can provide no useful information. An example of this kind of task is making sure the sample reservoir is connected to the sample inlet system.

ii. Stage 2--Interactive Setup:

Before data acquisition commences, several parameters must be specified to the computer. There are also further adjustments which must be made manually, about which reasonable decisions cannot be made without further information that the computer can help provide. The information that the user needs in order to choose appropriate parameters and to make final manual adjustments is gathered during the second stage of the experiment in three ways: by asking the computer to acquire it, by using the computer to examine previously collected data, or by visual observation of the instrument while the computer exercises some portion of its control over it. An example of the latter interaction is selecting the proper range for the photon ammeter by observing the meter while the computer scans the wavelength range of interest. This stage is characterized by a three-way interaction between user, computer, and instrument.

iii. Stage 3--Computer-Controlled Data Acquisition:

During the third, and usually longest, stage, the computer controls the data acquisition procedure, guided by parameters which were previously specified. The interaction is entirely between computer and instrument, so that the user's attention is not required, and he or she may use this time in other, hopefully fruitful, endeavors. The provision of real-time output does provide some measure of user-computer interaction, in that the user can occasionally check on the progress of the experiment and tell the computer to abort the rest of the experiment if it does not appear that continuing further would be productive.

iv. Stage 4--Interactive Data Reduction:

Once data acquisition is concluded, the raw data must be corrected for background signals, pressure variations, and so forth, and be converted into a photoionization efficiency curve. The necessary calculations are performed by the computer, but the user must make the decisions about the types of corrections that are desirable, and the way they should be performed. In addition, it is sometimes necessary to specify several empirical parameters to correct for scattered light effects. Thus, in the final stage of the experiment, the interaction is solely between the user and the computer.

2. Progression of Stages

Although it is possible to complete an experiment in a single pass through the four stages described above, the usual procedure is different. Typically, before starting a long data acquisition run, a much shorter run will be made (stage 3) and the results analyzed (stage 4) to make sure that the parameters which were specified are appropriate and that manual adjustments have been properly set. If necessary, the parameters or adjustments can be changed (stage 2). This entire procedure may be repeated until the controls and scan parameters are satisfactorily optimized; only then will the longer data run be initiated (final pass through stage 3). This iterative process takes only a small fraction of the total experiment time, and helps assure more nearly optimum results.

The ability to move rapidly from stage to stage of an experiment, especially from data acquisition to data reduction and back again, is possible only because of the use of an on-line computer. This ability creates the possibility of drastically reducing both the time required

to set up a data run and the number of runs performed with less-than-optimum parameters, and is one of the most important advantages of the MSU instrument. It is this aspect of its design, more than any other, which facilitates setting up experiments to an extent not possible with any of the other design approaches discussed in Chapter I.

3. Chapter Organization

The extent to which a computerized instrument realizes in practice any of its potential advantages depends strongly on the programs (software) used to run the instrument, since the design, convenience, and capability of the software determine most of the operating procedures used with the instrument. Accordingly, Section B of this chapter is devoted to a general overview of the software system that has been developed for use with the PIMS instrument. Section C contains more specific information about the stage-to-stage flow of a typical experiment and the operations performed in each stage. Detailed procedures and programming considerations for the interactive setup, computer-controlled data acquisition, and interactive data reduction stages are contained in Sections D, E, and F, respectively. Listings of the relevant software are included in Appendix E.

B. Overview of the PIMS Software System

1. Basic Considerations

The capabilities of a computerized instrument depend as much on the software that runs it as on the hardware of the computer-instrument interface. If the hardware is sufficiently complex, the software can be relatively simple. On the other hand, an instrument with comparable

capabilities could also be designed using more complex software and much simpler interface hardware. Several different factors influence the designer's decisions about this hardware-software tradeoff. For instance, a computer can execute only one instruction at a time, at some nearly constant (though rapid) rate. It may be difficult to design software which can control many simultaneous operations, or even a single series of operations if they must follow each other at a rate comparable to the cycle time of the computer. If such time limitations or conflicts are involved (they are not in the PIMS instrument, to any great degree), then the hardware-intensive approach has significant advantages.

Either hardware- or software-intensive approaches may have some degree of economic advantage, determined by the cost of a programmer's time relative to the cost of electronic components and assembly time. If neither economic or execution time considerations are overwhelming, and if the instrument for which the hardware and software are to be developed is well characterized and has well defined tasks, the decision about the relative amount of effort to devote to each can be based on the designer's personal preferences. The MSU photoionization mass spectrometer, however, was definitely not in this category. During much of the time that the basic interface and software design work was being done, the instrument was not yet completed, much less characterized, and the proposed operating procedures were tentative at best. In cases such as this, the relative ease with which software can be changed gives the simple hardware-complex software approach a decided advantage. A relatively software-intensive approach was therefore adopted for the PIMS instrument.

The existence of a hardware-software tradeoff is generally well

recognized, but there is also a second important tradeoff that should be kept in mind. It is the tradeoff between time spent programming the computer and time spent in running the instrument. It appears obvious that the latter is more important, until a distinction is drawn between "running the instrument" and "obtaining worthwhile experimental results". Ideally, the computer should act as a helpful, cooperative assistant, and allow the user to concentrate on the science while it takes care of all other details. All too often, however, the user finds the computer difficult to understand and unforgiving of minor mistakes, uncooperative if not downright hostile. The user is forced to spend most of his time trying to remember (scientifically) unimportant details of system operation,⁹ or trying to remember how to recover from errors made by his failure to do so correctly.¹⁰ The science can be almost forgotten, and "running the instrument" becomes equivalent to "fighting the computer", a far cry from "obtaining worthwhile experimental results". Although influenced to some extent by hardware, the difference between the ideal and the uncooperative computer systems is almost entirely in the software.

The hardware that influences the "cooperativity" of the computer is not the hardware that interfaces to the instrument, but the hardware of the computer itself and its peripheral devices. For instance, if the computer's interaction with the user is via a teletype or similar device, the amount of interaction can be severely limited by the maximum speed of the terminal. Also, if the memory of the computer is limited (and it always is), a decision must be made about the amount of room to devote

⁹"Let's see now. Is this one of the places I have to type a decimal point or not?"

¹⁰"Oh damn. It was. Now how do I get the @#\$%&*** thing to stop?"

to programs controlling user interaction or operation of the instrument; the latter must take precedence. However, if the system includes some sort of high speed terminal and some sort of high speed mass storage device (such as a floppy or hard disc drive), these problems become inconsequential, since the computer can both communicate with the user and load new program segments very rapidly. The question then becomes whether it is worth the extra time and effort involved to program the computer to be "cooperative" or even "facilitative".

If the instrument is meant to be used by only one person for a limited time before it is either abandoned or modified to such an extent that drastic software revisions become necessary, then it would not be worth the effort. The PIMS instrument was meant to be used by a growing number of people over a long time period. It seemed that the time spent in programming would be more than repaid by the time that the users of the instrument would save during the lifetime of the software, so I resolved to make the computer as convenient to use, and as "co-operative" as possible.

2. Specific Software Requirements

The considerations discussed above led to the two major objectives that have governed the development of all software for the PIMS system. First, the software should not limit the capabilities of the instrument. This means that the software must be versatile enough to perform any experiment that is likely to be desired, within the hardware limitations of the system. It must also allow the user to specify virtually all of the parameters that determine how the instrument is run.

The second objective is that the overall PIMS system should be

convenient to use. There are several requirements that the software must satisfy to meet this objective. It must allow the user to concentrate on the action he wants to have taken, rather than on how to get the computer to do so. The computer's requests for information from the user (its "prompts") should be simple, but also explanatory and consistent so that the user doesn't need to spend time trying to remember what is expected of him at a particular place in a particular program. All data reported to or entered by the user should be in units which are convenient to him; these units will not necessarily be those which are most convenient in the program.

The user should not have to enter long lists of parameters for each operation he wishes performed. This requirement might appear to conflict with that of versatility, but it need not conflict if extensive use is made of default values for parameters, and the programs are sophisticated enough to decide on the basis of information already provided what further information is necessary.

The objective of convenience also leads to the requirement that the programs should attempt to protect the user from his own mistakes as much as possible. The software should attempt to check all user input and give the user an opportunity to correct entries that appear to be mistakes, without having to re-enter other parameters. There are four types of potential user errors which should be guarded against. First, there are logic errors, such as an attempt to change the wavelength to a value which is beyond the physical range of the monochromator, or an attempt to read data from a floppy disc which is still on the user's desk. Secondly, there are setup errors, such as failure to properly initialize

the system or to turn over to the computer control of certain hardware which can also be controlled manually. Errors of the third type are due to non-familiarity with the detailed inner workings of the PIMS software. An example would be entering a floating point number which must (unknown to the user) later be converted to an integer number, but which is outside the range allowed integers by the software. The fourth type of error is a typographical error on the part of the user which results in the computer reading an entry that is not what was intended.

Since it is unlikely that even a very sophisticated program could successfully guard against all such mistakes, the software should also be written in such a way that it is easy for the user to correct mistakes that the computer does not catch, as soon as he realizes he has erred, even if the computer is in the midst of doing something else by then.

One final type of error which should be guarded against arises not from any "user malfunction", but from the fact that real data are unpredictable. Whenever actual data are used in a calculation, there is a possibility that a "bad" datum (i.e., one well outside the expected range) could cause the calculation to yield extremely inappropriate and undesirable results. If there is a chance of these inappropriate results adversely influencing the operation of the system, then the software should check the results of the appropriate calculations before using them.

3. General Description

The PIMS software system described in this chapter very successfully meets the requirements outlined above. Most of the programs in the PIMS system are written in FORTRAN II to take advantage of the relative

ease with which programs written in a high level language can be debugged and modified. Those portions of the system not written in FORTRAN are written in the SABR assembly language. The PIMS software system runs under Digital Equipment Corporation's OS/8 operating system.¹¹ References [138] and [139] describe the versions of OS/8, FORTRAN II and SABR which are used.

Since the PIMS programs would suffice to fill the memory of the computer several times over, extensive use is made of program chaining [138], a procedure by which a program running in the computer can cause a second program stored on the system mass storage device to be loaded into memory and executed. On the present floppy-based system, a program which occupies the full 12K core locations in the computer can be loaded automatically in only 5-6 seconds. This causes a noticeable but not unacceptable delay, and allows very large programs to be run on a computer with only a modest amount of memory.

The PIMS system was written to minimize the amount that the user must know about the OS/8 operating system. Once OS/8 is started and dated, the user asks the OS/8 Keyboard Monitor to run the program called PIMS8, which is the only program the user needs to call for explicitly. Besides this, the normal user only needs to be familiar enough with OS/8 to create and list directories on the floppies or tapes he uses for data storage, and to delete files which are no longer needed.

The instrument is run by choosing from the list of options which is presented by PIMS8. These options are listed and briefly described in Table V-1. After an option is selected, the system may require more

¹¹The computer system used is described in Chapter IV, Section A.

Table V-1. Main PIMS8 Options.

Name	Task(s)
CORRECT	Correct raw data files and store, print, or plot the corrected data. The data examined may be any of the twelve recognized data types (Table V-7). Each type may be corrected for backgrounds, reference variation, stray light, etc., in a number of ways, and either output or used to correct other types of data in the same file.
CLOSE	Close the monochromator shutter.
CURRENT WAVELENGTH	Report to the user the current wavelength setting of the monochromator, and the equivalent photon energy in eV.
INITIALIZE	Initialize the PIMS8 software. Ask the user to set the maximum stepping rate on the wavelength drive motor, initialize the software register which keeps track of the wavelength, etc.
MOVE	Move the monochromator to a new wavelength setting.
OPEN	Open the monochromator shutter.
PRESCALE	Allow the user to change the amount of prescaling performed by the pulse amplifier/prescaler circuitry (see Chapter IV).
RECORD	Perform data acquisition. Include provisions for wavelength scan, variable integration time, periodic reference and dark current measurements, real time output, etc.
SCAN	Scan over a selected wavelength region at a constant rate; do not perform any data acquisition.
TEST	Test the photon and ion data acquisition interfaces by repeatedly integrating photon and ion intensities for a preset time, and reporting the results to the user.
TIMED ACQUISITION	Acquire data, but time the points more accurately than is done in RECORD. Provisions for wavelength scan, variable integration time, reference and dark measurements and real time output are <u>not</u> included.
TRANSFER	Transfer information from an existing file to a specified device. Any or all types of information may be transferred in either PIMS8 format (Table V-4) or in ASCII. All information of the desired type, or only that information falling between specified limits may be transferred.

information before it takes the appropriate action. If the amount of information needed is relatively small, the computer simply asks the user to supply it. On the basis of the information supplied, the system decides whether still further information is required. If so, it asks for the further information; if not, it calls the appropriate subroutines or chains to the appropriate programs to perform the desired action.

If the amount of additional information required is potentially large, the system will not ask for it all directly. Rather, it will present the user with a second list of options. Many of the options identify parameters that the user may wish to specify. Each parameter has a default value, however, so that it is only necessary for the user to choose the corresponding option if he wishes to change the parameter from its default value. The user has the opportunity to specify every parameter, but often needs to specify very few of them. The other options presented to the user identify the various operations which the system undertakes, using the current set of parameters, when the corresponding option is chosen.

4. System Conventions

User-computer interaction consistently follows several important conventions in all PIMS8 system programs:

1. Termination of Options:

In general, when the execution of any option is finished, control returns to the point in PIMS8 where the user is asked to enter the name of the next main option he wishes to select. There are exceptions to this, for instance the CORRECT option, where the user frequently desires to

run the option repetitively without having to re-specify all of the parameters. Since all parameters are reset to their default values by a return to PIMS8, the return is not made automatically. Instead, in all such cases, the user is explicitly given the opportunity to return to PIMS8 or to repeat the execution of the option.

ii. Input Prompts:

To avoid confusion about what type of input the system is expecting, the following convention has been adopted for input prompts: no matter what other information the computer provides the user in a prompt, the last character of the prompt will always indicate the type of information the system is expecting. The six key characters and the types of information to which they correspond are listed in Table V-2.

iii. Special Characters:

Normally, the user's response to any prompt will be composed of the alphabetical characters A - Z or the numerals 0 - 9. There are seven additional, special characters which are also legal input after certain prompts. They are listed, with the conventions that have been adopted concerning their use, in Table V-3. Special note should be made of the CONTROL G character, whose primary purpose is to serve as an indication to the computer that the user wants it to ignore any other current input, to stop whatever it is doing, and let him select a different main option. In other words, it is the "Hold it! Stop! Cease and Desist! I didn't mean it" character. Since a CONTROL G character is not legal input after a numerical input prompt (: or =), the minus sign is used for the same purpose here. There are only three places in all the PIMS8 programs where a negative

Table V-2. PIMS8 Input Prompt Conventions.

Prompt	Expected Input ^a
->	The name of the option to be selected. At least the first two characters of the option name must be entered, although the entire name may be entered. The terminal bell rings as a part of the prompt.
:	Numerical information in integer format. The number entered must not have a decimal point, and must fall within the range -2047 +2047.
=	Numerical information in floating point format. The number entered must include a decimal point and may optionally include an exponent in the "E" format [138]. The range is usually limited by PIMS8 rather than FORTRAN, and depends on how the number is to be used. The user is notified if his input exceeds the desired range.
?	User agreement. A "Y" signifies yes, and any other character will be interpreted as a negative response.
#	Name of device and file to be used for input or output. The required format is DDDD:FFFFFF, where DDDD is any device name recognized by the OS/8 operating system [138], four characters or less. FFFFFFF is any legal file name of six characters or less (must begin with an alphabetical character). No extension may be specified, since the .DA extension is assumed. If no device is specified, the default device is assumed. No file name is necessary for non-file structured devices such as the terminal (TTY:). The colon is necessary after the device name and must not be used if no device name is entered.
%	Comments to be stored in the data file. The comments may contain any alpha-numeric characteris, and there may be any number of lines of 72 characters or less per line.

^a A carriage return is required to end every input line, except after the "?" prompt.

Table V-3. PIMS8 Special Character Conventions.

Character	Symbol	May be Used After	Meaning
CONTROL U ^a	↑U	->, # or %	Delete all of the input line entered before the ↑U.
RUBOUT		# or %	Delete the most recently entered character. May be used successively to delete as much as an entire input line.
LINEFEED	<LF>	# or %	Echo the input line as it now stands, then resume input.
CARRIAGE RETURN	<CR>	Any prompt except?	End of input line.
CONTROL C ^a	↑C	Any prompt	Leave the PIMS8 system and return to the OS/8 Keyboard Monitor immediately.
COLON	:	#	Previous characters on this input line are to be treated as a device name.
		%	End of final comment line, proceed with data acquisition.
CONTROL G ^a	↑G	->, # or ?	Ignore any previously entered characters on this line and consider execution of the current option to be complete.
		During execution of CORRECT, RECORD, TEST or TRANSFER	Discontinue further execution of the current option and consider the option to be completed.
MINUS SIGN	-	: or = ^b	Ignore any other characters on this line and consider current option complete

^aEntered by holding down the CONTROL key while striking the U, C or G key, as appropriate.

^bExcept when specifying data extrema, plot window extrema, or stray light correction factors in the CORRECT option.

numerical input might be desired;¹² these are the only places that a negative input does not have this special purpose.

iv. Data Storage:

Although the conventions adopted for data storage do not actually influence user-computer interaction directly, the user must be aware of them in order to make most efficient use of several of the main options. A typical data acquisition run will produce far more data than could possibly be stored in the computer's memory. Therefore, the data are usually stored in the output file point by point, as soon as they are acquired. Since there are several types of data acquired during most runs (background and reference data as well as photon and ion intensities, for instance) there must be some way of later distinguishing the types of data in the output file. This is accomplished by beginning every line of stored data with an integer line code, which is a key to the type of data stored on that line. Other system programs can then use the line codes to decide how to interpret the numbers stored on each line. The line codes as well as all data on the line are written in a "packed ASCII" format [138] in order to economize on storage space and time. Table V-4 identifies the line codes currently in use and the types of data to which they correspond.

C. Typical Operating Procedures

Nearly every experiment begins with at least some manual setup. The user must assure himself that the supplies of compressed air and lamp

¹²In setting data extrema on plot window extrema, as well as specifying stray light corrections, all in the CORRECT option.

Table V-4. PIMS8 Data Storage Conventions.^a

Line I2	Code ^b A2	Line Contents ^c	Line Format ^d
1	0A	Wavelength, PHO, ION, Elapsed Time, Integration Time	5R
2	0B	Wavelength, PHO, Elapsed Time, Integration Time	4R
3	0C	Wavelength, ION, Elapsed Time, Integration Time	4R
4	0D	Wavelength, BP, BI, Elapsed Time	4R
5	0E	Original Filename	1R
6	0F	Date (Month, Day, Year)	3I
7	0G	Scan Number (ISCNO)	1I
8	0H	Wavelength, SP, SI, Elapsed Time	4R
9	0I	Starting Wavelength, Ending Wavelength	2R
10	0J	Wavelength Increment	1P
11	0K	Minimum Integration Time, Maximum Integration Time, Desired Total Counts	3P
12	0L	Photon Full Scale, Ion Full Scale for Real-Time Plot	2R
13	0M	Number of Data Points per Background Measurement	1I
14	0N	Background Integration Time, Electrometer Settling Time	2P
15	0O	Reference Wavelength, Reference Integration Time	2R
16	0P	Number of Data Points per Reference Measurement	1I
17	0Q	Wavelength, RP, RI, Elapsed Time	4R
18	0R	Angstroms or Points per Inch for Real-Time Plot	1R
19	0S	1st Stray Light Wavelength, 2nd Stray Light Wavelength Stray Light Measurement Integration Time	3P
20	0T	Photon Prescaling Factor, Ion Prescaling Factor	2I
21	0U	X, Y Output from CORRECT Option	2R
98	A"	Comment	A
99	A#	End of File	0I

^aSee Reference [138] for a description of the A format used for packed ASCII under FORTRAN II.

^bThe entries under I2 give the integer value of the line code, the entries under A2 give the line code as it appears in the packed ASCII, A2 format.

^cPHO, ION, BP, BI, SP, SI, RP, RI are defined in Table V-7.

^dP formats are written out as (A2,n(A6)), where the initial A2 is for the line code and n is the number given in the table. I formats are written out as (A2, n(A2)). Comment lines are written as an initial A2 for the line code, followed by repeated A1 (single characters) until the end of the line, which is terminated with an "at" symbol (@).

gas are sufficient to last the duration of the experiment, that the water filters do not need changing, and that the chambers are properly evacuated. The lamp cooling water is then turned on and the interlock system checked to make sure the interlock/override switches are properly set (see Chapter III). The lamp power supply is turned on and while it is warming up, the entrance and exit slits are selected. Lamp gas is admitted to the lamp and the backpumping and flow rates are adjusted to give the desired pressure in the lamp. The lamp gas trap is cooled if desired and the output voltage of the power supply is turned up to start the lamp.

The sample reservoir is then connected to the sample inlet system, from which air is flushed before the leak valve is adjusted to give the desired sample pressure. The ion gauges are used to make sure that the pressures in the sample and quadrupole chambers are low enough to safely operate the high voltage devices, and are then turned off so they will not contribute to the ion background. Finally, the power supplies for the electron or photon multipliers, the mass filter and the other ion optics are turned on and allowed to warm up for a few minutes before being set to their approximate operating voltages.

Many adjustments made in the interactive setup stage require the PIMS8 system to change the wavelength setting of the monochromator. Therefore, the first task after running the PIMS8 program is to calibrate the software register which keeps track of wavelength. The operator reads the approximate wavelength from the mechanical wavelength readout on the monochromator and uses the INITIALIZE option to set the internal wavelength register to this approximate value. (The mechanical readout is readable to perhaps the nearest quarter of an angstrom. Its accuracy

is determined by the care with which it is adjusted when the grating is installed; an accuracy of 2-3 Ångstroms is not too difficult to obtain. This is a fairly good first approximation to use in calibrating the computer's wavelength register.)

The wavelength can be calibrated much more accurately by observation of atomic emission lines in the lamp output, many of which have very accurately known wavelengths [103]. The SCAN option is used to scan a small wavelength region around the wavelength where a suitable atomic line is expected to be found. When it is identified by observation of the electrometer meter during the scan, small MOVES can be used to set the monochromator to the wavelength of peak intensity. INITIALIZE can then be used once again to set the wavelength register to correspond to the accurately known emission line wavelength. This calibration can also be accomplished by using RECORD to record a small section of the lamp emission spectrum around the emission line of interest, then using CORRECT to identify the apparent peak wavelength of the line. The difference between the known wavelength and the apparent wavelength can be used to re-INITIALIZE to the correct wavelength.

The wavelength is next changed to a region of high ion intensity, and the quadrupole power supply is set for the desired mass and for minimum resolution (maximum throughput). TEST, OPEN and CLOSE are used to assure the user that ions are being detected, and to adjust the ion multiplier or counter threshold if necessary. Finally, TEST is used while adjusting the ion focussing controls for optimum ion intensity.

In order to maximize overall sensitivity, the throughput of the mass filter should be as high as possible (see Chapter II, Section F). Therefore, the quadrupole power supply is set for the minimum resolution

necessary to avoid interference from other ions. The resolution control is first set to give approximately one AMU resolution. The TIMED ACQUISITION option is used to acquire a mass spectrum of the region near the mass of interest, using the sweep output of an oscilloscope to scan the mass during data acquisition. CORRECT is used to examine the recorded data to determine the mass of the nearest interfering ion, and the resolution of the quadrupole is decreased as indicated. A second mass spectrum is usually obtained to make sure that the resolution is set only as high as necessary.

For reasons discussed earlier (see Chapter IV, Section E) it is necessary to adjust the photon electrometer so that it gives a non-zero reading with the shutter closed. When this is done, SCAN is used to make sure the electrometer gives a reading which is just on scale at the wavelength of highest intensity in the region of interest. If not, the electrometer or multiplier high voltage are adjusted as necessary. These adjustments ensure that the electrometer will remain "on scale" for the entire experiment, while using as much of its range as possible.

In order to specify reasonable data acquisition parameters, the user needs at least a rough idea of the variation of ion and photon intensities over the wavelength region to be scanned. The necessary information is usually obtained by once again venturing for a brief time into the data acquisition stage. RECORD is used to do a quick and very rough "junk" scan over the region, with points being acquired perhaps only every 50 Angstroms. The resulting data are examined with the TRANSFER option, and provide the information necessary to decide on reasonable data acquisition parameters. Since such junk runs take little

time, a second junk run is often made using the proposed parameters to make sure they are appropriate. When the user is satisfied that all manual adjustments and data acquisition parameters have been properly optimized, the computer-controlled data acquisition stage can be entered for the final time.

Most data acquisition is accomplished with the RECORD option. When this option is selected, the user is asked to provide several types of parameters -- the wavelength region and spacing of points, the variable integration time parameters, scaling parameters for the real time output, and parameters controlling the periodic reference and dark signal measurements. All such parameters are stored at the beginning of the data file, along with any comments describing the sample, the settings of manual controls or other documentation that the user may wish to enter (see Table V-4).

Once entry of comments is complete, the computer system assumes control of the data acquisition procedure. The user is free to leave, and need only return periodically to check the real time plot or to refill the lamp trap if one is being used.

When data acquisition is finished, the CORRECT option is used to convert the raw data into a corrected photoionization efficiency curve. The details of the procedure used to do this depend strongly on what the acquired data are like, though a very general procedure can be given. Typically, the photon background signal measurements are used to correct three other types of data -- stray light and photon reference data as well as the photon intensity data. The user must choose one of several available methods of making the correction, a decision which is usually

based on what a plot of photon background versus wavelength or time looks like. When the method is chosen, a plot of stray light data (corrected for photon background) is used to determine how the photon reference data should be corrected for stray light. A plot of (corrected) photon reference data is then used to decide the proper method for correcting the photon intensity data. A similar procedure is followed with the various ion data types, and finally the photon data (corrected for background, stray light, and reference) and ion data (corrected for background, stray light ions, and reference) are used to calculate the desired photo-ionization efficiency curve. This rather complicated-sounding procedure can often be shortened considerably, since the default correction methods often produce adequately-corrected data without the user having to specify any of the correction parameters at all. As with all PIMS8 system options, however, the user does have the opportunity to change any parameters necessary if the default values are not adequate.

D. Interactive Setup Options

This section contains more detailed descriptions of those main PIMS8 options that the user is likely to use during the interactive setup stage of the experiment: CLOSE, CURRENT WAVELENGTH, INITIALIZE, MOVE, OPEN, PRESCALE, SCAN, TEST and TRANSFER.

1. CLOSE

When the user selects this option, the system first checks to make sure that the shutter control circuitry (see Chapter IV, Section C) is in the automatic mode. If it is not, a message is printed asking the

user to correct the situation, and to once again choose one of the main options available. If the shutter control circuitry is already in the automatic mode, the system merely closes the shutter and asks for the next main option.

2. CURRENT WAVELENGTH

The PIMS8 programs do not control wavelength directly; rather they control the number and direction of steps taken by the stepping motor. Similarly, the system does not actually keep track of wavelength, but instead keeps track of a step count (stored in the variable WSCNT) which is defined as the number of steps the motor would have to take from zero Ångstroms (central image) to reach the present wavelength. The number of steps necessary to change the wavelength by one Ångstrom is stored in a second variable (WSPA). When the CURRENT WAVELENGTH option is selected, the wavelength is calculated from the current value of WSCNT and WSPA. The equivalent photon energy is calculated from the wavelength, and both wavelength and energy are reported to the user before he is asked for the next option.

3. INITIALIZE

This option is used to initialize the values of several important variables which are necessary for the proper execution of many other options. Therefore, this option is forced if the user attempts to choose any of these options before INITIALIZE has been selected. Most of the variables set by this option need be set only once each time PIMS8 is run, but one of them (WSCNT) may need to be changed several times before the wavelength is properly calibrated (see Section C). Every time this

option is chosen, a flag is tested to determine whether it has been selected previously. If it has, the user is only asked to supply the current wavelength which is used to set WSCNT to the corresponding step count. If the INITIALIZE option has not been previously selected, several additional tasks are performed:

The number of lines/mm on the grating in use and the gear ratio of the stepping motor wavelength drive (input by the user) are used to calculate the number of steps per angstrom (SPA) and the upper wavelength limit of the monochromator (UPLIM). All wavelengths entered by the user, in this and other options, are compared with UPLIM to make sure they do not exceed the physical limits of the monochromator wavelength drive. The user is asked to supply a value for ISCNO, which is a scan number that is recorded with every data file, and which is automatically incremented every time a file is recorded.

Finally, the user is asked to set the maximum stepping rate for the wavelength drive motor, and to set a flag that determines whether any acceleration is to be used (see MOVE). The number entered for the maximum stepping rate is compared with the upper limit recommended by the motor's manufacturer, and set to the recommended limit if the desired rate is in excess of that limit. Because of the non-infinite resolution of the real time clock (Appendix C), not all desired stepping rates are possible.¹³ The system calculates the (possible) stepping rate that comes closest to the desired rate input by the user. This rate is reported to the user for his approval before it is actually adopted; if it does not

¹³For instance, since the 0.01 millisecond time base is used to time the intervals between successive steps, rates between 1492 steps/second (0.67 msec per step) and 1515 steps/second (0.66 msec per step) cannot be used.

meet the user's approval, he is asked to enter another desired rate. If the proposed stepping rate does meet with the user's approval, it is adopted and program execution continues.

4. MOVE

When this option is selected, the system first reports the current wavelength and energy (exactly as in the CURRENT WAVELENGTH option) and then asks for the desired wavelength. The desired wavelength entered by the user is checked to make sure it lies between zero Ångstroms and the upper wavelength limit, UPLIM. If not, an error message is printed and the user is asked to repeat his entry. If the desired wavelength is within the physical limits of the monochromator, it is checked to make sure that it is above 500 Ångstroms. The reason for this is that the desired wavelength is read into the computer in a floating point format. If the user inadvertently omits the decimal point from his entry, the computer ends up with a very small number, significantly less than the intended entry. Since wavelengths below 500 Ångstroms are seldom used with the currently available light sources, it is likely that any entry the computer reads as less than this is in fact due to a user error. Whenever a wavelength entry apparently below 500 Ångstroms is encountered, the system echoes the wavelength it read in and asks the user if that is really what he meant. If not, the user is asked to repeat the entry.

Once the desired wavelength is entered, it is converted to an equivalent desired step count and the move is started. If the move is towards higher wavelength it will be made directly. If it is toward lower wavelength, MOVE will go 600 steps too far and then return to the desired step count to eliminate any possible gear backlash.

The speed at which the move is made depends on whether acceleration is permissible (decided the first time INITIALIZE was chosen). If no acceleration is used, all moves are made at a constant speed corresponding to the maximum permissible stepping rate. If acceleration is permitted, but the move is very short, it will be made at a constant rate equivalent to one fifth of the maximum permissible stepping rate. If the move is far enough to make it worthwhile, the motor will be accelerated to the maximum speed at the beginning of the move, and decelerated at the end.

When the desired wavelength is reached (within one step, or $.005 \text{ \AA}$), the wavelength and energy are again reported to the user before another option is requested.

5. OPEN

This option is exactly analogous to CLOSE, except that it causes the monochromator shutter to be opened.

6. PRESCALE

When this option is chosen, the system asks the user to specify the amount of prescaling to be done by any pulse amplifier/prescaler units that may be in use for measuring photon or ion intensities. The input is checked to make sure it corresponds to one of the four possibilities, and the prescale select registers in the counter interface are set accordingly.

7. SCAN

The first step in the execution of this option is to ask the user to supply an initial wavelength, a final wavelength and a scan rate

(in Ångstroms/second). The input wavelengths are checked for possible errors in a manner similar to that described for the MOVE option. In addition, the wavelengths are checked to make sure that the final wavelength is greater than the initial wavelength so that the scan will be in the direction that gives the best wavelength accuracy. For reasons discussed above, not all scan rates are possible (see INITIALIZE). The desired scan rate is converted to the closest possible scan rate; the actual beginning and ending wavelengths and scan rate, as well as the calculated total duration of the scan are reported to the user for his approval. After the user approves the proposed scan parameters, a MOVE is made to the starting wavelength, and a prompt is printed asking the user to type a carriage return when he wishes the scan to start. At the end of the scan, the current wavelength and energy are reported before the next main option is asked for.

8. TEST

The only additional user input required during this option is the desired integration period, which the system truncates to the nearest 0.1 seconds. Once it is specified, the dual 24 bit counter interface is used to integrate photon and ion counts for the desired period. At the end of the integration period, the results are reported to the user and the integration is started again, for the same period. This process continues until interrupted by the user (with a CONTROL G entry on the terminal).

The subroutine used to perform the actual integration is the same one used during data acquisition. It does not count the number of times the counter's overflow flag is set (see Chapter IV, Section G), but simply

checks the overflow error flags for an indication that the counters may have "rolled over" more than once. If an error flag is set, the subroutine returns the negative of the count read from the counter as an indication of possible error. Thus a negative count means that more than 33,554,432 (2^{25}) counts were accumulated during the integration period.

9. TRANSFER

This option is used to transfer data files, or portions of data files, to a second device or file. Since the second device need not be file structured, this option is useful for listing sections of previously recorded data on the terminal.

When the TRANSFER option is chosen, the user is presented with a second list of options (summarized in Table V-5) which allow the user to specify the parameters of the transfer. When the EXECUTE option is chosen, the transfer is performed, with the current values of these parameters. After completion of the transfer, control does not automatically return to PIMS8, but remains within TRANSFER, which allows the user to EXECUTE another transfer with the same or modified parameters. The user can return to the main PIMS8 option list by selecting the PIMS8 option of TRANSFER.

The INPUT option allows the user to specify the input device and file name. This name is eventually used to prepare the FORTRAN II system to read from the appropriate file using device-independent I/O [138]. If, due to some user error or oversight, the entered device name does not correspond to one of the legal device names recognized by the OS/8 system, or if the file name does not correspond to a data file actually on that device, an attempt to read the non-existent file would cause a fatal error in the execution of the TRANSFER program. This in turn would

Table V-5. TRANSFER Options.

Option	Default	Explanation
EXECUTE	----	Perform the transfer, using current parameters.
FORMAT	ASC11	Specify ASC11 or PIMS8 format for output.
INPUT	----	Specify and check input device and file names.
LIMITS	NONE	Specify wavelength or elapsed time limits for transferred data.
OUTPUT	TTY:	Specify and check output device and file names. Reverts to TTY: after EXECUTE.
PIMS8	----	Return to PIMS8.
TYPES	ALL	Specify data types to be transferred.

cause an immediate return to the OS/8 Keyboard Monitor and loss of wavelength calibration. To avoid such errors, use is made of a subroutine which checks to make sure the device is recognizable to OS/8 and that the desired input file can be found on that device. If a potential error is discovered, the user is asked to re-enter the input file and device name. The TRANSFER subroutine will refuse to attempt to EXECUTE a transfer until an acceptable input device and file specification has been entered.

The OUTPUT option allows the user to enter the name of the output device and file. The default output device is the terminal (TTY:), and this option need never be chosen unless the user desires to transfer data to some other device. If chosen, the same subroutine mentioned above is used to make sure that the specified output file does not already exist on the specified output device, which would also cause a fatal error. After completion of a transfer, the output device specification automatically reverts to TTY:.

The user may select the types of information to be transferred by choosing the TYPES option. For the purposes of TRANSFER, the various types of data are classified as follows (see also Table V-4):

Background:	All lines beginning with line code 4
Stray:	All lines beginning with line code 8
Reference:	All lines beginning with line code 17
Data:	All lines beginning with line codes 1, 2, or 3
Heading:	All other lines.

Any combination of these may be transferred. If the types to be transferred are not specified by the user, all types will be transferred.

Especially if the input file is very long, the user may not be interested in all the data of the desired type, but only a small portion of it. The LIMIT option allows either wavelength or elapsed time limits

to be placed on the transfer, so that only information of the desired type falling within the specified limits will be transferred. If the LIMIT option is not selected by the user, then all information of the desired type will be transferred.

Finally, the user may use the FORMAT option to specify whether the output data is to be written in ASCII or PIMS8 (packed ASCII) format. Regular ASCII is the default transfer format.

When the EXECUTE option is chosen, the input file is read a line at a time, and TRANSFER decides on the basis of its line code (Table V-4) whether or not to transfer the line. If it is one of the desired data types, and within the specified limits, if any, the entire line is transferred to the output file or device. If the output format is ASCII, headings are transferred with explanatory titles. Figure V-1 shows the result of using TRANSFER to output a portion of a file on the terminal in the PIMS8 standard format. Figure V-2 shows exactly the same data transferred in ASCII format. The PIMS8 format requires significantly less storage space (the reason that it was adopted in the first place!), but it is less meaningful to a human reader (the reason that ASCII format output is included in TRANSFER). ASCII format output is typically used when transferring information to the terminal for immediate inspection, or to a file which is to be listed on the lineprinter of another computer which has not been programmed to translate PIMS8 format information. PIMS8 format transfers are used whenever there is a chance that the new output file may in turn become an input file for TRANSFER or some other PIMS8 system option, such as CORRECT.

```

@EE21902
@F@H@C@C
@C@C
@T@A@A
@1QT"@@@Q\F Y8
@J0.3L3M
@KPT@@@@L,@@@Q?T@@@
@MD,
@NP/ @@@P@@@@
@SQJZ @@Q\Y@@P7 @@@
@QVP@@@P/ @@
@PBV
@LQ4Y@@@RM@@@
@RP.P@@@
@HQJZ @@@V6%@@Q1+28@@QNX 'P
@AQTI?7?Q\E/1.Q@@H@@Q0*X8UP5>Y8Z
@AQTBV%Q\CP0'Q3C6'@QTLN5"P5"Y8Z
@AQTD3LQ\H-35Q3F:FYQT"FY@P5X3L3
@AQT"GL2Q\J*K8Q$11G5QT7<%,P518Y8
@AQT"13XQ\K9/1Q3L?57QUMG(AP5CL3M
@AQT"K7?Q\MS10Q3NC,TQU1)C<P513L3
@AQT"NY%Q\0->Q3SY<#QU6N#P5E8Y8
@AQT"P3LQ\QX:DQ$URK3QVJHZ P46Y8Z
@AQT"SL2Q\S@8ZQ$ZPV$QV\;4P4"Y8Z
@AQT"U3XQ\T;R5Q$15K$QV,1+<P4P@@@
@AQT"W7?Q\V6%Q3_#KSQV?#G8P4P@@@
@AQT"ZY%Q\X1L Q2#S:GQWP1B1P4VY8Z
@AQT"3LQ\Z#D;Q$56JQW1R^P/3L3M
@AQT"_L2Q\^/\,Q$.JB@QW1E 1P/>Y8Z
@AQT"13XQ\^+1CQ$0LTQQ\@K^VP/Q8Y8
@AQT"#??Q -^TQ$4:F/Q\CX7_P/H@@@
@AQT"8Y%Q"1_1Q$7D1/Q\N<3P.<3L3
@AQT"(3LQ\SW1PQ$;5-1Q\U11<P/D3L3
@AQT"+L2Q\8YKSQ$>Z18Q\ZK8P.3L3M
@AQT"-3XQ\CGCQ$BNNQQ\#ESZP.18Y8

```

Figure V-1. TRANSFER Output in PIMS8 Format.

```

ORIGINAL FILENAME: E21902.DA
DATE: 8/ 7/1977
SCAN NUMBER: 7
PHOTONS PRESCALED BY: 1
IONS PRESCALED BY: 1
STARTING WAVELENGTH: 580.000 ANGSTROMS
ENDING WAVELENGTH: 1650.025 ANGSTROMS
WAVELENGTH INCREMENT: 0.075 ANGSTROMS
MANDATORY INTEGRATION TIME: 2.0 SECONDS
MAXIMUM INTEGRATION TIME: 300.0 SECONDS
DESIRED TOTAL COUNT: 30000.
ONE DARK MEASUREMENT EVERY 300 DATA POINTS
DARK INTEGRATION TIME: 30.0 SECONDS
ELECTROMETER SETTLING TIME: 10.0 SECONDS
FIRST BACKGROUND WAVELENGTH: 565.00000ANGSTROMS
LAST BACKGROUND WAVELENGTH: 1125.00000ANGSTROMS
BACKGROUNDINTEGRATION TIME: 60.0 SECONDS
REFERENCE WAVELENGTH: 800.00000 ANGSTROMS
REFERENCE INTEGRATION TIME: 30.0 SECONDS
ONE REFERENCE POINT EVERY 150 DATA POINTS
PHOTON FULL SCALE: 9000. CPS
ION FULL SCALE: 90000. CPS
REAL-TIME PLOT SCALE: 25.00 ANGSTROMS/INCH
8 0.5649999E+03 0.8771796E+03 0.1455179E+04 0.42150958E+03 0.47800000E+02
1 0.5799999E+03 0.1046966E+04 0.2054016E+04 0.49038875E+03 0.44300000E+02
1 0.5800749E+03 0.1053047E+04 0.2078826E+04 0.5364636E+03 0.43100000E+02
1 0.5801499E+03 0.1058363E+04 0.2103387E+04 0.5801985E+03 0.4419999E+02
1 0.5802249E+03 0.1066199E+04 0.2124140E+04 0.6238934E+03 0.40400000E+02
1 0.5802998E+03 0.1071608E+04 0.2151980E+04 0.6662382E+03 0.41600000E+02
1 0.5803749E+03 0.1077235E+04 0.2160961E+04 0.7072831E+03 0.4069999E+02
1 0.5804499E+03 0.1085995E+04 0.2203243E+04 0.7484680E+03 0.38800000E+02
1 0.5805249E+03 0.1093536E+04 0.2218273E+04 0.7882629E+03 0.36300000E+02
1 0.5805998E+03 0.1100051E+04 0.2258044E+04 0.8258578E+03 0.34000000E+02
1 0.5806749E+03 0.1107705E+04 0.2286647E+04 0.8610527E+03 0.34000000E+02
1 0.5807499E+03 0.1115411E+04 0.2303647E+04 0.8950975E+03 0.34000000E+02
1 0.5808249E+03 0.1123074E+04 0.2330488E+04 0.9295324E+03 0.34000000E+02
1 0.5808999E+03 0.1130192E+04 0.2366730E+04 0.9625773E+03 0.3119999E+02
1 0.5809749E+03 0.1138965E+04 0.2401253E+04 0.9941722E+03 0.3189999E+02
1 0.5810499E+03 0.1146735E+04 0.2433539E+04 0.1024717E+04 0.29100000E+02
1 0.5811249E+03 0.1154842E+04 0.2471263E+04 0.1053562E+04 0.28500000E+02
1 0.5811998E+03 0.1163093E+04 0.2480597E+04 0.1081746E+04 0.27800000E+02
1 0.5812750E+03 0.1169469E+04 0.2526713E+04 0.1109241E+04 0.2829999E+02
1 0.5813499E+03 0.1177573E+04 0.2547316E+04 0.1137636E+04 0.2719999E+02
1 0.5814249E+03 0.1184444E+04 0.2577777E+04 0.1164331E+04 0.2609999E+02

```

Figure V-2. TRANSFER Output in ASCII Format.

E. Data Acquisition Options

This section describes in some detail the options and procedures used to acquire data under the PIMS8 system. The first part describes the procedures used when the RECORD option is used to acquire data; succeeding parts contain some of the detailed considerations influencing RECORD's special features: variable integration time, periodic correction measurements and real-time data plotting. In the final part of this section, the procedures used under the TIMED ACQUISITION option are described and compared with those used in RECORD.

1. The RECORD Option

When the RECORD option is chosen, the user is asked to supply a number of parameters, the first of which is the storage device and file name. These names are checked in the same way output file names are checked in the TRANSFER option, and if they are all right, the user is asked for the wavelength parameters for the run. The desired initial and final wavelengths are entered and checked as in MOVE and SCAN. The user is then asked to supply the wavelength increment to be used for each point. The user may elect to acquire a number of points at the same wavelength. If the initial and final wavelengths he specifies are the same, he is asked to supply the number of points to be recorded. If points are to be recorded at different wavelengths, the system calculates the number of points that will be necessary to span the desired wavelength range at the desired increment. The actual final wavelength may exceed the desired value since the difference between the desired initial and final wavelengths may not correspond to an integral number of wavelength increments.

The next group of parameters the user specifies are those affecting the variable integration time feature of RECORD. They include the minimum, or mandatory, integration time, the maximum allowable integration time, and the desired total count, as well as the type of data to be recorded (photon data only, ion data only, or both).

The third group of parameters to be specified controls the periodic acquisition of various types of correction data. Since the photon electrometer must have some offset, background measurements are always necessary, and the user must specify the integration time for such measurements, as well as the frequency with which they are to be made. The computer also asks the user to supply a settling time which determines how long it waits for the electrometer to settle after closing or opening the shutter, etc. Reference measurements may not be necessary, so the user is asked whether they should be made. If so, the reference wavelength, integration time, and frequency of measurement must be specified. Similarly, stray light correction measurements parameters are only asked for if the user indicates that such measurements should be made. Stray light measurements can be made at either one or two wavelengths. The computer asks the user to supply two wavelengths; if they are the same, only one measurement is made. The user also supplies the integration time to be used for stray light measurements. Because such measurements provide only approximate corrections at best, there is no point in making them very frequently, and they are made only at the beginning and end of each experiment (if at all). Accordingly, the user does not have to specify the frequency with which the stray light measurements are made.

Finally, the user is asked to enter the highest count rates

expected for photons and ions during the run. (If only one type of data is to be recorded the computer asks for that count rate only.) These parameters are only used to scale the real time plot. The user is also asked to specify the abscissa scaling factor, either in points per inch or Angstroms per inch, depending on whether the points are to be acquired at the same or different wavelengths.

All of the above parameters are checked as they are entered to make sure they conform to the physical and software limitations of the system. Due to the discrete nature of the available wavelength or time intervals, the requirements specified by the user are often impossible to meet exactly. To make sure that nearest-increment agreement is satisfactory, as well as to give the user a chance to review all parameters, the actual parameters the system will use are reported on the terminal. If they meet with the user's approval, the various interfaces are checked to make sure they are not in a manual control mode, the run parameters are stored in the output file, and the user is allowed to append any comments he wishes.

A simplified flowchart of the actual data acquisition procedure, which commences as soon as the user finishes entering comments, is shown in Figure V-3. Every run begins with measurements for background, stray light and reference corrections (unless the run parameters indicate one or more of these measurements is not to be made). The variable integration time procedure used with each data point is simple: the signals are integrated for the minimum time, then checked to see if the desired total number of counts (CDES) has been accumulated. If not, a simple calculation is performed to decide whether CDES counts could be reached

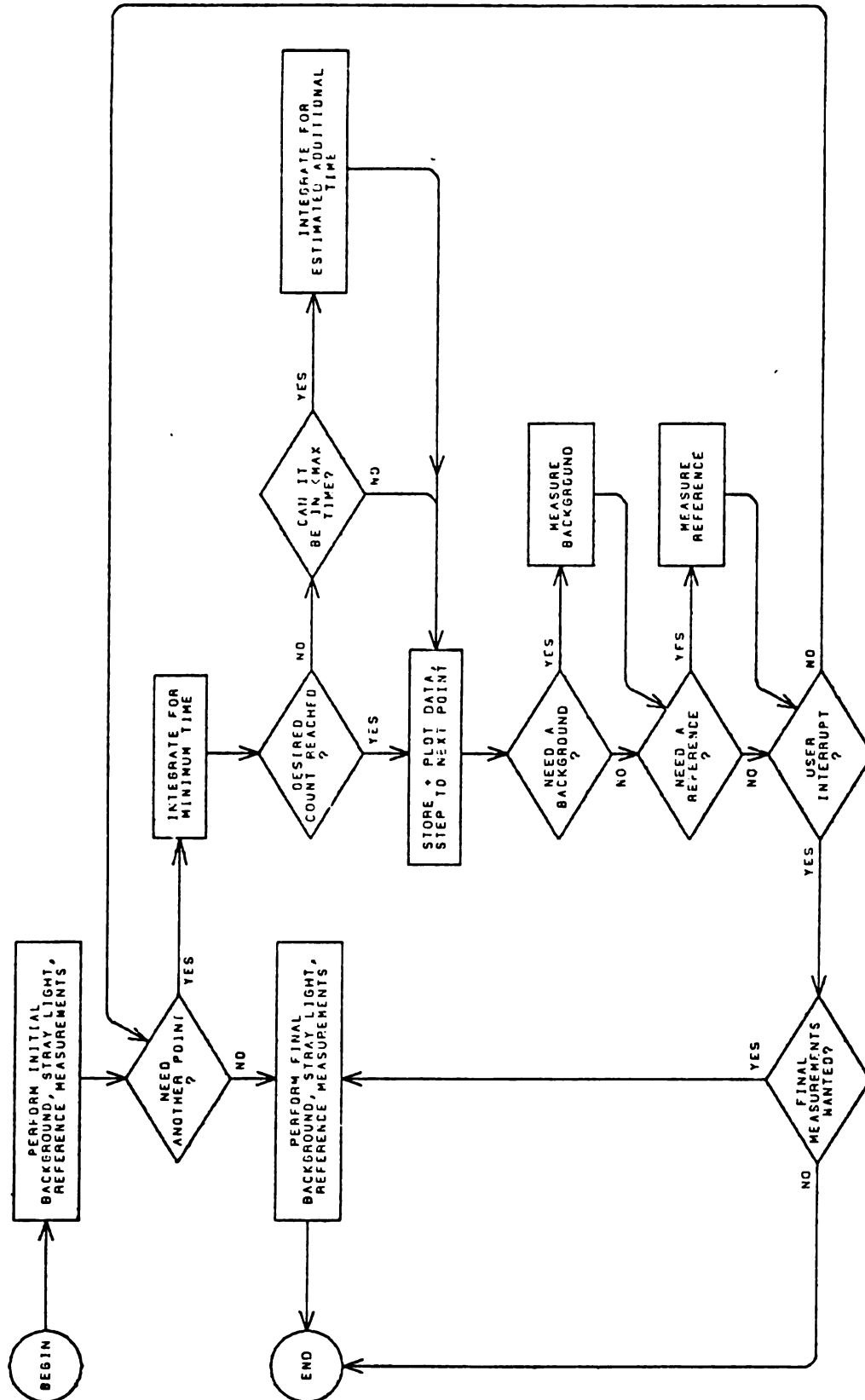


Figure V-3. RECORD Data Acquisition Flowchart.

if the integration time were to be extended to a total time not exceeding the maximum allowable time. If so, the signals are integrated for the necessary additional time. If CDES counts cannot be reached in the maximum allowable time, or if they have already been accumulated, the data are used as they stand after the minimum integration time. The decision about whether the desired number of counts have been accumulated is based on the ion counts unless only photon counts are being recorded. It is shown in the next part of this section that this procedure results in photon, ion or photoionization efficiency data being recorded at a nearly constant signal-to-noise ratio.

After each data point is acquired, it is stored and plotted. A check is made to determine whether another reference or dark signal measurement is needed; if so, the appropriate measurements are made and the resulting data stored. Before acquisition of another data point is started, the terminal is checked in case the user has requested that the rest of the run be aborted (by typing a CONTROL G). When all data points have been accumulated, final background, stray light, and reference measurements are made. If program execution is being ended because of user intervention, the user is given the opportunity to skip these last three measurements, since they often take a significant amount of time, which would be wasted if the user intends to throw out the rest of the data anyway.

When all data have been acquired and stored, the output file is closed and control is returned to the main PIMS8 program, which awaits the user's next option choice.

2. Variable Integration Time

The variable integration time (VIT) technique used with the MSU photoionization mass spectrometer is based on the assumption (discussed later) that the signal-to-noise ratio (S/N) of a measured photoionization efficiency is at least approximately proportional to the square root of the number of ion counts accumulated during the measurement.¹⁴ Therefore, the quality of the data can be increased by increasing the integration time.

The concept of intentionally varying the integration time so as to reach some reasonable tradeoff in S/N versus experiment duration is certainly not new. However, it is most often applied on a scan-by-scan basis, the integration time being set to a single value for the entire scan. As mentioned in Chapter I (Section E), VIT can be much more valuable if it can be applied on a point-by-point basis, often resulting in much more efficient use of experimental time. Any such technique is worth applying only when the time it saves is more than the time taken by the extra decision-making involved in setting the integration time at each point. This would be the case in any experiment where integration times are very long, or where decisions can be made in a relatively short time. The former is true of almost any PIMS experiment, and the latter is usually true of any computerized instrument. Thus,

¹⁴If the ion counts come from a pulse counter, then the proportionality constant is one. If the "ion counts" are instead output pulses from an analog-to-frequency converter such as the V-F converter described in Chapter IV, Section E, then the proportionality constant is likely to be far different. In all further discussion, it will be assumed that pulse counting techniques are being used, although it should be emphasized that the variable integration time technique described here is applicable to, and works well with, the V-F converter measurement system as well as the pulse counting system.

the decision to employ VIT on a point-by-point basis was easily made.

One way in which VIT could be implemented would be simply to measure the time that it takes to accumulate some desired number of counts. This simple method has the disadvantage that it includes no provision for recognizing points that are so weak in intensity that they would normally be considered worthless, but would instead attempt to record such points with the same S/N as all other points. This could result in extremely long integration times being used for data of little importance. The slightly more complex implementation of the VIT method used in RECORD has the advantage that such extremely weak points can be recognized after the first, relatively short, integration period; they are then acquired at a lower S/N.

In most PIMS experiments, it is the photoionization efficiency which should be acquired at a constant S/N, not the photon or ion intensities. This is easily shown to be equivalent, however, to acquiring the data in such a way as to maintain a constant S/N for the ion intensity data. The relative standard deviation of the PIE can be found very simply from the fact that the square of the relative standard deviation of a quotient is equal to the sum of the squares of the relative standard deviations of the divisor and dividend [140]:

$$\text{PIE} = I/P \quad \text{V-1}$$

so

$$\left(\frac{\sigma_{\text{PIE}}}{\text{PIE}}\right)^2 = \left(\frac{\sigma_I}{I}\right)^2 + \left(\frac{\sigma_P}{P}\right)^2 \quad \text{V-2}$$

Photoionization cross sections and typical operating pressures guarantee

that the detected photon flux will always be several orders of magnitude larger than the detected ion flux. Thus, the relative standard deviation of the photon data (whether measured by pulse counting or analog techniques) will be very much less than that of the ion data. Therefore,

$$\left(\frac{\sigma_{\text{PIE}}}{\text{PIE}}\right) = \left(\frac{\sigma_{\text{I}}}{\text{I}}\right) \quad \text{V-3}$$

If the usual definition of the S/N of a measurement (the ratio of the measured number to its standard deviation) is adopted, then the above equation gives

$$(S/N)_{\text{PIE}} = (S/N)_{\text{I}} \quad \text{V-4}$$

The VIT procedure used in the RECORD option attempts to integrate both photon and ion signals until a constant, desired ion count is reached. The ion count which is used in these calculations is not the total ion count, which includes contributions from background and stray light, but rather an ion count which has been approximately corrected for these factors. This is equivalent to integrating to a constant PIE signal-to-noise ratio, assuming that

$$(S/N)_{\text{I}} = \sqrt{\text{Integrated Ion Count}} \quad \text{V-5}$$

The above expression is not expected to be exactly true in the presence of background or stray light contributions to the signal, the exact expression being considerably more complicated (see reference [134] and Appendix D). However, it is shown in Appendix D that, under typical PIMS operating conditions, little error will be caused by assuming it to be valid. Appendix D also shows that somewhat more error is involved if

the total, uncorrected ion count is used as the basis for the VIT technique, as is often the case when point-by-point VIT is implemented on manual PIMS instruments (see Chapter I, Section F).

The ability of the variable integration time technique to obtain reasonable data efficiently when applied on a point-by-point basis is graphically illustrated in Figure V-4. Curve a in that figure was obtained using the variable integration time method described here. Curve b was obtained with a constant integration time equal to the minimum time used in obtaining curve a. As expected, both curves show a roughly comparable S/N near 800 Å, the peak of the lamp emission continuum (see Figure II-4). However, at lower and higher wavelengths, where the lamp emission is not so intense, curve a maintains a very nearly constant S/N, while that of curve b deteriorates. Better results could certainly be obtained using longer constant integration times, but the S/N of any PIE curve so obtained will still vary considerably. In order to obtain a PIE curve at least as good as that of curve a, the constant integration time would have to equal the longest time used in obtaining curve a, and the entire experiment would last for about 440 hours! Shorter constant integration times would allow shorter experiments, and would provide a better S/N than curve a near 800 Å, but a worse S/N at lower and higher wavelengths.

3. Periodic Correction Measurements

The use of Equation V-5 for calculating the S/N of the ion signal measurement amounts to assuming, among other things, that the ion background is negligible. Although often so small as to cause only very small

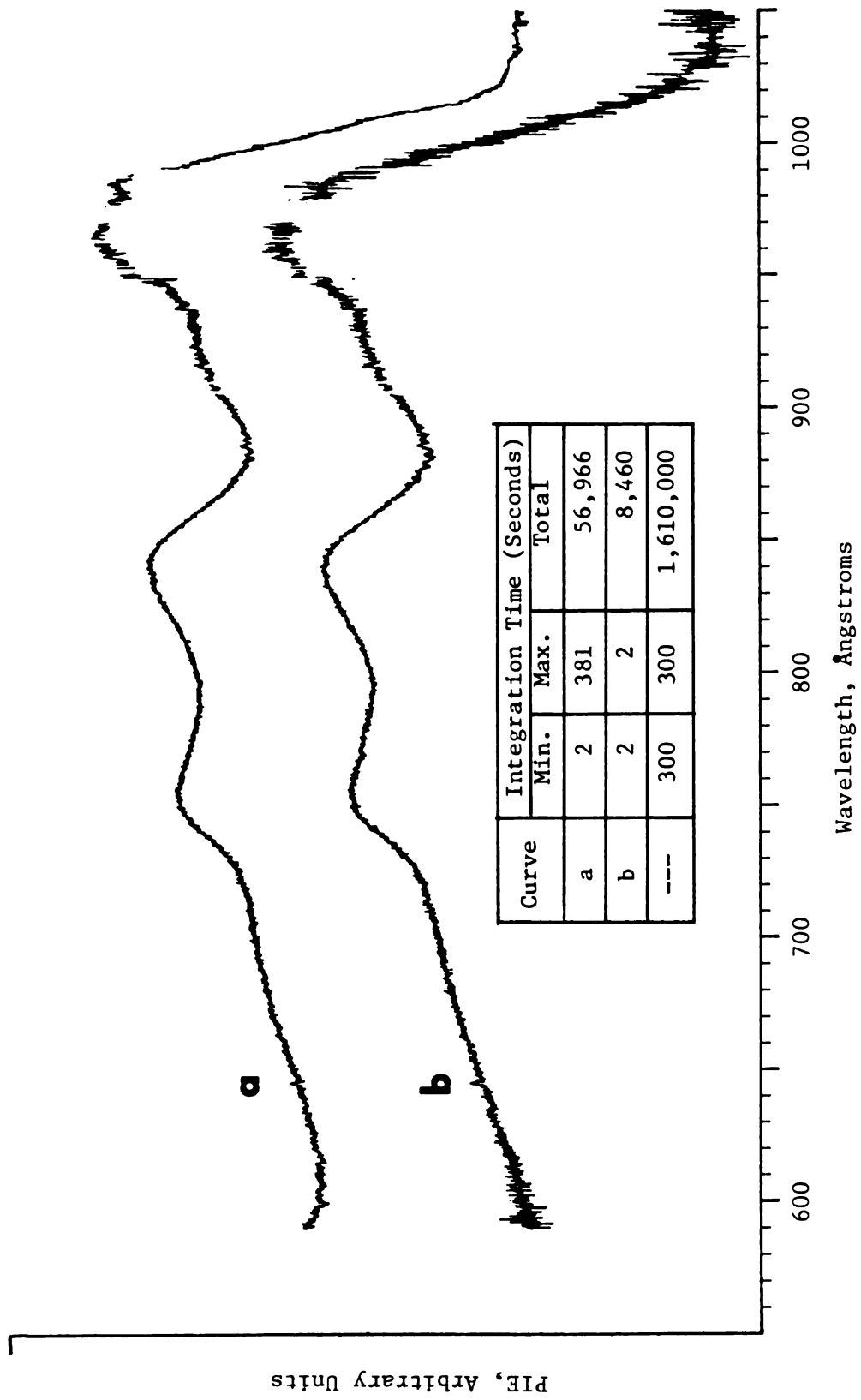


Figure V-4. Comparison of Variable and Constant Integration Times.

error in the S/N calculation, the ion background is not exactly zero, and the photon background (which includes the electrometer offset) is far from zero; these backgrounds must be measured in order to calculate an accurate photoionization efficiency.

If the backgrounds were constant, they would have to be measured only once. Unfortunately, they are usually not constant, and even if they were, slow drifts in the measurement system electronics would necessitate periodically remeasuring the apparent backgrounds [141,142]. When the signal to be measured is comparable in magnitude to the background, measurement time is spent most economically if equal and alternate time periods are spent measuring background and (background + signal) [143]. However, the background signals in PIMS are frequently much smaller than the desired ion or photon signals, so that a much lower fraction of the time need be spent measuring backgrounds.

The primary purpose for making periodic reference measurements is to allow the PIE curve which is later calculated from the data to be corrected for variation in sample pressure. Such variation is bound to occur, either because of a depleting sample reservoir or because of thermal or hysteresis effects in the variable leak which controls flow of sample into the ion source. The reference measurements also serve, however, to correct for slow changes in the relative sensitivities of the ion and photon transducers, as well as for drifts in the data acquisition electronics.

Both background and reference measurements need to be made often enough to adequately sample the changes for which they are intended to correct. Ideally, one would like to make such correction measurements at regular time intervals. However, since the programmable half of the

real-time clock (see Appendix C) is used for other measurements, and the desired time intervals are usually too long to measure easily with the decade half of the clock, the correction measurements are instead made after every N points, where N is specified by the user, and need not be the same for background and reference measurements. This method is slightly less satisfactory, although it is relatively easy to choose N so that the frequency of the measurements is adequate.

A second limitation of the presently used procedure is that the wavelength must be changed in order to make a reference measurement. Such changes, if made frequently, can add quite a bit of time to the duration of the experiment. As mentioned in Chapter IV, the planned addition of an interface for the capacitance manometer so that the pressure can be measured directly will be very useful, since wavelength changes to correct for the much slower detector aging can then be made much less frequently.

4. Stray Light Correction Measurements

It is shown in Appendix A that stray light effects can grossly distort both the broad and detailed shape of a measured PIE curve, and that it is therefore imperative that the data be corrected for stray light and ions due to stray light. Stray light arises from a number of sources, and its distribution depends on both the monochromator and the grating [36]. It can be roughly divided into two categories: "near scatter", which has a distribution of wavelengths near that to which the monochromator is set, and "far scatter", which consists of those wavelengths very different from the nominal wavelength setting of the monochromator.

Unfortunately, stray light and ions due to stray light are very difficult to measure, although the effects of stray light are sometimes very easily seen (see Appendix A). The data of Figure II-13, for instance, give a general indication of the far scatter distribution for the MSU instrument when used with the helium lamp, but can provide no information about the amount of near scatter, which cannot be distinguished from the continuum. Thus, although RECORD includes provisions for "measuring" stray light and stray light ions, such measurements are bound to be somewhat inaccurate, and provide only a starting point for later, empirical corrections that are made during data reduction (see Section F). Accordingly, if made at all, they are made only at the beginning and end of the experiment.

Typically, stray light corrections are most important when the intensity of the stray light is comparable to that of the "non-stray" light. This will be especially true near the extremes of the light source continuum (e.g., near 600 Å and 1000 Å with the helium lamp). As indicated in Figure II-13, the relative amount of stray light may be quite different at the two extremes, and so provision is made for measuring the stray light at two wavelengths (e.g., 500 Å and 1120 Å). For the purposes of the approximate corrections made during RECORD, a linear interpolation between (or extrapolation from) the two measurements is used. If only one stray light wavelength is used, then the measurement at that wavelength is used as a constant correction.

5. Real Time Output

The approximate photoionization efficiency curve which is plotted during data acquisition is not corrected for variations in reference

measurements. The only way this could be done would be to use the most recent reference measurement to calculate a correction factor, since the next reference measurement would not yet be available for interpolation. The resulting stepwise correction of the data has been found to introduce artificial steps in the real-time plot, which tend to confuse or obscure real structure in the PIE curve, and to make it difficult for the user to check on the quality of the data. The real-time plots are, however, corrected for background and stray light.

Figure V-5 shows strip chart recordings of the photon and ion signals during a constant-integration-time experiment on CF_3Cl^+ , as well as the real-time output from the experiment. Just prior to this experiment, an air leak had contaminated the helium supply for the lamp, and the atomic impurity lines observed are roughly an order of magnitude more intense than usual, resulting in a rather severely distorted real-time PIE curve (only partially corrected for stray light). Even though this real-time plot is among the worst plots obtained to date, it clearly gives a much better indication of the behavior of the PIE curve and the quality of the data being obtained than do the analog photon and ion signals. The comparison is even more favorable for low-intensity ions (which exhibit even more noise on a strip chart recording), for experiments using the hydrogen lamp (during which the photon intensity varies rapidly with wavelength, but stray light is less of a problem), or for experiments using variable integration time (which results in a non-linear wavelength scale on most strip chart recordings).

6. The TIMED ACQUISITION Option

The TIMED ACQUISITION option was added to the PIMS8 system for use in cases where variation in the time between points cannot be tolerated,

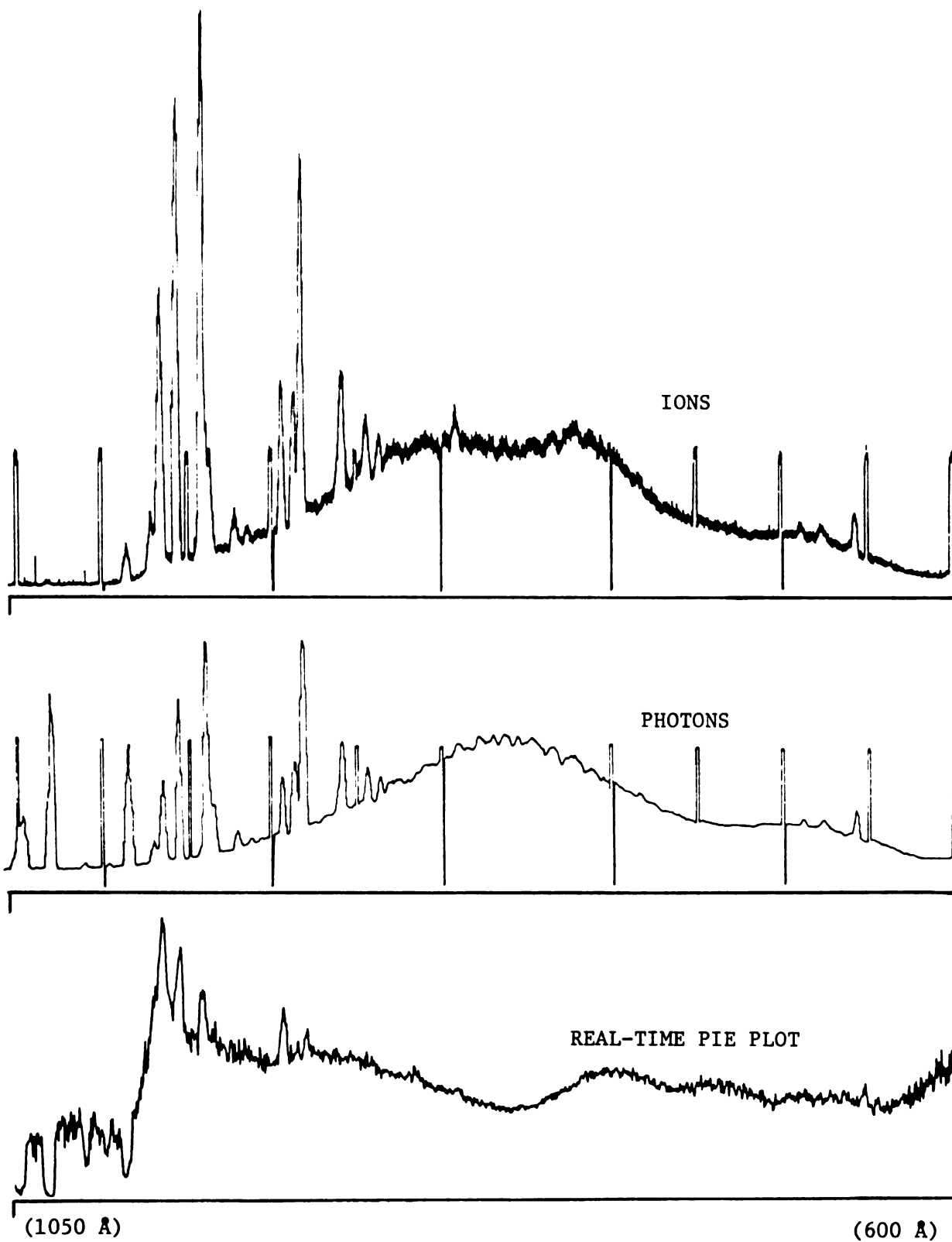


Figure V-5. Comparison of Photon and Ion Stripchart Recordings with PIMS8 Real-Time PIE Plot. CF_3Cl^+ from CF_3Cl (compare Figure VI-1). The regularly-spaced "spikes" in the photon and ion curves result from periodic reference and dark measurements.

and where the amount of time spent between points on data storage or other non-data acquisition chores should be as small as possible. It is used routinely for the acquisition of mass spectra, but is also useful for instrument characterization and performance tests.

TIMED ACQUISITION is a much simpler, less versatile option than RECORD. The major differences between the two options are described below:

- a) Limited number of points: Writing out the acquired data on a point-by-point basis requires very little time for most points, as they are actually stored in an output buffer by the OS/8 system. However, when the buffer becomes full, the entire buffer is written to the output device, which takes a significant amount of time. To avoid such variation, TIMED ACQUISITION uses its own, larger buffer, and limits data acquisition to the number of points that will fit in it (currently 550). The data are not actually stored on any mass storage device until the user requests that storage be done.
- b) No variable integration time: This feature is not available, since it conflicts with the requirements of constant time between successive points.
- c) No wavelength change: Wavelength cannot be changed between points, since to do so would require a significant amount of time. Also, the primary purpose for which TIMED ACQUISITION was written (acquiring mass spectra) does not require wavelength changes during a run.
- d) No real-time output: Plotting a data point can also take a significant and variable amount of time, even if a fast device such as a Tektronix terminal is used, and even assuming the calculations required to do the plotting would take negligible time. Therefore, no data are plotted during data acquisition, although all data are scaled and plotted immediately afterwards. Since most TIMED ACQUISITION runs are fairly short (less than 5 minutes), this is not a severe handicap.
- e) Correction Measurements: Only one background signal correction measurement is made, at the beginning of each run. No further background measurements are made, nor are stray light or reference measurements possible. However, to ensure correct functioning of the CORRECT option, a "dummy" reference measurement is stored at the beginning of the stored data file, the value used corresponding to the maximum value found during the run.

When the TIMED ACQUISITION option is selected, the user is asked to specify the total time and number of points desired for the run. The desired number of points is checked to make sure it doesn't exceed the internal buffer's storage capacity. The system calculates the integration time per point and the actual total scan time and reports these to the user for approval. If the user indicates that they are satisfactory, he is then asked to enter the type of data to be recorded (as in RECORD). The system checks the various interfaces to make sure they are not in a manual control mode, and then performs a 10 second background measurement. The user is asked to hit the carriage return key when he is ready for the scan to start. After this, the system waits five seconds and then rings the terminal bell three times at one second intervals. This allows the user to synchronize other devices (usually the sweep of the oscilloscope which controls the mass scan) to the start of data acquisition, which begins immediately after the third bell.

When data acquisition is completed, the user is given a choice of returning to PIMS8 to select another main option, repeating the data acquisition with the same parameters, doing a run with new parameters, having the just-acquired data plotted on the terminal, or storing the data currently in the buffer in a data file. If the plot option is selected, the data are automatically scaled to fill the terminal screen. If the data storage option is selected, the device and file names entered by the user are checked in the same way that output file names are checked in TRANSFER and RECORD before the output is accomplished.

F. Data Reduction Options

There are two versions of the PIMS8 data reduction program. One is run on the PDP 8/I computer under the PIMS8 system when the CORRECT option is chosen. The other version is a program called PIMS11, which is run under the DOS/BATCH operating system [144] on either of two PDP 11-type computers in the department.¹⁵ Typically, PIMS8-CORRECT is used when the user desires to move quickly between the data reduction stage of the experiment and the other stages. PIMS11 is used to take advantage of the superior graphics output devices on the two 11-type systems. Since the two programs are functionally equivalent, only one (PIMS11) will be described here.

PIMS11 is written almost entirely in FORTRAN, as described in Reference [144]. The conventions adopted for the PIMS8 system (described in Section B) have also been adopted for PIMS11, with only a few exceptions. The same input prompts are used, and the same special characters. However, both the ↑U and RUBOUT characters are legal after the : and = input prompts, while the <LF> character is not legal after the # prompt. The format to be used for specifying device and file names is also slightly different in the DOS/BATCH system (see Reference [144] and the INPUT entry in Table V-6). One further difference is that, while the user

¹⁵ One system is based on a Digital Equipment Corporation PDP 11/40 computer [96]. It includes a "home-built" X-Y incremental plotter [145] and a Tektronix 4010 terminal [98] as output graphics devices. The other system is the Departmental Cemcomgraph system, based on a Cal Data 135 computer (equivalent to DEC PDP 11/40) [146]. Its graphics output devices include a Versatec 1200A Electrostatic Printer/Plotter [145] and a Vector General 3D3I CRT Display [148].

Table V-6. PIMS11 Options.

Name	Description
***** Parameter Specification *****	
DIVISIONS	Specify the number of tick marks for the X and Y axes of any plots. The number of tick marks will be one more than the number of divisions specified, unless it is a non-integer number. Default = 1.0, 1.0
INPUT	Specify the input filename, in conventional DOS/BATCH format [144] of DEV:FILNAM.EXT[UIC], where DEV is a device name of three characters or less, FILNAM is the file name (six characters or less), EXT is an optional extension of three characters or less, and UIC is an optional user identification code. No default value.
ISTRAY	Specify new, or modify existing, USI data. PIMS11 will give the user the option of correcting a single entry, or entering all new data for the USI array. If the latter is chosen, PIMS11 will ask for the number of entries and whether the abscissa data will be in units of wavelength, energy, or time; it then will provide the prompts for entering the actual data. If only one entry is to be changed, PIMS11 will report the present value of the X, Y pair before asking for the new data. Default: No entries.
METHOD	Specify the method to be used to find the BCE of one of the twelve data types listed in Table V-7. PIMS11 asks for the number of the data type, and then the method to be used, which is specified by the first letter of its name: No Correction (Default) Latest Value Polynomial Interpolate If Polynomial is chosen, PIMS11 checks to make sure a FIT has been performed on the relevant data type. If Interpolate is chosen, it checks to see whether the necessary data is available in the computer memory. If not, the PREPARE option is automatically initiated for the desired data type.
PSTRAY	Analogous to ISTRAY, except the USP data array is affected. Default: No entries.

Table V-6. (Cont'd.)

Name	Description
SIZE	Specify the X, Y dimensions for any succeeding plots on either the graphics terminal or the plotter. These dimensions are in inches. Default = 8.0, 6.0
SKIP	Specify the number of points PIMS11 is to skip, between points which are plotted, listed, etc. E.g., if the user specifies 2 in this option, PIMS11 will plot only every other one of the data points which meet all other requirements (are within data and window extrema, etc.). Default: Do every point.
WINDOW	Specify the coordinates which are to correspond to the extreme edges of the plot. If the X-axis window extrema are set to 600 Å and 1000 Å, and the default plot size is used, then the X-axis scale factor will be 50 Å/inch. If window limits have never been specified by the user, then PIMS11 sets them equal to the current data extrema (specified in XLIMIT, YLIMIT) for each plot. Once set by the user, they remain set until changed. The user can, however, cause PIMS11 to return to automatic matching of the data extrema by specifying 0.0 for both window X-axis extrema. Default: Match data extrema.
XAXIS	Select the units of the abscissa: 1: Wavelength, in Ångstroms. (Default) 2: Elapsed time, in seconds. 3: Photon energy, in electron volts.
XLIMIT	Select new X-axis data extrema. When performing an execution type option (except a SEARCH of data or residuals), only data falling within this range will be considered. When plotting, data must be within both the window extrema and the data extrema to be plotted. Initial values = 0.0, 0.0 (No default values)
YAXIS	Select the type of data to be used for the ordinate in any plots, listings, etc. See Table V-7 for a description of the possible data types. Default: PIE (type 12).
YLIMIT	Similar to XLIMIT, but allows the Y-axis data extrema to be changed. Initial values = 0.0, 0.0 (No default values).

Table V-6. (Cont'd.)

Name	Description
YMOD	Select the Y-axis type modifier: 1: Data. (Default) 2: Fit. 3: Residuals. (Data - Fit) For instance, if YAXIS is to be photon intensity (PHO), and data is chosen for YMOD, then the actual (corrected) photon data will be plotted, listed, etc. However, if a fit is chosen for YMOD, then the output data will be calculated from the coefficients of the most recent polynomial equation fit to photon data. Current data and window extrema will apply, but the calculated data will not be further corrected. If residuals is chosen for YMOD, the difference between the (corrected) photon data and the calculated curve will be used.
***** Informational *****	
CURRENT	PIMS11 prints on the terminal the currently selected Y-axis data type and type modifier, the X-axis data type and the data and window extrema.
LOCATE	The graphics terminal cursors are called up, and may be moved by the user. When any key on the keyboard is struck, PIMS11 returns the current position of the cursors.
PARAMETERS	PIMS11 asks for an output file specification, in the same format as that used in INPUT for input files. (The lineprinter and keyboard may be used, here.) It then places in that file any of the current parameters affecting the currently-selected data type. It includes the current data and window extrema, as well as the correction methods to be used. For interpolative corrections, it includes the data stored in memory between which interpolations are performed. For polynomial corrections, it includes the values of the coefficients
***** Execution Options *****	
FIT	Read through the input file (or the data array in the case of USP or USI) searching for all data of the desired type that falls within the current X-axis data limits. Any such data found is to be corrected according to the current parameters, and is then to have a

Table V-6. (Cont'd.)

Name	Description
	polynomial equation fit to it. The order of the equation (≤ 9) is supplied by the user. The resulting coefficients and scaling parameters are stored for later use.
LIST	Similar to FIT, except the corrected data are listed on the lineprinter (and, of course, no fit is performed).
PLOT	Similar to FIT, except that the corrected data, if they fall within both data and window extrema are plotted.
PREPARE	Similar to FIT, except that the resulting data are stored in an array in the computer memory for easy access by PIMS11 when needed for interpolative type corrections. This option is automatically executed the first time the user selects the interpolative method for a given data type, in option METHOD. After that time, if the user wishes to change the data stored in memory for that data type, he must select this option himself. (If he merely chose the interpolative method again, PIMS11 would check to see if it had the required data. Finding data present in the array, it would not find it necessary to input any new values. When this option is chosen, PIMS11 is forced to replace the existing data with the newly-calculated data.)
SEARCH	Y-axis modifier = Data or Residuals: PIMS11 will go through the input file, as in FIT, keeping track of the largest and smallest abscissa and ordinate values encountered. When finished, it reports the extrema to the user and sets the data extrema equal to the new values. Y-axis modifier = Fit. The input file is not read, but ordinate data is calculated from the appropriate equation at one hundred points between the current X-axis data limits. The results are used to set new Y-axis data extrema, and the (old) X-axis and (new) Y-axis extrema are reported to the user.
STORE	Similar to LIST, except that PIMS11 first asks for an output device specification (as in PARAMETERS), and stores the data on that device (which may be the keyboard or lineprinter).

Table V-6. (Cont'd.)

Name	Description
***** Miscellaneous Options *****	
ERASE	Erase the screen of the graphics terminal. If a plot is not erased before another is done, the second is superimposed on the first.
QUIT	Terminate execution of PIMS11 and return to the DOS/BATCH keyboard monitor. Note that in PIMS8 CORRECT, this option is replaced with "PIMS8", which returns the user to the main PIMS8 option list.
RESTART	Restore all options to their default values.

is always expected to enter one number in response to the : or = prompts in PIMS8, PIMS11 will sometimes ask for more than one number in a single prompt. The user must then enter the first number, a comma, the second number and then the "end-of-line" carriage return.

PIMS11 accepts as input a file of raw, uncorrected data acquired under PIMS8.¹⁶ It calculates from these data a series of data points (X,Y pairs) which are output in one of three forms: as a lineprinter listing, as a plot, or as a new file suitable for plotting or listing at a later time. The abscissas of the output points may correspond to wavelength, photon energy, or elapsed time. The ordinates may correspond not only to photoionization efficiency, but also to any other type of data acquired during the experiment (photon background, ion reference, etc.), and may be corrected for variations in any other relevant data type.

The user controls the execution of the program by choosing from a list of twenty-five options, which are listed and described in Table V-6. Thirteen of these options allow the user to set execution parameters which control the calculations made by the computer. For instance, the user can select the type of data to output, which of the possible corrections should actually be made to the data before it is output and how they are made, and so forth. These options need only be selected if the user wishes to change the corresponding parameters from their default values.

¹⁶ Although the input data are in the PIMS8 format (see Table V-4), they must be stored on a medium which can be read by the DOS/BATCH system, since a floppy disc written on an 8-type computer cannot be read directly on the 11-type computer. The conversion from an 8-formatted floppy disc to an 11-formatted file on one of the 11-type computer's mass storage devices is accomplished with a program called PIP8, which runs under DOS/BATCH and was written by Dr. T. V. Atkinson.

Three of the options are informational, allowing the user to find the coordinates of features on data plots or to review the current status of certain parameters. Three miscellaneous options allow the user to terminate data reduction, restart the program (reset all parameters to their default values) or erase the graphics terminal screen.

When one of the options in the final group of six is selected, PIMS11 executes one pass through the input file; the details of the resulting operation depend on the option chosen, but are similar for all six. PIMS11 reads through the input file a line at a time. When a line is encountered which contains the type of raw data needed to calculate the desired ordinate data, the corresponding abscissa value is checked to make sure it is within limits specified with XLIMIT. If not, PIMS11 continues through the input file, but if the data are within the specified limits, the desired ordinate is calculated from the raw data. The calculation includes whatever corrections are indicated by the current parameters. Once the ordinate is calculated, the data point may be plotted, stored or listed. It may also be passed to subroutines which are used to find data extrema, to fit the data to a power series polynomial, or to prepare internal data arrays for use in later correction calculations on other data types.

There are twelve possible choices for the ordinate data type. These are described in Table V-7, which also includes (in column 4) the types of raw data from which the ordinate type is calculated and (in column 5) the other types of data which may be used as corrections in the calculations. For example, if the user wishes to examine a plot of reference photon intensity measurements (RP) versus time, the values plotted may be corrected for stray light or background, and may also be corrected

Table V-7. PIMS11 Data Types, Their Possible Corrections and BCE Calculation Methods.

1	2	3	4	5	6
No.	Symbol	Name and Description	Based On ^a Raw Data of Type	May Be Corrected For	Forbidden ^b BCE Method
1	USP	User-entered Stray Photon correction. May be used to adjust photon intensities for stray light empirically. See text and Appendix .	---	---	L
2	USI	User-entered Stray Ion correction. Similar to USP, but adjusts ion intensities for stray light effects.	---	---	L
3	BP	Photon Background. Photon signal measured with the shutter closed. If pulse counting is used, it is essentially the photon transducer dark count. If DC current measurement is used, it includes also the electrometer offset.	BP	---	---
4	BI	Ion Background. Ion signal analogous to BP	BI	---	---
5	SP	Stray Photons. This signal is measured with the shutter open, but at a wavelength where the lamp is not expected to produce any measurable output.	SP	EP	---
6	SI	Stray Ions. The ions signal measured at the same time as SP; supposedly the ions created by the stray light photons.	SI	BI	---
7	RP	Reference Photons. The photon intensity measured at the reference wavelength during a reference correction measurement.	RP	SP, BP, USP	---
8	RI	Reference Ions. The ion signal measured simultaneously with RP.	RI	SI, BI, USI	---
9	RPIE	Reference PIE. The PIE calculated from RP and RI. Not used as a correction for any other data type.	RP, RI	SP, BP, USP SI, BI, USI	N,L, P, I
10	PHO	Photon Intensity.	PHO	RP, SP, BP, USP	I
11	ION	Ion Intensity.	ION	RI, SI, BI, USI	I
12	PIE	Photoionization Efficiency. Ratio of ION/PHO.	PHO, ION	RP, SP, BP, USP RI, SI, BI, USI	N,L, P,I

^aThat is, data of the type given in Column 2 may be calculated for plotting, storing, etc., only when data of the type given in this column is encountered in the input file.

^bWhen finding the best current estimate of the data type given in Column 2, the methods listed in this column may not be used. In the case of USP and USI, this is because the "raw data" are not in the input file. In the case of PHO and ION, this is because of lack of storage space in memory and the large number of PHO and ION data points in a typical file. RPIE and PIE are never used to correct any data type, and hence all four methods of finding their BCEs are inappropriate.

^cUSP and USI must be entered by the user via the keyboard; they are never found in a stored data file.

with an empirical, user-entered correction factor. When PIMS11 encounters a line in the input file which contains raw RP data, it corrects that data using what shall be termed the "best current estimate" (BCE) of the value of SP, BP, and USP. The rather simple calculations performed are summarized in Table V-8.

Each of the best current estimates required for a given calculation may be obtained in one of four ways, chosen by the user via the METHOD option:

- a) N--No Correction: The ordinate data are not corrected at all for the relevant data type. This amounts to setting the BCE equal to 1.0 for RP or RI, or equal to 0.0 for any other data type.
- b) L--Latest Value Correction: The ordinate data are corrected using the latest value of the relevant data type encountered in the input file. The latest value will itself have been corrected for any other relevant data types, depending on the current execution parameters. The same BCE is used in all calculations until a new datum of the relevant type is encountered in the input file. Since USP and USI data do not exist in the input file, this method cannot be used to obtain their BCEs.
- c) P--Polynomial Correction. The BCE of the relevant data type is calculated from the coefficients of a power series polynomial which has been previously fit to the data of that type.
- d) I--Interpolative Correction: The BCE is obtained using a linear interpolation between (or extrapolation from) the two nearest points of the appropriate type. In order for the necessary data to be available, it must reside in computer memory and not in the input file. This is accomplished automatically the first time this method is chosen for a particular data type, and may be done at any time by use of the PREPARE option (see Table V-6).

The four different methods for evaluating the BCE of a given data type have different advantages, and the decision about which ones to use will depend heavily on the data. In practice, the "no correction" method is seldom used in calculating PIE curves, but is included because it allows the user to plot (list, etc.) the raw data quickly. The latest value

Table V-8. PIMS11 Calculations.^a

Output BP = Latest BP

Output BI = Latest BI

Output SP = Latest SP-BP'

Output SI = Latest SI-BI'

Output RP = Latest RP-BP'-SP'-UBP

Output RI = Latest RI-BI'-SI'-UBI

Output RPIE = RI'/RP'

Output PHO = (Latest PHO-BP'-SP'-UBP)(initial RP/RP')^b

Output ION = (Latest ION-BI'-SI'-UBI)(initial RI/RI')^b

Output PIE = ION'/PHO'

^aA primed symbol (e.g. DP') indicates that the best current estimate of that data type is used in the calculation (see text).

^bInitial RP and initial RI are just the values of RP' and RI' which are in use when the first output point is calculated.

method has the advantage of simplicity, and does not require a previous pass through the input file. Its major disadvantage is that if the corrections being made by this method are not very small, and if succeeding values of the data differ significantly, an artificial step may appear in the output data due to the sudden change in the correction factor used. On the other hand, when the corrections to be made are small or do not change too abruptly, such steps will be too small to observe, and the method may be used.

The polynomial method successfully eliminates artificial discontinuities, but is limited in that it can only be used with data that can be fit reasonably well by a polynomial of order nine or less. In order to check the accuracy of the fit, the YTYPE option allows the user to modify the ordinate data type to plot a curve calculated from the fit equation, the data, or the difference between the two.

The interpolative method also avoids steps in the corrected data, but may induce a slight change in slope. It has the advantage that it can be used with data which cannot be fit to a polynomial.

Besides the corrections based on recorded data from the input file additional corrections may be made using two types of correction factors input by the user (USI and USP). These corrections allow the user to more fully compensate for the effects of stray light, as discussed in Appendix A.

CHAPTER VI

SOME PRELIMINARY RESULTS

A. Introduction

A number of fluorocarbons referred to by such trade names as Freons, Genetrons, etc., have experienced wide use in recent years, notably in closed-cycle refrigeration systems and as propellants in aerosol spray cans. In 1974, Molina and Rowland [149] suggested that the protective ozone layer of the earth's upper atmosphere was being jeopardized by reactions involving chlorine atoms liberated by photolysis of several of these compounds after they had diffused into the upper atmosphere. This suggestion sparked wide-spread debate, both scientific and political, and news stories on the "Freon controversy" were particularly frequent during 1975 and 1976.

Early in 1976, enough of the PIMS instrument described in previous chapters had been assembled to allow the acquisition of crude photoionization efficiency data. Although the reactions of primary interest in the Freon/ozone system involve only neutral reactants and products, and therefore cannot be studied by PIMS, the publicity received by these compounds at that time did bring them to mind as possibilities for the first studies with the new instrument. PIMS studies of CF_4 had been reported by Noutary [150] and by Walter, *et al.* [151], who, in agreement with earlier electron impact studies, found no detectable amounts of the parent ion, but only fragment ions. Werner,

et al. [152] had reported a similar instability for CCl_4^+ , but found CBr_4^+ stable enough to provide a moderately intense signal (about five percent of the signal from the most intense ion). The only one of the "mixed chlorofluoromethanes" (CFCl_3 , CF_2Cl_2 , and CF_3Cl) which had been studied by PIMS was CF_3Cl [150]. In that study, no PIE curves were reported, but only appearance potentials for the parent ion and one fragment ion, CF_3^+ . Ionization potentials for all three of these compounds had been measured by photoionization, but without the benefit of mass analysis [24]. In view of the instability of the parent ions of CF_4 and CCl_4 , however, it was not entirely certain that the measured appearance potentials corresponded to the parent ion. The only appearance potentials available for the other ions from these compounds were electron impact mass spectrometric values, which are frequently of limited accuracy [3]. Therefore, a PIMS investigation of all three of the mixed chlorofluoromethanes was begun with the MSU instrument in an attempt to determine whether the parent ions of CF_2Cl_2 and CFCl_3 are stable, to confirm or improve the accuracy of existing appearance potential measurements, and to measure the appearance potentials of those fragmentation processes which had not yet been studied.

At nearly the same time, workers in two other laboratories began independent investigations, for similar reasons. The results of these other investigations, having been performed on instruments which were completed before the investigations were begun, have already been published. Ajello, Huntress and Rayermann (AHR) [153] have reported PIE curves and appearance potentials for a total of eleven ions from these three compounds, including the parent ions of CF_3Cl and

CF_2Cl_2 . They were unable to detect the parent ion of CFCl_3 at any wavelength. Jochims, Lohr, and Baumgartel (JLB) [154] have reported PIE curves for fifteen ions from these molecules, as well as absorption and resonance photoelectron spectra. Their PIE curves reveal a great deal of broad structure which is not so clear in the relatively noisy data of AHR. In addition, JLB used a synchrotron light source, and so were able to extend their studies to higher energies than AHR, in some cases to 25 eV. JLB were able to detect the presence of the CFCl_3^+ ion using undispersed radiation, but were unable to acquire a PIE curve for this ion.

The results which have been obtained to date with the MSU instrument are reported and discussed in this chapter, including PIE curves for thirteen ions from the mixed chlorofluoromethanes. These data were acquired during the course of the last two years, during which the MSU instrument and the operating procedures used with it have undergone dramatic improvement, and are hence of somewhat uneven quality. Recent runs, even on very weak ions, have produced far better results than the early runs on intense fragment ions. Most of the results presented here are similar to those of JLB and AHR, but the operating procedures used (see Chapter V) have allowed PIE curves to be obtained with even higher signal-to-noise ratios than those of JLB. The improved quality of the data reveals structure which is not always clear, and sometimes not even visible, in the earlier reports. Furthermore, the MSU instrument has proved to be sufficiently sensitive to obtain a PIE curve for the parent ion of CFCl_3 , albeit over a limited wavelength range thus far.

In Section B of this chapter the results which have been

obtained to date are presented, along with a discussion of several aspects of the results common to all three molecules. Discussion which is pertinent to the individual molecules is contained in the next three sections, and a brief summary of the findings is contained in Section F.

B. Results and General Discussion

1. Relative Intensities

The relative intensities of the various ions from CFCl_3 , CF_2Cl_2 , and CF_3Cl are summarized in Table VI-1. The light source for these measurements was the He I emission line at $584 \text{ \AA}$ (21.2 eV), excited by a DC discharge in about two torr of helium. The ion intensities were determined by measuring the DC current from the ion transducer with a picoammeter while varying the mass manually. Those ions with relative intensities less than about 0.5 were measured on picoammeter ranges more sensitive than 10^{-10} amperes. On these ranges, various noise sources produced a large fluctuation in the "DC" current, even when the maximum meter damping was used, making it difficult to estimate the true average current visually. The errors involved in making this estimate are the largest source of uncertainty in these results. Further errors arise from the fact that the results have not been corrected for either mass filter transmission or ion detector sensitivity. At best, these measurements are accurate to within $\pm 10\%$ or so, but relative intensities below 0.5 should be regarded as order of magnitude estimates only.

The observed intensities are influenced by the number of C-Cl

Table VI-1. Relative Abundance^a of Parent and Fragment Ions From CF_3Cl , CF_2Cl_2 , and CFCl_3 .^b

CF_3Cl		CF_2Cl_2		CFCl_3	
CF_3Cl^+	0.2	CF_2Cl_2^+	0.02	CFCl_3^+	0.004
CF_3^+	100	CF_2Cl^+	100	CFCl_2^+	100
CF_2Cl^+	47	CFCl_2^+	14	CCl_3^+	3.5
CF_2^+	5.0	CF_2^+	2.1	CFCl^+	2.8
CFCl^+	0.35	CFCl^+	1.8	CCl_2^+	2.3
CF^+	0.08	CCl_2^+	0.25	CF^+	1.1
CCl^+	(c)	CF^+	0.81	CCl^+	0.11
Cl^+	0.6	CCl^+	(c)	Cl^+	0.02
		Cl^+	0.08		

^aRelative abundance values below 0.5 are not reliable; see text.

^bSum of all chlorine isotope peaks, normalized to most intense ion as 100. Measured at 21.2 eV.

^cAt the mass resolution used, this ion was superimposed on the tail of the larger CF_2^+ peak, 3 AMU higher, and could not be reliably determined.

and C-F bonds in the molecule as well as by the fact that C-F bonds are stronger than C-Cl bonds. The relative bond energies are slightly more important than the number of C-F or C-Cl bonds, as indicated by the fact that the chlorine-loss fragment ion is always the most intense fragment, even for CF_3Cl . The loss of two chlorines is always more probable than loss of one chlorine and one fluorine, which in turn is preferred over the loss of two fluorines. Thus, the intensity of fragment ions from CF_2Cl_2 decreases as $\text{CF}_2^+ > \text{CFCl}^+ > \text{CCl}_2^+$, and production of CFCl^+ from the three compounds decreases in the order $\text{CFCl}_3 > \text{CF}_2\text{Cl}_2 > \text{CF}_3\text{Cl}$. It is interesting to note that the ratio of chlorine loss ion intensity to fluorine loss ion intensity (statistically weighted to account for the number of possible ways of forming each ion) is nearly the same, within experimental error, for each compound.

All three parent ions are formed, but only rarely. The relative intensities indicated in Table VI-1 for the parent ions are not believed to be accurate. It has been our experience in attempting to obtain PIE curves for the parent ions that the general order of intensity should be $\text{CF}_2\text{Cl}_2^+ > \text{CF}_3\text{Cl}^+ \gg \text{CFCl}_3^+$, and this is confirmed by the results of JLB.

2. Photoionization Efficiencies

The PIE curves for those ions which have been studied so far are presented in Figures VI-1 through VI-13. These PIE curves were obtained at room temperature (298 °K), generally with optical resolutions of 0.42 or 0.83 Å, and with sample pressures ranging from 0.5 to 1.5 mtorr. The experimental procedures described in Chapter V

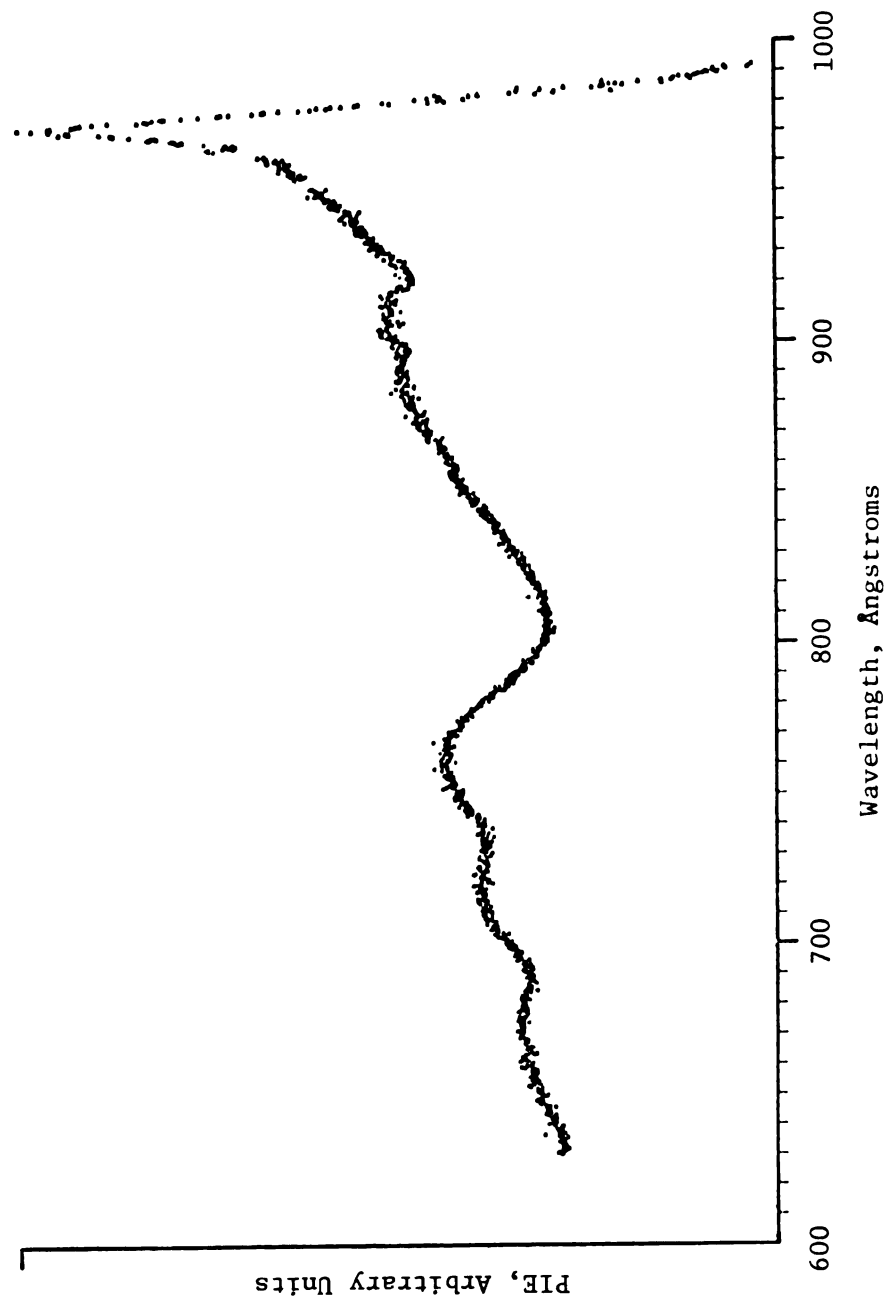


Figure VI-1. CF_3Cl^+ From CF_3Cl .

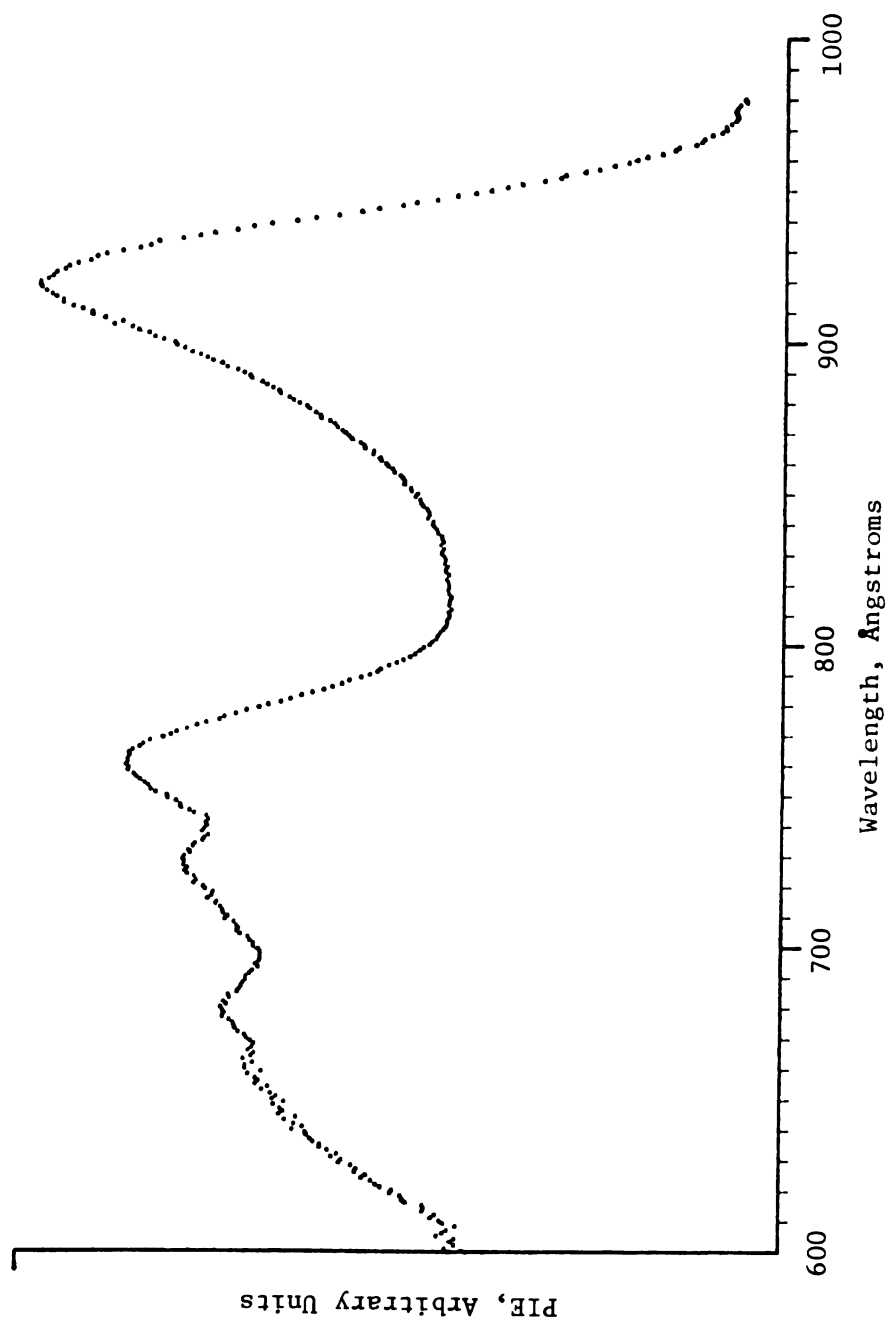


Figure VI-2. CF_3^+ From CF_3Cl .

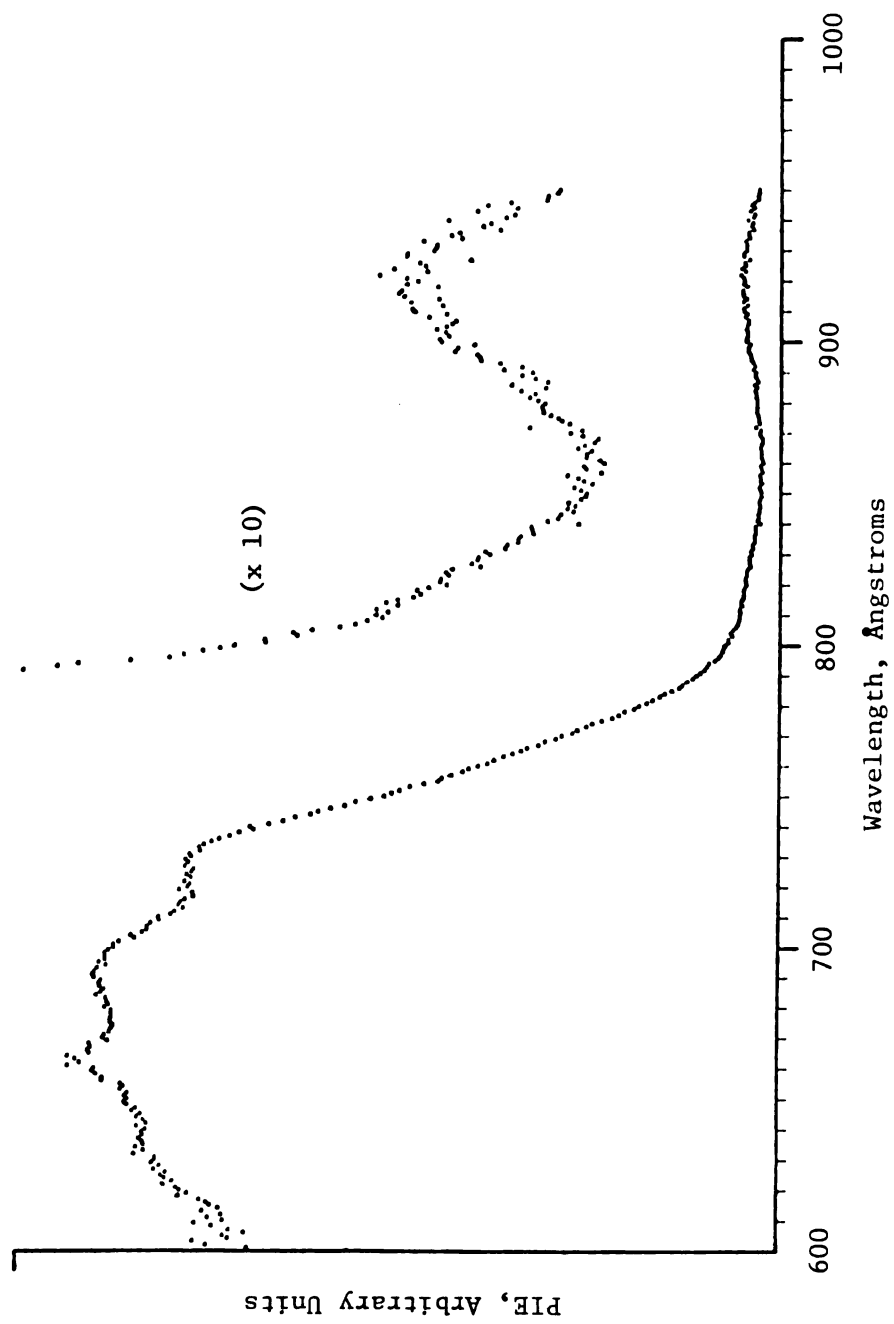


Figure VI-3. CF_3Cl^+ From CF_3Cl .

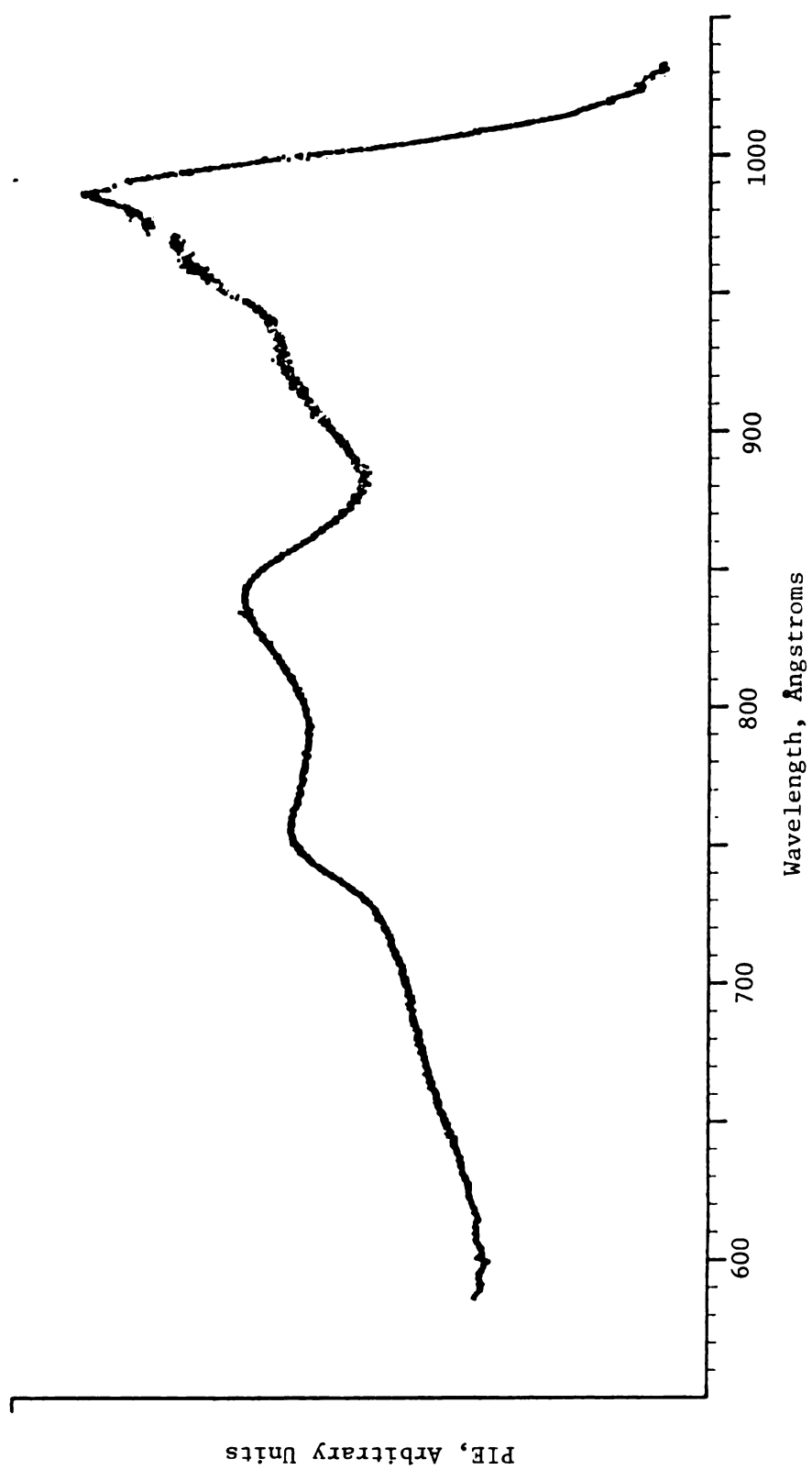


Figure VI-4. CF_2Cl^+ From CF_2Cl_2 .

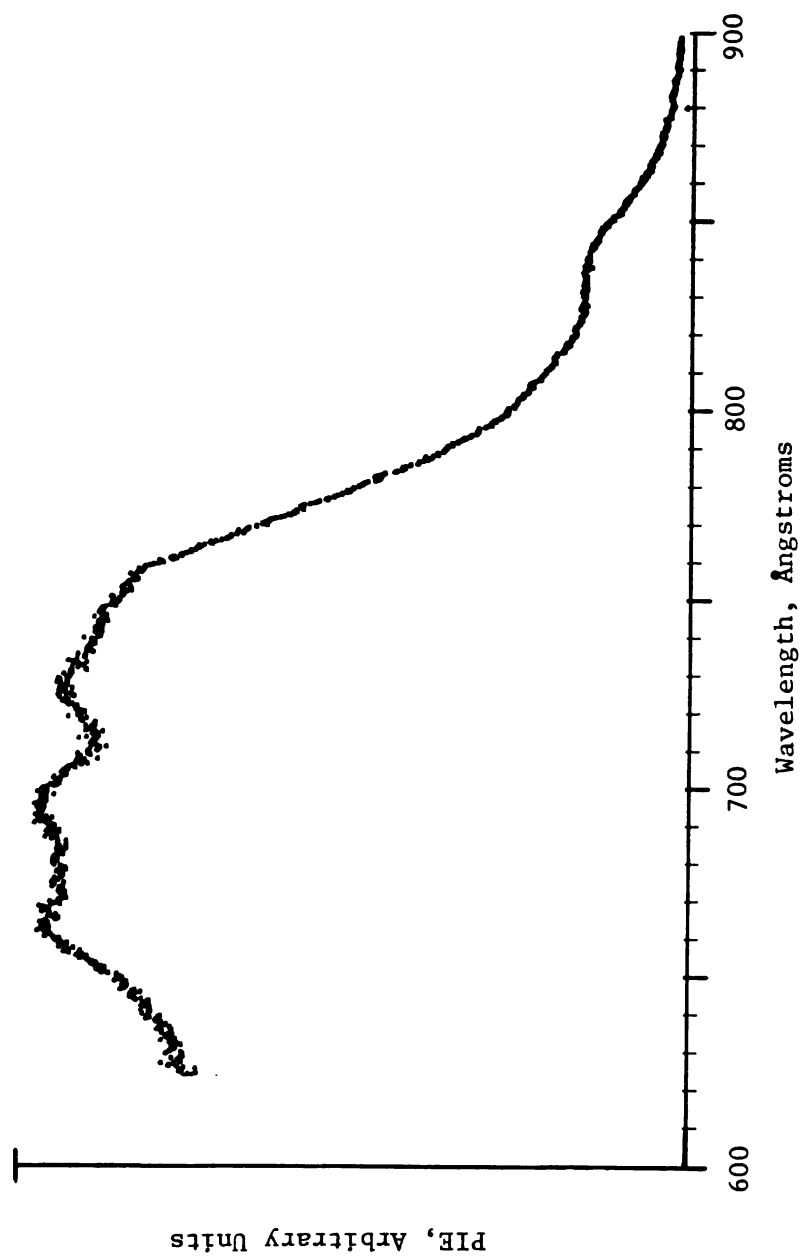


Figure VI-5. CF_2Cl_2^+ From CF_2Cl_2 .

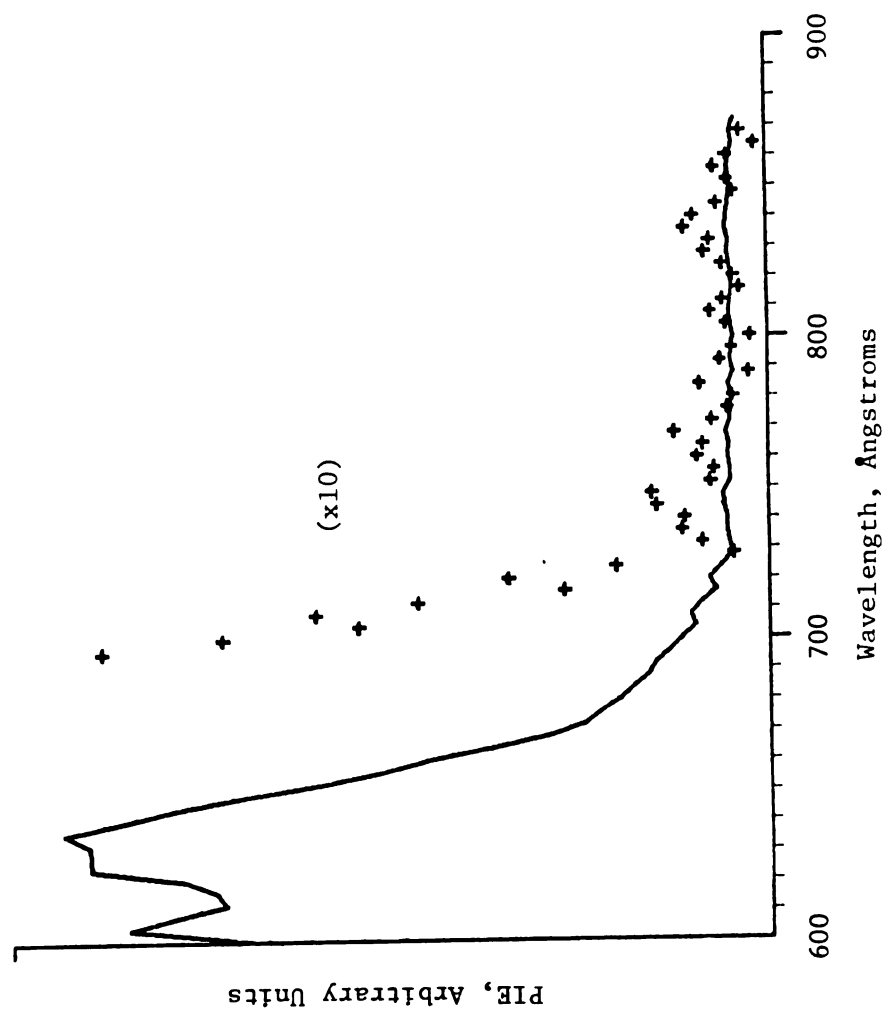


Figure VI-6. CF_2Cl_2 From CF_2Cl_2 .

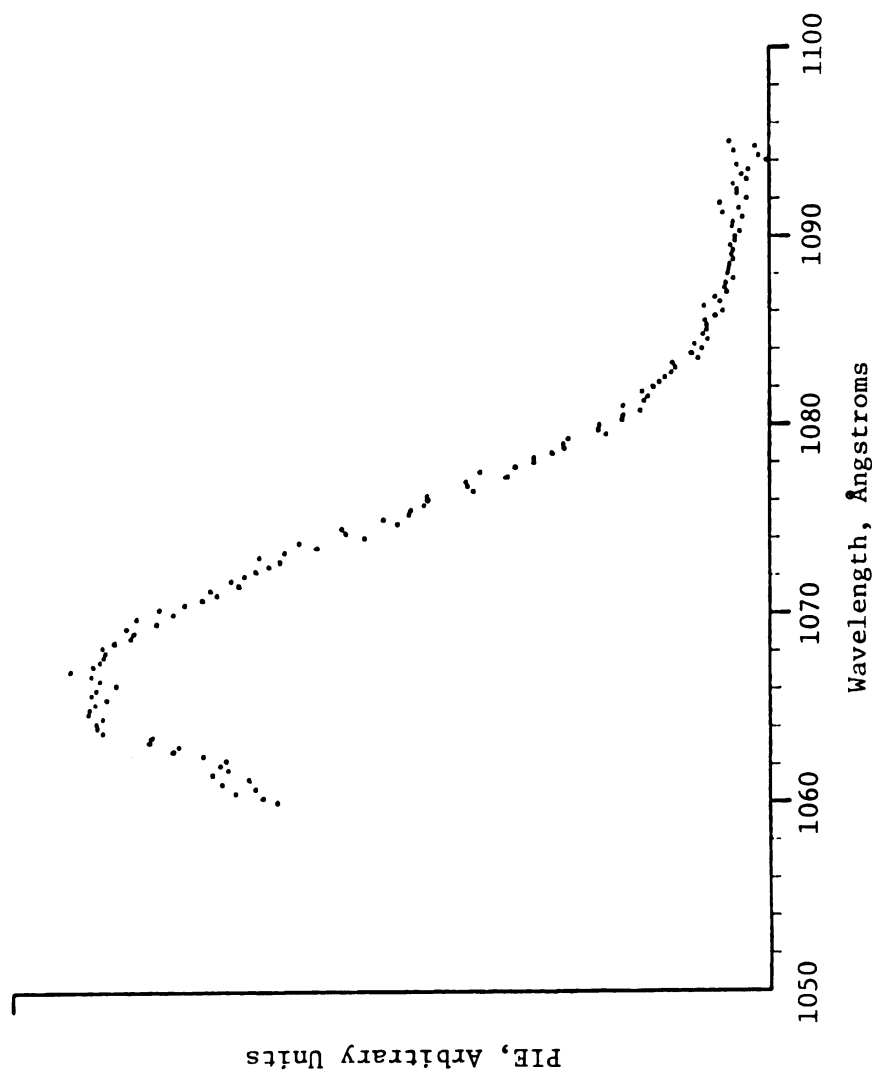


Figure VI-7. CFCl_3^+ From CFCl_3 .

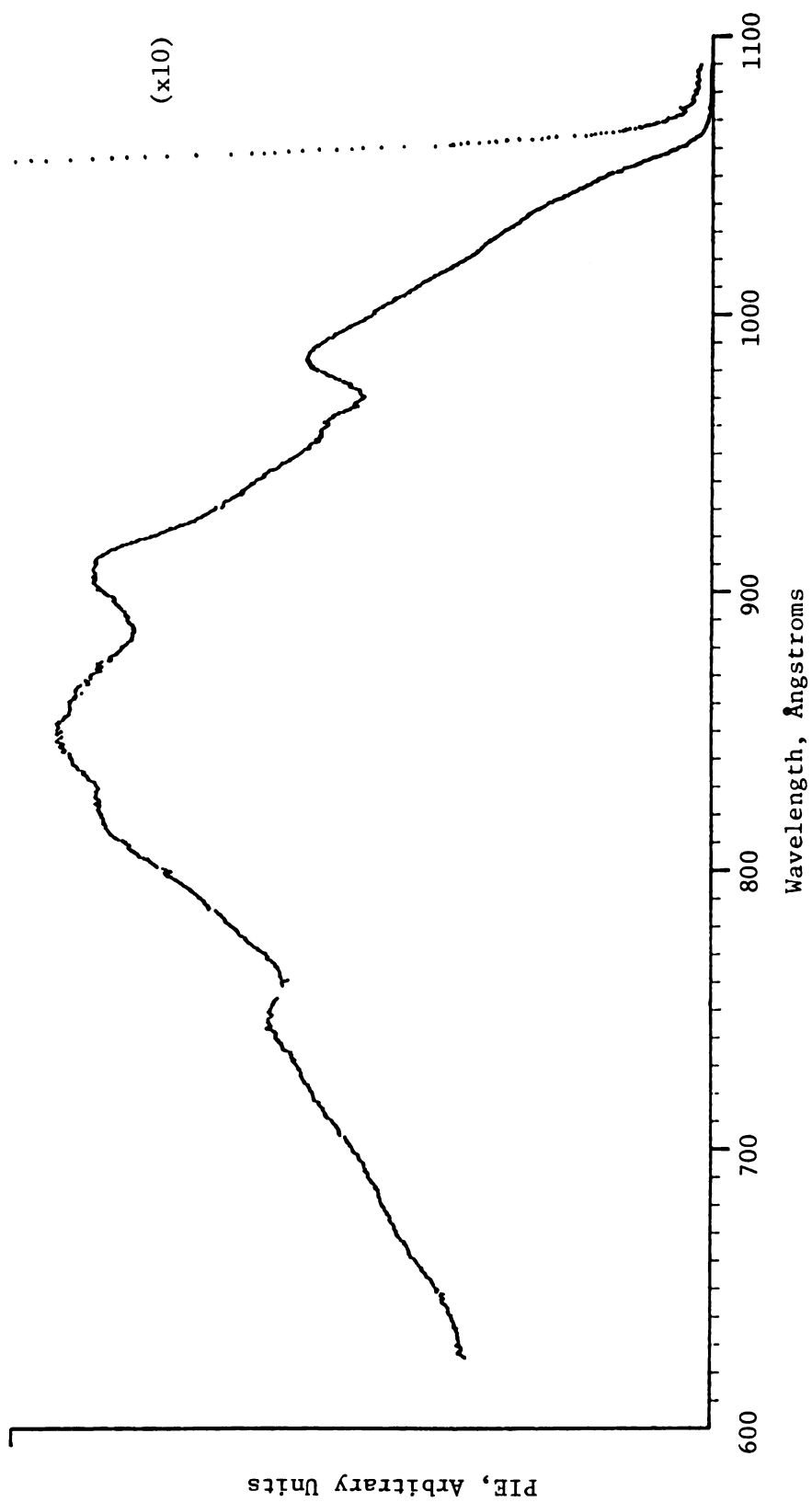


Figure VI-8. CFCl_2^+ From CFCl_3 .

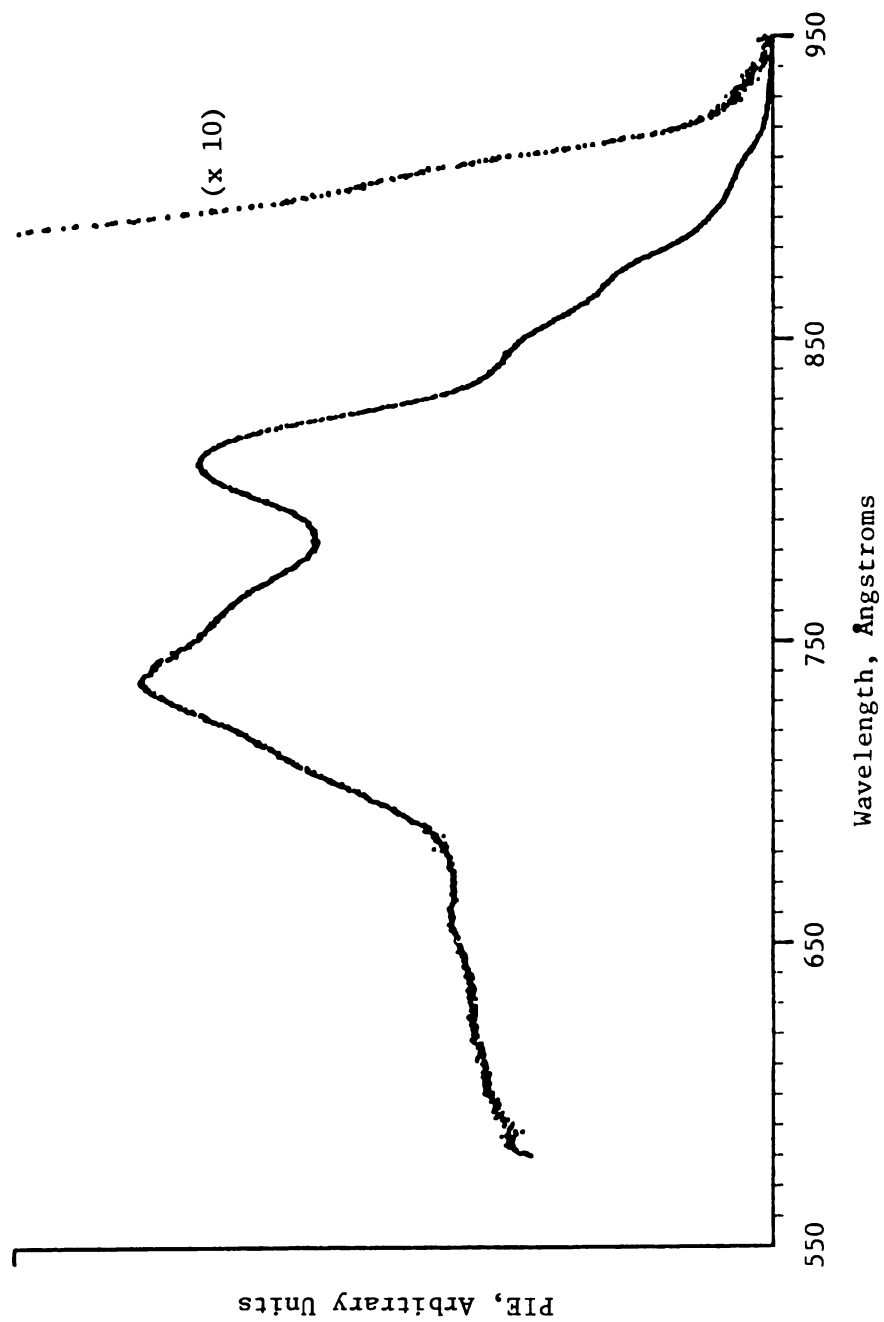


Figure VI-9. CCl_3^+ From CFC1_3 .

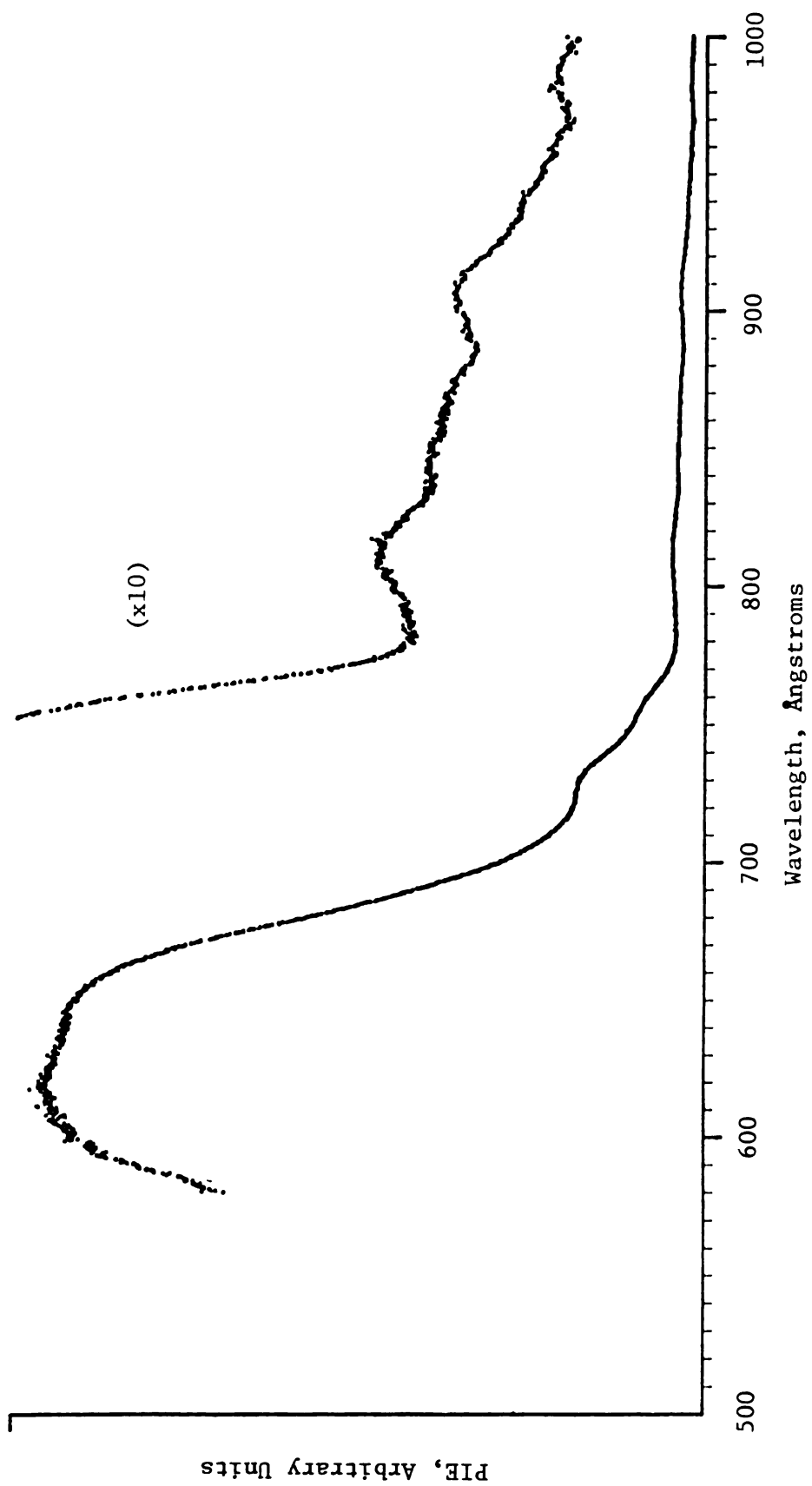


Figure VI-10. CFCl_3^+ From CFCl_3 .

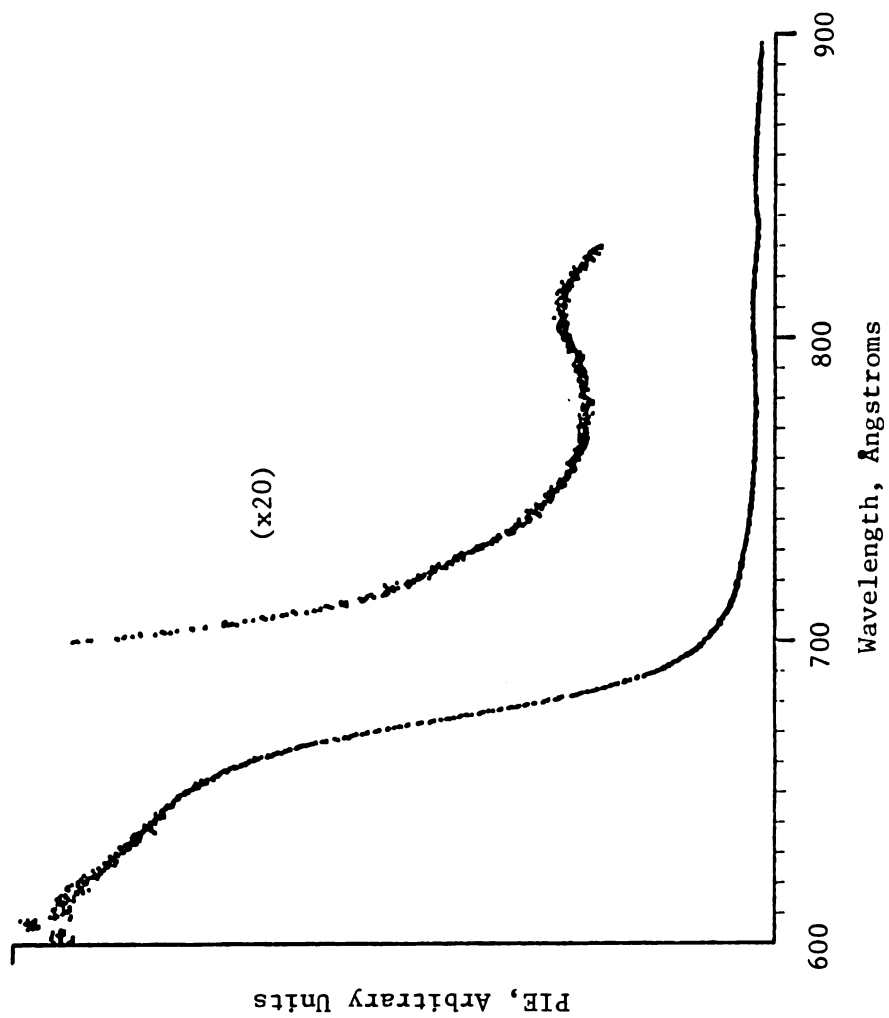


Figure VI-11. CCl_2^+ From CFCl_3 .

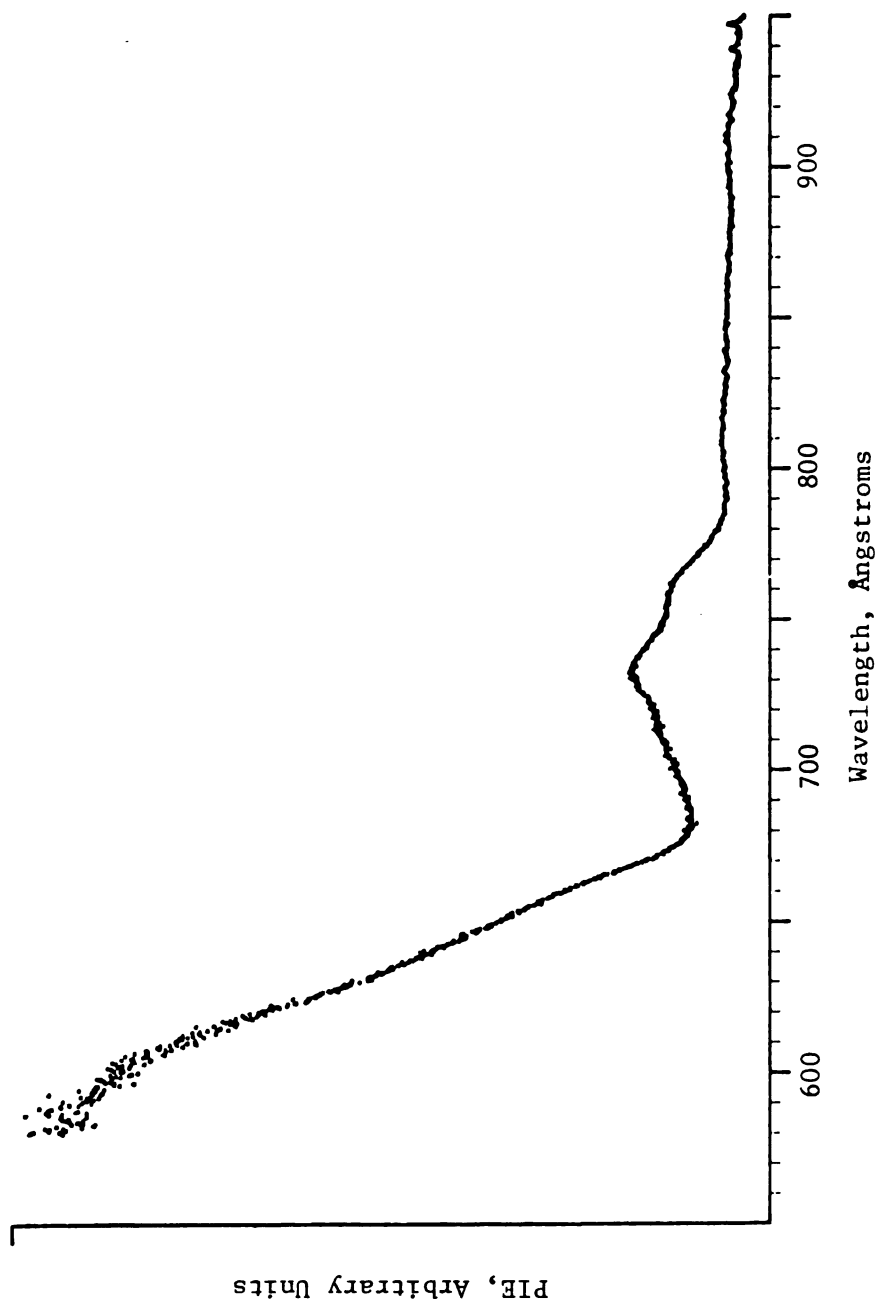


Figure VI-12. CF^+ From CFCl_3 .

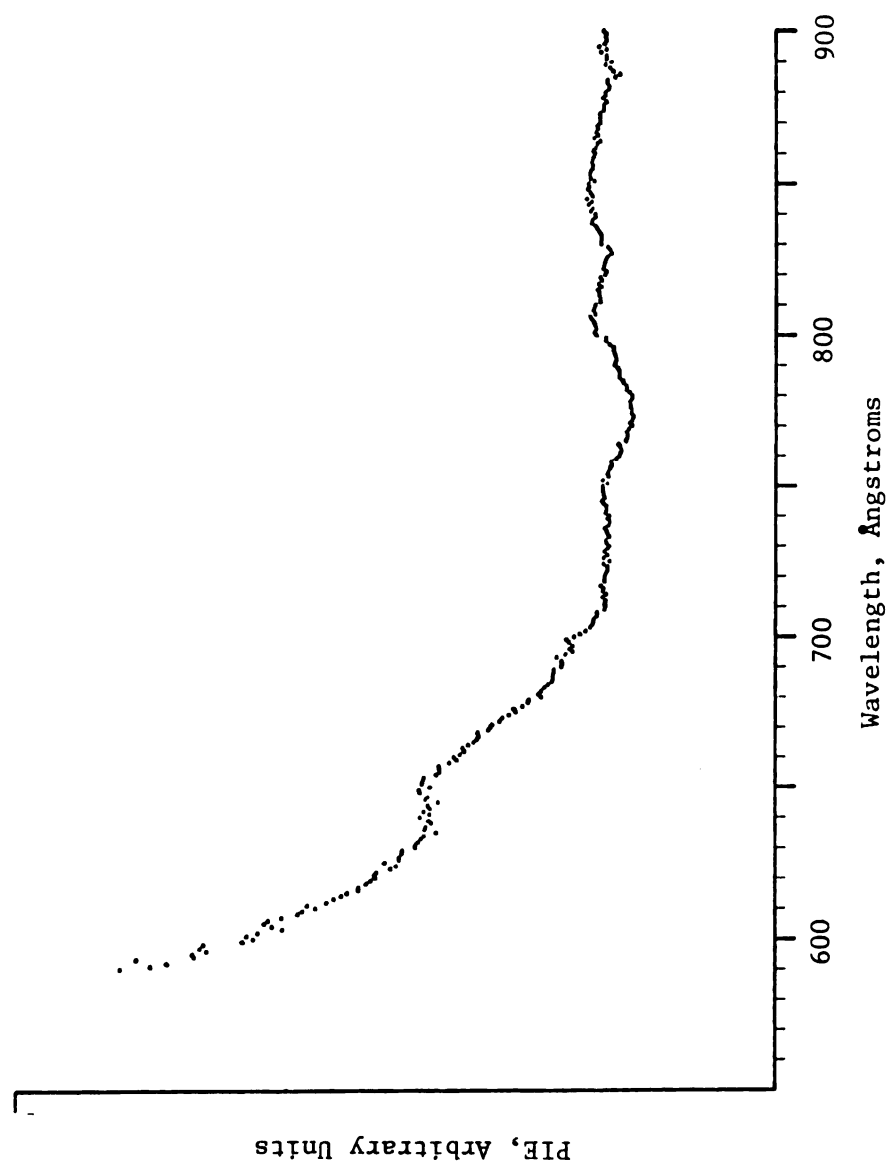


Figure VI-13. CCl_3^+ From CFCl_3 .

were used, except that some of the earliest experiments were performed before it was possible to make periodic correction measurements. None of these data have yet been corrected for the wavelength response of the copper-beryllium electron multiplier used to detect the photons. This introduces a distortion in the PIE curves which does not interfere with the measurement of thresholds or the interpretation of fine structure, but which tends to make the longer wavelength portions of the PIE curve appear too large relative to the lower wavelength portions. The results of JLB are recommended as a better indication of the overall shape of the PIE curves.

With only two exceptions (the CFCl_3^+ PIE curve and the threshold region of CFCl_2^+ from CFCl_3) all of these data were obtained using a helium lamp, despite the fact that several other PIE curves extend beyond 950 Å, where the helium continuum becomes very weak. The reason for this is that the powerful DC power supply now used to power the hydrogen lamp (see Chapter II) was not available until very recently. Although the only power supply available earlier (the Cober pulse generator) could be used to excite the hydrogen lamp in a pulsed mode, its limited duty cycle produced a many-lined continuum which was even weaker than the helium continuum at wavelengths below 1050 Å. Although long integration times permitted the acquisition of PIE curves to wavelengths as long as 1040 Å, even with the Cober-excited helium continuum, this is not far enough to reach some of the thresholds; the data above 1000 Å are not always of the desired quality. Thus, much of the interpretation in the following sections is regarded as tentative, pending experiments with the newer, more powerful light source power supplies, which will allow much more accurate long-wavelength

threshold measurements.

Ideally, one would like to determine from the PIE curves the energies corresponding to transitions from molecules in their lowest rotational, vibrational and electronic states to products which are also in their lowest states, i.e., the adiabatic transition for molecules at 0°K. In cases where the measured PIE curve exhibits a threshold which is nearly a step function (i.e., which rises from zero to its maximum in an energy range comparable to the photon energy bandwidth) it is possible to determine the adiabatic 0°K threshold fairly easily. The abruptness of the thresholds observed in the present results varies a great deal, but none of the thresholds are so sharp as to be considered step functions. At least part of the reason for the relatively slow onsets can be attributed to the effects of thermal excitation of the sample molecules. Guyon and Berkowitz [10] and Chupka [12] have shown that the effect of the internal thermal energy of molecules in a sample above absolute zero is to shift the entire PIE curve toward lower energies (compared with the 0°K PIE curve) and to add a low energy tail, which often approaches zero exponentially. The PIE curve is shifted by just the mean thermal energy of the molecule¹⁷ so that if

¹⁷ To be more precise, the PIE curve for a fragmentation process is shifted by the mean energy available to the particular process being observed. This quite often includes all the rotational energy, at least for simple bond breaking processes [12], and may or may not include vibrational energy, depending on how many excited vibrational levels are populated at the temperature of the sample [152]. For the molecules considered here, the shift is expected to be on the order of 0.10 eV.

the curve is approximately linear above threshold, an extrapolation of the linear portion will intersect zero at an energy which is lower than the 0°K appearance potential by the mean thermal energy. Contrary to the statement of AHR, any heat of formation calculated from the extrapolated threshold does not correspond to the 0°K heat of formation. Furthermore, the energy corresponding to the "end" of the thermal tail will be lower than the 0°K threshold by even more than the mean thermal energy (the exact amount depending on the sensitivity of the instrument) and should not be used to calculate the "room temperature" heat of formation. The 0°K appearance potential can be evaluated either by adding the mean thermal energy to the energy obtained by an extrapolation of the PIE curve near the threshold [151], or (preferably) by deconvoluting the internal energy distribution function of the molecule from the PIE curve [74]. In practice, the deconvolution is difficult, and instead the reverse procedure is used: the thermal energy distribution is convoluted with an estimated 0°K PIE curve, which is adjusted until a good fit is obtained [12,152]. The most accurate procedure for evaluating the room temperature heat of formation for an ion or fragment is to first evaluate the 0°K heat of formation, and from this calculate the heat of formation at the desired temperature [3].

Although the thermal energy of the molecules tends to make the measured appearance potentials lower than the 0°K appearance potentials, there are several reasons why the measured APs may be too high [2]. For instance, the ground state of the ion being observed may not be accessible in a vertical transition from the molecule so that the measured (vertical) appearance potential will be higher than the

sought-for (adiabatic) appearance potential. As a further example, the rate of many fragmentation processes depends on the amount of internal energy deposited in the molecule in excess of the minimum necessary energy. Thus, while fragmentation may occur at the thermochemical threshold, significant excess energy may be required to produce a measurable number of fragment ions in the few microseconds available between ionization and detection. In such cases, the PIE curve will approach zero only asymptotically, so that the measured AP will depend at least in part on the sensitivity of the instrument. There is some evidence (discussed in the following sections) that very significant amounts of excess energy may be involved in many of the thresholds reported here. In some cases, the amount of excess energy is obviously so large that any attempt to correct the data for the effects of thermal energy would be meaningless. In other cases, it is not clear whether significant excess energy is involved, and calculations should be performed to determine how much of the deviation from a step function can be attributed to thermal energy. These calculations have not yet been performed, since the PIE curves whose interpretation the calculations would help the most are also those with the longest wavelength thresholds, and higher quality data are needed before the calculations can be performed.

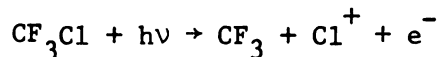
Almost all of the PIE curves in Figures VI-1 through VI-13 exhibit broad autoionization structure above threshold. Most of this structure corresponds to absorption maxima reported by JLB [154] and by Gilbert, *et al.* [155], but the high signal-to-noise ratio in some of the PIE curves presented here reveals several features not seen before.

Gilbert, et al. have assigned many of the absorption maxima to initial members of Rydberg series converging to excited electronic states. However, the structure is very broad and poorly resolved, and higher members of the series are not visible; the assignments must be regarded as tentative at best. The wavelengths of maxima observed in the PIE curves presented here are therefore reported with no attempt at interpretation.

The heats of formation which have been used in the calculations in the next three sections are summarized in Table VI-2. In addition, the following conversion factors have been used: $1 \text{ cm}^{-1} = 1.23981 \times 10^{-4} \text{ eV}$, and $1 \text{ eV molecule}^{-1} = 23.06 \text{ kcal mole}^{-1}$.

3. Formation of Positive Chlorine Ions

It is not surprising that the previous investigators have not reported the presence of Cl^+ among the fragment ions from these compounds. Chlorine ions are not among the most commonly observed positive ions, and they are formed only about as frequently as the parent ions from these compounds (see Table VI-1), so that they must be searched for rather carefully to be detected at all. In fact, they were overlooked in several previous relative intensity measurements made with the MSU instrument; the measurements reported in Table VI-1 were made very recently, and we have not yet had a chance to record a Cl^+ PIE curve. While Cl^+ could be formed via a simple, direct mechanism such as



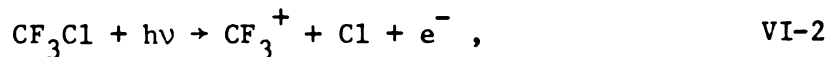
VI-1

Table VI-2. Data Used in Thermodynamic Calculations.^a

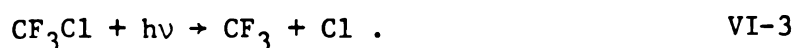
Species	ΔH_f° (kcal/mole)	Species	ΔH_f° (kcal/mole)
F	18.38	Cl ⁻	-54.64
F ⁻	-60.03	Cl ⁺	327.7
F ⁺	420.1	CCl	131
CF	60.1	CCl ₂	74.7
CF ₂	-43.6	CCl ₃	35.2
CF ₃	-111.7	CFC1 ₃	-65.2
FC1	-13.0	CF ₂ Cl ₂	-113.0
Cl	28.68	CF ₃ Cl	-164.8

^aAll data were taken from reference [159], except the values for CCl, CCl₂, and CCl₃, which are from reference [160].

there is also a second mechanism which may be important. Chlorine atoms are released not only in the fragmentation of parent ions,



but also in photodissociation to neutral products:



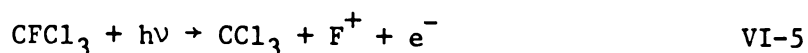
It is possible that the neutral chlorine atoms released in such processes are then ionized in a second step:



If the second mechanism (VI-2 and/or VI-3, followed by VI-4) is primarily responsible for the Cl^+ ions observed here, the intensity of the ion signal should vary as the square of the photon intensity. Unfortunately, the Cl^+ ions are difficult to detect even with the maximum available photon intensity, and the determination of the relation between photon intensity and Cl^+ ion intensity will probably prove to be too difficult. The two mechanisms can easily be distinguished another way, however. The appearance potential for Cl^+ produced as in VI-1 can be easily calculated (from data in Table VI-2) to be approximately 16.51 eV (750 Å). The Cl^+ appearance potential in the case of the second mechanism is just the ionization potential of Cl, or 14.21 eV (872 Å). While the PIE curve for Cl^+ formed according to VI-1 is expected to exhibit broad autoionization structure similar to that

observed in Figures VI-1 through VI-13, a PIE curve resulting from photoionization of chlorine atoms should exhibit a great deal of very sharp autoionization structure (which is typical of PIE curves of atoms). Thus, measurement of the Cl^+ PIE curves for each of these molecules should make it relatively easy to decide the relative importance of the two mechanisms, and such experiments will be performed very soon.

It should be noted that no positive fluorine ions could be detected at 21.2 eV (584 Å). Some processes, such as



have calculated appearance potentials above 21.2 eV (22.64 eV, or 549 Å in the case of VI-5). However, the ionization potential of fluorine atoms (17.4 eV, 711 Å) is less than the available energy and it should be possible to create F^+ ions via a mechanism similar to the second mechanism discussed above for production of Cl^+ . It is possible that F^+ ions are formed, but are fewer in number than the Cl^+ ions, and are therefore below the detection limit of the present measurements.

C. CF_3Cl

Three ions, including the parent ion, have been studied to date in PIMS experiments on CF_3Cl . Their PIE curves are shown in Figures VI-1 through VI-3. The appearance potentials which have been measured from these curves, as discussed below, are summarized and compared with the

values of other workers in Table VI-3.

1. CF_3Cl^+

The parent ion of CF_3Cl , although more intense than that of CFCl_3 , is nevertheless very weak, and it proved impossible to obtain data of the desired signal-to-noise ratio at threshold even with 400 second integration times. This PIE curve will soon be remeasured with the new hydrogen lamp; the present results allow only the determination of an upper limit for the ionization potential of 12.52 eV (990 Å). This value is at least compatible with the values reported by other workers (see Table VI-3). The slight differences between the appearance potentials reported by Noutary, JLB, and AHR probably stem more from the method used to determine the threshold (not specified by Noutary or JLB) than from any real differences in the data, as can be seen from the fact that the extreme values (1001 Å and 990 Å) have both been derived from the same PIE curve (that of AHR). From $\Delta H_f(\text{CF}_3\text{Cl})$ and the ionization potential reported here, $\Delta H_f(\text{CF}_3\text{Cl}^+)$ is calculated to be <124 kcal/mole.

Although the threshold has apparently not been reached in Figure VI-1, the signal-to-noise ratio between 625 and 980 Å is excellent, clearly revealing several broad peaks at approximately 671, 714, and 763 Å, as well as an even broader peak at 890 Å. The 714 and 763 Å features match peaks observed by JLB in absorption, but the two absorption maxima reported at 663 and 679 Å are not resolved in the parent ion PIE curve, and the 890 Å peak does not correspond to any absorption or photoelectron spectrum feature reported by JLB. None of this structure is apparent in the PIE curve of AHR, which is very

Table VI-3. Summary of CF_3Cl Appearance Potentials.

Ion	Appearance Potential, eV (A)			
	JLB 154	Noutary 150	AHR 153	This work
CF_3Cl^+	12.45 (996)	12.43 (1001) ^a	12.39 (1001) ^b 12.52 (990) ^c	<u>≤</u> 12.52 (990)
CF_3^+	12.55 (988)	12.57 (982) ^a	12.65 (980) ^b 12.81 (968) ^c	<u>≤</u> 12.78 (970)
CF_2Cl^+	14.25 (870)	---	collisional only	<u>≤</u> 14.42 (860)
CF_2^+	18.85 (658)	---	18.84 (685)	----
CFC1	----	----	----	----
CF	----	----	----	----
CC1	----	----	----	----

^aThere is a discrepancy between the wavelength and energy reported by Noutary. The wavelengths have been assumed correct, and the energies shown calculated from them.

^bDetermined by the first measurable signal.

^cDetermined by extrapolation of linear portion of the PIE curve.

noisy and extends only from threshold to 800 Å. Although JLB have drawn a smooth curve through their CF_3Cl^+ PIE data (which is also fairly noisy), indicating that structure is present, their experimental scatter is so severe that some of the structure indicated must be as much from analogy with the more intense CF_3^+ ion PIE curve as from the CF_3Cl^+ data.

2. CF_3^+

The PIE curve for CF_3^+ , shown in Figure VI-2, exhibits auto-ionization peaks at 661, 678, 725, and 762 Å, all of which match absorption maxima reported by JLB quite closely. The large peak at about 920 Å appears to correspond to the 931 Å absorption maximum, considering that the threshold is so nearby that it is truncated on the long wavelength side. As in the case of the parent ion, the PIE curve does not extend far enough to be completely certain about the threshold; pending further experiments, an upper limit of 12.78 eV (970 Å) has been adopted for $\text{AP}(\text{CF}_3)$. Combined with heats of formation for Cl and CF_3Cl , this yields $\Delta H_f(\text{CF}_3^+) \leq 101$ kcal/mole. By way of contrast, the $\Delta H_f(\text{CF}_3^+)$ calculated from the appearance potential of JLB (the lowest of those in Table VI-3) is ≈ 96 kcal/mole.

Since there is a large geometry change between the CF_3Cl molecule and the CF_3^+ ion (which is planar [156,157]), one might expect that the minimum of the CF_3^+ potential well might be inaccessible in a vertical transition from CF_3Cl . This would result in the production of excited CF_3^+ ions, the creation of fragments with significant kinetic energy, or both, and the measured appearance potential would be too high. There is some evidence to support this conjecture. Noutary

[150] has measured the AP for CF_3^+ ions for a series of compounds, CF_3X , where $\text{X} = \text{I}, \text{Br}, \text{Cl}, \text{CF}_3, \text{H}, \text{and F}$. The heats of formation for CF_3^+ calculated from the results span a range of some 30 kcal/mole, with that from CF_3I ($\Delta H_f(\text{CF}_3^+) = 87.3$ kcal/mole) being the lowest. The differences were interpreted as showing that excess energy is involved at the fragmentation thresholds of the other compounds. Even assuming the excess energy involved in the fragmentation of CF_3I to be zero [158], the excess energy for CF_3Cl (calculated using the AP reported by JLB) would be 8.7 kcal/mole (0.38 eV), thus placing the true thermochemical appearance potential of CF_3^+ at 12.17 eV (1018 Å). This is believed to be a more accurate estimate of the thermochemical AP for CF_3^+ from CF_3Cl , but it is below the measured ionization potential of the molecule! Since the adiabatic IP is expected to be less than the thermochemical AP for any fragment ion, this implies that the measured IP is not the adiabatic IP, but only a vertical IP which is higher than the adiabatic IP by at least 0.35 eV.

The fact that the parent ions of CF_4 and CCl_4 have never been observed has been interpreted to mean that the geometry of the ion is so different from that of the molecule that vertical transitions from the latter can only sample the steeply repulsive portion of the ion's potential curve, which lies above the dissociation limit [151,152]. It would not seem unreasonable to expect a similar geometry change for the parent ions of the mixed chlorofluoromethanes, although perhaps to a lesser degree due to the lower symmetry of the molecules. Such a change could make the CF_3Cl^+ potential minimum inaccessible in a vertical transition, resulting in a measured ionization potential which,

like the measured AP for CF_3^+ , is too high. The parent ion PIE curves of both AHR and JLB exhibit what at first appears to be a large thermal tail just below the apparent onset at about 980 Å. However, in neither case does the reported PIE curve extend to long enough wavelengths for the slope of the PIE curve to reach zero, and it is possible that the PIE curve actually continues to significantly longer wavelengths, albeit very weakly. If it does continue further than a few tenths of an eV below the apparent threshold (and thus cannot be ascribed entirely to a thermal tail), it would support the conjecture that the true CF_3^+ appearance potential and CF_3Cl ionization potentials are lower than so far reported. Both of these thresholds are being remeasured more carefully to determine whether they do provide any such evidence.

Since the ionization potential of CF_3 is accurately known (IP = 9.17 eV [157]), it is possible to calculate the expected CF_3^+ appearance potential from

$$\text{AP}(\text{CF}_3^+) = \Delta H_f(\text{CF}_3) + \text{IP}(\text{CF}_3) + \Delta H_f(\text{Cl}) - \Delta H_f(\text{CF}_3\text{Cl}) \quad \text{VI-6}$$

which yields an expected AP of 12.72 eV (975 Å). (Note that this is 0.17 eV lower than the expected AP calculated by JLB using 298°K, rather than 0°K heats of formation.) The expected AP is consistent with the tentative AP reported here. However, it is higher than the experimental AP reported by JLB, which is believed more accurate than ours, but possibly (as discussed above) still too high. The experimental AP is never expected to be lower than the "true" AP by more than the mean thermal energy of the molecule, but the discrepancy between JLB's experimental AP and the calculated AP is greater than this, indicating

that at least one of the values used in calculating the expected AP must be in error, most likely the heat of formation of CF_3 .

3. CF_2Cl^+

AHR report that CF_2Cl^+ is formed only via some collisional process involving the most intense ion, CF_3^+ , and that therefore its PIE curve (not shown by them) appears to have the same appearance potential and shape as that of the CF_3^+ ion. They cite the fact that the intensity of the CF_2Cl^+ ion signal is proportional to the square of the sample pressure as evidence for a collisional process. JLB report an appearance potential for CF_2Cl^+ at 14.25 eV (870 Å), and make no mention of collisional fragmentation. The reason for this discrepancy probably lies in the fact that the PIE curve of CF_2Cl^+ is fairly weak below 800 Å. (see Figure VI-3). AHR did not report the wavelength region over which they searched for this ion, but many of their PIE curves extend only to 800 Å (15.5 eV), and they may simply have not looked far enough. Although no pressure-variation study has yet been performed on the MSU instrument, it seems likely that most of the CF_2Cl^+ ions formed at wavelengths longer than 860 Å in our results are due to collisions involving the CF_3^+ ions. As a result of this collisional process, the appearance potential is difficult to determine, but an upper limit can be set at the minimum in the PIE curve at 860 Å (14.42 eV). From this, $\Delta H_f(\text{CF}_2\text{Cl}^+)$ is calculated to be ≤ 149 kcal/mole.

The PIE curve exhibits peaks at 630, 665, 692, and 735 Å, and a small shoulder just above the threshold, at about 825 Å. The same structure can be observed in the CF_2Cl^+ PIE curve of JLB. While most of this structure is attributed to autoionization, not all of these

peaks correspond to absorption maxima; in particular, the shoulder at $\approx 820 \text{ \AA}$ seems to correspond with a small peak in the photoelectron spectrum of JLB.

D. CF_2Cl_2

To date, PIE curves for only three of the ions from CF_2Cl_2 have been measured with the MSU photoionization mass spectrometer, although several other ions have sufficient intensity to be observed. The results are shown in Figures VI-4 through VI-6, and the estimated appearance potentials from these curves are reported in Table VI-4.

1. CF_2Cl^+

The PIE curve for this ion (Figure VI-4) exhibits autoionization peaks at 750, 845, 920, and 960 $\text{\AA}$ which are also observed in the results of JLB and, much less clearly, in the results of AHR. Once again, the poor intensity of the helium continuum in the threshold region prevents an accurate assessment of the appearance potential, which is estimated to be $<12.15 \text{ eV}$ ($>1020 \text{ \AA}$). From this value, ΔH_f (CF_2Cl^+) is calculated to be $\leq 138 \text{ kcal/mole}$. At least within the rather large uncertainty of the preliminary results, this value agrees with the value determined from the appearance potential of CF_2Cl^+ from CF_3Cl .

2. CFCl_2^+

The PIE curve for this ion (Figure VI-5) is again very similar to the data of AHR and JLB, except for having a slightly better signal-to-noise ratio. The autoionization structure matches that observed by JLB, although it is more clearly revealed; the peak locations are

Table VI-4. Summary of CF_2Cl_2 Appearance Potentials.

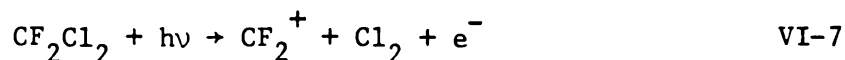
Ion	Appearance Potential, eV ($\text{\AA}$)			
	JLB [154]	AHR [153]		This Work
CF_2Cl_2^+	11.75 (1055)	11.75 (1055) ^a	11.81 (1050) ^b	----
CF_2Cl^+	12.10 (1024)	11.99 (1034) ^a	12.13 (1022) ^b	$\leq$ 12.15 (1020)
CFC1_2^+	14.15 (876)	13.81 (898)		$\leq$ 14.20 (823)
CF_2^+	17.22 (720)	16.98 (730)		17.03 (728)
CFC1^+	----	----		----
CCl_2^+	----	----		----
CF^+	17.65 (702)	----		----
CCl^+	----	----		----

^aFirst measurable signal.^bExtrapolation of linear portion of PIE curve.

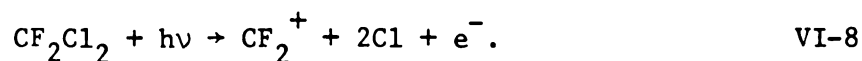
approximately 662, 695, 725, 748, and 840 Å. The PIE curve approaches the baseline very gradually in the threshold region, making it very difficult to estimate the appearance potential. This may indicate the presence of significant kinetic shift, and the estimated appearance potential of 14.20 eV (873 Å) is at best only an upper limit. The heat of formation for this ion, calculated from this AP, is 196 kcal/mole. This is considerably higher than that for the same ion calculated from the CFCl_3 data discussed in the next section (173 kcal/mole), confirming the fact that considerable excess energy is involved in its production from CF_2Cl_2 . It should be noted that the PIE curve has not quite reached zero even at 900 Å, and it is difficult to know how much further it may continue. The PIE curve reported by JLB exhibits a nearly constant intensity from about 920 Å to 975 Å (13.5 to 12.7 eV), and then a further, gradual decline to near zero at about 1100 Å (11.27 eV). JLB do not appear to feel that this long, low intensity extension of the PIE curve is real, having quoted an appearance potential at 14.15 eV (876 Å). Since the apparent appearance potentials reported by JLB and in this thesis are evidently too high, however, it is possible that the "extension" of the PIE curve in their results is real (and they do not say that it is an artifact). It is unlikely to be due to any collisional process (as in the case of CF_2Cl^+ from CF_3Cl), since it extends below the apparent thresholds both for the parent ion and for CF_2Cl^+ . It is important to reinvestigate the CFCl_2^+ PIE curve in the region of 850-1100 Å to determine if this extension is real, since if it is, it would imply that the measured appearance potentials for the parent ion and for CF_2Cl^+ are significantly too high.

3. CF_2^+

Usually, the interpretation of the PIE curves for the higher-energy fragments is much more difficult than that for the lower-energy fragments, since kinetic shift and other effects are much more likely to obscure the threshold. That does not seem to be the case for CF_2^+ , however, which exhibits a fairly sharp threshold at 17.03 eV (728 Å), as shown in Figure VI-6. There are several ways in which the CF_2^+ ions may be created, for instance



or



The heat of formation calculated for CF_2^+ assuming the first process is 279.7 kcal/mole, while that calculated from the second process is 222.4 kcal/mole. The latter value matches the value of 217.5 kcal/mole calculated for this ion by JLB from their CF_3Cl data. It is in even closer agreement with the literature value of 224 kcal/mole [159]. This latter value was calculated from IP (CF_2), determined by electron impact mass spectrometry and if inaccurate would be expected to be too high. If the data presented here are interpreted as involving ion pair formation, the heat of formation calculated for CF_2^+ would be even higher than the value calculated using the process in Equation VI-2. Clearly, the process being observed is that shown in VI-7.

E. CFCl_3

The PIE curves for the parent ion and six fragment ions from CFCl_3 are presented in Figures VI-7 through VI-13. The determination of appearance potentials from many of the fragment ion curves, among the earliest acquired with the MSU instrument, is hampered by an instrumental artifact which was not discovered and corrected until some time after these data were acquired. As an example, Figure VI-10 shows a greatly expanded plot of the CFCl^+ PIE curve. It is apparent that the baseline below the apparent threshold is not flat, but instead has a shape similar to that of the CFCl_2^+ PIE curve (Figure VI-8). All of the fragment ions formed at higher energies than CFCl_2^+ exhibit similar low-intensity, long wavelength tails below their apparent onsets, all tails having approximately the same shape. The ion intensity in these pre-threshold regions varies as the square of the sample pressure, indicating that the ions arise from some collisional process involving the CFCl_2^+ ions, which are present in abundance above their threshold. The energy necessary to create CFCl^+ ions from CFCl_2^+ ions at the latter's threshold is far in excess of the energy that should be available to the species in the ion source from thermal energy. However, these data were acquired at a time when a poor electrical connection to the repeller in the ion source forced us to operate at higher nominal repeller voltages than desired. While the repeller itself was not actually at the higher voltage, the wire in the ion source that serves as the connection to the repeller was, and it is likely that the CFCl_2^+ ions could have gained the necessary energy from the fields in the ion source. The amount of collision-related fragmentation is extremely

small at the sample pressures used (less than 2% of even the weakest fragment ion PIE curve), and it is a tribute to the sensitivity of the instrument that it could be detected at all. However, the presence of even this small amount of background makes it extremely difficult to determine accurate appearance potentials for some of the fragment ions, whose PIE curves have rather gradual onsets. The appearance potentials which could be determined from the present data are summarized in Table VI-5.

1. CFCl_3^+

The PIE curve for the parent ion of CFCl_3 (Figure VI-7) has a gradual onset starting at about 1095 Å, but the initial rise terminates at only slightly shorter wavelengths, where formation of the CFCl_2^+ fragment ion begins. The fact that the parent ion can be detected at all on the mass spectrometer's time scale shows that its potential surface has a well deep enough to support at least the zero point energy of the molecular ion. If the change in equilibrium geometry between the molecule and ion were small, one would expect the parent ion to be formed with a relatively high probability, at least over a small wavelength range. The fact that it is formed so rarely indicates that there is a considerable geometry change, and that most of the transitions sample the steeply repulsive part of the surface above the dissociation limit at about 1065 Å (11.64 eV). The drastic change in geometry makes it unlikely that the CFCl_3^+ ions are created in the ground vibrational state, unless the potential well is so shallow that it contains only one level. Thus, the reported ionization potential is unlikely to correspond to the adiabatic IP, and the heat of

Table VI-5. Summary of CFCl_3 Appearance Potentials.

Ion	Appearance Potential, eV ($\text{\AA}$)		
	JLB [154]	AHR [153]	This Work
CFCl_3^+	----	----	~ 11.45 (1083)
CFCl_2^+	11.65 (1064)	11.57 (1072) ^a 11.66 (1063) ^b	11.59 (1070)
CCl_3^+	13.50 (918)	13.25 (936)	≤ 13.05 (950)
CFCl^+	16.00 (775)	16.02 (774)	< 16.64 (745)
CCl_2^+	17.00 (729)	17.12 (724)	< 16.98 (730)
CF^+	15.70 (790)	----	< 15.79 (785)
CCl^+	20.5 (605)	----	(c)

^aFirst measurable signal.

^bExtrapolation of linear portion of PIE curve.

^cToo close to low wavelength limit of light source to determine an AP.

formation calculated from it ($\Delta H_f(\text{CFCl}_3^+) = 199 \text{ kcal/mole}$) is probably an upper limit. Until appropriate calculations are performed, it is unclear how much of the slowly rising part of the PIE curve is due to thermal energy broadening, and the threshold reported in Table VI-5 must be considered very tentative. Experiments with cooled samples and better statistics, which might reveal any vibrational structure, would also help with the interpretation considerably, and further experiments are planned.

2. CFCl_2^+

The PIE curve reported here for this ion (Figure VI-8) corresponds well in overall shape with the curve reported by JLB. Autoionization peaks are observed at 742, 815, 848, 910, 960, and 984 Å. The PIE curve reported by AHR extends only to 800 Å, but even in the region of 800-900 Å seems to have an overall shape quite different from the PIE curves reported here and by JLB. AHR also report vibrational structure with a spacing of about 4 Å near threshold for this ion; such structure is not observed by JLB. Although the threshold region of the CFCl_2^+ PIE curve shown in Figure VI-8 does exhibit several very small step-like features, these features all fall at wavelengths corresponding to peak intensities of the hydrogen many-lined light source used, and are thus ascribed to the effects of incomplete stray light corrections (see Appendix A). The threshold region should be reexamined, but the probability of resolving 4 Å vibrational steps when thermal tailing of the threshold is of the same order of magnitude seems small, and the "vibrational structure" reported by AHR should probably be regarded as artificial.

The heat of formation for CFCl_2^+ calculated from the appearance potential reported here is 173 kcal/mole. As mentioned earlier, this is lower than $\Delta H_f(\text{CFCl}_2^+)$ calculated from the AP of this ion from CF_2Cl_2 , indicating that excess energy is involved in creating the ion from that compound. It is not known whether any excess energy is involved in creating the ion from CFCl_3 . However, if the suggestions made in the preceeding section are true, and the appearance potentials of the parent ions and first few fragment ions from CF_3Cl and CF_2Cl_2 are all too high, then it might be expected that the appearance potential measured for CFCl_2^+ from CFCl_3 could also be too high. The reference data necessary to calculate an expected appearance potential are not available, and so the question must presently remain unresolved. In the future, if the ionization potential of CFCl_2 cannot be measured directly, it would be interesting to measure the appearance potential of CFCl_2^+ from CFCl_2I which would be expected to involve less excess energy (analogous to the case of CF_3^+ from CF_3Cl and CF_3I [150], discussed in Section C).

3. CCl_3^+

The PIE curve for CCl_3^+ shown in Figure VI-9 is among those most recently acquired with the MSU instrument, and is typical of the high-quality data which can now be obtained with the instrument. It reveals large autoionization peaks at 735 and 808 Å which are also apparent in the results of JLB, but also several much weaker, less well resolved peaks or shoulders at 658, 708, 762, 845, 870, and 905 Å. An expanded plot of the threshold region (also shown in Figure VI-9) reveals a very gradual onset which begins below 13.05 eV (950 Å). The appearance potential reported here results in a calculated $\Delta H_f(\text{CCl}_3^+)$ of 217

kcal/mole. Werner, et al. [152] have measured the appearance potential of CCl_3^+ from CCl_4 , and derive an upper limit on the heat of formation of the CCl_3^+ ion of 208.8 ± 0.6 kcal/mole. Evidently, even the CCl_3^+ appearance potential reported here, which is lower than those reported by JLB and AHR, includes at least several tenths of an electron volt of excess energy.

4. Other Fragments

The PIE curves for CFCl^+ , CCl_2^+ , CF^+ , and CCl^+ (Figures VI-10 through VI-13) generally confirm the results of JLB and AHR. For reasons mentioned at the beginning of this section, the determination of appearance potentials from these PIE curves is extremely difficult, despite the excellent signal-to-noise ratio of the data. Although approximate appearance potentials are reported in Table VI-5, they are considered to be too unreliable to be used in calculating heats of formation. The threshold for CCl^+ , in fact, is so close to the low wavelength limit of the light source that it is impossible to tell from the data of Figure VI-13 whether a significant portion of the first rise in the PIE curve or only a few percent of it is being observed. Therefore, no appearance potential at all is reported for CCl^+ ; the results of JLB (which extend to 24 eV, or $516 \text{ \AA}$) are judged to be much more reliable in this case.

F. Summary

The data presented in this chapter show that the parent ions of CF_3Cl , CF_2Cl_2 , and CFCl_3 can all be formed, and are stable at least

for the few microseconds that it takes for them to travel from the ion source to the detector. However, the parent ions are formed only very rarely compared to the chlorine loss ion fragment from each molecule, so that if mass analysis is not used, the onset of the parent ion will be totally obscured by the low-energy tail of the much more intense fragment ion. Thus, the ionization potentials reported by Watanabe [24] (measured by photoionization without mass analysis) should be interpreted instead as the appearance potentials for the first fragmentation processes.

The heats of formation which have been derived from the present results (discussed in previous sections) are summarized in Table VI-6. Most of these are reported as upper limits. In the case of fragments resulting from processes with high-energy appearance potentials, the uncertainty is due to the very gradual onsets exhibited by the PIE curves. Difficulty in determining higher-energy appearance potentials is not uncommon in PIMS studies of polyatomic molecules. Not so typical, however, is the uncertainty indicated in Table VI-6 for heats of formation of parent ions and fragment ions with the lowest appearance potentials, which can often be determined quite precisely by PIMS. The inadequacy of the light source used in these studies at wavelengths above $1000 \text{ \AA}$ is the cause of only part of the uncertainty. It has been suggested in previous sections that the measured ionization potentials of the compounds as well as the first fragment appearance potentials for CF_2Cl_2 and CF_3Cl may be considerably higher than the adiabatic ionization and appearance potentials. This interpretation is by no means certain, and both JLB and AHR report the measured thresholds for CF_3Cl^+ and CF_2Cl_2^+ as adiabatic ionization potentials. The

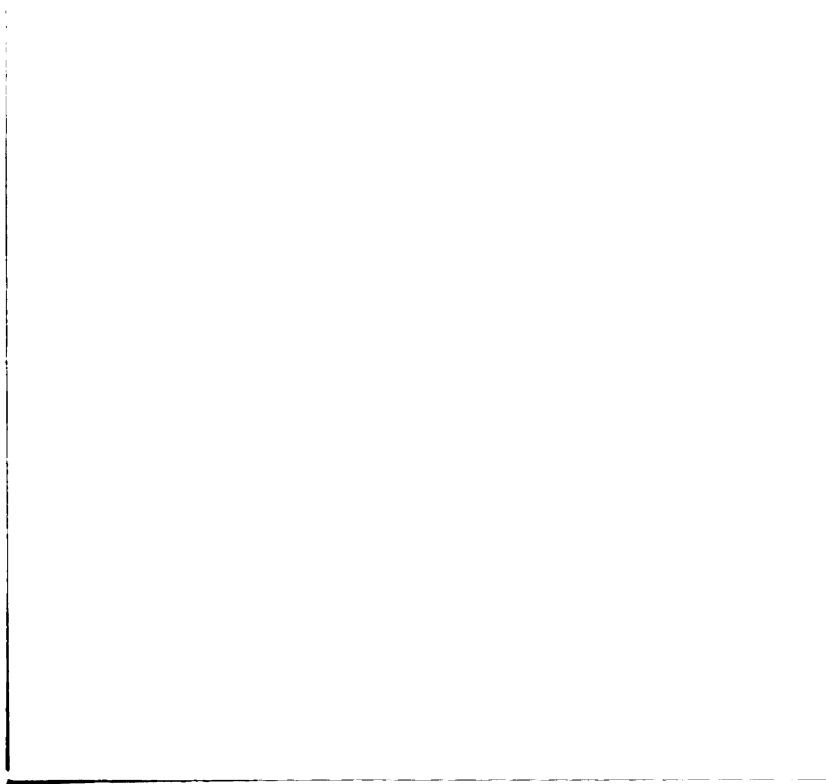
Table VI-6. Summary of Heats of Formation (kcal/mole) Found in This Work.

CF_3Cl	CF_2Cl_2	CFC1_3
$\text{CF}_3\text{Cl}^- \leq 124$	$\text{CF}_2^+ \quad 222.4$	$\text{CFC1}_3^+ \quad 199$
$\text{CF}_3^+ \leq 101$	$\text{CFC1}_2^+ \quad 196$	$\text{CFC1}_2^+ \quad 173$
$\text{CF}_2\text{Cl}^+ \leq 149$	$\text{CF}_2\text{Cl}^+ \leq 138$	$\text{CCl}_3^+ \leq 217$

new DC and pulsed light source power supplies which have been installed only recently should allow the already good quality of the data to be improved, especially in the region of the molecular ionization potentials, and further experiments in the threshold region of the parent ions of these compounds and of similar halocarbons are being undertaken to determine whether the interpretation suggested here can be better supported.

APPENDICES

Page 341



5/10/84/36
2 p. 5

APPENDIX A

ARTIFACTS

One of the characteristics most desired for the VUV light sources used in PIMS is that their output have a smooth distribution of intensity with wavelength. There are several reasons for this, one of them being that the measured PIE curves often exhibit artificial structure at wavelengths corresponding to sharp structure in the light source output. The purpose of this appendix is to examine three of the reasons for such artificial structure, and to point out how a moderate amount of sharp structure in the light source output can actually be very useful in ensuring proper interpretation of the data and alignment of the instrument.

A. Finite Slit Widths

In order to obtain sufficient photon intensity to perform an experiment, the monochromator entrance and exit slits must have a finite width. The photon beam leaving the monochromator is therefore not absolutely monochromatic, and the finite width of the monochromator slit function can cause artificial structure in the PIE curve which is measured. It is assumed here that the slit function of the monochromator can be adequately represented as a triangular function which has a width much greater than the actual linewidth of any emission line being observed. For instance, if the light source is assumed to emit only a single line at $500 \text{ \AA}$, the apparent emission spectrum measured with a resolution of $2 \text{ \AA}$ (FWHM) will appear as shown in Figure A-1. A finite

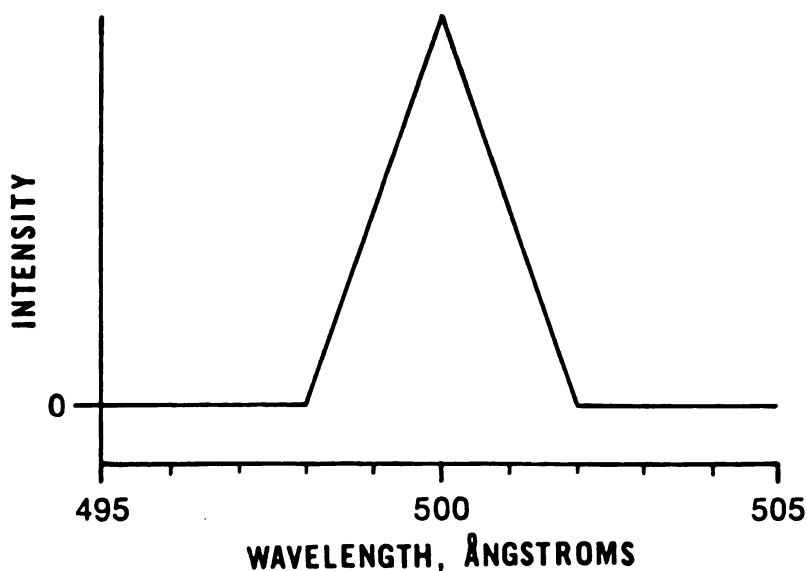


Figure A-1. The Assumed Slit Function.

photon intensity will be measured when the monochromator is set to any wavelength between 498 and 502 Å. However, the photons which are detected at 501 Å actually have a wavelength of 500 Å, and any ions created by these photons will be created with an efficiency equal to the true PIE at 500 Å, not the PIE at 501 Å.

Obviously, if the light source emits no continuum, but only lines, accurate PIE information can only be obtained at wavelengths of the emission lines. This is true even if the slit function is sufficiently wide that nearby lines appear to overlap, as indicated in Figure A-2. In this case, the PIE measured in regions where there is some light intensity but no overlap equals the true PIE at the wavelength of the nearest line; the PIE measured in the region of overlap is intermediate between the true PIEs at the line wavelengths. The actual shape of the PIE curve at other wavelengths has no effect (since there are

no photons of other wavelengths) as indicated by the fact that both of the postulated "true" PIE curves in Figure A-2 result in the same measured PIE curve.

The situation is slightly different if the sharp line is superimposed on a true continuum, as in Figure A-3. In this case, the measured PIE curve is correct, except near the sharp line, where it is biased toward the value of the true PIE at the wavelength of the line. If the line is very weak compared to the continuum, or if the slope of the PIE curve in the region is small, the effect will be very small. However, if the true PIE curve is changing rapidly and the line is intense, the effect may be significant. In any case, the maximum error is the difference between the actual PIE and the PIE at the wavelength of the emission line, and this effect will be negligible unless the PIE curves exhibits very weak, sharp structure with which the artificial structure can become confused.

B. Scattered Light

Although scattered light is difficult to measure accurately, it can grossly distort the overall shape of a PIE curve if it is not adequately corrected for in the data reduction process. For the time being let the effects of finite slit width (discussed in the preceding section) and of near scatter be ignored. Then let

P = the number of non-scattered light photons transmitted by the sample,

PIE = the photoionization efficiency at the wavelength of the P photons,

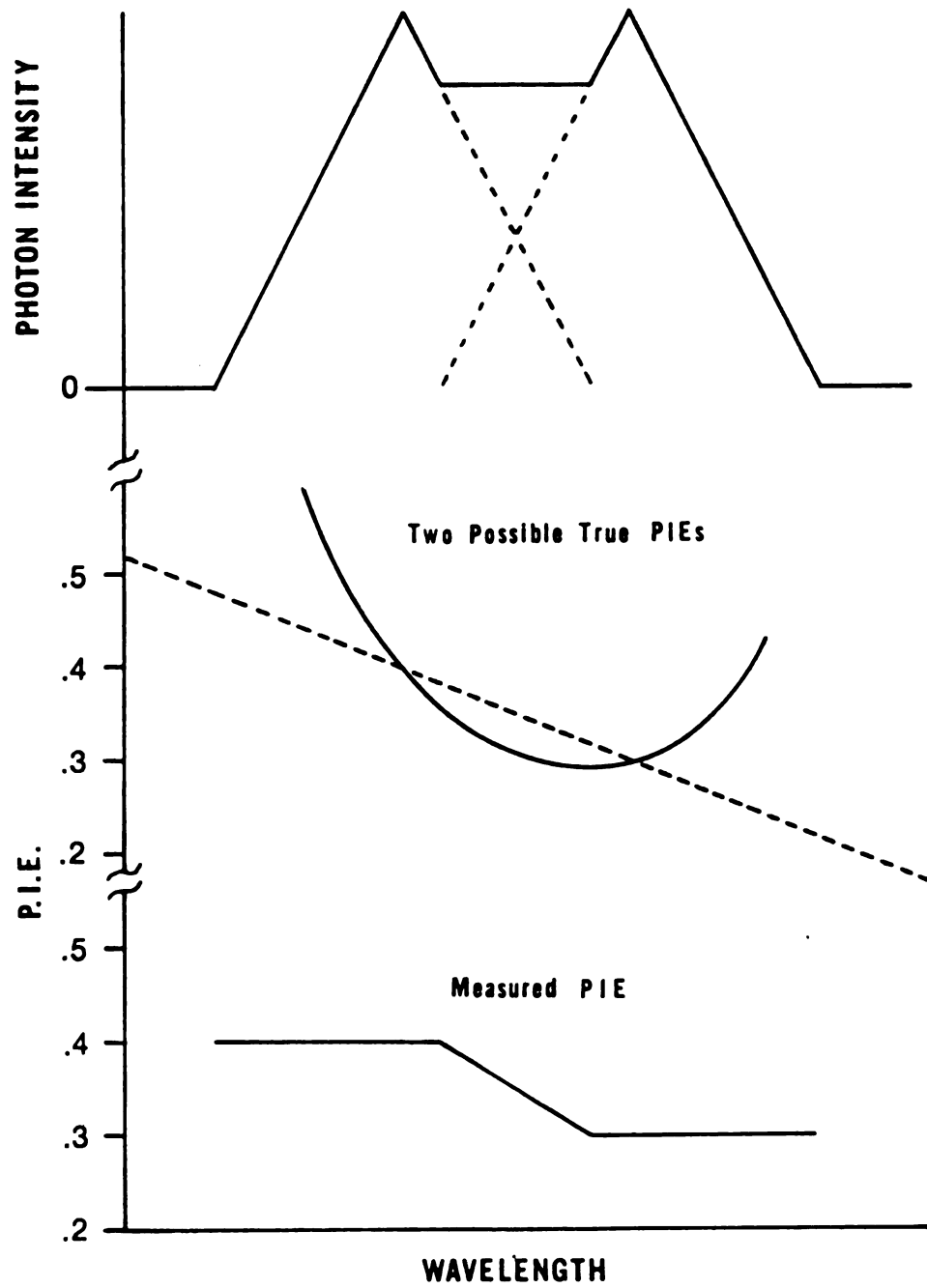


Figure A-2. Measured PIE for Two Overlapping Lines.

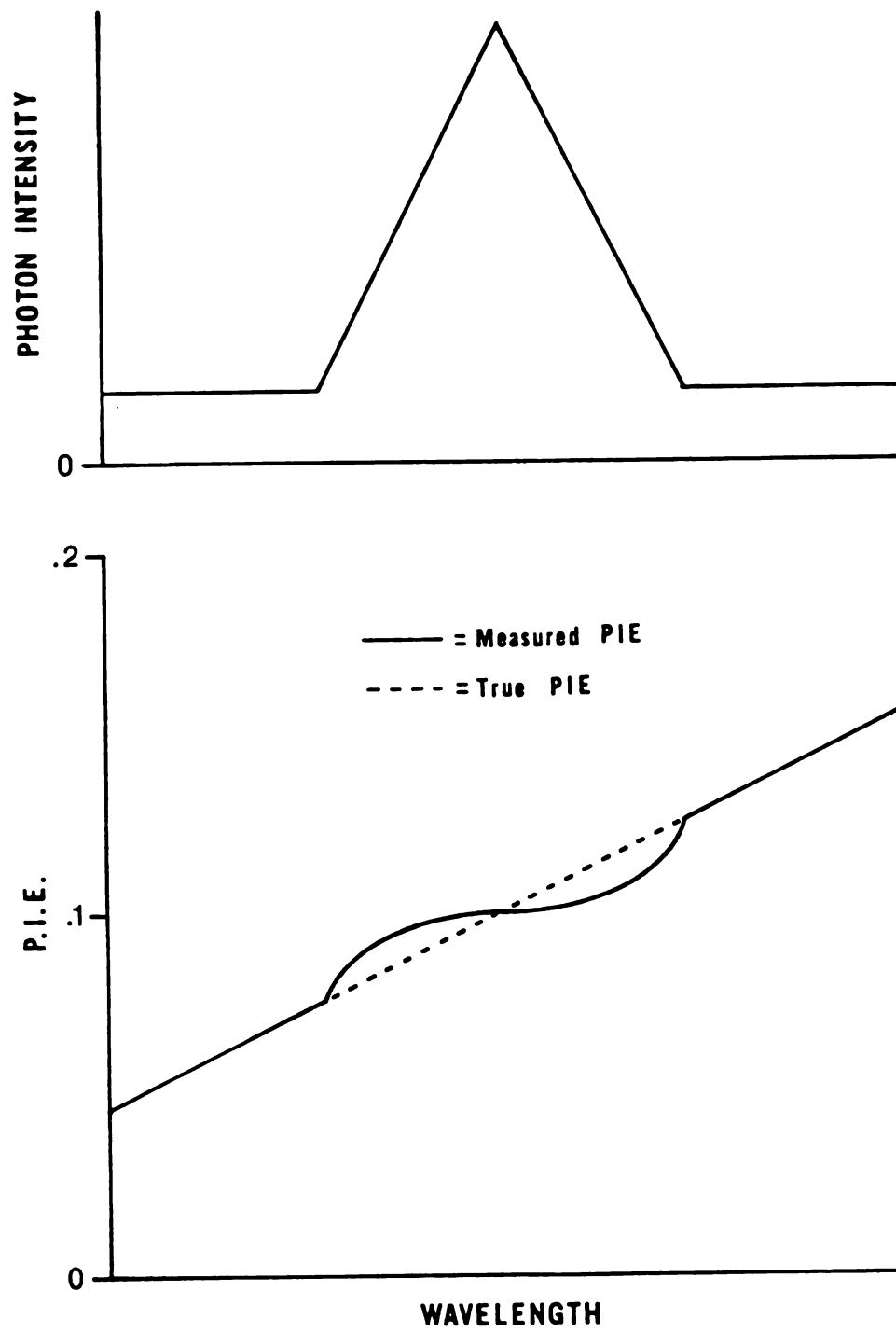


Figure A-3. Measured PIE for a Line Superimposed on a Continuum.

SP = the number of scattered light photons transmitted by the sample. (Assumed to have a wide distribution of wavelengths at any monochromator setting, and constant intensity over any small wavelength interval),

SPIE = a weighted average of the true photoionization efficiencies over the wavelengths contributing to SP, so that (SP)(SPIE) will equal the number of ions created by the scattered photons.

Since (P)(PIE) equals the number of ions created by the non-scattered photons, the photoionization efficiency that is calculated from the data when no corrections are made for stray light will be

$$PIE_{calc} = \frac{P(PIE) + SP(SPIE)}{P + SP} \quad A-1$$

Over much of the wavelength range of a continuum source such as the Hopfield continuum, $P \gg SP$, so that $PIE_{calc} = PIE$. Near the ends of the continuum, however, P and SP become comparable, so that PIE_{calc} and PIE may differ greatly. (unless $SPIE = PIE$, which is expected to happen only fortuitously). Depending on the relative magnitudes of PIE and SPIE in such regions, the calculated PIE could be either too high or too low.

One of the advantages of sharp structure in the light source output is that it makes the presence of unaccounted-for scattered light very obvious. SP and SPIE will remain constant over the width of an atomic line, but the ratio P/SP will not, especially if the continuum is weak. This results in the atomic line being mirrored in the calculated PIE curve, with the value of PIE_{calc} at the peak of the line (where P is greatest) being closest to the true PIE. The effect is illustrated in Figure A-4 for data calculated from Equation A-1, and

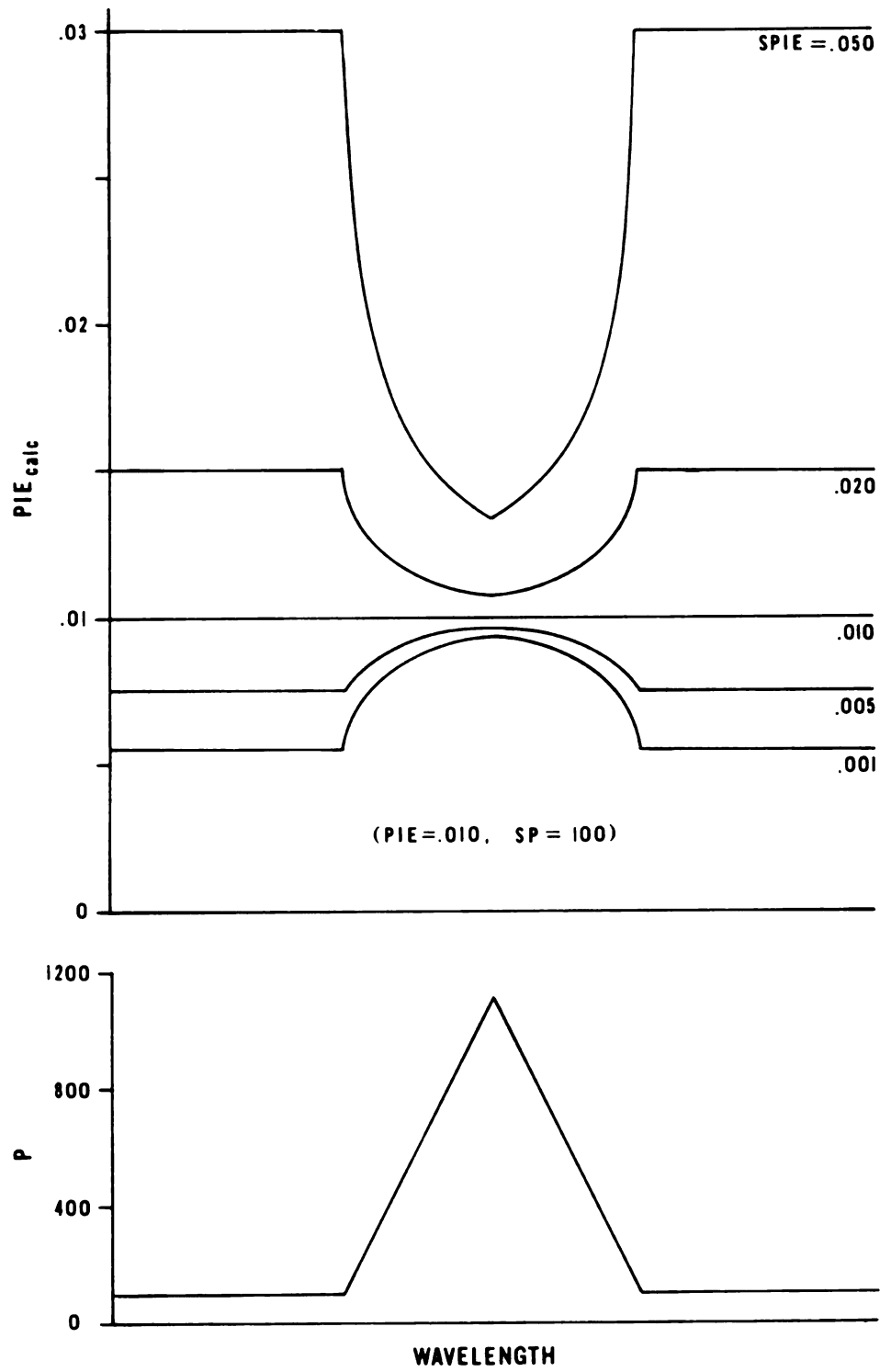


Figure A-4. Calculated Stray Light Effects.

in Figure A-5a for actual data. When such sharp structure is present, it becomes a fairly easy matter to correct the data for the effects of scattered light. The stray light correction factors which can be input by the user in the CORRECT option of PIMS8 (or in PIMS11) essentially allow the user to compensate for SP in the denominator of Equation A-1, and for (SP)(SPIE) in the numerator. The correction factors can be varied until the sharp atomic lines in the light source output are no longer evident in the calculated PIE curve, at which time the user can be confident that the scattered light has been adequately compensated for, as in Figure A-5b.

Near scatter photons are not expected to remain constant as was assumed for SP in the preceding discussion. However, since such photons have wavelengths close to the nominal wavelength of the monochromator setting, any ions created by them will be created with nearly the same efficiency as the true PIE at that wavelength, and therefore the effect of near scatter will usually be small.

C. Detector Misalignment

If an accurately calibrated monochromator is set to, say, $1000 \text{ \AA}$, then the $1000 \text{ \AA}$ photons reaching the grating from the entrance slit will be focused into an image of the entrance slit at the exit slit. After emerging from the exit slit, they will diverge horizontally at an angle which depends on the area of the grating that is illuminated and on its focal length. Similarly, $1000.5 \text{ \AA}$ photons will be focused to an image of the entrance slit which is slightly displaced with respect to

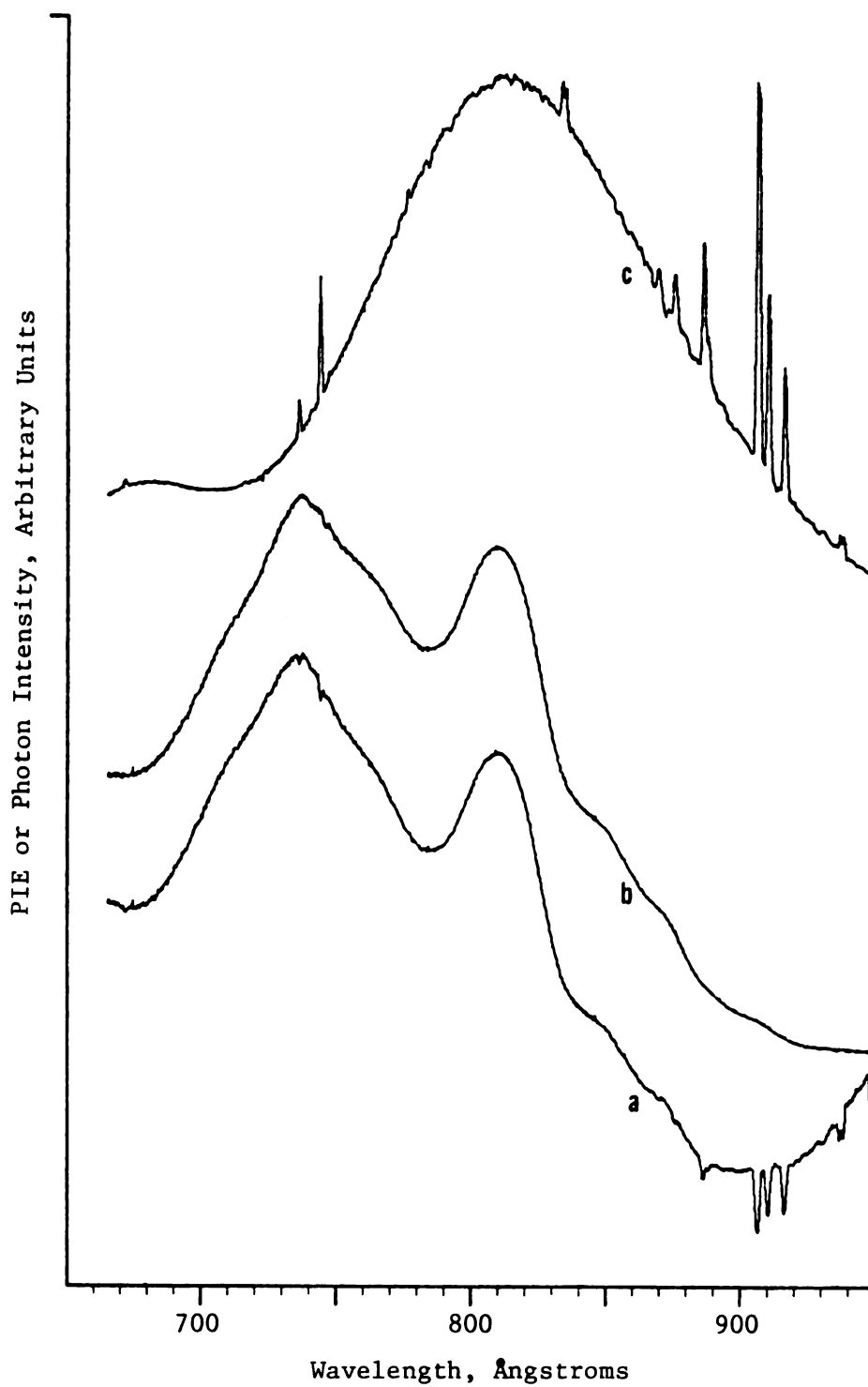


Figure A-5. Experimentally Observed Stray Light Effects. Curve a: PIE curve without stray light corrections; curve b: PIE curve with stray light corrections; curve c: photon intensity.

the 1000 Å slit image. If the exit slit is sufficiently wide, some of these photons may also pass through the exit slit, also diverging at the same angle. Thus, a distribution of wavelengths will pass the exit slit. If the photon detector is not wide enough to intercept the entire photon beam, the wavelength distribution of the photons striking the detector may differ slightly, depending on which portion of the beam it samples (i.e., on its physical position).

Similarly, ions will usually be formed over the entire width of the photon beam; depending on the ion optics used, ions may be collected preferentially from a portion of the ionization region. It is possible, then, that the detected photons are not the same ones that have passed through the region from which ions are collected. When a sharp line is scanned, two effects are possible. The FWHM of the line measured from the ion intensity vs. wavelength curve may differ from the FWHM measured from the photon curve, or the maxima may occur at slightly different wavelengths. In the first case, the PIE curve will exhibit an extra peak on each side of the photon or ion maximum (see Figure A-6). In the second case, the PIE curve will exhibit a "derivative-shaped" peak (Figure A-7), the size of which may vary, but can be quite large. The only cure for such artifacts is to more carefully align the detectors.

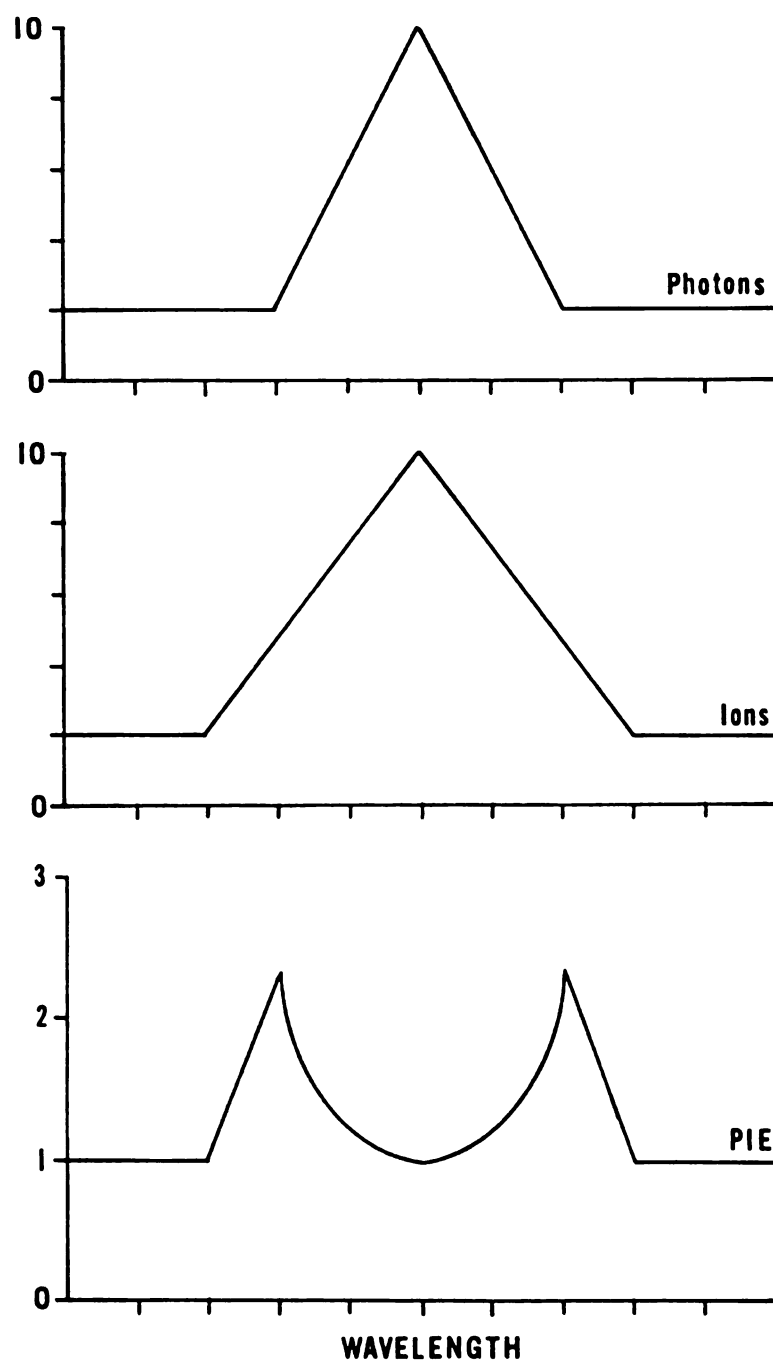


Figure A-6. Calculated PIE for Differing Ion and Photon Linewidths.

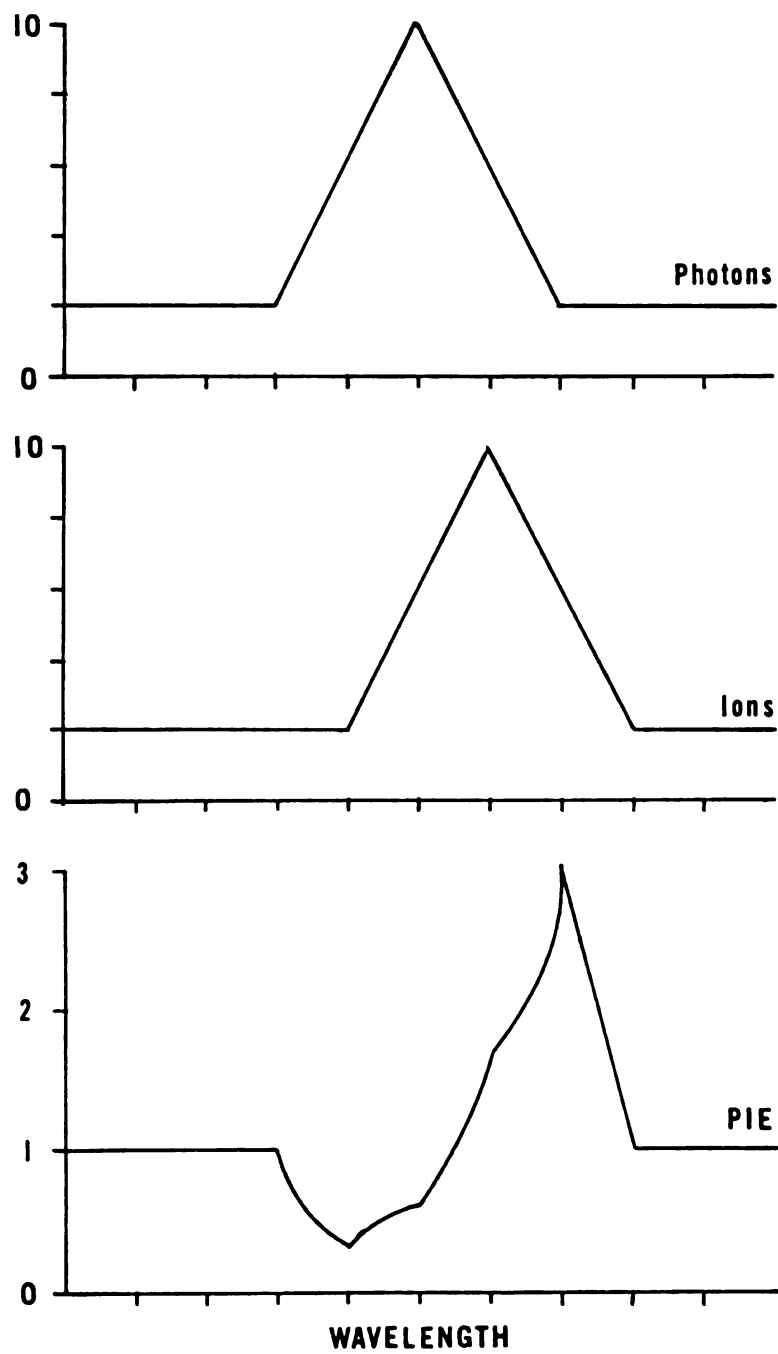


Figure A-7. Calculated PIE for Offset Ion and Photon Peaks.

APPENDIX B

BASIC PDP 8/I INTERFACING

Although it is possible for peripheral devices to access the memory of the computer directly, all of the interfaces described in Chapter IV make use of programmed input and output (I/O) in which data is transferred a word at a time through the computer's main arithmetic register (the accumulator, or AC) under program control. The PDP 8-type computers use a 12 bit word for both data and instructions. These numbers are usually written as four digit octal numbers between 0000 and 7777. All I/O instructions begin with an octal 6 (6xxx), and it is the responsibility of each external device to make sure that it responds only to the appropriate instructions.

All of the interface electronic circuits for the PIMS instrument are built on printed circuit or wire-wrap boards which mount in a card cage in the computer rack. The card cage provides access (via bussed edge-connector sockets) to all the signals which are necessary for programmed I/O. Although the signal descriptions provided below apply strictly only to the PDP 8/I, signals differing in only minor respects may be obtained from newer 8-type computers (such as the 8/E) via a positive I/O bus interface card. All of the interface circuits described in Chapter IV will therefore work (and have been used) with the newer types of PDP 8 computers.

The card cage connectors provide power (+5V, +15V, and -15V)

and ground connections for all of the cards. In addition, the following signal lines to and from the computer are provided:

$\overline{\text{AC IN 00}}^* - \overline{\text{AC IN 11}}^*:$ ¹⁸ These lines are used to transfer data into the computer through its accumulator register. They must be driven by open collector gates whose outputs are always high when not active. During any I/O instruction, an inclusive OR is performed on each bit of the AC and the corresponding bit on the AC IN lines, with the result being stored in the AC. Thus, if the AC is first cleared, and only one device places information on the $\overline{\text{AC IN}}$ bus, the result is that the complement of that information is transferred to the AC during the I/O instruction. Note that bit 11 is the least significant bit on a PDP 8-type computer, and bit 00 the most significant.

BAC 00 - BAC 11: These twelve lines carry the current contents of the accumulator register. They are always active, and are used to export control words or data to peripheral devices from the computer.

BMB 03 - BMB 11: These lines carry the nine least significant bits of the memory buffer, which contains the instruction that is currently being executed. These lines are always active, and are monitored by each external device to determine when an I/O instruction requiring its action is being executed.

IOP1, IOP2, IOP4: These three lines provide pulses for timing I/O

¹⁸ An asterisk after a signal name indicates that the signal is driven by an open-collector gate. A bar over the signal name means that the signal is low level logic (i.e., low when true); otherwise the signal is assumed to be high when true.

operations. The pulses are generated only during an I/O instruction and hence indicate to the peripheral device that the current instruction is an I/O instruction. Although always generated in the sequence IOP1, IOP2, IOP4, which of the pulses is generated depends on the least three significant bits of the I/O instruction. A 6xx1 instruction generates only IOP1; a 6xx2 generates IOP2; a 6xx3 generates IOP1, then IOP2, etc.

CLA*: When this line is pulled low at the start of an I/O instruction, it causes the accumulator to be cleared (all bits set = 0) immediately. It is used to ensure that the AC is cleared before data are transferred into it.

SKP*: This line is active only during I/O instructions. If low during any I/O instruction, it causes the computer to skip the next instruction in the program. The SKP* line is used to cause conditional branching in the program depending on the status of special signals ("flags") in the peripheral devices.

PI*: This line is active at any time the computer's program interrupt facility is enabled by the appropriate software command. When pulled low, the computer begins its program interrupt routine as soon as the current instruction is finished. Due to the difficulty of using program interrupt with the OS/8 operating system, however, none of the PIMS software or interfaces make use of this feature.

RUN: This line is low whenever the computer is running a program.

INIT: This line goes momentarily high when the computer is started, and at the beginning of each new program.

The edge-connector contact assignments for the signals described above are shown in Figure B-1.

The middle two (octal) digits of the BMB are decoded in each peripheral device to provide a signal that is high only when a specific two digits appear, such as x47x in the example shown in Figure B-2. This signal indicates to the device that (if the instruction is an I/O instruction), it is the device selected to perform some task, and the signal is therefore called the Device Select (DS) signal. If a DS signal is gated with an IOP signal (which occurs only during an I/O instruction) to perform some task, then the desired action will occur only during the proper I/O instruction. As an example, if three NAND gates and a flip-flop are connected as shown in Figure B-3, then a 6471 instruction will set the flip-flop, a 6472 instruction will clear it, and a 6474 instruction will cause it to change state. When this technique is used, there are only three possible actions for each DS (one corresponding to each IOP); instructions such as 6476 simply cause more than one of the three. If, however, the last three bits of the BMB are decoded as shown in Figure B-4 to give several signals called Function Select (FS) signals, and DS, FS, and IOP signals are gated together for each task, then seven specific tasks will be possible for each DS code. This makes much more efficient use of the available Device Select codes, and is the technique used in virtually all of the interfacing for the PIMS instrument. In most of the circuit diagrams in Chapter IV, the derivation of the DS and FS signals is not shown. In cases where "BIOP" is used instead of "IOP", it merely means that the IOP signal has been buffered on the circuit board so as not to exceed the drive capacity of the gate which drives the IOP signals down the card cage bus.

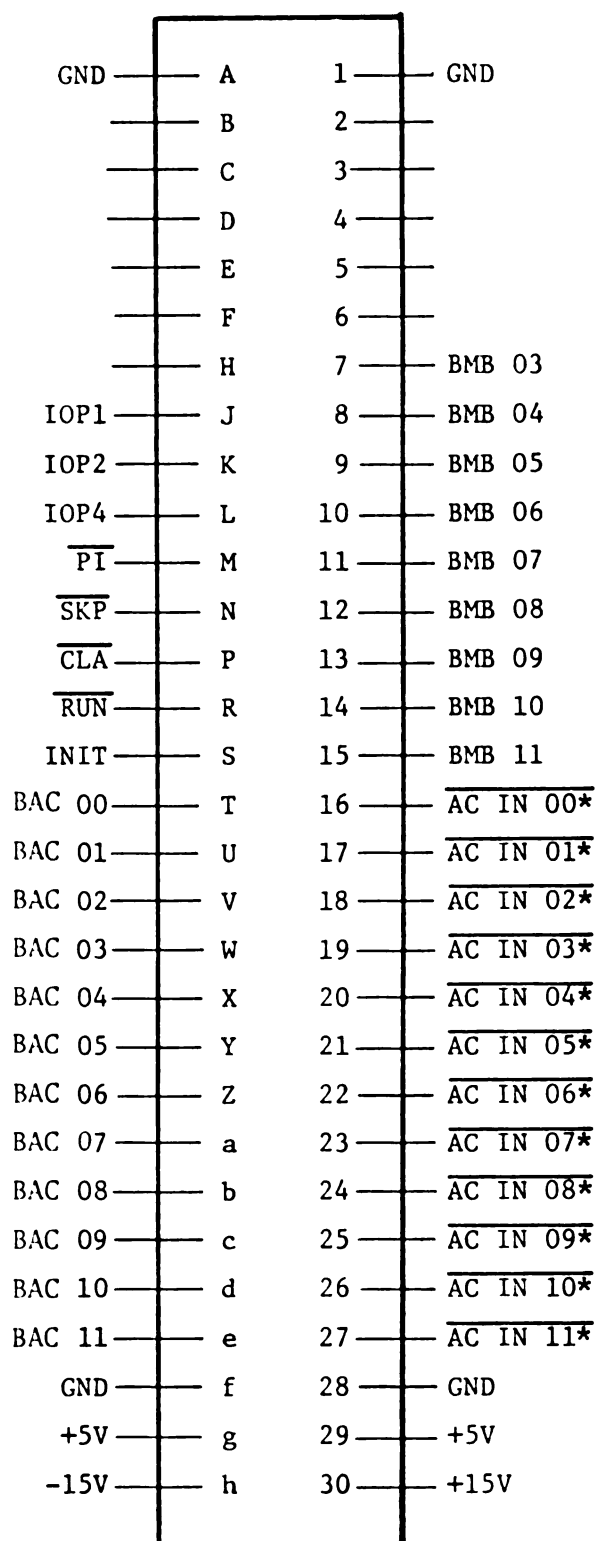


Figure B-1. Edge Connector Contact Assignments.

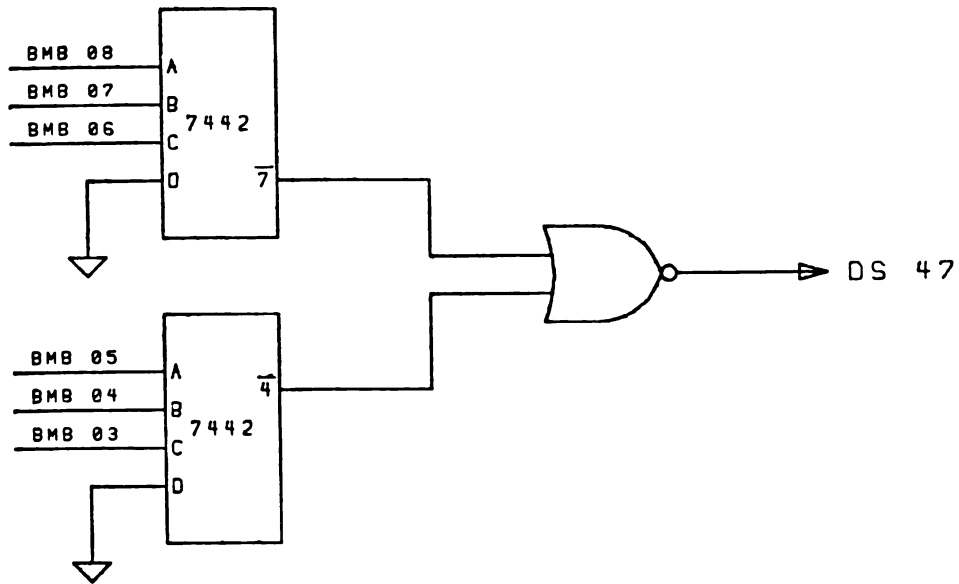


Figure B-2. Generation of a Device Select Signal.

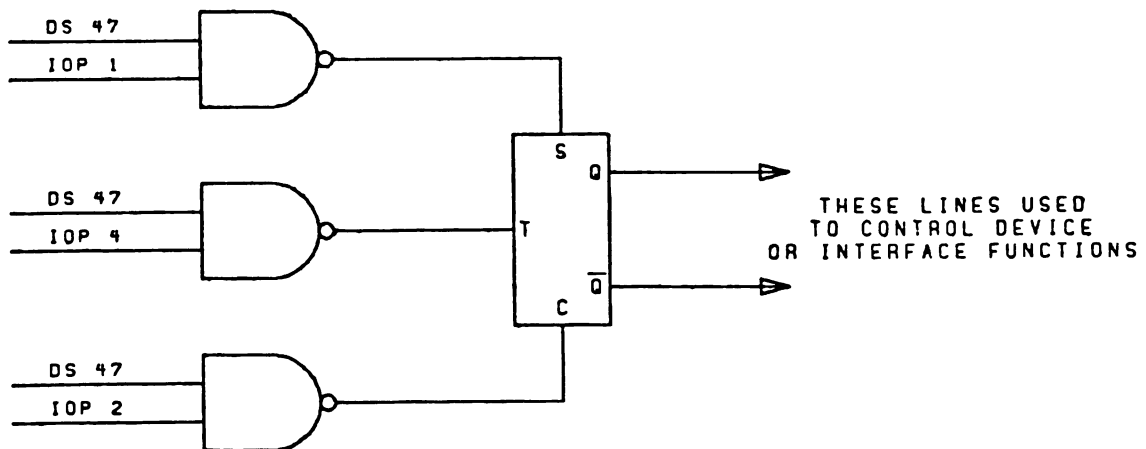


Figure B-3. Gating of DS and IOP Signals.

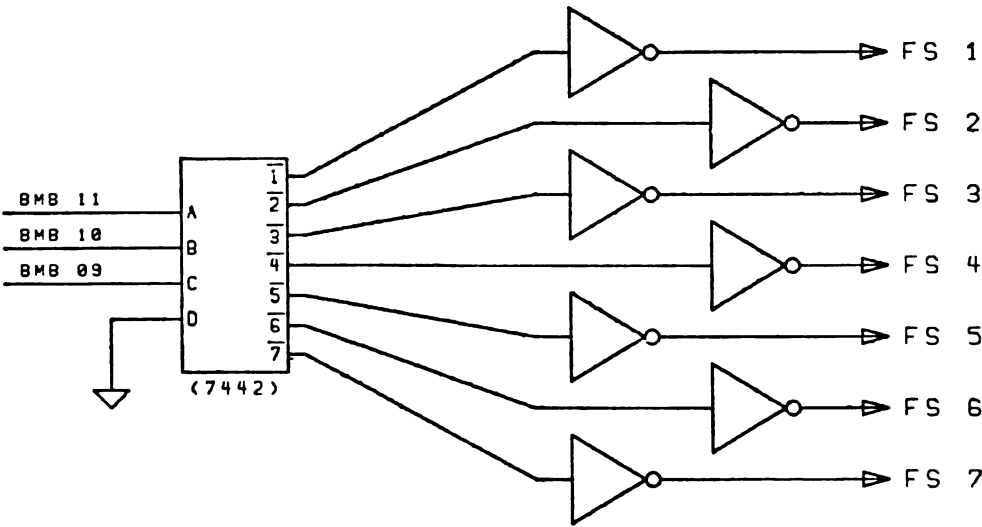


Figure B-4. Generation of Function Select Signals.

APPENDIX C

THE REAL TIME CLOCK

A. Introduction

It is critically important that the computer be able to measure time accurately if it is to perform its tasks properly. There are several types of "clocks" which can be used to measure time in conjunction with computer interfaced instrumentation, each of which has advantages and disadvantages that must be weighed carefully when deciding which type of clock to use for a particular application. Several types of clocks and their relative merits have been discussed by Hahn and Enke [161] in a report which also contains a brief description of a clock that was the prototype for the clock used in the PIMS experiments, and described in this appendix.

Using the terminology of reference [161], this clock can be classified as a mainframe clock -- a versatile, general-purpose clock which interacts solely with the computer (rather than with the instrument or experiment) and is available for use in any of the experiments to which the computer is interfaced. It provides the computer with an accurate time standard which is independent of the computer's cycle time and timing variations, hence the name "real time" clock. One of the unique features of this clock is that it can be regarded and used as two independent clocks, timing two different processes simultaneously.

In terms of clock-computer interactions, one of these clocks (the "Decade Clock") is a simple variable time clock, and the other (the "Programmable Clock") is a combination clock which can function either as a more sophisticated variable time clock, or as an elapsed time clock.

The clocks derive their timing signals from a single crystal-controlled oscillator. They function by raising flags or causing program interrupts at the end of the time period they have been set to measure (or when the counter overflows in the elapsed time mode). In addition, each clock will raise an error flag if a second time period has elapsed since the flags were last cleared.

The Decade Clock can generate time periods ranging from 10^{-6} to 10^2 seconds, in decade steps. The time periods that can be generated by the Programmable Clock can be expressed as $A \times B$, where $A = 1, 2, 3, \dots, 4096$ and $B = 1 \text{ min}, 10 \text{ min}, 1 \text{ hr}, \text{ or } 10^{-6} \text{ to } 10^2 \text{ seconds in decade steps}$. In the elapsed time mode, the Programmable Clock can measure time periods given by $A \times B$, where A and B are defined as above, except that if the overflow and error flags are regarded as extra counter bits, A can range up to 12,288 instead of only 4096.

The front panel of the clock contains light emitting diode (LED) indicators that are used to display the status of the various flags as well as the time base which is selected for each clock. A toggle switch on the front panel controls the decoding of the I/O instructions from the computer as explained in Section C of this appendix. Except for the front panel components, the clock is built on a single circuit board which is mounted in the card cage described in Appendix B.

Section B of this appendix describes the operation of the clock

at the block diagram level. This is sufficient background to understand Section C, which explains the various I/O commands the computer uses to control the clock and gives several short program examples. The final section contains complete schematics for the real-time clock.

B. Basic Operation

A block diagram of the clock is shown in Figure C-1. The Command Decoder is responsible for monitoring the computer instruction that is being executed, and providing the rest of the clock circuitry with various load, clear and drive signals at the appropriate times. The Control Register (CR) is a 12 bit register that can be loaded from the computer. Its contents determine which time bases are selected, whether the program interrupt lines are enabled, and the operating mode of the Programmable Clock.

The heart of the clock is a crystal-controlled 1 MHz oscillator whose output signal is connected to the inputs of two TIME BASE SELECT circuits. Each of these circuits contains a digital divider chain, a multiplexer, and output and control circuits. The output of each circuit is a square wave with a period that is some multiple of the 1 usec oscillator period. These output signals are called the TIME BASE (TB) signals. The periods of the TB signals are determined by seven bits of the CR. Three bits (DEC SELECT) control the Decade TB signal, and four bits (PGM SELECT) control the Programmable TB signal. Clearing the internal digital dividers of the TIME BASE SELECT circuits assures that the next falling edge of each TB signal will occur as close as possible to one full period later.

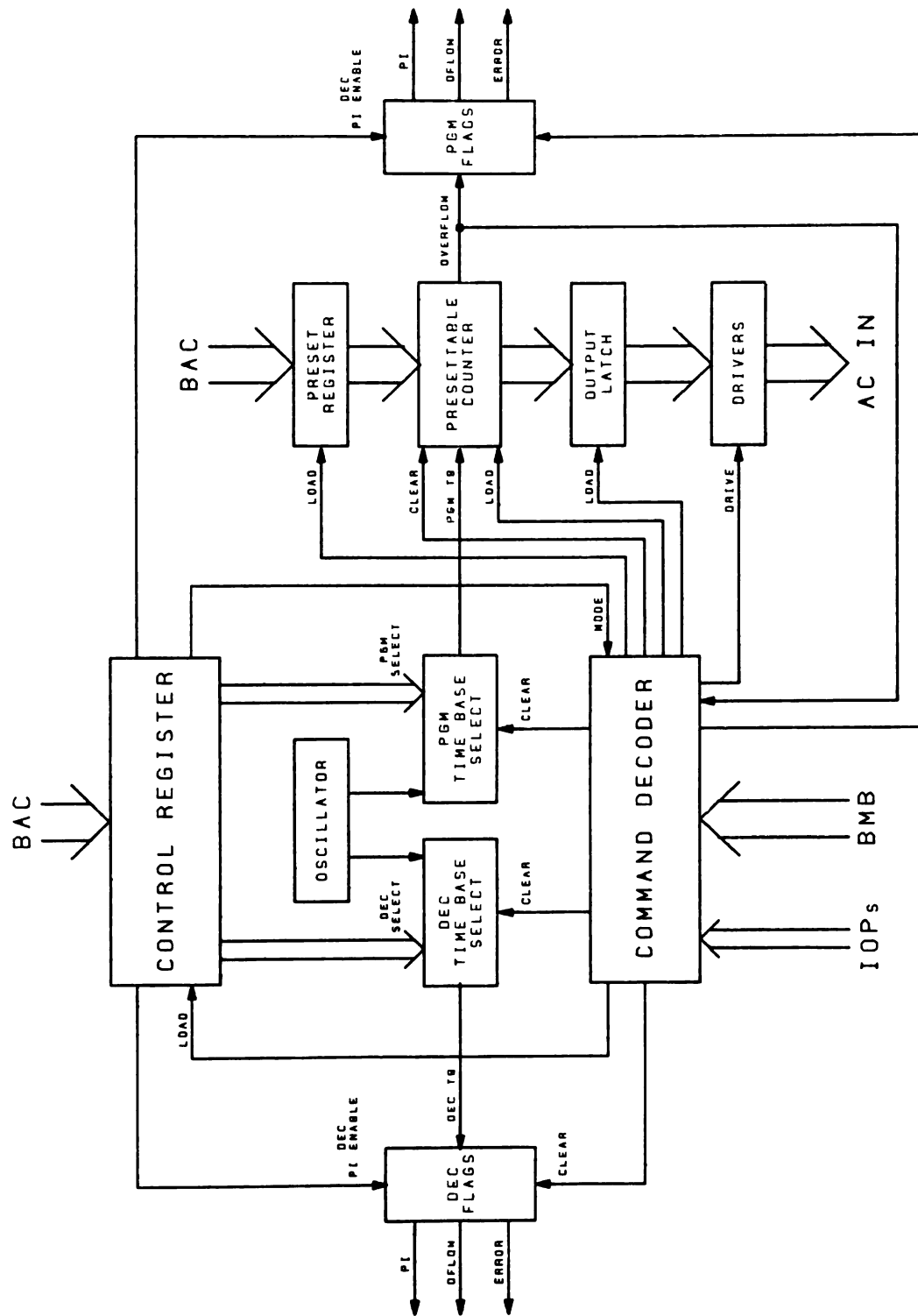


Figure C-1. Block Diagram of the Real-Time Clock.

1. Decade Clock

If the Decade flags have all been cleared and the DEC TIME BASE SELECT circuitry has just been cleared, nothing will happen until the end of one full Decade time period. At that time, the falling edge of the DEC TB signal will cause the Decade flag (OFLOW) to be set. This flag may be checked by the computer. The computer may also clear the flag, in which case it will be set again at the end of the next time period. If the OFLOW flag has not been cleared by the end of a second time period, the second TB signal falling edge will clear the OFLOW flag and set the error flag (ERROR). The ERROR flag will remain set until cleared by the computer, but the OFLOW flag will continue to be alternately set and cleared by succeeding falling edges of the Decade TB signal.

A program interrupt (PI) is asserted by the Decade Clock whenever the OFLOW flag is set and the Decade PI ENABLE bit of the CR is also set.

2. Programmable Clock:

The MODE bit of the CR determines how the Programmable Clock operates. In the variable mode, the 12 bit PRESET REGISTER (PR) is loaded from the computer. Its contents are then loaded into the PRESETTABLE COUNTER (PC), and the PGM TIME BASE SELECT counters and PGM FLAGS are cleared. The falling edges of the PGM TB signal do not set flags directly, but rather increment the PC. The number to which the PC was preset determines how many periods of the PGM TB signal will go by before the PC overflows. The PC overflow triggers the PGM FLAG circuit and causes the PC to be loaded once more with the contents of the PR so that the next overflow will occur after a period equal to the first. The PGM FLAG circuit operation is exactly analogous to the DEC FLAG circuit.

When the Programmable Clock is used in the elapsed time mode, the I/O instruction that clears the flags also clears the PC, which is not loaded from the PR, but rather begins to count the number of PGM TB signal periods that elapse. At some later time, the contents of the counter can be latched into the OUTPUT LATCH and then read into the computer via the DRIVERS. The 12 bit number so obtained can be used to calculate the time that elapsed between the clear instruction and the latch instruction. In this mode, the OFLOW and ERROR flags of the Programmable Clock may be used as an additional $1\frac{1}{2}$ counter bits, though each must be read in separately.

C. Programming

This section contains a detailed explanation of the various I/O instructions used by the clock, as well as a few short sample programs illustrating a few of the possible uses of the clock.

The actual implementation of the I/O instructions in the clock depends on the position of the front panel toggle switch. When the switch is in the dual start position, the Decade and Programmable Clocks may be started independently of each other. If the switch is in the single start position, the COMMAND DECODER interprets the computer's I/O instructions differently, with the result that both clocks must be started simultaneously. This latter mode was included solely to maintain strict software compatability with an earlier version of the clock.

1. Instructions

Table C-1 contains a list of the I/O commands used by the clock, with a description of each. The list of I/O commands is followed by the bit assignments for the CONTROL REGISTER, in Table C-2.

Table C-1. Clock I/O Instructions.

<u>Octal</u>	<u>Mnemonic</u>	<u>Single/Dual</u>	<u>Explanation</u>
6121	LPSET	both	Load Preset Register Loads the contents of the Accumulator into the PRESET REGISTER, from which the counter will be loaded. Since the counter counts to overflow, the two's complement of the number of counts desired must be loaded into this register. (E.g., for a period of 174_8 counts, the Accumulator should contain -174_8 , or 7604_8 .)
6122	CLTB	Single	Clear Time Bases Clears the counters in both of the TIME BASE SELECT circuits so that the first clock intervals will be as close as possible to their full length.
6123	ICNTR	Single	Initialize Counter Clears the PRESETTABLE COUNTER to zero if in the elapsed time mode, or presets it to the number in the PRESET REGISTER if in the variable time mode. Both OFLOW and both ERROR flags are also cleared.
6122	STDEC	Dual	Start the Decade Clock Clears the counters in the DECADE TIME BASE SELECT circuit and clears both Decade flags.
6123	STPGM	Dual	Start the Programmable Clock Clears the PRESETTABLE COUNTER to zero if in the elapsed time mode, or presets it to the value of the PRESET REGISTER if in the variable time mode. Clears

Table C-1. (Cont'd.)

<u>Octal</u>	<u>Mnemonic</u>	<u>Single/Dual</u>	<u>Explanation</u>
			the counters in the PROGRAMMABLE TIME BASE SELECT circuit and clears both Programmable flags.
6124	STCLK	both	Start the Clock A combination of 6122 and 6123. Clears all flags and TIME BASE SELECT circuits. Clears of presets the PRESETTABLE COUNTER, as appropriate.
6125	LCNTR	both	Latch the Counter Latches the contents of the PRESETTABLE COUNTER into the OUTPUT LATCH. Gating is arranged so that a count occurring during the latch instruction has time to ripple through the counter before the latch is closed.
6126	RLACH	both	Read Output Latch Clears the Accumulator, then drives the contents of the OUTPUT LATCH into the Accumulator.
6127	RCNTR	both	Read the Counter A combination of 6125 and 6126. Breaking this instruction into two steps allows the clock to be latched upon the occurrence of some event, and to be examined later during the program.
6131	LCTRL	both	Load Control Register Loads the CONTROL REGISTER with the current contents of the Accumulator. See Table C-2 for bit assignments.
6132	POSKP	both	PGM OFLOW Skip Skips the next instruction if the PGM clock ERROR flag is set.
6134	PFLCL	both	PGM Flag Clear Clear both Programmable Clock flags.
6135	DOSKP	both	DEC OFLOW Skip Skip the next instruction if the Decade Clock OFLOW flag is set.

Table C-1. (Cont'd.)

<u>Octal</u>	<u>Mnemonic</u>	<u>Single/Dual</u>	<u>Explanation</u>
6136	DESKP	both	DEC ERROR Skip Skip the next instruction if the Decade Clock ERROR flag is set.
6137	DFLCL	both	DEC Flag Clear Clear both Decade Clock flags.

Table C-2. Control Register Bit Assignment.

0	1	2	3	4	5	6	7	8	9	10	11
PGM TB C	PGM TB B	PGM TB A	MODE	PGM PI	PGM TB D			DEC PI	DEC TB C	DEC TB B	DEC TB A
MODE = 1 Variable Time MODE = 0 Elapsed Time PI = 1 Interrupt Enabled PI = 0 Interrupt Disabled											
PGM TB				PERIOD		DEC TB					
D	C	B	A			C	B	A			
0	0	0	0	1 usec		0			0	0	
0	0	0	1	10 usec		0			0	1	
0	0	1	0	100 usec		0			1	0	
0	0	1	1	1 msec		0			1	1	
0	1	0	0	10 msec		1			0	0	
0	1	0	1	100 msec		1			0	1	
0	1	1	0	1 sec		1			1	0	
0	1	1	1	10 sec		1			1	1	
1	0	0	0	100 sec							
1	0	0	1	1 min							
1	0	1	0	1 hour							
1	0	1	1	10 min							

2. Programming Examples

The following are several short example programs, or rather sections of programs, which illustrate the several ways in which computer software can use the clock to provide various timing functions.

i. Elapsed Time Measurement:

The goal here is to measure the time between event 1 and event 2. These events are signalled by the raising of two flags, which are tested by the instructions IOSKP1 and IOSKP2. Assume the time interval is known to be about 2 seconds, so the counter will be incremented every 100 usec for maximum accuracy.

```

CLA CLL
TAD (2000    /set counter to 100 usec, elapsed time mode
LCTRL      /load the CR
IOSKP1     /wait for event 1
JMP .-1
STCLK      /start the clock
IOSKP2     /wait for event 2
JMP .-1
RCNTR      /read counter into Accumulator

```

The Accumulator now contains the number of 0.00001 second periods that elapsed between event 1 and event 2.

ii. Variable Time Measurement:

Data is to be acquired at a rate of one point every 1.0 millisecond after an external trigger tested with IOSKP. After data acquisition check to make sure another millisecond has not already passed, which would lead to timing errors.

```

CLA CLL
TAD (0003      /1.0 msec time base
LCTRL         /load CR
IOSKP         /wait for start
JMP .-1
STDEC         /start decade clock
A, DOSKP      /wait for decade OFLOW flag
JMP A
.
.
.
Data Acquisition Routine
.
.
.
DESKP         /has another msec passed while getting data?
SKP           /no; skip past error exit
JMP ERROR     /yes; jump to error routine
DFLCL        /clear flags and
JMP A         /go back to wait for next 1.0 msec period.

```

iii. Combined Operation:

After an initial external event triggers data acquisition (through IOSKP), twenty data points are to be acquired, one every 1.0 msec. This is to be followed by another I/O sequence beginning 23.76 msec after the initial trigger. The Programmable Clock times the 23.76 msec, while the Decade Clock times the 1.0 msec intervals. Either the single or dual start modes could be used here.

```

CLA CLL
TAD (1403      /PGM; 10 usec time base, variable time mode
                /DEC; 1.0 msec time base
LCTRL
CLA CLL
TAD (3270      /2376 counts of 10 usec = 23.76 msec
LPSET
CLA CLL
TAD (7760      /-2010
DCA CNT        /counter for no. of data points
IOSKP         /wait for start
JMP .-1
STCLK         /start both clocks
A, DOSKP      /wait for 1.0 msec flag

```



```

    JMP A
    .
    .
    .
    Data Acquisition
    .
    .
    .
    DFLCL          /clear decade flags
    ISZ CNT        /have twenty points yet?
    JMP A          /no; get another point
    PESKP          /yes; did data acquisition take too long?
    SKP            /no; skip error exit
    JMP ERROR      /yes; take error exit
B,  POSKP         /wait for 23.76 msec flag
    JMP B
    .
    .
    .
    Second I/O sequence
    .
    .
    .

```

D. Circuit Description

1. Control Register

Figure C-2 is the schematic for the control register as well as the decoding for the TB Select LED indicators on the front panel. (In this and all other schematics in this appendix, the numbers in parentheses refer to IC locations on the clock printed circuit board, and are included only for completeness of documentation.) The control register is simply two 74174 hex D flip-flop ICs, which latch the contents of the accumulator on the positive-going edge of the $\overline{\text{LPSET}}$ pulse from the command decoder. Three 7442 BCD-to-Decimal decoder ICs are used to decode the latched DEC SELECT and PGM SELECT bits in such a way that the correct front panel LEDs (corresponding to the time base which has been selected for each

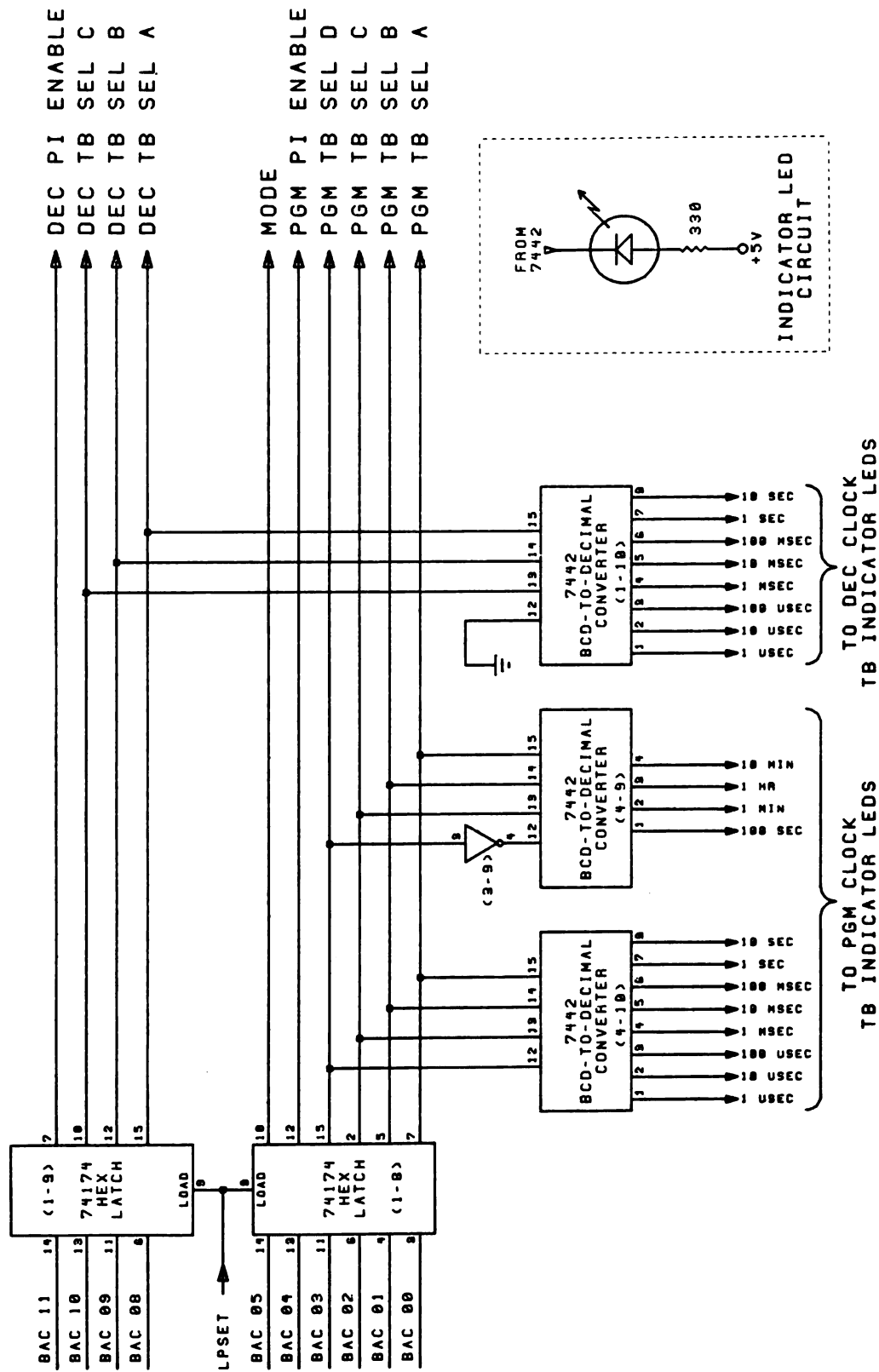


Figure C-2. Control Register and LED Indicator Decoding Circuits.

clock) are lit.

All front panel LEDs are connected exactly the same way, as shown in the inset of Figure C-2. The 330 Ohm resistors limit the current to a value that will not exceed the current-sink capacity of a standard TTL gate. Note that when the output of the gate driving the LED is low, the LED is on.

2. Command Decoder

The command decoder contains circuitry to decode the BMB signals into device select and function select signals. Since this circuitry is similar to that shown in Appendix B, it is not shown here. The remainder of the command decoder circuitry is shown in Figures C-3 and C-4.

Figure C-3 shows the circuitry that generates the various drive and latch signals. The $\overline{\text{LPSET}}$ and $\overline{\text{LCTRL}}$ circuits need no explanation. The DRIVE signal is derived from either of two I/O instructions. The DRIVE signal is inverted and used to pull the $\overline{\text{CLA}^*}$ line low, assuring that the AC is cleared before data is transmitted to the computer. The $\overline{\text{LATCH}}$ signal is also generated by either of two I/O commands. The $\overline{\text{LATCH}}$ circuit includes a monostable to make sure that any PGM TB pulse which occurs during either of these I/O commands will have time to ripple through the presettable counter before the latch is closed. During the I/O command (more precisely, during IOP4), the signal at A is high, which enables the monostable. Should a PGM TB pulse end while the monostable is enabled, its falling edge will trigger the monostable; the $\overline{Q}$ output of the monostable then prolongs the $\overline{\text{LATCH}}$ signal 250 nsec from that time, giving the counter time to settle.

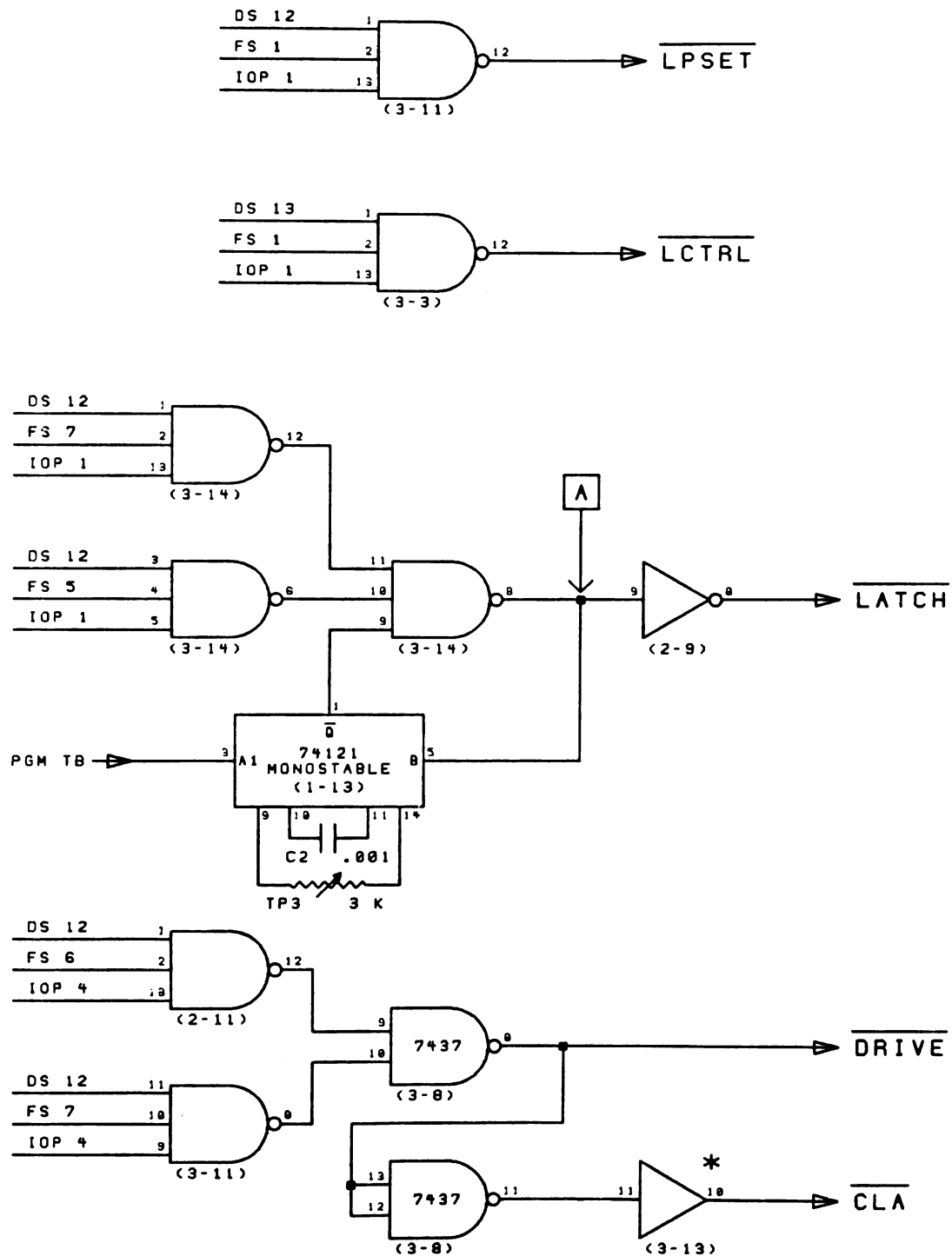
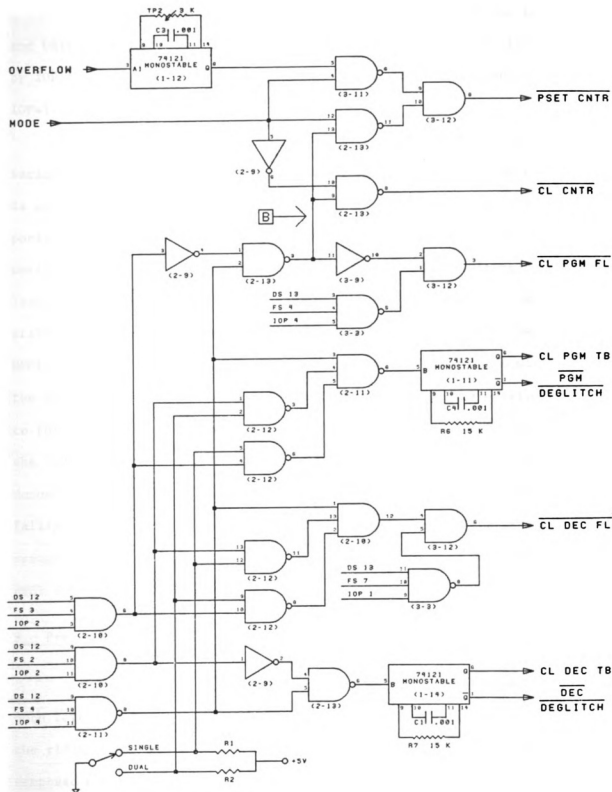


Figure C-3. Latch/Drive Control Circuits (Command Decoder).



Note that although a 6127 instruction triggers both the LATCH and DRIVE signals, different IOPs are used so that the latch (triggered by IOP1) has time to settle before the DRIVE signal is applied (during IOP4).

The circuit that controls the clearing and presetting of the various flags and counters is shown in Figure C-4. Most of the circuit is concerned only with decoding the DS and FS signals according to the position of the front panel toggle switch, however a few things are worth pointing out. The time base select ICs demand a clear pulse at least 10 microseconds long to clear the internal counters, and two monostables are used to lengthen the I/O commands appropriately. The two DEGLITCH signals will be discussed in Part 4. The signal at B will clear the PGM flags and also clear or set the counter. It is gated with MODE to form PSET CNTR, and with MODE to form CL CNTR. The falling edge of the last bit of the presettable counter is used as the overflow signal. Monostable 1-12 is used to provide a pulse of 200 nsec, triggered by the falling edge of the last bit (i.e., it is used as an edge-to-level converter). The output of this monostable is gated with MODE to give the PSET CNTR signal when the counter overflows.

3. Preset Latch, Presettable Counter, Output Latch and Drive

These portions of the clock circuit are shown in Figure C-5. Both latches are composed of 74174 hex D flip-flops, which latch on the rising edge of the appropriate signals. The presettable counter is composed of three 74197 presettable binary counter ICs in series, with the most significant bit (MSB) providing the falling edge that creates

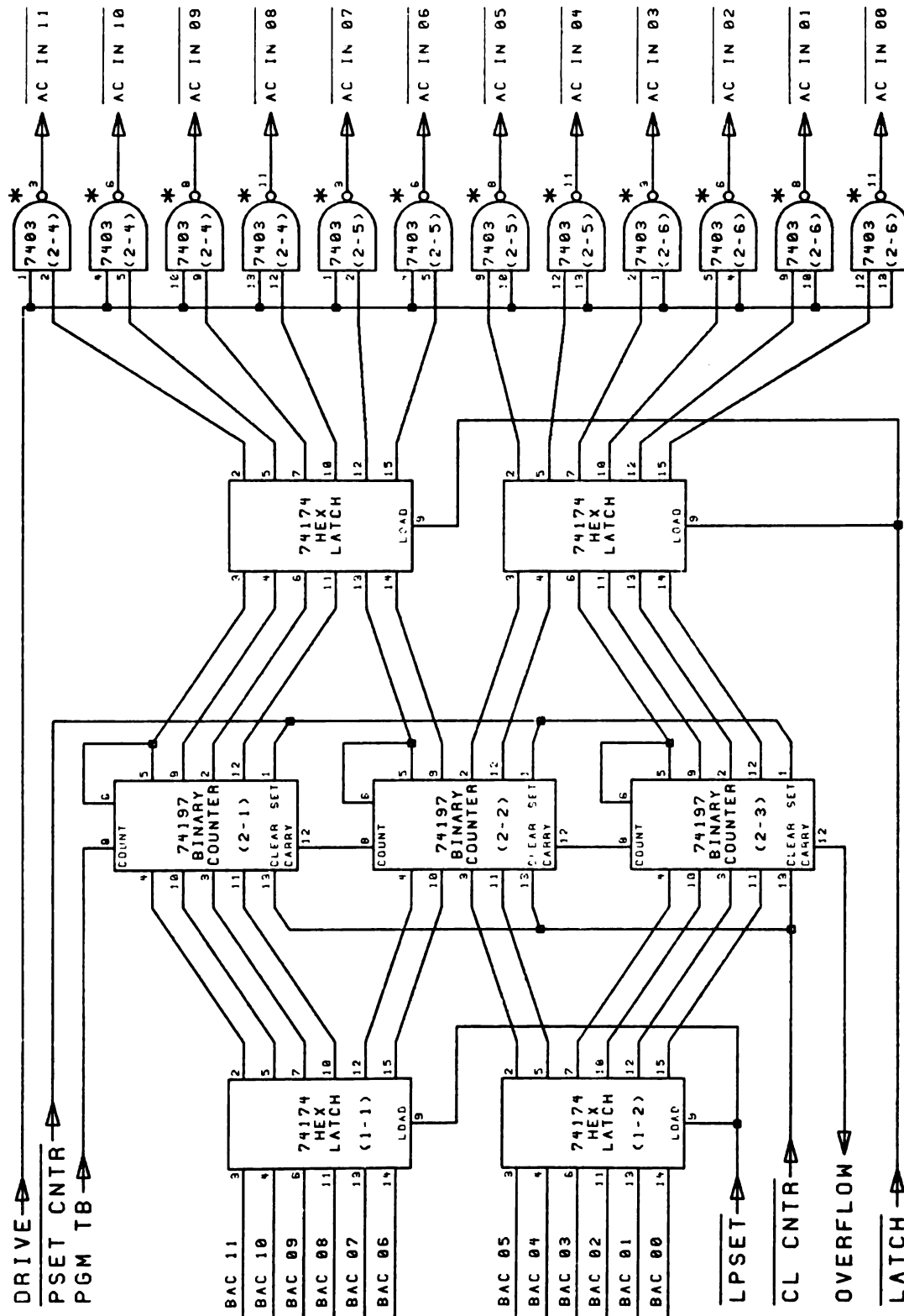


Figure C-5. Latch/Counter/Latch/Driver Circuits (Programmable Clock).

the OVERFLOW signal. The DRIVER is composed of twelve 7403 two input, open collector output NAND gates which connect directly to the AC IN* bus of the card cage.

4. Oscillator and Time Base Select

Figure C-6 shows the 1 MHz crystal oscillator and dual time base select circuit. The oscillator comprises a crystal, two capacitors, and three resistors, plus part of one MK5009P integrated circuit [162]. The oscillator is connected to the input of the first 5009 internally, and externally to the second 5009. The 5009 ICs contain all the frequency dividers and multiplexers required to convert the 1 MHz input to the desired output frequency, which is presented at TIME OUT. The rest of this figure is self explanatory, except the reason for using the DE-GLITCH signals. When a 10 usec pulse is applied to the RESET 0 input of the 5009 (to clear the internal counters), the TIME OUT signal goes low, but does not remain low. Figure C-7 shows what actually happens. The glitch (approximately 2 usec wide) always occurs, and if the rest of the clock were allowed to "see" it, it would be counted as an extra period of the time base. This would result in flags being set before the clear pulse is over, etc. So the complement of the appropriate clear pulse is used to gate the TB output, preventing the glitch from getting to the rest of the circuit.

5. Flag Circuits

The circuit for the Decade Clock Flags is shown in Figure C-8. The circuit for the Programmable Clock Flags is entirely analogous, and

Figure C-6. Oscillator and Time Base Select Circuits.

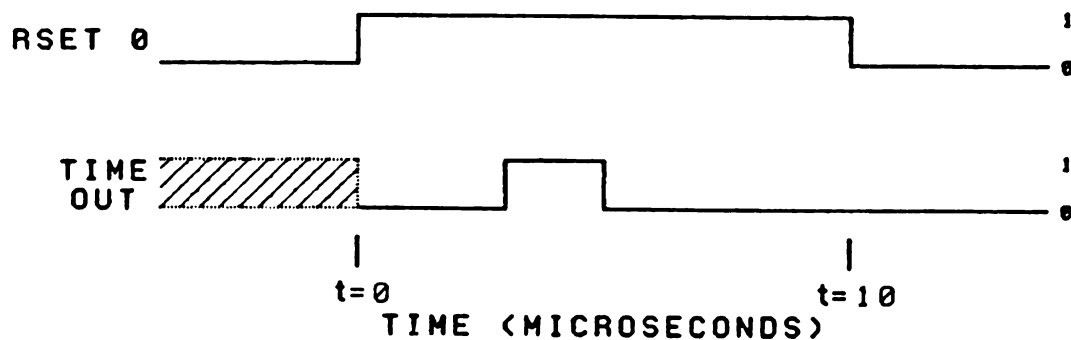


Figure C-7. MK5009 Reset Glitch.

is not shown. Since the TIME OUT signal may be high before a clear pulse occurs (see Figure C-7), it is possible that the DEGLITCH signal applied to NAND gate 3-8 in Figure C-6 will cause a falling edge at the input to the OVERFLOW flag flip-flop (Figure C-8) at the beginning of the clear pulse. To prevent this edge from triggering the flags, the DEGLITCH signal is also applied to the J and K inputs of the flag and error flag flip-flops (which are 74H100 type, edge-triggered flip-flops).

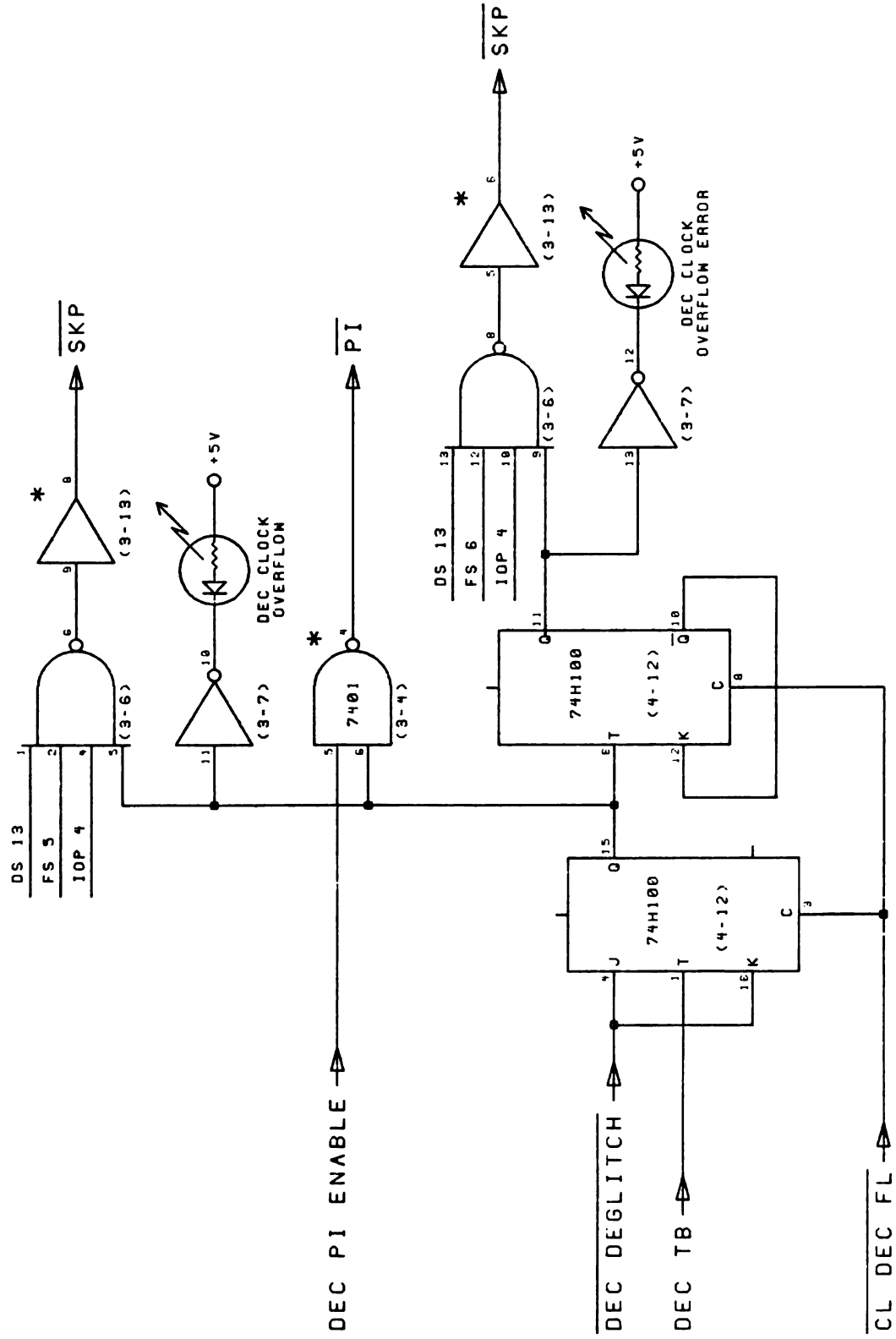


Figure C-8. Decade Clock Flag Circuit.

APPENDIX D

SIGNAL-TO-NOISE RATIO EXPRESSIONS

The goal of the variable integration time method described in Chapter V is to acquire each point at the same signal-to-noise ratio (S/N). In practice, a variation of a few percent in the SNR of the acquired data does not interfere with the realization of the major advantages of the technique. This allows two simplifying assumptions to be made, each of which reduces both the complexity of the necessary programming and the calculation time involved. The first assumption, implicit in the method outlined in Chapter V, is that the preliminary integration yields data of sufficient quality to calculate the amount of additional integration time necessary to within the desired accuracy. The second assumption is that the S/N expression upon which the calculation is based is also sufficiently accurate. In this appendix, the accuracy of the S/N expression used is discussed.

Ingle and Crouch [101, 134] have derived very complete expressions for photon counting signal-to-noise ratios, which are also applicable to ion counting measurements. Equation 37 from Reference [101] gives the S/N as

$$S/N = \frac{N_{ot} - (N_{ob} + N_{od})}{(\sigma_{Not}^2 + \sigma_{Nob}^2 + \sigma_{Nod}^2)^{1/2}} \quad D-1$$

where N_{ot} , N_{ob} , and N_{od} are the observed number of total, background, and dark counts, respectively. The final S/N expression used in any particular application can usually be considerably simplified by taking into account certain practical considerations. For instance, dark count

rates of electron multipliers used as ion transducers are typically only a few counts per minute (see Chapter II, Section G), so that for any reasonable signal intensity, Equation D-1 becomes

$$S/N = \frac{N_{ot} - N_{ob}}{(\sigma_{Not}^2 + \sigma_{Nob}^2)^{1/2}} \quad D-2$$

The total variance terms in the denominator of Equation D-2 are each composed of contributions from quantum, flicker, readout, and excess noise [101]. The contributions of excess noise can be made negligible by setting the counter threshold for most stable operation (see Chapter IV, Section F). Since the lamp output is very stable when proper operating conditions are used, source flicker noise is small. Furthermore, since the background signal in PIMS arises only from scattered light, background flicker noise will also be negligible. Finally, readout noise of one count can also be neglected in comparison to the number of counts usually collected during the preliminary integration period. The only remaining significant noise source is quantum noise, so that Equation D-2 reduces [101] to

$$S/N = \frac{N_{ot} - N_{ob}}{(N_{os} + 2N_{ob})^{1/2}} = \frac{N_{os}}{(N_{os} + 2N_{ob})^{1/2}} \quad D-3$$

where the number of observed "signal" counts, N_{os} , are assumed to be the difference between N_{ot} and N_{ob} .

Equation D-3 is frequently encountered in the photon counting literature. It is valid only when the simplifications discussed above are legitimate, and when the integration times used in measuring N_{ot} and N_{ob} are equal, an assumption implicit in the derivations in

references [101, 134]. This is usually not the case during PIMS experiments performed by PIMS8-RECORD, but Equation D-3 can be extended to non-equal integration times as follows. Let N_{ot} be the total number of counts accumulated in an integration time of t seconds. It is assumed to be the sum of background counts (N_b) and the desired signal counts (N_s). We would like to calculate the signal as

$$S = N_{ot} - N_b = (N_s + N_b) - N_b. \quad D-4$$

We do not know N_b , however, and in its place we use an estimate based on an integration of background alone for t' seconds. The relationship between the estimate used in the calculation, N_{eb} , and the observed background counts, N'_{ob} , is

$$N_{eb} = N'_{ob} \left(\frac{t}{t'} \right) \quad D-5$$

so

$$S \equiv N_{ot} - N_{eb} = (N_s + N_b) - N'_{ob} \left(\frac{t}{t'} \right). \quad D-6$$

The signal variance is found by application of standard propagation of error mathematics [140] to Equations D-5 and D-6:

$$\sigma_s^2 = \sigma_{N_s}^2 + \sigma_{N_b}^2 + \sigma_{N_{eb}}^2 \quad D-7$$

$$\left(\frac{\sigma_{N_{eb}}}{N_{eb}} \right)^2 = \left(\frac{\sigma_{N_{ob'}}}{N_{ob'}} \right)^2 + \left(\frac{\sigma_t}{t} \right)^2 + \left(\frac{\sigma_{t'}}{t'} \right)^2 \quad D-8$$

Since the measurement of t and t' is very precise, the relative standard deviations of these quantities are insignificant, and rearrangement of Equation D-8 yields

$$\sigma_{N_{eb}}^2 = \sigma_{N'_{ob}}^2 \left(\frac{N_{eb}}{N'_{ob}} \right)^2 = \sigma_{N'_{ob}}^2 \left(\frac{N_{ob}(t/t')}{N'_{ob}} \right)^2$$

D-9

$$\sigma_{N_{eb}}^2 = \sigma_{N'_{ob}}^2 \left(\frac{t}{t'} \right)^2.$$

The signal-to-noise ratio can now be written (from Equations D-6, D-7, and D-10) as

$$S/N = \frac{N_{ot} - N_{eb}}{(\sigma_{N_s}^2 + \sigma_{N_b}^2 + \sigma_{N'_{ob}}^2 \left(\frac{t}{t'} \right)^2)^{1/2}} = \frac{(N_s - N_b) - N_{eb}}{(\sigma_{N_s}^2 + \sigma_{N_b}^2 + \sigma_{N'_{ob}}^2 \left(\frac{t}{t'} \right)^2)^{1/2}}.$$

D-11

The variance terms in Equation D-11 are composed of elements similar to those making up the variance terms in Equations D-1 and D-2. Therefore, the same sort of simplifications can be made, and

$$S/N = \frac{(N_s + N_b) - N_{eb}}{[N_s + N_b + N_{ob} \left(\frac{t}{t'} \right)^2]^{1/2}} = \frac{N_s + N_b - N_{eb}}{[N_s + N_b + N_{eb} \left(\frac{t}{t'} \right)^2]^{1/2}}. \quad D-12$$

Since Equation D-6 is based on the assumption that $N_b = N_{eb}$, Equation D-12 can be reduced to

$$S/N = \frac{N_s}{[N_s + N_b \left(1 + \frac{t}{t'} \right)]^{1/2}} \quad D-13$$

which is the equivalent of Equation D-3, with the integration times for background and (background + signal) included explicitly. The variable integration time method described in Chapter V is based on the assumption that

$$S/N = N_s^{1/2} \quad D-14$$

and the usual manually-applied variable integration time method assumes

$$S/N = N_t^{1/2} . \quad D-15$$

Equation 13 can be used to calculate the amount that the actual S/N will differ from the desired S/N when counts are accumulated to a constant N_s or N_t . The results of such calculations are shown in Table D-1 for the case where $t' = 10 t$, which is typical of the relationship between background and preliminary integration times used in PIMS8-RECORD.

One important conclusion is that, in the presence of background (i.e., stray light) signal, integrating to a constant N_s is more nearly equivalent to integrating to a constant S/N than is the constant N_t method. Also, assuming that a 10% variation in the S/N of the acquired data is acceptable (actually +0%, -10%), then the calculations used in PIMS8-RECORD are perfectly adequate even if the background amounts to nearly 25% of the actual signal.

Table D-2. Error Involved in Using Simplified S/N Expressions.

Ratio of N_b/N_s	Percentage Error ^a	
	Constant N_s	Constant N_t
.01	0.5	1.0
.02	1.1	2.1
.05	2.6	5.0
.10	5.1	9.5
.15	7.4	13.6
.25	11.4	20.8
.50	19.7	35.1
1.00	31.0	51.2

^aThe true S/N is taken to be that calculated from Equation D-13. The entries show the percentage by which the true S/N is lower than the S/N calculated from D-14 (Constant N_s) and D-15 (Constant N_t).

APPENDIX E

SOFTWARE

A. Introduction

This Appendix contains listings of nearly all of the programs and subroutines necessary to the operation of the Michigan State University photoionization mass spectrometer. The main program of the PIMS8 system, PIMS8.FT, is presented first in Section B. It is followed by the remaining PIMS8 system programs and subroutines, in alphabetical order. The only programs not included are those in the SCSYS and TEKLIB subroutine libraries, and those involved in executing the CORECT option under the PIMS8 system. The three subroutines in SCSYS, written by B.K. Hahn, allow data to be plotted on a strip chart recorder with analog Y-axis input and stepper motor driven chart paper. The TEKLIB subroutines allow plotting on the Tektronix T4010 terminal, and are available from DECUS [163]. The CORECT option subroutines are not included because Section C contains all of the PIMS11 programs and subroutines, which accomplish the same tasks but on a PDP-11 computer (see Chapter V).

The first file in Section C, PIMS11.ODL, is used to specify to the DOS/BATCH program LINK [144] how the various subroutines are to be loaded. PIMSCO.FTN, used to initialize common variables, is next, followed by PIMS00.FTN, PIMS1A.FTN, and PIMS1B.FTN, which together form the "main" PIMS11 program. The remaining subroutines follow in alphabetical order, omitting only those device-specific programs which handle output to

whatever graphics devices are used.

The final section of this Appendix contains definitions of the common variables used in the PIMS8 and PIMS11 programs.

PIMS8.FT, CONT'D.

```

S      CLA CLL
S      TAD I DATE
S      DCA TEMP
S      TAD TEMP
S      SZA
S      JMP EX
S      JMP ER
S      CPAGE 2
S      SDATE, 6211
S      7666
S      STEMP, 0
S      SER, NOP
S      WRITE (1,2000)
5      CALL EXIT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DATE IS SET, SO UNPACK AND SAVE IN IDAY,IMO,IYR.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SEX,    AND (7
S      DCA \IYR
S      TAD TEMP
S      RAR,RTR
S      AND (37
S      DCA \IDAY
S      TAD TEMP
S      CLL RAL; RTL; RTL
S      AND (17
S      DCA \IMO
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WRITE OPTIONS AND PROMPT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10      IF (ITEK - 1) 13,11,13
11      CALL ERASE
        CALL POSIT (0,0)
        WRITE (1,2001) IMO, IDAY, IYR
        WRITE (1,2018)
        GO TO 15
13      IF (INITF - 1) 14,15,14
14      WRITE (1,2016) IMO, IDAY, IYR
15      CALL BELL
        WRITE (1,2002) IOPT,
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      READ IN THE DESIRED OPTION AND BRANCH TO THE CORRECT PART
C      OF PIMS8.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        CALL LETIN (IDUM)
        IF (IDUM - 3) 20,5,20
20      IF (IDUM - 7) 21,10,21
21      IF (IDUM - 590) 22,700,22
22      IF (IDUM - 204) 24,50,24
24      IF (IDUM - 297) 25,100,25
25      IF (IDUM - 213) 26,150,26
26      IF (IDUM - 847) 27,200,27
27      IF (IDUM - 976) 28,250,28
28      IF (IDUM - 1042) 29,300,29
29      IF (IDUM - 1157) 30,400,30
30      IF (IDUM - 1219) 31,450,31
31      IF (IDUM - 1285) 32,500,32
32      IF (IDUM - 1289) 33,550,33
33      IF (IDUM - 1293) 10,600,10

```

PIMS8.FT, CONT'D.

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CLOSE SHUTTER.  CHECK INTERFACE FIRST.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
50      CONTINUE
S      SHSKP
S      JMP A
S      NOP
55      WRITE (1,2011)
        GO TO 15
SA,     SHCL
S      CLA CLL
        GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CORRECT DATA.  CHAIN TO PROGRAM CORECT.FT
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
100     IGOBK = 0
        CALL CHAIN ('CORECT')
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CURRENT.  OUTPUT WAVELENGTH, ENERGY
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
150     IF (INITF - 1) 700,155,700
155     CALL EOUT
        GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MOVE.  REPORT CURRENT WAVELENGTH, READ DESIRED WAVELENGTH.
C      CHECK TO MAKE SURE THE DESIRED WAVELENGTH IS REASONABLE, AND
C      THEN MOVE TO THE NEW WAVELENGTH.  REPORT NEW WAVELENGTH AND
C      ENERGY.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
200     IF (INITF - 1) 700,205,700
205     CALL EOUT
        READ (1,2006) RDES
        IF (RDES) 15,210,210
210     CALL WAVCK (0.,RDES,UPLIM)
        IF (RDES) 200,220,220
220     RDES = RDES * WSPA
        CALL DFIX (RDES,IH,IL)
        CALL DFLOT (IH,IL,RDES)
        CALL MOVE (RDES)
        GO TO 150
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      OPEN SHUTTER.  CHECK INTERFACE FIRST.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
250     CONTINUE
S      SHSKP
S      JMP B
S      NOP
        WRITE (1,2011)
        GO TO 55
SB,     SHOP
S      CLA CLL
        GO TO 15

```

PIMS8.FT, CONT'D.

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      PRESCALE SELECT.  INPUT THE AMOUNT OF PRESCALING TO BE USED
C      ON THE ION AND PHOTON COUNT INPUT.  EACH IS CHECKED TO MAKE
C      SURE IT CORRESPONDS TO AN AMOUNT OF PRESCALING THAT IS
C      ACTUALLY AVAILABLE (CURRENTLY THE COUNTS MAY BE PRESCALED IN
C      THE PULSE COUNTERS BY 1,2,4, OR 16) BEFORE BEING ADOPTED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
300      READ (1,2012) IDUM
          IF (IDUM) 15,305,301
301      IF (IDUM - 1) 302,305,302
302      IF (IDUM - 2) 303,306,303
303      IF (IDUM - 4) 304,307,304
304      IF (IDUM - 16) 300,308,300
305      IPREA = 1
          IDUM = 0
          GO TO 309
306      IPREA = 2
          IDUM = 1
          GO TO 309
307      IPREA = 4
          IDUM = 2
          GO TO 309
308      IPREA = 16
          IDUM = 3
309      CONTINUE
          S      CLA CLL
          S      TAD \IDUM
          S      PRESA
          S      CLA CLL
310      READ (1,2013) IDUM
          IF (IDUM) 15,315,311
311      IF (IDUM - 1) 312,315,312
312      IF (IDUM - 2) 313,316,313
313      IF (IDUM - 4) 314,317,314
314      IF (IDUM - 16) 310,318,310
315      IPREB = 1
          IDUM = 0
          GO TO 319
316      IPREB = 2
          IDUM = 1
          GO TO 319
317      IPREB = 4
          IDUM = 2
          GO TO 319
318      IPREB = 16
          IDUM = 3
319      CONTINUE
          S      CLA CLL
          S      TAD \IDUM
          S      PRESB
          S      CLA CLL
          GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      RECORD.  CHAIN TO RECORD.FT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
400      IF (INITF - 1) 700,405,700
405      IGOBK = 0
          CALL CHAIN ('RECORD')
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SCAN.  CALL SUBROUTINE SCAN.FT
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
450      IF (INITF - 1) 700,455,700
455      CALL SCAN

```

PIMS8.FT. CONT'D.

```

GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      TEST COUNTERS. READ INTEGRATION TIME, CONVERT TO INTEGER
C      NUMBER OF 0.100 SECOND PERIODS, THEN LOOP UNTIL G.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
500    READ (1,2009) RTEM
        IF (RTEM) 15,15,505
505    RTEM = RTEM * 10.
        CALL DFIX (RTEM,IH,IL)
510    CALL COUNT (IH,CNTP,CNTI)
        WRITE (1,2010) CNTP, CNTI
        CALL KCHEK (IH)
        IF (IH) 510,510,515
515    WRITE (1,2017) IH
        GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      TIMED DATA ACQUISITION. CHAIN TO ACQUIR.FT
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
550    IF (INITF - 1) 700,551,700
551    IF (ITEK) 560,560,555
555    CALL ERASE
560    CALL CHAIN ('ACQUIR')
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      TRANSFER FILES. CHAIN TO TRANS.FT
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
600    CALL CHAIN ('TRANS')
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      INITIALIZE. IF FIRST CALL, READ IN ALL INITIALIZATION
C      VARIABLES, OTHERWISE JUST A NEW WAVELENGTH.
C      NOTE THAT LINE 802 SHOULD BE DELETED AND RLINE INPUT
C      FROM THE KEYBOARD IF GRATINGS OF DIFFERENT DISPERSION ARE
C      USED. (PRESENTLY ONLY 1200 L/MM GRATINGS ARE AVAILABLE)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
700    IF (INITF - 1) 705,805,705
705    READ (1,2003) RTEM
        IF (RTEM) 15,710,710
710    RTEM = 100000. / RTEM
        IF (RTEM - 66.) 715,720,720
715    RTEM = 66.
720    IF (RTEM - 4096.) 730,730,725
725    RTEM = 4096.
730    CALL DFIX (RTEM,IH,MWATE)
        CALL DFLOT (IH,MWATE,RTEM)
        RTEM = 100000. / RTEM
        WRITE (1,2007) RTEM, IDUM,
            CALL YESNO (IDUM)
            IF (IDUM - 3) 735,5,735
735    IF (IDUM - 7) 740,15,740
740    IF (IDUM - 89) 705,745,705
745    WRITE (1,2008) IDUM,
            CALL YESNO (IDUM)
            IF (IDUM - 3) 750,5,750
750    IF (IDUM - 7) 755,15,755
755    IF (IDUM - 89) 760,765,760
760    IACCL = 0
        GO TO 770
765    IACCL = 1
770    WRITE (1,2015) IDUM,
            CALL YESNO (IDUM)
            IF (IDUM - 3) 775,5,775
775    IF (IDUM - 7) 780,15,780

```


PIMS8.FT, CONT'D.

```

780     IF (IDUM - 89) 785,790,785
785     ITEK = 0
        GO TO 800
790     ITEK = 1
800     READ (1,2020) REDUC
        IF (REDUC) 15,800,801
801     IF (REDUC- 10.) 802,802,800
802     RLINE = i200.
        WSPA = (RLINE*REDUC) / 30.
        UPLIM = 3600000. / RLINE
        READ (1,2004) ISCNO
        IF (ISCNO) 15,805,805
805     READ (1,2005) RTEM
        IF (RTEM) 15,810,810
810     IF (RTEM - UPLIM) 815,815,805
815     WSCNT = RTEM * WSPA
        INITF = 1
        GO TO 150
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     FORMAT STATEMENTS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2000     FORMAT ('PLEASE DATE SYSTEM')
2001     FORMAT ('PIMS8  VERSION 6.B      ',I2,'/',I2,'/197',I1,'//5X
1'CLOSE SHUTTER',11X,'PRESCALE SELECT'/5X,'CORRECT DATA',12X
2,'RECORD DATA'/5X,'CURRENT WAVELENGTH',6X,'SCAN WAVELENGTH')
2018     FORMAT (5X,'INITIALIZE PIMS8',8X,'TEST COUNTERS'/5X,'MOVE
1 WAVELENGTH',9X,'TIMED ACQUISITION'/5X,'OPEN SHUTTER',12X,
2'TRANSFER FILES')
2002     FORMAT ('->',I0)
2003     FORMAT ('MAXIMUM SCAN RATE, STEPS/SECOND = ',F8.2)
2004     FORMAT ('NEXT SCAN TO BE NUMBER : ',I4)
2005     FORMAT ('CURRENT WAVELENGTH = ',F10.4)
2006     FORMAT ('DESIRED WAVELENGTH = ',F10.4)
2007     FORMAT ('ACTUAL = ',F10.3,' , OKAY? ',I0)
2008     FORMAT ('USE ACCELERATION ? ',I0)
2009     FORMAT ('INTEGRATION PERIOD [SECONDS] = ',F10.1)
2010     FORMAT ('P= ',E15.8,' I= ',E15.8)
2011     FORMAT ('PLEASE PUT SHUTTER CONTROL IN AUTO MODE!')
2012     FORMAT('PRESCALE PHOTONS BY : ',I5)
2013     FORMAT ('PRESCALE IONS BY : ',I5)
2014     FORMAT ('USING ION COUNTING ? ',I0)
2015     FORMAT ('ARE YOU USING THE TEK TERMINAL? ',I0)
2016     FORMAT (I2,'/',I2,'/197',I1,'//CLOSE      CORRECT      CURRENT
1  INIT      MOVE      OPEN//PRESCALE  RECORD      SCAN
2  TEST      TIMED      TRANSFER')
2017     FORMAT (I0)
2020     FORMAT ('GEAR RATIO ONE TO = ',F15.7)
        END

```

```

CCCC ACALC.FT CCCC 1/23/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      FUNCTION ACALC.FT          1/23/77          E. J. DARLAND          CC
CC
CC      PURPOSE:
CC      TO CALCULATE DARK-CORRECTED PHOTON INTENSITY
CC      ION INTENSITY, OR PHOTOIONIZATION EFFICIENCY FOR USE
CC      IN PROGRAM ACQUIR.FT. ONE OF THE FOLLOWING
CC      CALCULATIONS IS PERFORMED:
CC
CC      
$$Y = ACALC = \frac{(CNTI / TIMEP) - DI}{(CNTP / TIMEP) - DP}$$

CC
CC      
$$Y = ACALC = (CNTP / TIMEP) - DP$$

CC
CC      
$$Y = ACALC = (CNTI / TIMEP) - DI$$

CC
CC      USAGE:  Y = ACALC (ITYPE)
CC
CC      PARAMETER DESCRIPTION:
CC      ITYPE:  DETERMINES WHETHER PIE, PHOTON OR ION TYPE
CC              OF CALCULATION IS TO BE DONE BY ACALC:
CC              1:  PIE
CC              2:  PHOTON
CC              3:  ION
CC
CC      CORE OCCUPIED:          2 PAGES
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      FUNCTION ACALC (ITYPE)
      COMMON I1,BAKP,BAKI,CNTP,CNTI,TIMEP
      DIMENSION I1(21)
      GO TO (10,20,30) ITYPE
10      ACALC = ((CNTI / TIMEP) - BAKI)/((CNTP/TIMEP) - BAKP)
      RETURN
20      ACALC = (CNTP / TIMEP) - BAKP
      RETURN
30      ACALC = (CNTI / TIMEP) - BAKI
      RETURN
      END

```

```

CCCC ACQUIR.FT CCCC 1/23/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      PROGRAM ACQUIR.FT          1/23/77          E. J. DARLAND          CC
CC
CC      PURPOSE:
CC
CC      TO RECORD PHOTON AND ION INTENSITIES AS A FUNCTION
CC      OF TIME. FOR THIS PURPOSE, ACQUIR.FT IS SOMEWHAT MORE
CC      ACCURATE THAN RECORD.FT. (SEE CHAPTER V FOR A
CC      DESCRIPTION OF THE DIFFERENCES BETWEEN THE TWO.)
CC
CC      USAGE:
CC      ACQUIR.FT IS AUTOMATICALLY CHAINED INTO THE
CC

```

```

CC COMPUTER'S MEMORY WHEN THE 'TIMED ACQUISITION' OPTION CC
CC IS CHOSEN. CC
CC NOTE THAT THE ELAPSED TIME RECORDED WITH EACH CC
CC POINT IS MERELY THE SUM OF THE INTEGRATION PERIODS CC
CC SINCE THE BEGINNING OF THE RUN. IN OTHER WORDS, IT CC
CC IS ASSUMED THAT THE TIME SPENT BETWEEN POINTS STORING CC
CC THE DATA IS MUCH LESS THAN THE MINIMUM 0.10 SECOND CC
CC INTEGRATION TIME. CC
CC
CC INTERNAL VARIABLES: CC
CC   DEV: DEVICE NAME FOR DATA STORAGE. CC
CC   ELAPS: ELAPSED TIME FROM START OF THE RUN. CC
CC   FILNAM: FILE NAME FOR DATA STORAGE. CC
CC   IDATA: DETERMINES TYPE OF DATA TO BE RECORDED: CC
CC           1: PIE (BOTH PHOTONS AND IONS) CC
CC           2: PHOTONS ONLY CC
CC           3: IONS ONLY CC
CC   TIME: TOTAL INTEGRATION TIME, SUMMED OVER ALL CC
CC          POINTS. CC
CC   XSCAL: ABSCISSA SCALING FACTOR FOR PLOT. CC
CC   YXCAL: ORDINATE SCALING FACTOR FOR PLOT. CC
CC
CC   Y: GENERAL PURPOSE REAL VARIABLE. CC
CC   IOPT: GENERAL PURPLSE INTEGER VARIABLE. CC
CC
CC SUBROUTINES CALLED: ACALC KCHEK CC
CC                   COUNT LIMIT CC
CC                   DELAY POSIT CC
CC                   DFIX STOR CC
CC                   DFLOT YESNO CC
CC                   FILE CC
CC
CC                   <FROM TEKLIB>: BELL CC
CC                   <CHAINS TO>: PIMS8 CC
CC
CC MEMORY OCCUPIED: 15 PAGES FIELD 1 CC
CC                  3 PAGES FIELD 0 CC
CC                  (FOR DEVIDE INDEPENDENT I/O) CC
CC
CC LOADING INSTRUCTIONS: LOAD WITH THE /I/O OPTIONS, AND CC
CC                      LOAD SUBROUTINE FILE NEXT. CC
CC
CC REQUIRED OPERATION CODES: CC
CC S OPDEF SHCL 6441 /CLOSE SHUTTER CC
CC S OPDEF SHOP 6442 /OPEN SHUTTER CC
CC S SKPDF SHSKP 6443 /SKP IF SHUTTER ON MANUAL CC
CC S SKPDF ABSKP 6434 /SKP IF COUNTER ON MANUAL CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
COMMON WSCNT,WSPA,R1,R2,IPREA,IPREB,ITEK,ISCNO,I3,IMO,
1IDAY,IYR,IPT,BAKP,BAKI,CNTP,CNTI,TIMEP,NPTS,ITIM
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C READ IN NEW PARAMETERS, TOTAL TIME AND NUMBER OF POINTS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
29 IF (ITEK) 22,22,21
21 CALL POSIT (1,0)
22 READ (1,1001) TIME
IF (TIME) 155,20,25
25 IF (ITEK) 27,27,26
26 CALL POSIT (2,0)
27 READ (1,1002) NPTS
IF (NPTS) 155,25,30
30 IF (NPTS - 550) 31,31,25
31 TIME = (10. * TIME) / FLOAT(NPTS)
CALL DFIX (TIME,IH,ITIM)
CALL DFLOT (0,ITIM,TIME)

```

[illegible]

ACQUIR.FT. CONT'D.

```
C  
C      FIND MAXIMA AND MINIMA OF ACQUIRED DATA, BACKGROUND CORRECTED.  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
95    IF (ITEK) 155,155,95  
     BAKP = BAKP / 10.  
     BAKI = BAKI / 10.  
     IPT = 1  
     CALL STOR (1)  
     Y = AALCALC (IDATA)  
     YMIN = Y  
     YMAX = Y  
     DO 125 IPT = 2,NPTS  
     CALL STOR (1)  
     Y = AALCALC (IDATA)  
     IF (Y-YMAX) 110,110,105  
105   YMAX = Y  
110   IF (YMIN - Y) 125,125,120  
120   YMIN = Y  
125   CONTINUE  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C  
C      SET UP SCALING AND DRAW BOX AROUND SCREEN.  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
     YSCAL = 600. / (YMAX - YMIN)  
     XSCAL = 1023. / FLOAT (NPTS)  
     CALL FDIS (0,0,0)  
     CALL FDIS (1,1023,0)  
     CALL FDIS (1,1023,600)  
     CALL FDIS (1,0,600)  
     CALL FDIS (1,0,0)  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C  
C      DO THE PLOT.  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
DO 150 IPT = 1, NPTS  
CALL STOR (1)  
Y = AALCALC (IDATA)  
Y = (Y - YMIN) * YSCAL  
X = XSCAL * FLOAT (IPT)  
IY = LIMIT (0,Y,600)  
IX = LIMIT (0,X,1023)  
150 CALL FDIS (-1,IX,IY)  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C  
C      ASK FOR RETURN TO PIMS8, SCAN AGAIN, NEW PARAMETERS, OR  
C      STORE DATA.  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
153 IF (ITEK) 156,156,157  
157 CALL POSIT (0,0)  
156 READ (1,1000) IOPT  
IF (IOPT) 153,155,160  
160 IF (IOPT - 4) 163,163,155  
165 GO TO (175,45,20,180) IOPT  
175 CALL CHAIN ('PIMS8')  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C  
C      STORING DATA--GET FILE SPECIFICATINS.  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
180 CALL POSIT (7,0)  
IERR = 0  
CALL FILE (DEV, FILNAM, IERR)  
IERR = IERR + 3  
GO TO (155,33,180,190) IERR
```

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WRITE OUT DATA.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
190    CALL OOPEN (DEV, FILNAM)
      IREC = 5
      WRITE (4,1011) IREC, FILNAM
      IREC = 6
      WRITE (4,1012) IREC, IMO, IDAY, IYR
      IREC = 7
      WRITE (4,1012) IREC, ISCNO
      ISCNO = ISCNO + 1
      IREC = 14
      X = 10.
      Y = 5.
      WRITE (4,1011) IREC, X, Y
      IREC = 20
      WRITE (4,1012) IREC, IPREA, IPREB
      IREC = 4
      WAVE = WSCNT / WSPA
      ELAPS = (TIMEP / 2.)
      WRITE (4,1011) IREC, WAVE, BAKP, BAKI, ELAPS
      IREC = 17
      CNTP = 1.0
      WRITE (4,1011) IREC, WAVE, CNTP, CNTP, ELAPS
      DO 230 IPT = 1, NPTS
      ELAPS = ELAPS + TIMEP
      CALL STOR (1)
      CNTP = CNTP / TIMEP
      CNTI = CNTI / TIMEP
      GO TO (200,210,220) IDATA
200    WRITE (4,1011) IDATA, WAVE, CNTP, CNTI, ELAPSE, TIMEP
      GO TO 230
210    WRITE (4,1011) IDATA, WAVE, CNTP, ELAPSE, TIMEP
      GO TO 230
220    WRITE (4,1011) IDATA, WAVE, CNTI, ELAPSE, TIMEP
230    CONTINUE
      IREC = 99
      WRITE (4,1011) IREC
      CALL OCLOSE
      GO TO 155
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      FORMAT STATEMENTS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1000   FORMAT('PIMS [1] REPEAT [2] NEW [3] OR STORE [4] : ',I2)
1001   FORMAT('TOTAL SCAN TIME = ',E15.8)
1002   FORMAT('NUMBER OF POINTS : ',I5)
1003   FORMAT('I4, POINTS AT ',F5.1, ' SEC. = ',F9.1, ' SEC. TOTAL
      1. OKAY? ',I0)
1004   FORMAT('PIE [1], PHO [2] OR ION [3] : ',I2)
1005   FORMAT('SHUTTER NOT IN AUTO MODE.')
1006   FORMAT('COUNTER NOT IN AUTO MODE.')
1007   FORMAT ('<CR> TO START',I2)
1011   FORMAT(A2,5A6)
1012   FORMAT(6A2)
      END

```

COUNT.SB

```

//// COUNT.SB //// 1/22/77 //////////////////////////////////////
//
// SUBROUTINE COUNT.SB      1/22/77      E.J. DARLAND      //
//
// PURPOSE:
//   TO USE THE 24 BIT DUAL COUNTER TO INTEGRATE PHOTON
//   AND ION INTENSITY FOR A PERIOD OF TIME BETWEEN 0.0 AND
//   409.6 SECONDS. THE USER OR CALLING PROGRAM MUST MAKE
//   SURE THE COUNTER IS SET TO AUTO OR 24-BIT MODE.
//
// USAGE:  CALL COUNT ( ISEC,CNTA,CNTB)
//
// PARAMETER DESCRIPTION:
//   ISEC = INTEGER NUMBER OF 0.100 SECOND
//   INTERVALS TO INTEGRATE THE COUNTS,
//   0<=ISEC<=4096.
//   CNTA = THE NUMBER OF COUNTS ACCUMULATED IN
//   CHANNEL A. IF A COUNTER OVERFLOW
//   ERROR HAS OCCURED, CNTA WILL BE SET
//   TO THE NEGATIVE OF THE COUNT READ IN
//   FROM THE COUNTER, WHICH WILL BE TOO
//   SMALL BY AT LEAST 1 0000 0000 .
//
//   CNTB = THE NUMBER OF COUNTS ACCUMULATED IN
//   CHANNEL B. IF AN OVERFLOW ERROR IS
//   ENCOUNTERED, CNTB IS ALSO NEGATED.
//
// MEMORY OCCUPIED:      1 PAGE
//
//
//
//
// OPDEF ITAD 1400      /INDIRECT TAD TO FOOL SABR
// OPDEF IDCA 3400      /INDIRECT DCA TO FOOL SABR
// OPDEF LPSET 6121     /LOAD CLOCK PRESET REGISTER
// OPDEF STCLK 6124     /START THE CLOCK
// OPDEF LCTRL 6131     /LOAD CLOCK CONTROL REGISTER
// SKPDF POSKP 6132     /SKIP ON PCM CLOCK OVERFLOW FLAG
// SKPDF PESKP 6133     /SKIP ON PCM CLOCK ERROR FLAG
// OPDEF PFLCL 6134     /CLEAR PCM CLOCK FLAGS
// SKPDF SKPAF 6411     /SKIP ON COUNTER A OVERFLOW FLAG
// SKPDF SKPAE 6412     /SKIP ON COUNTER A ERROR FLAG
// OPDEF CLAF 6413      /CLEAR COUNTER A FLAGS
// OPDEF RLSA 6414      /READ LEAST SIG. 12 BITS OF COUNT A
// OPDEF RMSA 6415      /READ MOST SIG. 12 BITS OF COUNT A
// SKPDF SKPBF 6421     /SKIP ON COUNTER B OVERFLOW FLAG
// SKPDF SKPBE 6422     /SKIP ON COUNTER B ERROR FLAG
// OPDEF CLBF 6423      /CLEAR COUNTER B FLAGS
// OPDEF RLSB 6424      /READ LEAST SIG. 12 BITS OF COUNT B
// OPDEF RMSB 6425      /READ MOST SIG. 12 BITS OF COUNT B
// OPDEF ABOPN 6431     /OPEN A,B COUNTER GATE
// OPDEF ABCLS 6432     /CLOSE A,B COUNTER GATE
// OPDEF ABCLR 6433     /CLEAR A,B COUNTERS
// SKPDF ABSKP 6434     /SKIP IF COUNTERS ARE NOT ON AUTO
// OPDEF ABS24 6436     /SET COUNTERS TO 24 BIT FLAGS
//
//
//
//
// STORAGE
//
//
// EXP,      2330
// ISEC,      0
// SET,      3400
// TEM1,      0
// TEM2,      0
// K0001,    0001
// K4002,    4002

```

COUNT.SB, CONT'D.

```

////////////////////////////////////
/
/      ENTRY BLOCK FOR ARGUMENT PASSAGE.
/
////////////////////////////////////
/      ENTRY COUNT
COUNT, BLOCK 2
////////////////////////////////////
/
/      GET THE FIRST ARGUMENT, AS WELL AS ADDRESSES AND DATA FIELDS
/      FOR THE OTHER TWO ARGUMENTS, WHICH ARE PASSED BACK LATER
/
////////////////////////////////////
/      CLA CLL
/      TAD COUNT
/      DCA A
A,    HLT          /THEN CDF MAIN
/      ITAD COUNT#
/      DCA B        /B IS NOW CDF ISEC
/      INC COUNT#
/      ITAD COUNT#
/      DCA ISEC     /ISEC IS NOW ADDR ISEC
/      INC COUNT#
/      ITAD COUNT#
/      DCA C        /C IS NOW CDF CNTA
/      INC COUNT#
/      ITAD COUNT#
/      DCA TEM1     /TEM1 IS NOW ADDR CNTA
/      INC COUNT#
/      ITAD COUNT#
/      DCA D        /D IS NOW DEF CNTB
/      INC COUNT#
/      ITAD COUNT#
/      DCA TEM2     /TEM2 IS NOW ADDR CNTB
/      INC COUNT#
B,    HLT          /BY NOW IS CDF ISEC
/      ITAD ISEC
/      DCA ISEC     /NOW ISEC IS TRULY ISEC
/      AND I B      /CDF CURRENT
////////////////////////////////////
/
/      INITIALIZE COUNTER.
/
////////////////////////////////////
/      ABS24
/      ABCLS
/      ABCLR
/      CLAF
/      CLBF
////////////////////////////////////
/
/      IF ISEC = 0, RETURN. OTHERWISE, SET UP PGM CLOCK
/      FOR THE REQUIRED NO. OF 100 MSEC WAIT PERIODS.
/
////////////////////////////////////
/      CLA CLL
/      TAD ISEC
/      SNA          /ZERO?
/      JMP RETN     /YES:GO BACK
/      CIA          /NO: FORM COMPLEMENT
/      LPSET        /AND LOAD PRESET REGISTER.
/      CLA CLL
/      TAD SET      /SET CONTROL REGISTER TO 100 MSEC PERIODS,PGM.
/      LCTRL
////////////////////////////////////
/
/      START CLOCK AND OPEN COUNTER GATE.
/
////////////////////////////////////

```


COUNT.SB, CONT'D.

```

      STCLK
      ABOPN
////////////////////////////////////
/
/      LOOP TO WAIT FOR END OF CLOCK PERIOD, THEN CLOSE GATE.
/
////////////////////////////////////
E,      POSKP
        JMP E
        ABCLS
////////////////////////////////////
/
/      RETURN CNTA, CNTB
/
////////////////////////////////////
RETN,   CLA CLL
C,      HLT          /BY NOW CDF CNTA
        TAD EXP      /GET EXPONENT
        SKPAF        /WAS THERE A COUNTER FLAG?
        SKP
        TAD K0001    /YES: SET 25TH BIT = 1
        SKPAE        /NO: WAS THERE AN ERROR?
        SKP
        TAD K4002    /YES: CHANGE SIGN BIT, SET 26TH BIT = 1
        IDCA TEM1    /NO: NOW RETURN FIRST WORD OF CNTA
        INC TEM1
        RMSA         /GET MOST SIG. 12 BITS (EXCLUDING FLAG BITS?)
        IDCA TEM1    /AND RETURN.
        INC TEM1
        RLSA         /SIMILARLY FOR LEAST SIGNIFICANT
        IDCA TEM1
D,      HLT          /BY NOW CDF CNTB; GO THROUGH THE SAME PROCEDURE
        TAD EXP      /AS FOR CNTA IN ORDER TO RETURN CNTB.
        SKPBF
        SKP
        TAD K0001
        SKPBE
        SKP
        TAD K4002
        IDCA TEM2
        INC TEM2
        RMS3
        IDCA TEM2
        INC TEM2
        RLSB
        IDCA TEM2
        CLA CLL      /PREPARE TO RETURN TO MAIN PROGRAM:
        AND I C      /GET SABR TO CDF CURRENT WITH INDIRECT.
        RETRN COUNT  /AND RETURN.
        END

```

DELAY.SB

```

//// DELAY.SB 2/28/76 //////////////////////////////////////
//
// SUBROUTINE DELAY.SB 2/28/76 E.J. DARLAND //
//
// PURPOSE: //
// TO CAUSE A DELAY IN PROGRAM EXECUTION RANGING FROM //
// 0.0 TO 409.6 SECONDS. //
//
// USAGE: CALL DELAY (IVAL) //
//
// PARAMETER DESCRIPTION: //
// IVAL = NO. OF 0.1 SEC INTERVALS IN THE TOTAL DELAY. //
//
// MEMORY OCCUPIED: 1 PAGE //
//
////////////////////////////////////
OPDEF ITAD 1400
OPDEF LPSET 6121
OPDEF LCTRL 6131
OPDEF STCLK 6124
SKPDF PCSKP 6132
////////////////////////////////////
/
/ ENTRY BLOCK FOR ARGUMENT PASSAGE
/
////////////////////////////////////
ENTRY DELAY
DELAY, BLOCK 2
////////////////////////////////////
/
/ GET THE ARGUMENT
/
////////////////////////////////////
CLA CLL
TAD DELAY
DCA A
A, HLT /THEN CDF MAIN, ADDR IVAL, AND IVAL
ITAD DELAY#
DCA B /B IS NOW CDF IVAL
INC DELAY#
ITAD DELAY#
DCA A
INC DELAY#
B, HLT /THEN CDF IVAL
ITAD A
DCA A /A IS NOW IVAL
////////////////////////////////////
/
/ EXECUTE
/
////////////////////////////////////
TAD (5405 /.1SEC TIMEBASE, VARIABLE MODE
LCTRL
CLA CLL
TAD A
CIA
LPSET
STCLK
C, POSKP
JMP C
CLA CLL
AND I A
CLA CLL
RETRN DELAY
END

```

DFIX.SB

```

//// DFIX.SB //// 9/24/75 //////////////////////////////////////
//
// SUBROUTINE DFIX.SB      9/24/75      E.J. DARLAND      //
//
// PURPOSE:
//   THE REAL-TO-INTEGER FIX ROUTINE AVAILABLE IN THE
//   FORTRAN II LIBRARY CREATES SINGLE PRECISION (12 BIT)
//   TWO'S COMPLEMENT INTEGERS. THERE ARE MANY PLACES IN THE
//   PIMS8 PROGRAMS WHERE EITHER A SINGLE OR DOUBLE PRECISION
//   INTEGER WAS NEEDED IN STRAIGHT BINARY CODE. THIS
//   SUBROUTINE CONVERTS OS/8 FORTRAN II REAL VARIABLES INTO
//   DOUBLE PRECISION (24 BIT) STRAIGHT BINARY INTEGERS.
//
// USAGE:  CALL DFIX (RVAR, IH, IL)
//
// PARAMETER DESCRIPTION:
//   RVAR =  THE FORTRAN REAL VARIABLE TO BE CONVERTED.
//           SEE NOTE BELOW ON ACCEPTABLE RANGE.
//   IH   =  THE MOST SIGNIFICANT 12 BITS OF THE
//           RESULTING INTEGER.
//   IL   =  THE LEAST SIGNIFICANT 12 BITS.
//
// ***** CAUTION *****
//
//   DFIX WORKS BY TAKING THE MOST SIGNIFICANT 24
//   BITS OF THE MANTISSA OF RVAR (IE, IT TRUNCATES THE
//   LAST THREE BITS OF MANTISSA) AND ROTATING THEM INTO
//   THE DOUBLE PRECISION INTEGER IH,IL BY A NUMBER
//   OF PLACES EQUIVALENT TO RVAR'S BINARY EXPONENT.
//   THIS PROCEDURE RESULTS IN CORRECT CONVERSIONS
//   PROVIDING THAT RVAR HAS A VALUE BETWEEN 0 OCTAL
//   (0 DECIMAL) AND 77777777 OCTAL (16777215 DECIMAL),
//   INCLUSIVE. THIS SUBROUTINE DOES NOT CHECK TO SEE
//   IF RVAR IS ACTUALLY WITHIN THIS RANGE!!!
//
// MEMORY OCCUPIED:      1 PAGE
//
//
// OPDEF ITAD 1400      /INDIRECT TAD TO FOOL SABR
// OPDEF IDCA 3400      /INDIRECT DCA TO FOOL SABR
//
//
// CONSTANT AND VARIABLE STORAGE
//
// V1,      0
// V2,      0
// V3,      0
// M7770,   7770
// M0007,   0007
// M3770,   3770
// N200,    7600
//
//
// ENTRY BLOCK FOR SUBROUTINE ARGUMENT PASSAGE
//
// ENTRY DFIX
// DFIX,    BLOCK 2
//
// GET THE FIRST ARGUMENT (RVAR)
//
// CLA CLL
// TAD DFIX
// DCA A

```

DFIX.SB, CONT'D.

```

A,      HLT                      /THEN CDF MAIN, ADDR ARG1, IH
        ITAD DFIX#
        DCA B                    /B IS NOW CDF ARG
        INC DFIX#
        ITAD DFIX#
        DCA A                    /A IS NOW ADDR ARG1
        INC DFIX#
B,      HLT                      /THEN CDF ARG, IL
        ITAD A
        DCA V1
        INC A
        ITAD A
        DCA V2
        INC A
        ITAD A
        DCA V3
        DCA A
        DCA B                    /CLEAR A,B FOR USE AS IH, IL
////////////////////////////////////
/
/      ROTATE MANTISSA RIGHT BY 3 BITS TO GET A 24 BIT
/      MANTISSA. NOTE THAT THIS AUTOMATICALLY TRUNCATES THE
/      LAST THREE BITS.
/
////////////////////////////////////
        CLA CLL
        TAD M7770
        AND V3
        RTR
        RAR
        DCA V3
        TAD M0007
        AND V2
        RTR
        RTR
        TAD V3
        DCA V3                    /V3 IS NOW LOWER 12 BITS OUT OF 24
        TAD M7770
        AND V2
        RTR
        RAR
        DCA V2
        TAD M0007
        AND V1
        RTR
        RTR
        TAD V2
        DCA V2                    /V2 IS NOW UPPER 12 BITS OUT OF 24
////////////////////////////////////
/
/      EXTRACT EXPONENT FROM V1
/
////////////////////////////////////
        CLA CLL
        TAD M3770
        AND V1
        RTR
        RAR
        TAD N200
        SPA                      /CHECK FOR NEGATIVE EXPONENT
        JMP RET                  /EXP<0; RETURN WITH I=0
        CIA                      /EXP OK. FORM NEGATIVE FOR USE IN:
        DCA V1                    /V1, AS COUNTER FOR NO. OF ROTATIONS.
////////////////////////////////////
/
/      ROTATE MANTISSA EXPONENT BITS LEFT INTO A,B (IH, IL)
/
////////////////////////////////////
C,      CLA CLL

```

DFIX.SB, CONT'D.

```

DCA F
DCA G
TAD V3
JMS ROT
DCA V3
TAD V2
JMS ROT
DCA V2
TAD B
JMS ROT
DCA B
TAD A
JMS ROT
DCA A
ISZ V1          /DONE EXP ROTATIONS YET?
JMP C           /NO:GO BACK TO DO ANOTHER
/////////////////
/
/      RETURN VALUES OF IH, IL
/
/////////////////
RET,    CLA CLL
        TAD DFIX
        DCA F
        CLA CLL
        TAD A
        JMS D
        TAD B
        JMS D
        CLA CLL
        RETRN DFIX
/////////////////
/
/      SUBROUTINE TO RETURN ARGUMENTS TO MAIN PGM.
/
/////////////////
D,      0          /ALSO STORAGE FOR ROT
        DCA ROT
F,      HLT          /THEN OL, THEN CDF MAIN
        ITAD DFIX#
        DCA G
        INC DFIX#
        ITAD DFIX#
        DCA V1      /V1 IS ADDR ARGN
        INC DFIX#
G,      HLT          /THEN NL, THEN CDF ARG
        TAD ROT
        IDCA V1
        JMP I D      /RETURN, ALSO CDF CURRENT?
/////////////////
/
/      SUBROUTINE TO ROTATE ARGUMENTS LEFT ONE BIT, AND FIX
/      NEW 12TH BIT DEPENDING ON THE LINK IN THE LAST
/      ROTATION.
/
/////////////////
ROT,    0          /ALSO STORAGE FOR D
        RAL
        DCA D
        SNL
        JMP E
        CLA CLL
        IAC
        DCA G
E,      TAD F
        TAD D
        DCA D
        TAD G
        DCA F

```

DFIX.SB, CONT'D.

```
DCA G
TAD D
JMP I ROT      /ALSO CDF CURRENT (?)
END
```

```
//// DFLLOT.SB //// 9/26/75 //////////////////////////////////////
////
//      SUBROUTINNE DFLLOT.SB      9/25/75      E.J. DARLAND      //
//
//      PURPOSE:
//      THE INTEGER-TO-REAL FLOAT ROUTINE PROVIDED IN THE OS/8 //
//      FORTRAN II LIBRARY IS INCAPABLE OF HANDLING STRAIGHT BINARY //
//      DOUBLE PRECISION (24 BIT) INTEGERS. THIS SUBROUTINE WILL //
//      CONVERT SUCH AN INTEGER INTO A FORTRAN II REAL NUMBER. //
//
//      USAGE: CALL DFIX (IH,IL,RVAR)
//
//      PARAMETER DESCRIPTION:
//      IH:      MOST SIGNIFICANT 12 BITS OF INPUT INTEGER. //
//      IL:      LEAST SIGNIFICANT 12 BITS OF INPUT INTEGER. //
//      RVAR:     RETURNED AS REAL NUMBER EQUAL TO THE //
//      DOUBLE PRECISION INTEGER IH,IL.
//
//      ***** CAUTION *****
//
//      THE INTEGER IH,IL AND THE RETURNED REAL NUMBER,
//      RVAR ARE ALL ASSUMED TO BE POSITIVE!!
//
//      MEMORY OCCUPIED:      1 PAGE
//
//      //////////////////////////////////////
//      OPDEF ITAD 1400      /INDIRECT TAD TO FOOL SABR
//      OPDEF IDCA 3400      /INDIRECT DCA TO FOOL SABR
//      //////////////////////////////////////
//
//      CONSTANT AND VARIABLE STORAGE
//
//      //////////////////////////////////////
//      IH,      0
//      IL,      0
//      OL,      0
//      NL,      9
//      EXP,     1770
//      P23,     0031
//      N1,      7777
//      N2,      7776
//      //////////////////////////////////////
//
//      ENTRY BLOCK FOR ARGUMENT PASSAGE
//
//      //////////////////////////////////////
//      ENTRY DFLLOT
//      DFLLOT,  BLOCK 2
//      //////////////////////////////////////
//
//      GET THE FIRST TWO ARGUMENTS
//
//      //////////////////////////////////////
//      CLA CLL
//      TAD DFLLOT
//      DCA A
//      TAD DFLLOT
//      DCA RETN
```

DFLOT.SB, CONT'D.

```

A,      HLT                /THEN CDF MAIN, ROTATION COUNTER
        ITAD DFLOT#
        DCA B              /B IS NOW CDF IH
        INC DFLOT#
        ITAD DFLOT#
        DCA IH            /IH IS NOW ADDR IH
        INC DFLOT#
        ITAD DFLOT#
        DCA C              /C IS NOW CDF IL
        INC DFLOT#
        ITAD DFLOT#
        DCA IL            /IL IS NOW ADDR IL
        INC DFLOT#
B,      HLT                /THEN CDF IH, 2ND LINK, 2ROT COUNTER
        ITAD IH
        DCA IH            /IH IS NOW VALUE OF IH
        HLT
C,      HLT                /THEN CDF IL, THIRD WORD
        ITAD IL
        DCA IL            /IL IS NOW VALUE OF IL
////////////////////////////////////
/
/      SET UP COUNTERS, ETC. FOR ROTATIONS
/
////////////////////////////////////
        DCA C              /NOW THIRD WORD
        DCA B              /NOW 2LINK
        TAD P25
        DCA A              /NOW ROTATION COUNTER
////////////////////////////////////
/
/      ROTATE IL INTO IH INTO C UNTIL 2LINK (THE LINK OF THE
/      SECOND WORD, IH) IS 1 FOR THE FIRST TIME. THIS MEANS
/      THAT TWO MORE ROTATIONS MUST BE DONE FOR THE 24 BITS
/      TO BE IN THE PROPER POSITION. IF LINK2 IS NOT A 1
/      AFTER 25 ROTATIONS, THEN ALL BITS MUST BE 0, AND RVAL
/      IS RETURNED AS 0.00E00
/
////////////////////////////////////
AGIN,   CLA CLL
        DCA OL
        TAD IL            /THIS THREE INSTRUCTION SEQUENCE ROTATES
        JMS ROT           /THE WORD IN QUESTION ONCE LEFT AND SETS
        DCA IL            /THE LS BIT AS APPROPRIATE AFTER THE LAST
                          /ROTATION.

        TAD IH
        JMS ROT
        DCA IH
        TAD OL            /SAVE 2LINK FOR LATER EXAMINATION
        DCA B
        TAD C
        JMS ROT
        DCA C              /3 WORD ROTATION NOW COMPLETE
        TAD B              /CHECK 2LINK
        SZA
        JMP CON1          /2LINK = 1; GO TO FINAL 2 ROTATIONS
        TAD A              /2LINK = 0; CHECK NO. OF ROTATIONS SO FAR
        TAD N1
        SNA
        JMP ZERO          /NROT = 25; SET RVAR = 0
        DCA A              /NROT < 25; SAVE A, ROT AGAIN
        JMP AGIN
CON1,   CLA CLL
        TAD N2
        DCA B              /B IS NOW 2-ROTATION COUNTER
CON2,   CLA CLL
        DCA OL
        TAD IL
        JMS ROT
        DCA IL

```

DFLOT.SB, CONT'D.

```

TAD IH
JMS ROT
DCA IH
TAD C
JMS ROT
DCA C
ISZ B          /DONE 2 ROTATIONS YET?
JMP CON2       /NO; DO AGAIN
CLA CLL        /YES; SET UP THIRD WORD
TAD A
RTL
RAL
TAD EXP
TAD C
DCA C          /C IS NOW CORRECT MS WORD OF REAL VAR.
JMP RETN
/////////////////////////////////////////////////////////////////
/
/   SET IH, IL = 0 IF NECESSARY; C IS STILL 0
/
/////////////////////////////////////////////////////////////////
ZERO,  DCA IH
       DCA IL
/////////////////////////////////////////////////////////////////
/
/   RETURN RVAR TO MAIN PROGRAM
/
/////////////////////////////////////////////////////////////////
RETN,  HLT          /THEN CDF MAIN, ADDR RVAR
       ITAD DFLOT#
       DCA E        /E IS NOW CDF RVAR
       INC DFLOT#
       ITAD DFLOT#
       DCA RETN     /RETN IS NOW ADDR RVAR
       INC DFLOT#
E,     HLT          /THEN STORAGE FOR ROT, CDF RVAR
       TAD C
       IDCA RETN
       INC RETN
       TAD IH
       IDCA RETN
       INC RETN
       TAD IL
       IDCA RETN
       AND I E      /GET SABR TO CDF CURRENT
       CLA CLL
       RETRN DFLOT
/////////////////////////////////////////////////////////////////
/
/   SUBROUTINE TO ROTATE LEFT. MAKE SURE LINK=0 FIRST!!!
/
/////////////////////////////////////////////////////////////////
ROT,   0
       RAL
       DCA E
       SNL
       JMP F
       CLA CLL
       IAC
       DCA NL
F,     TAD OL
       TAD E
       DCA E
       TAD NL
       DCA OL
       DCA NL
       TAD E
       JMP I ROT
END

```


EOUT.FT

```

CCCC EOUT.FT CCCC 12/1/75 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      SUBROUTINE EOUT.FT      12/1/75      E.J. DARLAND      CC
CC      PURPOSE:      CC
CC      TO TYPE THE CURRENT WAVELENGTH AND EQUIVALENT PHOTON      CC
CC      ENERGY ON THE TERMINAL.      CC
CC      USAGE:  CALL EOUT      CC
CC      MEMORY OCCUPIED:      ONE PAGE      CC
CC      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SUBROUTINE EOUT
COMMON WSCNT, WSPA
RLAM = WSCNT / WSPA
REN = 12398.541 / RLAM
WRITE (1,1000) RLAM, REN
RETURN
1000  FORMAT(' ',F10.4,' A      ',F12.4,' EV')
END

```

```

//// FILE.SB //// 2/11/77 //////////////////////////////////////
////
////      SUBROUTINE FILE.SB      2/11/77      E.J. DARLAND      //
////      PURPOSE:      //
////      TO INPUT FROM THE KEYBOARD DEVICE AND FILE NAMES FOR      //
////      USE IN DEVICE INDEPENDENT I/O UNDER OS/8 FORTRAN II. THE      //
////      INPUT ROUTINE INCORPORATES A SIMPLE EDITOR AND RESPONDS TO      //
////      SPECIAL CHARACTERS ^C AND ^G. SIMPLE SYNTAX CHECKING IS      //
////      PERFORMED, AS ARE SEVERAL SIMPLE TESTS TO MAKE SURE THAT      //
////      OUTPUT IS NOT ATTEMPTED TO AN EXISTING FILE OR READ ONLY      //
////      DEVICE, ETC.      //
////      USAGE:  CALL FILE (DEV, FIL, IMODE)      //
////      PARAMETER DESCRIPTION:      //
////      DEV:  RETURNED AS THE DEVICE NAME IN PACKED ASCII      //
////      (4 CHARACTERS IN FIRST TWO WORDS).      //
////      FIL:  RETURNED AS THE FILE NAME IN PACKED ASCII      //
////      (6 CHARACTERS).      //
////      IMODE:  I/O MODE AND RETURN FLAG:      //
////      ON CALL:      //
////      0:  OUTPUT IS INTENDED TO THE DEV      //
////      AND FIL ENTERED.      //
////      1:  INPUT IS INTENDED FROM THE DEV      //
////      AND FIL ENTERED.      //
////      ON RETURN:      //
////      -2:  ^C ENCOUNTERED; DEV AND FIL      //
////      ARE NOT VALID.      //
////      -1:  ^G ENCOUNTERED; DEV AND FIL      //
////      NOT VALID.      //
////      0:  SYNTAX ERROR OR INAPPROPRIATE      //
////      DEVICE. INTENDED I/O WOULD      //
////      FAIL IF IT WERE ATTEMPTED,      //
////      SO DON'T TRY IT.      //
////      1:  ALL IS OKAY, AND DEV AND FIL      //
////      ARE VALID.      //
////      MEMORY OCCUPIED:      8 PAGES IN FIELD ZERO      //
////      ALSO LOCATIONS 00125 00126      //

```

FILE.SB, CONT'D.

PROGRAM DESCRIPTION:

THIS SUBROUTINE MUST BE LOADED INTO FIELD 0 IN ORDER TO FUNCTION PROPERLY. IF IT IS NOT, IT WILL TYPE AN ERROR MESSAGE ('FILE.SB LOAD ERROR') AND EXIT TO THE OS/8 KEYBOARD MONITOR.

FILE WILL WRITE THE PROMPT '*' WHEN CALLED, AND THEN READ CHARACTERS FROM THE KEYBOARD AND STORE THEM IN ITS CHARACTER BUFFER UNTIL ONE OF THE FOLLOWING SPECIAL CHARACTERS IS TYPED, OR UNTIL THE 12TH NON-SPECIAL CHARACTER IS TYPED:

RETURN: END OF INPUT. FILE SORTS OUT THE DEVICE AND FILE NAMES, ECHOES <CR>, <LF> AND RETURNS TO THE MAIN PROGRAM.

^C: CONTROL C. FILE ECHOES '^C', <CR>, <LF>, AND RETURNS TO THE MAIN PROGRAM.

^G: CONTROL G. FILE ECHOES <BELL>, <CR>, <LF>, AND RETURNS TO THE MAIN PROGRAM.

^U: CONTROL U. RESTART. FILE ECHOES <CR>, <LF> AND THEN RESTARTS. THIS HAS THE EFFECT OF COMPLETELY IGNORING THE LINE TYPED PRIOR TO THE ^U.

RUBOUT: DELETE LAST CHARACTER. FILE ECHOES A BACKSLASH AND DELETES THE MOST RECENTLY ENTERED CHARACTER FROM ITS BUFFER UNLESS THERE IS ONLY ONE CHARACTER LEFT, IN WHICH CASE IT RESTARTS AS IN ^U.

THE INPUT EXPECTED AFTER THE * PROMPT IS:

DDDD:FFFFFF

WHERE DDDD IS A DEVICE NAME OF 1 TO 4 CHARACTERS AND FFFFFFFF IS A FILE NAME OF 0 TO 6 CHARACTERS. IF THE DEVICE NAME IS OMITTED, DEVICE 'DSK' IS ASSUMED. IF THE DEVICE IS A READ ONLY OR WRITE ONLY DEVICE, FILE NAME IS ASSUMED TO BE ZERO. NOTE THAT 0 SYMBOLS DO NOT HAVE TO BE TYPED TO FILL THE NAMES TO 4 OR 6 CHARACTERS.

SUMMARY OF ERROR MESSAGES:

FILE.SB LOAD ERROR	FILE NOT LOADED IN FILLD 0.
SYNTAX ERROR	DEVICE NAME OR FILE NAME LONGER THAN 4 OR 6 CHARACTERS RESPECTIVELY, OR MORE THAN 11 CHARACTERS ON INPUT LINE, OR NULL INPUT, ETC.
DEVICE NOT FOUND	ATTEMPT TO LOAD DEVICE HANDLER FAILED.
FILE ALREADY EXISTS	OUTPUT INTENDED TO A FILE WHICH ALREADY EXISTS.
FILE NOT FOUND	INPUT INTENDED FROM A FILE WHICH COULD NOT BE FOUND ON THE SPECIFIED DEVICE.
READ ONLY DEVICE	OUTPUT INTENDED TO A READ-ONLY DEVICE.

```
//      WRITE ONLY DEVICE          INPUT INTENDED FROM A WRITE //
//                                ONLY DEVICE.                   //
//                                                                //
// ANY OF THE ABOVE ERRORS (EXCEPTING THE FIRST) WILL CAUSE //
// IMODE TO BE RETURNED AS 0.                                     //
//                                                                //
////////////////////////////////////////////////////////////////////
ABSVM PNTD 0125
ABSVM PNTI 0126
OPDEF ITAD 1400           /INDIRECT TAD TO FOOL SABR
OPDEF IDCA 3400           /INDIRECT DCA TO FOOL SABR
OPDEF IJMP 5400           /INDIRECT JMP TO FOOL SABR
OPDEF RIF 6224            /READ INSTRUCTION FIELD
DEV,    D1                /POINTER TO:
D1,     0                 /STORAGE FOR PACKED ASCII DEV
D2,     0
D3,     0
FIL,    F1                /POINTER TO:
F1,     0                 /STORAGE FOR PACKED ASCII FIL
F2,     0
F3,     0
F4,     0401              /EXTENSION .DA ASSUMED !!
IMODE,   0
DADD,    0                /DEV ADDRESS
FADD,    0                /FIL ADDRESS
IADD,    0                /IMODE ADDRESS
GET,     0
        ITAD FILE#
        INC FILE#
        IJMP GET
//////////////////////////////////////////////////////////////////
/
/ ENTRY BLOCK. MAKE SURE IF = 0. IF NOT, GO TO ERROR
/ ROUTINE.
/
//////////////////////////////////////////////////////////////////
FILE,    ENTRY FILE
        BLOCK 2
        CLA CLL
        RIF
        SZA
        JHP ERR1
//////////////////////////////////////////////////////////////////
/
/ GET THE DATA FIELDS AND ADDRESSES OF THE ARGUMENTS FOR LATER
/ USE.
/
//////////////////////////////////////////////////////////////////
MCDF,    TAD FILE
        DCA MCDF
        HLT               /NOW CDF MAIN
        JMS GET
        DCA DCDF           /NOW CDF DEV
        JMS GET
        DCA DADD           /NOW ADDRESS DEV
        JMS GET
        DCA FCDF           /NOW CDF FIL
        JMS GET
        DCA FADD           /NOW ADDRESS FIL
        JMS GET
        DCA ICDF           /NOW CDF IMODE
        JMS GET
        DCA IADD           /NOW ADDRESS IMODE
```

FILE.SB, CONT'D.

```

////////////////////////////////////
/
/      GET IMODE FROM THE MAIN PROGRAM.
/
////////////////////////////////////
ICDF,   HLT           /CDF IMODE
        ITAD IADD
        DCA IMODE
        JMP START    /SKIP PAST RETURN SUBROUTINES AND START.
////////////////////////////////////
/
/      HERE'S WHERE DEV, FIL AND IMODE ARE RETURNED TO THE MAIN
/      PROGRAM. THIS SECTION IS LOCATED ON THE FIRST PAGE TO
/      KEEP IT CLOSE TO THE ARGUMENT STORAGE SO THAT THE IDCA
/      WILL NOT CREATE A 'DOUBLE INDIRECT' BY REFERENCING AN OFF-PAGE
/      VARIABLE.
/
////////////////////////////////////
RETURN, JMS CRLF      /ECHO CRLF
DCDF,   HLT           /CDF DEV
        TAD D1        /GET FIRST WORD OF DEV
        IDCA DADD      /AND DROP OFF
        INC DADD       /INCREMENT POINTER
        TAD D2        /AND REPEAT FOR 2ND, 3RD WORDS
        IDCA DADD
        INC DADD
        TAD D3
        IDCA DADD      /DEV NOW PASSED.
FCDF,   HLT           /CDF FIL: GO THROUGH THE SAME THING HERE.
        TAD F1
        IDCA FADD
        INC FADD
        TAD F2
        IDCA FADD
        INC FADD
        TAD F3
        IDCA FADD      /FIL NOW PASSED.
        TAD ICDF       /SET UP TO PASS IMODE BACK
        DCA ICDF
ICDFF,  HLT           /CDF IMODE
        TAD IMODE
        IDCA IADD      /IMODE PASSED.
        6201           /CDF CURRENT
        RETRN FILE     /AND RETURN TO CALLING PROGRAM
////////////////////////////////////
/
/      THE FOLLOWING SUBROUTINE IS USED TO PACK TWO WORDS FROM THE
/      LOCATIONS POINTED TO BY PNTI INTO THE WORD POINTED TO BY
/      PNT0. THE TWO INPUT WORDS ARE CLEARED AND BOTH POINTERS
/      ARE LEFT POINTING TO THE NEXT WORD IN THEIR RESPECTIVE ARRAYS.
/
////////////////////////////////////
PAK,    0
        TAD I PNTI     /GET THE FIRST WORD
        AND P77        /STRIP TO 6 BITS
        RTL;RTL;RTL    /ROTATE TO UPPER HALF OF WORD
        DCA TEMP       /AND STORE
        DCA I PNTI     /CLEAR THE INPUT BUFFER
        INC PNTI       /AND INCREMENT POINTER
        TAD I PNTI     /GET THE SECOND WORD
        AND P77        /STRIP
        TAD TEMP       /AND ADD TO FIRST WORD
        DCA I PNT0     /STORE
        DCA I PNTI     /AND CLEAR INPUT BUFFER WORD
        INC PNTI       /INCREMENT POINTERS
        INC PNT0
        JMP I PAK      /AND RETURN

```

FILE.SB, CONT'D.

```

////////////////////////////////////
/
/      THE FOLLOWING SUBROUTINES USE PAK TO LOAD THE CONTENTS OF
/      NBUFF INTO DEV AND FIL.
/
////////////////////////////////////
DPACK,  0
        CLA CLL
        TAD DEV          /POINTER TO STORAGE
        DCA PNT0         /SET UP FOR PAK
        TAD NPNT         /POINTER TO INPUT
        DCA PNTI         /SET UP FOR PAK
        JMS PAK          /PACK 6 WORDS INTO 3
        JMS PAK
        JMS PAK
        JMP I DPACK      /AND RETURN.
FPACK,  0
        CLA CLL
        TAD FIL
        DCA PNT0
        TAD NPNT
        DCA PNTI
        JMS PAK
        JMS PAK
        JMS PAK
        JMP I FPACK
////////////////////////////////////
/
/      ***** START *****
/
/      HERE'S WHERE THE ACTION REALLY STARTS, NOW THAT WE'VE JUMPED
/      PAST ALL THAT OTHER GARBAGE THAT WE WON'T USE TILL LATER.
/
/      CLEAR BUFFERS AND ICNT
/
////////////////////////////////////
START,  TAD M21          /-17 DECIMAL
        DCA ICNT        /COUNT NO. OF PLACES IN BOTH BUFFERS
        TAD CPNT        /POINTER TO ADDRESS OF CBUFF
        DCA PNT0        /POINTER TO ADDRESS OF NEXT WORD
ST1,    DCA I PNT0      /CLEAR A WORD OF BUFFER
        ISZ ICNT        /LAST WORD TO DO?
        JMP ST1         /NO: DO ANOTHER
        DCA ICNT        /YES: MAKE DOUBLE SURE ICNT = 0
        TAD P243        /ASCII #: WRITE PROMPT
        JMS TYPE
////////////////////////////////////
/
/      GET A NEW CHARACTER FROM THE KEYBOARD.  STRIP IT TO 7 BIT
/      ASCII AND STORE IN TEMP.
/
////////////////////////////////////
CHAR,   KSF             /KEYBOARD HIT YET?
        JMP CHAR        /NO: WAIT SOME MORE
        KRB             /YES: READ THE CHARACTER
        AND P177        /MASK
        DCA TEMP        /AND STORE
////////////////////////////////////
/
/      CHECK TO SEE IF IT'S ONE OF THE SPECIAL CHARACTERS.  IF SO
/      BRANCH TO THE RIGHT PART OF FILIN, IF NOT FALL THROUGH TO THE
/      NEXT SECTION.
/
////////////////////////////////////
        TAD TEMP        /GET CHARACTER
        TAD M3          /-3 OCTAL
        SNA             /IS IT A ^C (0033)?
        JMP CC          /YES
        TAD M4          /NO, -4 OCTAL

```

FILE.SB, CONT'D.

```

SNA          /IS IT A ^G (0007)?
JMP CG       /YES
TAD M3       /NO
SNA          /IS IT A LF (0012)?
JMP LF       /YES
TAD M3       /NO
SNA          /IS IT A CR (0015)?
JMP CR       /YES
TAD M10      /NO
SNA          /IS IT A ^U (0025)?
JMP CU       /YES
TAD M152     /NO
SNA          /IS IT A RUBOUT (0177)?
JMP RUB      /YES
////////////////////////////////////
/
/   IF IT'S NOT A SPECIAL CHARACTER, WE WIND UP HERE.  IF THIS IS
/   THE TWELFTH CHARACTER, GO TO THE SYNTAX ERROR SECTION.
/   SINCE THAT'S MORE THAN CAN POSSIBLE BE IN A DEVICE AND FILE
/   NAME.  IF NOT, ECHO THE CHARACTER, STRIP TO 6 BIT ASCII AND
/   STORE IN CBUF.
/
////////////////////////////////////
GOCHA, CLA CLL
TAD ICNT     /CHECK ICNT
TAD M13      /-11 DECIMAL
SNA          /IS THIS NO. 12?
JMP ERR2     /YES: ERROR
CLA CLL      /NO: ECHO CHARACTER
TAD TEMP
JMS TYPE
TAD CPNT     /CALCULATE STORAGE ADDRESS,
TAD ICNT     / = CBUF + ICNT,
DCA PNT0     /AND STORE IN POINTER
TAD TEMP     /GET CHARACTER
DCA I PNT0   /STORE.
INC ICNT     /INCREMENT CHARACTER COUNTER
JMP CHAR     /AND GO GET ANOTHER CHARACTER.
////////////////////////////////////
/
/   SPECIAL: CONTROL C.
/
/   ECHO ^C AND SET IMODE = 2.  RETURN.
/
////////////////////////////////////
CC, CLA CLL
TAD P336     /ASCII ^
JMS TYPE
TAD P303     /ASCII C
JMS TYPE
TAD M1
DCA IMODE
JMP RETURN
////////////////////////////////////
/
/   SPECIAL: CONTROL G
/
/   ECHO BELL.  SET IMODE = 3 AND RETURN.
/
////////////////////////////////////
CG, CLA CLL
TAD P7       /ASCII BELL
JMS TYPE
TAD M2       /3 DECIMAL
DCA IMODE
JMP RETURN

```

FILE.SB, CONT'D.

```

////////////////////////////////////
/
/      SPECIAL: CONTROL U
/
/      ECHO CRLF AND RESTART.
/
////////////////////////////////////
CU,      JMS CRLF
          JMP START
////////////////////////////////////
/
/      SPECIAL: RUBOUT
/
/      ECHO A BACKSLASH.  IF THIS WOULD BE DELETING THE LAST
/      REMAINING CHARACTER, SIMPLY RESTART.  OTHERWISE, DECREMENT
/      THE CHARACTER COUNTER, ICNT.  THEN RETURN TO THE CHARACTER
/      FE1CH SECTION.
/
////////////////////////////////////
RUB,     CLA CLL
          TAD P334          /ASCII BACKSLASH
          JMS TYPE
          TAD ICNT          /TEST ICNT
          TAD M2            /-2 DECIMAL
          SPA              /IF ICNT - 2 .LT. ZERO, ONLY 1 CHARACTER LEFT
          JMP CU            /YUP. ONLY ONE, SO RESTART
          CLA CLL          /MORE THAN ONE LEFT.
          TAD ICNT          /DECREMENT ICNT
          TAD M1            /-1
          DCA ICNT
          JMP CHAR          /AND GO FOR ANOTHER CHARACTER.
////////////////////////////////////
/
/      SPECIAL: LINE FEED
/
/      ECHO CRLF. THEN ECHO ALL CHARACTERS SO FAR RECORDED.
/
////////////////////////////////////
LF,      CLA CLL
          JMS CRLF
          DCA JCNT          /CLEAR "CHARACTERS TYPED" COUNTER
LF1,     TAD JCNT          /CHECK TO SEE IF ALL ARE ECHOED YET
          CIA
          TAD ICNT          /ICNT - JCNT
          SNA              /IF EQUAL, HAVE DONE ALL
          JMP CHAR          /SO GO BACK TO READING
          CLA CLL          /START CALCULATING ADDRESS OF NEXT CHARACTER
          TAD CPNT          /POINTER TO CBUF
          TAD JCNT          / = CBUF + JCNT
          DCA PNT0          /AND LEAVE IN POINTER
          TAD I PNT0        /GET THE CHARACTER
          JMS TYPE          /AND ECHO IT
          INC JCNT          /INCREMENT COUNTER
          JMP LF1          /NO: DO ANOTHER
////////////////////////////////////
/
/      VERY SPECIAL: CARRIAGE RETURN.
/
/      DONE WITH INPUT.  TILL NOW, ICNT HAS POINTED TO THE NEXT
/      CHARACTER, SINCE IT WAS INCREMENTED AS SOON AS A CHARACTER
/      WAS READ.  DECREMENT IT SO THAT IT CORRESPONDS TO THE
/      LAST CHARACTER IN THE BUFFER.
/
/      SEARCH THROUGH ALL THE CHARACTERS IN THE BUFFER UNTIL A
/      COLON IS FOUND.  IF THERE IS NO COLON, GO TO NOCOL WHERE
/      THE ASSUMED DEVICE WILL BE SYS AND ALL INPUT WILL BE

```

FILE.SB, CONT'D.

```

/      ASSUMED TO BE THE FILE NAME.  IF A COLON IS FOUND, GO TO
/      COLON, WHERE BOTH DEVICE AND FILENAMES ARE SORTED OUT.
/
////////////////////////////////////
CR,    CLA CLL
      TAD M1          /ICNT = ICNT - 1
      TAD ICNT
      DCA ICNT
      DCA JCNT        /INITIALIZE "CHARACTER SEARCHED" COUNTER
CR1,   CLA CLL        /SEE IF THERE ARE MORE CHARACTERS TO SEARCH
      TAD JCNT
      CIA
      TAD ICNT        /ICNT - JCNT; IF LESS THAN 0, NO MORE
      SPA            /POSITIVE IF MORE
      JMP NOCOL       /NONE LEFT, AND NO COLON FOUND
      CLA CLL
      TAD CPNT        /CALCULATE ADDRESS OF NEXT CHARACTER TO
      TAD JCNT        /BE LOOKED AT AS BEFORE (EG, LF).
      DCA PNT0        /STORE IN POINTER.
      TAD I PNT0      /GET A CHARACTER
      TAD M72         /-72 OCTAL; -COLON.
      SNA            /IS IT A COLON?
      JMP COLON       /YES: EXIT LOOP
      INC JCNT        /YES: INCREMENT COUNTER AND LOOK AGAIN
      JMP CR1
////////////////////////////////////
/
/      NO COLON IN INPUT.  ASSUME ALL INPUT IS FILENAME, AND
/      CHECK TO MAKE SURE IT ISN'T MORE THAN 6 CHARACTERS LONG.
/      IF IT IS, TAKE THE ERROR EXIT.  IF ALL IS OKAY, SET NBUFF TO
/      'DSK' AND PACK INTO DEV.  THEN MOVE THE FILNAME INTO
/      NBUFF AND PACK INTO FIL.
/
////////////////////////////////////
NOCOL, CLA CLL
      TAD ICNT        /TEST ICNT.  MUST BE LESS THAN 6
      TAD M6          /-6
      SNA            /IS IT OKAY?
      JMP ERR2        /NO: SYNTAX ERROR
      CLA CLL        /YES.
      TAD NPNT        /POINTER TO NAME BUFFER
      DCA PNT0
      TAD P104        /ASCII D
      DCA I PNT0
      INC PNT0
      TAD P123        /ASCII S
      DCA I PNT0
      INC PNT0
      TAD P113        /ASCII K
      DCA I PNT0      /OBUFF NOW = DSK@@@
      JMS DPACK       /PACK INTO DEV AND CLEAR NBUFF
      TAD CPNT
      DCA PNTI        /POINTER TO WORD IN CBUFF
      TAD NPNT
      DCA PNT0        /POINTER TO WORD IN OBUFF
      DCA JCNT        /CLEAR "CHARACTER TRANSFERRED" COUNTER
NO1,   TAD JCNT       /TRANSFER A CHARACTER?
      CIA
      TAD ICNT
      SPA            /WELL?
      JMP DUN1        /NO, TRANSFER DONE.
      CLA CLL        /YES:
      TAD I PNTI      /FETCH FROM CBUFF
      DCA I PNT0      /AND LEAVE IN OBUFF
      INC PNTI        /INCREMENT POINTERS AND COUNTER
      INC PNT0
      INC JCNT
      JMP NO1        /AND REPEAT

```


FILE.SB, CONT'D.

```

////////////////////////////////////
/
/      DONE WITH TRANSFER TO NBUFF, SO PACK INTO FIL.  THEN CHECK TO
/      MAKE SURE FIL IS NOT ALL 0'S, WHICH WOULD MEAN A NULL INPUT
/      LINE (A SYNTAX ERROR).
/
////////////////////////////////////
DUN1,   JMS FPACK      /PACK INTO FIL AND CLEAR NBUFF
        TAD F1         /GET FIRST WORD
        SZA            /IS IT ZERO?
        JMP NO2        /NO: NOT A NULL INPUT LINE
        TAD F2         /YES: HAVE TO CHECK WORD 2
        SZA            /IS IT ZERO?
        JMP NO2        /NO: NOT A NULL INPUT LINE
        TAD F3         /YES: HAVE TO CHECK LAST WORD
        SNA            /IS IT ZERO?
        JMP ERR2       /YES: NULL INPUT LINE IS AN ERROR
NO2,    JMP CHEK       /NO: GO TO CHECK FOR DEVICE HANDLER
////////////////////////////////////
/
/      THERE IS A COLON.  FIRST, SET ICOL = NUMBER OF THE CHARACTER
/      THAT IS THE COLON.  THEN MAKE SURE THAT THE DEVICE NAME
/      (BEFORE-COLON CHARACTERS) ARE FEWER THAN FIVE.  IF NOT, GO
/      TO THE SYNTAX ERROR SECTION.  LIKewise, IF THE FILE NAME
/      CHARACTERS EXCEED SIX, GO TO THE SYNTAX SECTION.
/      IF ALL IS OKAY, TRANSFER THE APPROPRIATE CHARACTERS TO
/      NBUFF AND PACK THEM INTO DEV ANF FIL.
/
////////////////////////////////////
COLON,  CLA CLL
        TAD JCNT
        DCA ICOL      /ICOL SET NOW.
        TAD ICOL      /CHECK TO MAKE SURE THERE ARE MORE THAN
        TAD M1        /0 AND LESS THAN 5 CHARACTERS IN DEVICE
        SPA           /NAME.
        JMP ERR2
        TAD M4
        SMA           /OKAY?
        JMP ERR2      /NO: TAKE THE ERROR ROUTE
        CLA CLL       /YES: CHECK NO. OF CHARACTERS IN FILENAME
        TAD ICOL
        CIA
        TAD M7        /-7 DECIMAL
        TAD ICNT      /ICNT - ICOL MUST BE .LT. 7
        SMA           /IS IT?
        JMP ERR2      /NO: ERROR
        CLA CLL       /YES: OKAY TO TRANSFER
        TAD NPNT      /NBUFF ADDRESS POINTER
        DCA PNT0
        TAD CPNT      /CBUFF ADDRESS POINTER
        DCA PNTI
        DCA JCNT      /CLEAR "CHARACTER TRANSFERRED" COUNTER
CO1,    TAD JCNT      /DO A TRANSFER?
        CIA
        TAD ICOL      /CHECK ICOL - JCNT - 1
        TAD M1
        SPA
        JMP DUN2      /NO: DONE WITH THIS TRANSFER
        CLA CLL
        TAD I PNTI    /YES: TRANSFER A CHARACTER
        DCA I PNT0
        INC PNTI
        INC PNT0
        INC JCNT
        JMP CO1       /DO THE LOOP AGAIN.
DUN2,   JMS DPACK     /TRANSFER DONE. PACK DEV, CLEAR NBUFF
        TAD ICOL      /SET UP TRANSFER FROM ICOL + 1 TO ICNT
        IAC
        DCA JCNT

```

FILE.SB, CONT'D.

```

      TAD JCNT
      TAD CPNT
      DCA PNTI          /JCNT SET TO START RIGHT, AS IS PNTI
      TAD NPNT
      DCA PNT0          /PNT0 NOW SET
CO2,  TAD JCNT          /DO ANOTHER TRANSFER?
      CIA
      TAD ICNT
      SPA              /?
      JMP DUN3          /NO: DONE
      CLA CLL
      TAD I PNTI        /YES: TRANSFER
      DCA I PNT0
      INC PNTI
      INC PNT0
      INC JCNT
      JMP CO2           /GO LOOP AGAIN
DUN3, JMS FPACK         /PACK INTO FIL AND CLEAR NBUF
      JMP CHEK

```

```

////////////////////////////////////
/
/   LOCK USR IN CORE
/

```

```

////////////////////////////////////
P200, 0200
P7700, 7700
CHEK,  CLA CLL
      CPAGE 5
      6212          /CIF 1
      JMS I P7700    /JUMP TO USR
      10            /FUNCTION CODE FOR USRIN

```

```

////////////////////////////////////
/
/   LOAD DEVICE HANDLER TO MAKE SURE IT EXISTS.
/

```

```

      CLA CLL
      TAD D1          /SET UP DEVICE NAME IN FE1,FE2
      DCA FE1
      TAD D2
      DCA FE2
      TAD HPNT        /PUT POINTER TO HANDLER STORAGE IN FE3
      IAC            /SET BIT 11 FOR 2-PAGE HANDLER
      DCA FE3
      CPAGE 10
      6212          /CIF 1
      JMS I P200      /USR CALL
      1              /FUNCTION CODE FOR FETCH CALL
FE1,  0
FE2,  0
FE3,  0
      JMP ERR3        /ERROR RETURN: DEVICE NOT FOUND
      CLA CLL        /NORMAL RETURN: CONTINUE

```

```

////////////////////////////////////
/
/   GET DEVICE CONTROL WORD FROM TABLE IN FIELD 1 AND DETERMINE
/   IF THE DEVICE IS READ ONLY, WRITE ONLY, OR FILE STRUCTURED.
/

```

```

////////////////////////////////////
      TAD P7757
      TAD FE2          /LOCATION = 1 7757 + DEV. NO.
      DCA W1
      DUMMY WORD
      TAD I WORD
      DCA TEMP        /STORE CONTROL WORD
      TAD TEMP
      CLL RAL          /CHECK BIT 0
      SZL             /IS IT FILE STRUCTURED?
      JMP FILST        /YES

```

FILE.SB, CONT'D.

```

      RAL          /NO
      SZL          /IS IT READ ONLY?
      JMP RONLY   /YES
      RAL          /NO
      SZL          /IS IT WRITE ONLY?
      JMP WONLY   /YES
      JMS FPACK    /NO: MUST BE READ/WRITE NON-FILE, SO MAKE SURE
      CLA CLL      /FIL IS 0, THEN RETURN.
      IAC          /SET IMODE = 1
      DCA IMODE
      JMP OUT
      CPAGE 2
WORD, 6211
W1, 0
////////////////////////////////////
/
/      IT IS A FILE STRUCTURED DEVICE.  DO A LOOKUP ON THE FILE TO
/      SEE IF IT'S THERE.
/
////////////////////////////////////
FILST, CLA CLL
      TAD FIL      /POINTER TO FILE NAME
      DCA L1
      TAD FE2      /GET DEVICE NO. IN ACCUMULATOR
      CPAGE 10
      6212        /CIF 1
      JMS I P200   /USR CALL
      2           /FUNCTION CODE FOR LOOKUP
L1, 0
L2, 0
      JMP RERR     /ERROR RETURN: NO FILE FOUND
      CLA CLL      /NORMAL:FILE FOUND
      TAD IMODE
      SNA          /INTENDING INPUT?
      JMP ERR4     /NO:CAN'T OUTPUT TO EXISTING FILE
      CLA CLL      /YES:IMODE OKAY
      JMP OUT
RERR, CLA CLL
      TAD IMODE
      SZA          /INTENDING OUTPUT?
      JMP ERR5     /NO:CAN'T INPUT FROM NONEXISTING FILE
      IAC          /YES:SET IMODE = 1, ALL IS OKAY
      DCA IMODE
////////////////////////////////////
/
/      DISMISS USR FROM CORE, THEN RETURN
/
////////////////////////////////////
OUT,  CPAGE 5
      6212        /CIF 1
      JMS I P200   /FUNCTION CODE FOR USROUT
      11
      JMP RETURN
////////////////////////////////////
/
/      READ ONLY DEVICE.  MAKE SURE INPUT IS INTENDED, IF SO SET
/      FIL = 0
/
////////////////////////////////////
RONLY, CLA CLL
      TAD IMODE
      SNA          /INTEND INPUT?
      JMP ERR6     /NO:CAN'T WRITE ON READ ONLY DEVICE
      JMS FPACK    /YES: SET FIL TO 0
      JMP OUT      /MODE ALREADY = 1

```

FILE.SB, CONT'D.

```

////////////////////////////////////
/
/      WRITE ONLY DEVICE.  MAKE SURE OUTPUT IS INTENDED, IF SO SET
/      FIL = 0
/
////////////////////////////////////
WONLY,  CLA CLL
        TAD IMODE
        SZA          /INTEND OUTPUT?
        JMP ERR7      /NO: CAN'T READ FROM WRITE ONLY DEVICE
        IAC          /YES: SET IMODE = 1
        DCA IMODE
        JMS FPACK      /SET FIL = 0
        JMP OUT
////////////////////////////////////
/
/      ROUTINES TO TYPE ERROR MESSAGES.
/
////////////////////////////////////
ERR1,   CLA CLL
        TAD ME1
        JMS MESSAGE
        CALL 0, EXIT
ME1,    MES1
MES1,   TEXT /%# FILE.SB LOAD ERROR! %#/
ERR2,   CLA CLL
        TAD ME2
        JMS MESSAGE
        CLA CLL
        DCA IMODE
        JMP RETURN
ME2,    MES2
MES2,   TEXT / SYNTAX ERROR! /
ERR3,   CLA CLL
        TAD ME3
        JMS MESSAGE
        JMP SET
ME3,    MES3
MES3,   TEXT / DEVICE NOT FOUND! /
ERR4,   CLA CLL
        TAD ME4
        JMS MESSAGE
        JMP SET
ME4,    MES4
MES4,   TEXT / FILE ALREADY EXISTS! /
ERR5,   CLA CLL
        TAD ME5
        JMS MESSAGE
SET,    CLA CLL
        DCA IMODE
        JMP OUT
ME5,    MES5
MES5,   TEXT / FILE NOT FOUND! /
ERR6,   CLA CLL
        TAD ME6
        JMS MESSAGE
        JMP SET
ME6,    MES6
MES6,   TEXT / READ ONLY DEVICE! /
ERR7,   CLA CLL
        TAD ME7
        JMS MESSAGE
        JMP SET
ME7,    MES7
MES7,   TEXT / WRITE ONLY DEVICE! /

```

FILE.SB, CONT'D.

```

////////////////////////////////////
/
/      SUBROUTINES FOR HANDLING OUTPUT TO THE TELETYPE.
/
////////////////////////////////////
/
/      CPAGE 7
TYPE, 0 /TYPES ONE CHARACTER ON TERMINAL
T1,   TSF /TERMINAL READY?
      JMP T1 /NO:CHECK AGAIN
      TLS /YES, TRANSMIT CHARACTER
      CLA CLL
      JMP I TYPE
      CPAGE 10
CRLF, 0 /TYPES CARRIAGE RETURN, LINE FEED
      CLA CLL
      TAD P215 /ASCII CR
      JMS TYPE
      TAD P212 /ASCII LF
      JMS TYPE
      JMP I CRLF
/
/      MESSAGE. PRINTS A MESSAGE ON THE TTY WHICH IS STORED IN A TEXT
/      STATEMENT. ENTER WITH THE ADDRESS OF THE MESSAGE IN THE
/      ACCUMULATOR. TREATS % AS A CARRIAGE RETURN AND # AS A LINE
/      FEED.
/
////////////////////////////////////
/
/      CPAGE 43
MESSAGE, 0 /STORE ADDRESS OF MESSAGE
      DCA PNT0
      TLS /INITIALIZE TTY
MS1,   TAD I PNT0 /GET A WORD
      RTR;RTR;RTR /ROTATE FIRST CHARACTER INTO BITS 6-11
      JMS TYPCH /TYPE CHARACTER
      TAD I PNT0 /GET SECOND CHARACTER
      JMS TYPCH /AND TYPE
      ISZ PNT0 /ADVANCE POINTER
      JMP MS1 /TO ANOTHER
TYPCH, 0 /ROUTINE TO TYPE A CHARACTER
      AND P77 /MASK OUT UPPER 6 BITS
      SNA /IF 0, TERMINATE MESSAGE
      JMP I MESSAGE /AND RETURN TO MAIN PROGRAM.
      TAD M40 /IF NOT ZERO, FIGURE OUT FIRST TWO BITS
      SPA /WILL BE A 200 CODE
      TAD P100 /WILL BE A 300 CODE
      TAD M5 /CHECK FOR CR
      SNA
      TAD M30 /CHANGE % TO CR
      TAD P2 /CHECK FOR LF
      SNA
      TAD M31 /CHANGE # TO LF
      TAD P243 /FORM THE STANDARD ASCII CODE
      TLS /PRINT THE CHARACTER
T2,   TSF /WAIT FOR TTY READY
      JMP T2
      CLA CLL /AND RETURN
      JMP I TYPCH
/
/      STORAGE
/
////////////////////////////////////
/
ICNT, 0
JCNT, 0
ICOL, 0
CPNT, CBUFF
NPNT, NBUFF
BPNT, HAND

```

FILE.SB, CONT'D.

```

TEMP,    0
M1,      7777
M2,      7776
M3,      7775
M4,      7774
M5,      7773
M6,      7772
M7,      7771
M10,     7770
M13,     7765
M21,     7757
M30,     7750
M31,     7747
M40,     7740
M72,     7706
M152,    7626
P2,      0002
P7,      0007
P77,     0077
P100,    0100
P104,    0104
P113,    0113
P123,    0123
P177,    0177
P212,    0212
P215,    0215
P243,    0243
P303,    0303
P334,    0334
P336,    0336
P7757,   7757
CBUFF,   BLOCK 13
NBUFF,   BLOCK 6
LAP
PAGE
HAND,    BLOCK 200
        BLOCK 200
        EAP
        END

```

```

//// GOSTP.SB //// 4/5/77 //////////////////////////////////////
////
// SUBROUTINE GOSTP.SB V2 4/5/77 E.J. DARLAND //
//
// PURPOSE: //
// THIS SUBROUTINE DRIVES THE MONOCHROMATOR WAVELENGTH //
// STEPPING MOTOR AT A CONSTANT SPEED, USING THE REAL TIME //
// CLOCK TO CONTROL THE NUMBER OF STEPS PER SECOND. //
//
// USAGE: CALL GOSTP (NDIR,NSTPH,NSTPL,NWATE) //
//
// PARAMETER DESCRIPTION: //
// NDIR: CONTROLS DIRECTION OF MOTOR: //
// 0: INCREASING WAVELENGTH //
// OTHER: DECREASING WAVELENGTH //
// NSTPH: MOST SIGNIFICANT 12 BITS OF NUMBER OF STEPS //
// TO TAKE. //
// NSTPL: LEAST SIGNIFICANT 12 BITS OF NO. OF STEPS. //
// NWATE: NUMBER OF 0.01 MILLISECOND WAIT PERIODS //
// BETWEEN STEPS. SHOULD BE 67 OR GREATER; IF //
// NOT IT IS SET TO 67 BY GOSTP. //
//

```

```
// MEMORY OCCUPIED: 1 PAGE
//
/////////////////////////////////////////////////////////////////
OPDEF ITAD 1400 /INDIRECT TAD TO FOOL SABR
OPDEF LPSET 6121 /LOAD CLOCK PRESET REGISTER
OPDEF STCLK 6124 /START REAL TIME CLOCK
OPDEF LCTRL 6131 /LOAD CLOCK CONTROL REGISTER
SKPDF POSKP 6132 /SKIP ON PCM CLOCK OVERFLOW FLAG
SKPDF PESKP 6133 / " " " " " ERROR FLAG
OPDEF PFLCL 6134 /CLEAR PROGRAMMABLE CLOCK FLAGS
/////////////////////////////////////////////////////////////////
/
CONSTANT AND VARIABLE STORAGE
/
/////////////////////////////////////////////////////////////////
NDIR, 0
NSTPH, 0
NSTPL, 0
NWATE, 0
C, 0
FSTEP, 6401
RSTEP, 6402
SET, 1400
M103, 7676 /-67 DECIMAL
P103, 9193 /+67 DECIMAL
M1, 7777
/////////////////////////////////////////////////////////////////
/
ENTRY BLOCK FOR ARGUMENT PASSAGE
/
/////////////////////////////////////////////////////////////////
ENTRY GOSTP
GOSTP, BLOCK 2
/////////////////////////////////////////////////////////////////
/
GET THE ARGUMENTS
/
/////////////////////////////////////////////////////////////////
CLA CLL
TAD GOSTP
DCA A
JMS FETCH
DCA NDIR
JMS FETCH
DCA NSTPH
JMS FETCH
DCA NSTPL
JMS FETCH
DCA NWATE
/////////////////////////////////////////////////////////////////
/
INSERT CORRECT OPCODE DEPENDING ON NDIR
/
/////////////////////////////////////////////////////////////////
TAD NDIR
SZA
JMP UP
TAD FSTEP
JMP REDY
UP, CLA CLL
TAD RSTEP
REDY, DCA STEP
/////////////////////////////////////////////////////////////////
/
SET PROGRAMMABLE CLOCK TO 10 USEC TIME BASE, VARIABLE
TIME MODE.
//
```

GOSTP.SB, CONT'D.

```

      TAD SET
      LCTRL
////////////////////////////////////
/
/      MAKE SURE NWATE IS NOT LESS THAN 67
/
////////////////////////////////////
      CLA CLL
      TAD NWATE
      SPA                      /THIS TO PREVENT COMPUTER FROM CONSIDERING
      JMP D                      /NUMBERS BETWEEN 4000 AND 7777 AS NEGATIVE
      TAD M103
      SNA
      JMP D
      CLA CLL
      TAD P103
      DCA NWATE
D,    CLA CLL
////////////////////////////////////
/
/      SET PRESET REGISTER FOR DESIRED NO. OF WAIT PERIODS, START
/      THE CLOCK.
/
////////////////////////////////////
      TAD NWATE
      CIA
      LPSET
      STCLK
////////////////////////////////////
/
/      ENTER THE LOOP THAT TAKES THE REQUIRED NO. OF STEPS
/
////////////////////////////////////
CHECK, CLA CLL                /SEE IF NSTP = 0
      TAD NSTPH
      SZA
      JMP E
      CLA CLL
      TAD NSTPL
      SZA
      JMP E
      RETRN GOSTP
E,    POSKP
      JMP E
      PFLCL
STEP, HLT
      CLA CLL
      TAD NSTPL
      TAD M1
      DCA NSTPL
      RAL
      TAD NSTPH
      TAD M1
      DCA NSTPH
      JMP CHECK
////////////////////////////////////
/
/      SUBROUTINE TO FETCH ARGUMENTS
/
////////////////////////////////////
FETCH, 0
A,    HLT
      ITAD GOSTP#
      DCA B
      INC GOSTP#
      ITAD GOSTP#
      DCA C
      INC GOSTP#
B,    HLT

```



```

////////////////// KCHEK.SB //////////////////// 12/17/73 ////////////////////
////////////////// SUBROUTINE KCHEK.SB 12/17/75 E.J. DARLAND ////////////////////
////////////////// PURPOSE: ////////////////////
////////////////// THIS SUBROUTINE IS CALLED TO DETERMINE WHETHER A ^G
////////////////// (CONTROL G) HAS BEEN TYPED AT THE KEYBOARD. IF NO
////////////////// CHARACTER HAS BEEN TYPED, OR IF ANY OTHER CHARACTER THAN
////////////////// A ^G HAS BEEN TYPED, NO ACTION IS TAKEN BEYOND CLEARING
////////////////// THE KEYBOARD FLAG. IF A ^G HAS BEEN TYPED, IT IS ECHOED
////////////////// BEFORE RETURNING TO THE MAIN PROGRAM.
//////////////////
////////////////// USAGE: CALL KCHEK (IDUM)
//////////////////
////////////////// PARAMETER DESCRIPTION:
////////////////// IDUM: RETURNED TO THE MAIN PROGRAM AS:
////////////////// 0: NO ^G HAS BEEN TYPED.
////////////////// 1: A ^G HAS BEEN TYPED.
//////////////////
////////////////// MEMORY OCCUPIED: 1 PAGE
//////////////////
//////////////////
OPDEF ITAD 1400 /INDIRECT TAD TO FOOL SABR
OPDEF IDCA 3400 /INDIRECT DCA TO FOOL SABR
ADDR, 0 /STORAGE
STOR, 0 /STORAGE
M0177, 0177 /7-BIT MASK
C7771, 7771 /-^G
C0007, 0007 /^G
ENTRY KCHEK /ENTRY BLOCK FOR ARGUMENT PASSAGE
KCHEK, BLOCK 2
CLA CLL
KSF /KEYBOARD HIT?
JMP ZERO /NO: RETURN 0
KRB /YES: GET THE CHARACTER
AND M0177 /MASK IT TO 7-BIT ASCII
TAD C7771 /-^G
SZA /IS IT A CONTROL G?
JMP ZERO /NO: RETURN 0
CLA CLL /YES: ECHO BELL
TLS /INITIALIZE TTY
TAD C0007
A, TSF /TTY READY YET?
JMP A /NO:CHECK AGAIN
TLS /YES: TYPE A BELL
CLA CLL /AND RETURN A 1
IAC
DCA STOR
JMP RETN
ZERO, CLA CLL /A ZERO IS TO BE RETURNED
DCA STOR
RETN, CLA CLL
TAD KCHEK
DCA CDFM /CDFM IS NOW CDF MAIN
CDFM, HLT

```

KCHEK.SB, CONT'D.

```

ITAD KCHEK#
DCA CDFA          /CDFA IS NOW CDF ARGUMENT
INC KCHEK#
ITAD KCHEK#
DCA ADDR          /ADDR IS NOW THE ADDRESS OF ARG
INC KCHEK#
HLT
TAD STOR
IDCA ADDR
AND I CDFA        /GET SABR TO CDF CURRENT
CLA CLL
RETRN KCHEK       /AND RETURN TO MAIN PROGRAM
END

```

[illegible]

LETIN.SB, CONT'D.

```

                OPDEF ITAD 1400 /INDIRECT TAD TO FOOL SABR
                OPDEF IDCA 3400 /INDIRECT DCA TO FOOL SABR
ICNT,          0
IC,            0003           /7 BIT ^C
IG,            0007           /7 BIT ^G
IU,            0025           /7 BIT ^U
ICR,           0015           /7 BIT CR
ILF,           0012           /7 BIT LF
IVAL,          0
ITEM,          0

```

////////////////////////////////////

```

/
/      ENTRY BLOCK FOR ARGUMENT PASSAGE
/

```

////////////////////////////////////

```

                ENTRY LETIN
LETIN, BLOCK 2

```

////////////////////////////////////

```

/      SET IVAL, CHARACTER COUNTER TO ZERO
/

```

////////////////////////////////////

```

ST,            CLA CLL
                DCA ICNT           /CHARACTER COUNTER
                DCA IVAL
                KCC                /INITIALIZE KEYBOARD

```

////////////////////////////////////

```

/      GET A CHARACTER AND STORE TEMPORARILY AS SEVEN BIT ASCII.
/

```

////////////////////////////////////

```

A,            KSF
                JMP A
                KRB
                AND (0177
                DCA ITEM

```

////////////////////////////////////

```

/      CHECK TO SEE IF IT'S ONE OF THE SPECIAL CHARACTERS.
/

```

////////////////////////////////////

```

                TAD ITEM           /CHECK FOR ^C
                CIA
                TAD IC
                SNA
                JMP SC             /IT IS A ^C
                CLA CLL           /IT ISN'T, CHECK FOR ^G
                TAD ITEM
                CIA
                TAD IG
                SNA
                JMP SC             /IT IS A ^G
                CLA CLL           /IT ISN'T, CHECK FOR ^U
                TAD ITEM
                CIA
                TAD IU
                SNA
                JMP SU             /IT IS A ^U
                CLA CLL           /IT ISN'T, CHECK FOR CR
                TAD ITEM
                CIA
                TAD ICR
                SNA
                JMP CR             /IT IS A CR

```

LETIN.SB, CONT'D.

```

////////////////////////////////////
/
/      MADE IT THROUGH THE CHAIN, SO IT'S NOT A SPECIAL.  STORE IN THE
/      APPROPRIATE HALF OF IVAL IF IT'S THE FIRST OR SECOND CHARACTER.
/      THEN INCREMENT COUNTER AND ECHO.  OTHERWISE JUST INCREMENT THE
/      COUNTER AND ECHO THE CHARACTER.
/
////////////////////////////////////
          CLA CLL
          TAD ICNT
          SNA
          JMP C1          /IT IS THE FIRST CHARACTER
          TAD (7777       /-1
          SNA
          JMP C2          /IT IS THE SECOND CHARACTER
B,        CLA CLL        /HERE IS WHERE THE COUNTER IS INCREMENTED AND
          INC ICNT       /THE CHARACTER IS ECHOED.
          TAD ITEM
          JMS TYPE
          JMP A          /NOW GO GET ANOTHER CHARACTER.
C1,       TAD ITEM       /FIRST CHARACTER:
          AND (0077      /MASK TO SIX BITS,
          RTL;RTL;RTL    /AND ROTATE TO UPPER HALF OF WORD.
          DCA IVAL       /STORE IN IVAL
          JMP B          /AND GO TO ECHO THE CHARACTER.
C2,       TAD ITEM       /SECOND CHARACTER:
          AND (0077      /MASK TO SIX BITS
          TAD IVAL       /ADD TO IVAL (UPPER SIX BITS)
          DCA IVAL       /AND STORE IN IVAL AGAIN.
          JMP B          /IVAL NOW COMPLETE; GO TO ECHO CHARACTER.
////////////////////////////////////
/
/      HERE'S WHERE THE SPECIAL CHARACTERS ARE HANDLED.
/
////////////////////////////////////
SC,       CLA CLL        /^C; SET IVAL = ^C
          TAD IC
          DCA IVAL
          TAD (0136      /^; ECHO ^C
          JMS TYPE
          TAD (0103      /C
          JMS TYPE
          JMS CRLF       /ECHO CR, LF.
          JMP RET        /AND RETURN ARGUMENT.
////////////////////////////////////
SC,       CLA CLL        /^G; SET IVAL = ^G
          TAD IC
          DCA IVAL
          TAD (0007      /ECHO BELL
          JMS TYPE
          JMS CRLF       /ECHO CR, LF
          JMP RET        /AND RETURN ARGUMENTS.
////////////////////////////////////
SU,       CLA CLL        /^U; ECHO CR, LF, THEN RESTART
          JMS CRLF
          JMP ST
////////////////////////////////////
CR,       CLA CLL        /CR; ECHO CRLF
          JMS CRLF       /THEN RETURN ARGUMENT
////////////////////////////////////
/
/      THIS SECTION RETURNS THE ARGUMENT TO THE MAIN PROGRAM.
/
////////////////////////////////////
RET,      CLA CLL
          TAD LETIN      /HOLDS CDF MAIN
          DCA CHNG
CHNG,     HLT
          ITAD LETIN*    /GET CDF ARGUMENT

```

LETIN.SB, CONT'D.

```

DCA ADF          /AND STORE IN PROGRAM
INC LETIN#
ITAD LETIN#      /GET ARGUMENT ADDRESS
DCA ICNT         /STORE IN ICNT
INC LETIN#       /SET UP FOR RETURN ADDRESS
ADF, HLT         /BY NOW A CDF ARGUMENT INSTRUCTION
TAD IVAL         /GET THE ARGUMENT
IDCA ICNT        /AND STORE IN MAIN PROGRAM
CLA CLL
AND I ADF        /GET SAER TO CDF CURRENT
RETRN LETIN      /AND RETURN TO MAIN PROGRAM
////////////////////
/
/      ECHO CR, LF ROUTINE
/
////////////////////
CRLF, 0
      CLA CLL
      TAD ICR
      JMS TYPE
      TAD ILF
      JMS TYPE
      JMP I CRLF
////////////////////
/
/      CHARACTER TYPE TO TERMINAL SUBROUTINE.
/
////////////////////
TYPE, 0
Z,     TSF
      JMP Z
      TLS
      CLA CLL
      JMP I TYPE
      END

```

```

CCCC LIMIT.FT CCCC 5/2/76 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      SUBROUTINE LIMIT.FT      5/2/76      E.J. DARLAND      CC
CC
CC      PURPOSE:
CC      PROVIDES A BOUND REAL-TO-INTEGER CONVERSION.  IF THE CC
CC      REAL NUMBER INPUT IS WITHIN THE INPUT INTEGER LIMITS, IT IS CC
CC      MERELY FIXED AND RETURNED.  IF IT IS OUTSIDE THE LIMITS, IT CC
CC      IS SET EQUAL TO THE NEAREST LIMIT.
CC
CC      USAGE:  INEW = LIMIT ( ILOW, ROLD, IHI)
CC
CC      PARAMETER DESCRIPTION:
CC      ILOW:  LOWER LIMIT
CC      ROLD:  REAL NUMBER TO BE CONVERTED
CC      IHI:  UPPER LIMIT
CC
CC      MEMORY OCCUPIED:      2 PAGES
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      FUNCTION LIMIT ( ILOW, ROLD, IHI)
      IF (FLOAT ( IHI) - ROLD) 10,40,20
10      LIMIT = IHI
      GO TO 45
20      IF (ROLD - FLOAT ( ILOW)) 30,40,40
30      LIMIT = ILOW
      GO TO 45

```

```

CCCC MOVE.FT CCCC 4/5/75 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC SUBROUTINE MOVE.FT V2 4/5/75 E.J. DARLAND CC
CC PURPOSE: CC
CC TO MOVE THE MONOCHROMATOR TO THE WAVELENGTH SPECIFIED BY THE CALLING PROGRAM. THE DETAILS OF HOW THE MOVE IS CC
CC MADE DEPEND ON WHETHER MOVE IS EVER ALLOWED TO ACCELERATE THE STEPPING MOTOR: CC
CC NO ACCELERATION: CC
CC ALL MOVES ARE MADE AT THE MAXIMUM ALLOWABLE STEPPING RATE. IF THE MOVE IS TOWARD LOWER WAVE- CC
CC LENGTH, AN INTENTIONAL OVERSHOOT OF 600 STEPS IS MADE, THEN THE MOTOR IS STEPPED BACK UP 600 STEPS CC
CC TO ELIMINATE GEAR BACKLASH. CC
CC ACCELERATION: CC
CC MOVES ARE MADE SIMILARLY, EXCEPT THAT IF THE MOVE IS MORE THAN 510 STEPS, THE STEPPING MOTOR CC
CC WILL BE ACCELERATED TO THE MAXIMUM RATE (AND DECELERATED FROM IT). IF THE MOVE IS LESS THAN CC
CC 510 STEPS IT WILL BE MADE AT A CONSTANT RATE EQUAL TO ONE FIFTH THE MAXIMUM ALLOWED RATE. CC
CC USAGE: CALL MOVE (WDES) CC
CC PARAMETER DESCRIPTION: CC
CC WDES: THE STEP COUNT OF THE DESIRED FINAL WAVELENGTH. CC
CC ***** CAUTION ***** CC
CC THE CALLING PROGRAM MUST MAKE SURE THAT WDES IS A REAL, POSITIVE NUMBER BETWEEN 0. AND CC
CC 16,777,215 (DECIMAL), INCLUSIVE, WHICH CORRESPONDS TO THE CORRECT INTEGER STEP COUNT. CC
CC INTERNAL VARIABLES: CC
CC ISL = AN ARRAY CONTAINING THE NUMBER OF STEPS DIFFERENT STAGES OF STEPPING MOTOR ACCEL- CC
CC ERATION. CC
CC IWATE = AN ARRAY CONTAINING THE NUMBER OF 0.1 MILLISECOND WAIT PERIODS PER STEP FOR EACH CC
CC STAGE OF ACCELERATION. CC
CC IH = MOST SIGNIFICANT 12 BITS OF DOUBLE PRECISION INTEGER CONTAINING NUMBER OF CC
CC STEPS TO TAKE. CC
CC IL = LEAST SIGNIFICANT 12 BITS OF THE INTEGER IH, IL. CC
CC RCO = NUMBER OF STEPS TO TAKE. CC
CC SUBROUTINES CALLED: DFIX, DFLOT, COSTP CC
CC MEMORY OCCUPIED: 5 PAGES CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SUBROUTINE MOVE (WDES)
COMMON WSCNT,WSPA,UPLIM,MWATE,IACCL
DIMENSION IWATE(8),ISL(8)
OPDEF LCTRL 6131 /LOAD CLOCK CONTROL REGISTER

```

MOVE.FT, CONT'D.

```

S      OPDEF STCLK 6124          /START CLOCK
S      SKPDF DESKP 6136         /SKP ON DECADE ERROR FLAG
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DECIDE WHETHER TO USE THE ACCELERATION SECTION.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      IF (IACCL) 100,100,1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      ACCELERATION PERMISSABLE.
C
C      SET UP ISL, IWATE FOR ACCELERATION. THEIR VALUES DEPEND ON
C      MWATE. THEN CHECK THE DIRECTION OF THE MOVE, AND RETURN IF
C      ALREADY THERE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1      DO 5 I=1,8
          J = 9-I
          ISL(I) = 2*(I-1)
          IWATE(J) = ISL(I) + MWATE
5      CONTINUE
          RGO = WDES - WSCNT
          IF (RGO) 20,10,35
10     RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      MOVE IS NEGATIVE SO WANT TO GO 600 TOO FAR. SET RGO FOR
CC      NEEDED NUMBER OF STEPS + 600 - NUMBER NEEDED FOR ACCELERATION.
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
20     RGO = 90. - RGO
          CALL DFIX (RGO, IH, IL)
          CALL DFLOT (IH, IL, RGO)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      ACCELERATE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      DO 25 I = 1,8
          CALL GOSTP (1,0, ISL(I), IWATE(I))
25     CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DO MIDDLE RGO STEPS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      CALL GOSTP (1, IH, IL, MWATE)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DECELERATE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      DO 30 I = 1,8
          J = 9 - I
          CALL GOSTP (1,0, ISL(J), IWATE(J))
30     CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WAIT 0.2 SECONDS FOR TURNAROUND
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
S      CLA CLL
S      TAD A
S      LCTRL
S      STCLK
SB,    DESKP

```

MOVE.FT, CONT'D.

```

S      JMP B
S      SKP
SA,    5005
S      CLA CLL
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      NOW GO FORWARD 600 STEPS, SLOWLY.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      RT = 5. * FLOAT (MWATE)
      CALL DFIX (RT, IH, IL)
      CALL GOSTP (0,0,600,IL)
      WSCNT = WSCNT - RGO + 90.
      GO TO 10
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      MOVE IS POSITIVE.  SEE IF IT'S BIG ENOUGH TO WARRANT
CC      ACCELERATING.
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
35      IF (RGO - 510.) 40,40,45
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      TOO FEW STEPS TO ACCELERATE.  GO SLOWLY.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
40      RT = 5. * FLOAT (MWATE)
      CALL DFIX (RT, IH, IL)
      IH = IFIX (RGO)
      CALL GOSTP (0,0,IH,IL)
      WSCNT = WSCNT + RGO
      GO TO 10
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MORE THAN 510. STEPS. SET UP AND THEN ACCELERATE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
45      RGO = RGO - 510.
      CALL DFIX (RGO, IH, IL)
      CALL DFLOT (IH, IL, RGO)
      DO 50 I=1,8
      CALL GOSTP(0,0,ISL(I),IWATE(I))
50      CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MIDDLE, CONSTANT VELOCITY STEPS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL GOSTP (0, IH, IL, MWATE)
      CONTINUE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DECELERATE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 55 I=1,8
      J = 9-I
      CALL GOSTP(0,0,ISL(J),IWATE(J))
55      CONTINUE
      WSCNT = WSCNT + RGO + 510.
      GO TO 10

```


MOVE.FT, CONT'D.

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      NO ACCELERATION TO BE USED.
C
C      DECIDE ON DIRECTION OF MOVE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
100      RGO = (WDES - WSCNT)
          IF (RGO) 110,10,120
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MOVE IS TOWARD LOWER WAVELENGTH.  GO 600 TOO FAR.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
110      RGO = 600. - RGO
          CALL DFIX (RGO, IH, IL)
          CALL DFLOT (IH, IL, RGO)
          CALL GOSTP (1, IH, IL, MWATE)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WAIT 0.2 SECONDS FOR TURNAROUND
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
S      CLA CLL
S      TAD C
S      LCTRL
S      STCLK
SD,    DESKP
S      JMP D
S      SKP
SC,    5005
S      CLA CLL
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      AND RETURN 600 STEPS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          CALL GOSTP (0,0,600,MWATE)
          WSCNT = WSCNT + 600. - RGO
          GO TO 10
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MOVE IS TOWARD HIGHER WAVELENGTH.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
120      CALL DFIX (RGO, IH, IL)
          CALL DFLOT (IH, IL, RGO)
          CALL GOSTP (0, IH, IL, MWATE)
          WSCNT = WSCNT + RGO
          GO TO 10
          END

```

```

//// POSIT.SB //// 11/15/75 //////////////////////////////////////
////
//      SUBROUTINE POSIT.SB      11/15/75      E.J. DARLAND      //
//
//      PURPOSE:
//      TO POSITION THE ALPHA CURSOR OF THE TEKTRONIX 4010
//      TERMINAL A GIVEN NUMBER OF LINES AND SPACES FROM THE HOME
//      (UPPER LEFT) POSITION.  THIS ALLOWS INTENTIONAL POSITIONING
//      OF FORTRAN OUTPUT AT A KNOWN PLACE ON THE TERMINAL SCREEN.
//
//      USAGE:  CALL POSIT (NLIN, NSPAC)
//

```

POSIT.SB, CONT'D.

```

//      PARAMETER DESCRIPTION:                                //
//      NLIN:  THE NUMBER OF LINES DOWN FROM THE HOME        //
//              POSITION THAT THE CURSOR WILL BE MOVED.        //
//      NSPAC:  THE NUMBER OF SPACES FROM THE LEFT MARGIN    //
//              THAT THE CURSOR WILL BE MOVED.                //
//
//      MEMORY OCCUPIED:      1 PAGE                            //
//
////////////////////////////////////////////////////////////////
//      OPDEF ITAD 1400      /INDIRECT TAD TO FOOL SABR
NLIN,    0
NSPAC,    0
LF,      0012      /ASCII LINE FEED
SP,      0040      /ASCII SPACE
US,      0037      /TEK TERMINAL CONTROL CHARACTER
CS,      0035      /
B1,      0267      /
B2,      0377      /
B3,      0249      /
B4,      0300      /
////////////////////////////////////////////////////////////////
/
/      ENTRY BLOCK FOR ARGUMENT PASSAGE
/
////////////////////////////////////////////////////////////////
/      ENTRY POSIT
POSIT,   BLOCK 2
////////////////////////////////////////////////////////////////
/
/      GET THE ARGUMENTS
/
////////////////////////////////////////////////////////////////
/      CLA CLL
/      TAD POSIT
/      DCA A
A,      HLT      /THEN CDF MAIN
/      ITAD POSIT#
/      DCA B      /B IS NOW CDF NLIN
/      INC POSIT#
/      ITAD POSIT#
/      DCA NLIN   /NLIN IS NOW ADDR NLIN
/      INC POSIT#
/      ITAD POSIT#
/      DCA C      /C IS NOW CDF NSPAC
/      INC POSIT#
/      ITAD POSIT#
/      DCA NSPAC  /NSPAC IS NOW ADDR NSPAC
/      INC POSIT#
B,      HLT      /THEN CDF NLIN
/      ITAD NLIN
/      CMA        /FORM NEGATIVE-1 OF NLIN
/      DCA NLIN   /AND STORE
C,      HLT      /THEN CDF NSPAC
/      ITAD NSPAC
/      CMA
/      DCA NSPAC  /NSPAC IS NOW -NSPAC - 1
////////////////////////////////////////////////////////////////
/
/      MOVE CURSOR TO HOME POSITION
/
////////////////////////////////////////////////////////////////
/      CLA CLL
/      TAD CS      /CS = 0035; PUTS T4019 IN GRAPHIC MODE
/      JMS CHOUT
/      TAD B1      /B1 = 0267; B1-B4 ARE EQUIVALENT OF POSITION
/      JMS CHOUT   / OF (0,767) ON SCREEN. IE HOME
/      TAD B2      /B2 = 0377
/      JMS CHOUT

```

POSIT.SB, CONT'D.

```

TAD B3          /B3 = 0240
JMS CHOUT
TAD B4          /B4 = 0300
JMS CHOUT
TAD US          /US = 0037; PUTS T4010 IN ALPHA MODE
JMS CHOUT
//////////////////
/
/      NOW ARE AT THE HOME POSITION.  SKIP DOWN NLIN LINES.
/
//////////////////
D,      ISZ NLIN
      SKP
      JMP E
      TAD LF
      JMS CHOUT
      JMP D
//////////////////
/
/      NOW SPACE OVER NSPAC SPACES
/
//////////////////
E,      ISZ NSPAC
      SKP
      JMP F
      TAD SP
      JMS CHOUT
      JMP E
//////////////////
/
/      DONE. RETURN TO MAIN PROGRAM
/
//////////////////
F,      CLA CLL
      RETRN POSIT
//////////////////
/
/      SUBROUTINE TO OUTPUT A CHARACTER TO THE T4010
/
//////////////////
CHOUT,  0
G,      TSF
      JMP G
      TLS
      CLA CLL
      JMP I CHOUT
      END

```

```

CCCC RECORD.FT CCCC 4/8/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      PROGRAM RECORD.FT      V3      4/8/77      E.J. DARLAND      CC
CC
CC      PURPOSE:
CC      THIS IS THE PROGRAM WHICH INPUTS AND CHECKS THE      CC
CC      PARAMETERS FOR DATA ACQUISITION UNDER THE RECORD OPTION OF CC
CC      PIMS8.  WHEN ALL NECESSARY PARAMETERS ARE INPUT AND CHECKED CC
CC      THE OUTPUT FILE IS OPENED, AND RECORD STORES THE PARAMETERS CC
CC      THERE ALONG WITH ANY ADDITIONAL COMMENTS THE USER DESIRES CC
CC      TO TYPE IN. RECORD THEN CHAINS TO PROGRAM RECOR2.FT,      CC
CC      WHICH HANDLES THE ACTUAL DATA ACQUISITION.      CC
CC      VARIABLE INTEGRATION TIME, PERIODIC CORRECTION      CC
CC      MEASUREMENTS AND REAL-TIME OUTPUT ARE THREE OF THE SPECIAL CC
CC      FEATURES OF RECORD, AND ARE DESCRIBED IN SOME DETAIL IN      CC
CC      CHAPTER V.      CC

```

RECORD.FT, CONT'D.

```

CC          USAGE:  RECORD IS AUTOMATICALLY CHAINED INTO MEMORY WHEN      CC
CC          THE RECORD OPTION OF PIMS8 IS CHOSEN.                          CC
CC          SUBROUTINES CALLED:      DFIX          TRUNC                      CC
CC          DFLOT          WAVCK                      CC
CC          FILE          YESNO                      CC
CC          POSIT                      CC
CC          <FROM TEKLIB>      ERASE                      CC
CC          <CHAIN TO>:      RECOR2          PIMS8                      CC
CC          MEMORY OCCUPIED:      22 PAGES PLUS ONE PAGE OF COMMON          CC
CC          STORAGE.                      CC
CC          LOADING INSTRUCTIONS:  LOAD WITH THE /1/0 IOPIONS, AND          CC
CC          LOAD SUBROUTINE FILE NEXT. NOTE                      CC
CC          THAT THE /0 OPTION COSTS 3 MORE                      CC
CC          PAGES OF CORE.                      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
S          SKPDF ABSKP 6434          /SKIP IF COUNTER IS ON MANUAL
S          SKPDF SHSKP 6443          /SKIP IF SHUTTER IS ON MANUAL
COMMON WSCNT, WSPA, UPLIM, MWATE, IACCL, INITF, IPREA, IPREB, ITEK
1, ITEK, ISCNO, IGOBK, IMO, IDAY, IYR, DEV, FILNAM, START, FINSH, STEPS
2, IWAVE, DATAS, TMIN, TMAX, CDES, IREC, TIMES, WBAK1, TIMEB, WREF
3, TIMER, NPPR, TIMED, NPPD, FSPHT, FSION, CSTP, WBAK2
FSPHT = 0.
FSION = 0.
GO TO 35
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C          DECIDE WHETHER TO RETURN TO PIMS8 OR RECORD MORE DATA.
C          (THIS SECTION SKIPPED FIRST TIME THROUGH.)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10          WRITE (1,2000) IOPT,
            CALL YESNO (IOPT)
            IF (IOPT - 3) 15,25,15
15          IF (IOPT - 7) 20,10,20
20          IF (IOPT - 89) 35,30,35
25          CALL EXIT
30          CALL CHAIN ('PIMS8')
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C          GET STORAGE FILE SPECIFICATIONS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
35          IF (ITEK) 45,45,40
40          CALL ERASE
            CALL POSIT (0,0)
45          IOPT = 0
            CALL FILE (DEV, FILNAM, IOPT)
            IOPT = IOPT + 3
            GO TO (10,25,45,50) IOPT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C          GET STARTING WAVELENGTH.  CHECK TO MAKE SURE IT IS
C          WITHIN THE RANGE OF THE MONOCHROMATOR, AND CONVERT IT TO A
C          STEP COUNT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
50          READ (1,2001) START
            IF (START) 10,55,55
55          CALL WAVCK (0., START, UPLIM)
            IF (START) 50,60,60
60          START = TRUNC (START * WSPA)

```

RECORD.FT, CONT'D.

```

      RT = START / WSPA
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GET FINAL WAVELENGTH. CHECK TO MAKE SURE IT IS GREATER OR
C      EQUAL TO THE STARTING WAVELENGTH AND ALSO WITHIN THE
C      MONOCHROMATOR'S RANGE. CONVERT TO STEP COUNT, AND DECIDE IF
C      THE USER SHOULD BE ASKED FOR A DELTA WAVELENGTH OR A NUMBER
C      OF POINTS (IE, DELTA WAVELENGTH = 0.0).
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
75      READ (1,2002) FINSH
      IF (FINSH) 10,80,80
80      CALL WAVCK (RT,FINSH,UPLIM)
      IF (FINSH) 75,85,85
85      FINSH = TRUNC(FINSH * WSPA)
      IF (FINSH - START) 75,100,125
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      ASK FOR THE NUMBER OF POINTS TO RECORD.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
100     IWAVE = 0
      READ (1,2003) DATAS
      IF (DATAS) 10,100,105
105     STEPS = 0.0
      GO TO 170
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GET THE DELTA WAVELENGTH AND CHECK TO MAKE SURE IT
C      IS MORE THAN ZERO AND LESS THAN THE RANGE OF THE
C      MONOCHROMATOR.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
125     IWAVE = 1
      READ (1,2004) STEPS
      IF (STEPS) 10,125,130
130     IF (STEPS - UPLIM) 135,125,125
135     STEPS = TRUNC (STEPS * WSPA)
      IF (STEPS) 10,125,130
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      FIND THE ACTUAL FINAL WAVELENGTH THAT WILL CORRESPOND
C      TO AN INTEGRAL NUMBER OF STEPS OF THE SIZE SPECIFIED IN THE
C      PREVIOUS STEP. THIS WILL BE EQUAL TO OR GREATER THAN THE
C      INPUT FINAL WAVELENGTH.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
150     DATAS = (FINSH - START) / STEPS
155     DATAS = TRUNC (DATAS)
      RT = ((DATAS - 1.) * STEPS) + START
      IF (RT - FINSH) 160,165,165
160     DATAS = DATAS + 1.
      GO TO 155
165     FINSH = RT
170     STEPS = STEPS / WSPA
      START = START / WSPA
      FINSH = FINSH / WSPA
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GET THE MINIMUM INTEGRATION TIME.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
175     READ (1,2005) TMIN
      IF (TMIN) 10,185,180
180     TMIN = (TRUNC(TMIN * 10.))/10.
      IF (TMIN) 183,185,190
185     WRITE (1,2006) TMIN
      GO TO 175
190     IF (TMIN - 409.5) 200,200,195

```

RECORD.FT, CONT'D.

```

195      WRITE (1,2007) TMIN
        GO TO 175
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      READ AND CHECK THE MAXIMUM INTEGRATION TIME.  IF IT IS
C      THE SAME AS THE MINIMUM TIME, DON'T BOTHER WITH THE DESIRED
C      COUNT IN THE NEXT SECTION.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
200      READ (1,2008) TMAX
        IF (TMAX) 10,200,205
205      TMAX = (TRUNC(TMAX * 10.)) / 10.
        IF (TMAX - TMIN) 210,225,215
210      WRITE (1,2006) TMAX
        GO TO 200
215      IF (TMAX - 409.5) 250,250,220
220      WRITE (1,2007) TMAX
        GO TO 200
225      CDES = 0.0
        GO TO 300
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      READ THE DESIRED COUNT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
250      READ (1,2009) CDES
        IF (CDES) 10,250,300
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GET THE DATA TYPE TO BE RECORDED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
300      READ (1,2010) IREC
        IF (IREC) 10,305,305
305      IF (IREC - 3) 325,325,300
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GET THE ELECTROMETER SETTLING TIME.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
325      READ (1,2011) TIMES
        IF (TIMES) 10,330,330
330      TIMES = (TRUNC (TIMES * 10.)) / 10.
        IF (TIMES) 325,335,335
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GET THE BACKGROUND MEASUREMENT PARAMETERS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
335      WRITE (1,2032) IREC,
        CALL YESNO (IOPT)
        IF (IOPT - 3) 336,25,336
336      IF (IOPT - 7) 337,10,337
337      IF (IOPT - 89) 338,339,338
338      TIMEB = -1.0
        GO TO 400
339      READ (1,2036) WBAK1
        IF (WBAK1) 10,339,340
340      CALL WAVCK (20.,WBAK1,UPLIM)
        IF (WBAK1) 339,339,345
345      WBAK1 = (TRUNC (WBAK1 * WSPA)) / WSPA
346      READ (1,2037) WBAK2
        IF (WBAK2) 10,346,347
347      CALL WAVCK (WBAK1,WBAK2,UPLIM)
        IF (WBAK2) 346,346,348
348      WBAK2 = (TRUNC(WBAK2 * WSPA)) / WSPA
350      READ (1,2014) TIMEB
        IF (TIMEB) 10,350,355
355      TIMEB = (TRUNC (TIMEB * 10.)) / 10.

```

RECORD.FT, CONT'D.

```

      IF (TIMEB) 360,360,365
360    WRITE (1,2006) TIMEB
      GO TO 350
365    IF (TIMEB - 409.5) 400,400,370
370    WRITE (1,2007) TIMEB
      GO TO 350
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GET THE REFERENCE MEASUREMENT PARAMETERS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
400    WRITE (1,2033) IOPT,
      CALL YESNO (IOPT)
      IF (IOPT - 3) 401,25,401
401    IF (IOPT - 7) 402,10,402
402    IF (IOPT - 89) 403,404,403
403    TIMER = -1.0
      GO TO 450
404    READ (1,2013) WREF
      IF (WREF) 10,404,405
405    CALL WAVCK (20.,WREF,UPLIMD
      IF (WREF) 404,404,410
410    WREF = (TRUNC (WREF * WSPA)) / WSPA
415    READ (1,2014) TIMER
      IF (TIMER) 10,415,420
420    TIMER = (TRUNC (TIMER * 10.)) / 10.
      IF (TIMER) 425,425,430
425    WRITE (1,2006) TIMER
      GO TO 415
430    IF (TIMER - 409.5) 445,445,435
435    WRITE (1,2007) TIMER
      GO TO 415
445    READ (1,2015) RT
      IF (RT) 10,445,446
446    IF (RT - 2047.) 449,449,447
447    WRITE (1,2007) RT
      GO TO 445
449    NPPR = IFIX (RT)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GET THE DARK MEASUREMENT PARAMETERS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
450    WRITE (1,2017)
452    READ (1,2014) TIMED
      IF (TIMED) 10,452,455
455    TIMED = (TRUNC (TIMED * 10.)) / 10.
      IF (TIMED) 460,460,465
460    WRITE (1,2006) TIMED
      GO TO 452
465    IF (TIMED - 409.5) 480,480,470
470    WRITE (1,2007) TIMED
      GO TO 452
480    READ (1,2015) RT
      IF (RT) 10,480,485
485    IF (RT - 2047.) 495,495,490
490    WRITE (1,2007) RT
      GO TO 480
495    NPPD = IFIX (RT)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GET REAL-TIME PLOT SCALING FACTORS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
500    WRITE (1,2018)
      GO TO (505,505,510) IREC
505    READ (1,2019) FSPHT
      IF (FSPHT) 10,500,510
510    GO TO (515,520,515) IREC

```

[illegible]

RECORD.FT, CONT'D.

```

        IF (IDUM - 3) 685,25,685
685      IF (IDUM - 89) 10,690,10
690      CONTINUE
S        SHSKP
S        JMP A
S        NOP
        WRITE (1,2029)
        GO TO 700
SA,      ABSKP
S        JMP B
S        NOP
        WRITE (1,2030)
700      WRITE (1,2028)
        CALL YESNO (1OPT)
        GO TO 690
SB,      NOP
        CALL CHAIN ('RECOR2')
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C        FORMAT STATEMENTS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2000      FORMAT('RETURN TO PIMS8? ',10)
2001      FORMAT('BEGINNING WAVELENGTH =      ',F11.4)
2002      FORMAT('ENDING WAVELENGTH =      ',F11.4)
2003      FORMAT('NUMBER OF POINTS =      ',F7.0)
2004      FORMAT('WAVELENGTH INCREMENT =      ',F11.4)
2005      FORMAT('MINIMUM INTEGRATION TIME = ',F8.1)
2006      FORMAT(E11.4, ' TOO SMALL!')
2007      FORMAT(E11.4, ' TOO LARGE!')
2008      FORMAT('MAX. INTEGRATION TIME =      ',F8.1)
2009      FORMAT('DESIRED MINIMUM COUNT = ',F9.0)
2010      FORMAT('PIE [1] PHOTONS [2] OR IONS [3] : ',12)
2011      FORMAT('SETTLING TIME, SECONDS =      ',F8.1)
2012      FORMAT('/'BACKGROUND [STRAY LIGHT] CORRECTION PARAMETERS')
2013      FORMAT('          WAVELENGTH =      ',F10.3)
2014      FORMAT('          INTEGRATION TIME = ',F8.1)
2015      FORMAT('          POINTS / MEASUREMENT = ',F7.0)
2016      FORMAT('/'REFERENCE CORRECTION PARAMETERS')
2017      FORMAT('/'DARK CORRECTION PARAMETERS')
2018      FORMAT('/'REAL-TIME PLOT SCALING')
2019      FORMAT('PHOTON FULL SCALE =      ',F7.0)
2020      FORMAT('ION FULL SCALE =      ',F7.0)
2021      FORMAT('POINTS PER INCH FOR PLOT =      ',F8.1)
2022      FORMAT('ANGSTROMS / INCH FOR PLOT =      ',F8.1)
2023      FORMAT(A4,':',A6'.DA')
2024      FORMAT(10X,'RECORDING ',10)
2025      FORMAT('PIE'/)
2026      FORMAT('PHOTONS ONLY'/)
2027      FORMAT('IONS ONLY'/)
2028      FORMAT('OKAY ? ',10)
2029      FORMAT('SET SHUTTER TO AUTO MODE!!!')
2030      FORMAT('SET DUAL COUNTER TO AUTO MODE!!!')
2031      FORMAT('          POINTS / MEASUREMENT = ',18)
2032      FORMAT('/'BACKGROUND MEASUREMENTS?',10)
2033      FORMAT('/'REFERENCE MEASUREMENTS?',10)
2034      FORMAT('/'NO BACKGROUND MEASUREMENTS')
2035      FORMAT('/'NO REFERENCE MEASUREMENTS')
2036      FORMAT('          FIRST WAVELENGTH =      ',F10.3)
2037      FORMAT('          LAST WAVELENGTH =      ',F10.3)
        END

```

RECOR2.FT

```

CCCC RECOR2.FT CCCC 4/14/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                     CC
CC      PROGRAM RECOR2.FT      V2.B      4/14/77      E.J. DARLAND      CC
CC                                     CC
CC      PURPOSE:                                                         CC
CC          TO EXECUTE A DATA ACQUISITION RUN GUIDED BY THE             CC
CC          PARAMETERS PASSED FROM PROGRAM RECORD.FT. VARIABLE           CC
CC          INTEGRATION TIME, REAL-TIME GRAPHIC OUTPUT AND               CC
CC          PERIODIC CORRECTION MEASUREMENTS ARE FEATURES WHICH          CC
CC          MAY BE USED. FOR A FULL DESCRIPTION OF THE RECORD           CC
CC          OPTION, SEE CHAPTER V.                                       CC
CC                                     CC
CC      USAGE: RECOR2 IS CHAINED INTO MEMORY AUTOMATICALLY AT THE       CC
CC          CONCLUSION OF PROGRAM RECORD, WHICH IS RUN IN               CC
CC          RESPONSE TO THE 'RECORD' OPTION OF PIMSB.                   CC
CC                                     CC
CC      INTERNAL VARIABLES:                                              CC
CC          DCNT: NUMBER OF DATA POINTS ACQUIRED SO FAR.               CC
CC          ELAPS: TOTAL ELAPSED TIME SINCE THE BEGINNING OF             CC
CC                  THE RUN, IN SECONDS.                                CC
CC          IBTIM: NUMBER OF 0.10 SECOND WAIT PERIODS IN THE            CC
CC                  BACKGROUND INTEGRATION TIME.                        CC
CC          IDCNT: NUMBER OF DATA POINTS TAKEN SINCE THE               CC
CC                  LATEST DARK COUNT MEASUREMENT.                     CC
CC          IDTIM: NUMBER OF 0.10 SECOND WAIT PERIODS IN THE            CC
CC                  DARK INTEGRATION TIME.                              CC
CC          IRCNT: NUMBER OF DATA POINTS TAKEN SINCE THE              CC
CC                  LATEST REFERENCE MEASUREMENT.                      CC
CC          IRTIM: NUMBER OF 0.10 SECOND WAIT PERIODS IN THE            CC
CC                  REFERENCE INTEGRATION TIME.                         CC
CC          ISECT: MEASUREMENT SECTION FLAG:                             CC
CC                  1: INITIAL CORRECTION MEASUREMENTS                 CC
CC                  2: REGULAR MEASUREMENTS                             CC
CC                  3: FINAL CORRECTION MEASUREMENTS                   CC
CC          ISETTL: ELECTROMETER SETTLING FLAG:                          CC
CC                  0: DELAY FOR SETTLING IS NOT                        CC
CC                      NECESSARY.                                       CC
CC                  1: A DELAY MUST BE INCLUDED BEFORE                 CC
CC                      ACQUIRING MORE DATA.                            CC
CC          ISTIM: NUMBER OF 0.10 SECOND WAIT PERIODS IN THE            CC
CC                  ELECTROMETER SETTLING TIME.                         CC
CC          MINT:  NUMBER OF 0.10 SECOND WAIT PERIODS IN THE            CC
CC                  MINIMUM (MANDANTORY) INTEGRATION TIME               CC
CC          SCB1:  STEP COUNT CORRESPONDING TO THE FIRST                CC
CC                  BACKGROUND WAVELENGTH.                               CC
CC          SCB2:  STEP COUNT CORRESPONDING TO THE SECOND               CC
CC                  BACKGROUND WAVELENGTH.                               CC
CC          SCINIT: STEP COUNT CORRESPONDING TO THE STARTING             CC
CC                  WAVELENGTH.                                          CC
CC          SCLAST: STEP COUNT CORRESPONDING TO THE MOST                 CC
CC                  RECENT DATA WAVELENGTH.                            CC
CC          SCREF:  STEP COUNT CORRESPONDING TO THE REFERENCE            CC
CC                  WAVELENGTH.                                          CC
CC          SCSTEP: STEP COUNT CORRESPONDING TO THE                      CC
CC                  WAVELENGTH INCREMENT BETWEEN POINTS.               CC
CC          TIMB:   ONE HALF OF THE BACKGROUND INTEGRATION              CC
CC                  TIME.                                                CC
CC          TIMD:   ONE HALF OF THE DARK INTEGRATION TIME.              CC
CC          TIMR:   ONE HALF OF THE REFERENCE INTEGRATION               CC
CC                  TIME.                                                CC
CC          TNES:   TIME NECESSARY TO GET DESIRED COUNTS.               CC
CC          TIMET:  TOTAL INTEGRATION TIME ACTUALLY USED.                CC
CC                                     CC
CC      SUBROUTINES CALLED: COUNT      LIMIT      CC
CC                          DELAY      MOVE      CC
CC                          DFIX      TCALC      CC
CC                          DFLOT      WTEXT      CC
CC                          GOSTP      YESNO      CC

```

RECOR2.FT, CONT'D.

```

CC                                     KCHEK                                     CC
CC      <FROM SUBR>:                                     CC
CC      SCEND   SCPLT   SCST                                     CC
CC      <CHAINS TO>:                                     CC
CC      PIMS8                                           CC
CC
CC      NECESSARY OPERATION CODES:                                     CC
CC
S      OPDEF RLOAD 6072          /LOAD PLOTTER D/A FROM AC          CC
S      OPDEF PEN   6071          /CHANGE STATUS OF PLOTTER PEN      CC
S      OPDEF LCTRL 6131          /LOAD CLOCK CONTROL REGISTER      CC
S      OPDEF LPSET 6121          /LOAD CLOCK PRESET                 CC
S      OPDEF STCLK 6124          /START CLOCK                       CC
S      SKPDF POSKP 6132          /SKIP ON PGM CLOCK OFLOW.         CC
S      OPDEF SHCL  6441          /CLOSE SHUTTER                     CC
S      OPDEF SHOP  6442          /OPEN SHUTTER                      CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
COMMON WSCNT, WSPA, UPLIM, MWATE, IACCL, INITF, IPREA, IPREB, ITEK
1, ISCNO, IGOBK, IMO, IDAY, IYR, DEV, FILNAM, START, FINSH, STEPS
2, IWAVE, DATAS, TMIN, TMAX, CDES, IREC, TIMES, WBAK1, TIMEB, WREF
3, TIMER, NPPR, TIMED, NPPD, FSPHT, FSION, CSTP, WBAK2
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      OPEN THE OUTPUT FILE AND RECORD THE SCAN PARAMETERS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CALL OOPEN (DEV, FILNAM)
  ICODE = 5
  WRITE (4,2000) ICODE, FILNAM
  ICODE = 6
  WRITE (4,2002) ICODE, IMO, IDAY, IYR
  ICODE = 7
  WRITE (4,2002) ICODE, ISCNO
  ICODE = 20
  WRITE (4,2002) ICODE, IPREA, IPREB
  ICODE = 9
  WRITE (4,2000) ICODE, START, FINSH
  ICODE = 10
  WRITE (4,2000) ICODE, STEPS
  ICODE = 11
  WRITE (4,2000) ICODE, TMIN, TMAX, CDES
  ICODE = 13
  WRITE (4,2002) ICODE, NPPD
  ICODE = 14
  WRITE (4,2000) ICODE, TIMED, TIMES
  IF (TIMEB) 11,10,10
10    ICODE = 19
    WRITE (4,2000) ICODE, WBAK1, WBAK2, TIMEB
11    IF (TIMER) 13,12,12
12    ICODE = 15
    WRITE (4,2000) ICODE, WREF, TIMER
13    ICODE = 16
    WRITE (4,2002) ICODE, NPPR
    ICODE = 12
    WRITE (4,2000) ICODE, FSPHT, FSION
    ICODE = 18
    WRITE (4,2000) ICODE, CSTP
    CCNT = 0.
    IF (IWAVE) 15,15,14
14    CCNT = ((START / CSTP) * 600.)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      APPEND USER COMMENTS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
15    WRITE (1,2003) ICODE,
      CALL WTEXT

```

```
C
C      SET THE INTERNAL VARIABLES BEFORE EXECUTION.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    IDCNT = -1
    ISECT = 1
    IRCNT = -1
    DCNT = 0.0
    ELAPS = 0.0
    TEMP = TIMES * 10.
    CALL DFIX (TEMP,IH,ISTIM)
    TEMP = TIMER * 10.
    CALL DFIX (TEMP,IH,IRTIM)
    TEMP = TIMED * 10.
    CALL DFIX (TEMP,IH,IDTIM)
    TEMP = TIMEB * 10.
    CALL DFIX (TEMP,IH,IBTIM)
    TIMB = TIMEB / 2.0
    TMR = TIMER / 2.0
    TIMD = TIMED / 2.0
    TEMP = TMIN * 10.
    CALL DFIX (TEMP,IH,MINT)
    SCB1 = WBAK1 * WSPA
    SCB2 = WBAK2 * WSPA
    SCINIT = START * WSPA
    SCREF = WREF * WSPA
    SCSTP = STEPS * WSPA
    BP1 = 0.
    BP2 = 0.
    BI1 = 0.
    BI2 = 0.
    DP = 0.
    DI = 0.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      THIS IS THE START OF THE DATA ACQUISITION LOOP. FIRST DECIDE
C      IF THERE ARE ANY CORRECTION MEASUREMENTS TO BE MADE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
100   ISETTL = 0
       ID = 0
       IB1 = 0
       IB2 = 0
       IR = 0
       GO TO (130,103,130) ISECT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WE ARE IN THE MIDDLE SECTION OF DATA ACQUISITION. IE, WE ARE
C      NOT INVOLVED IN THE PRELIMINARY CORRECTION MEASUREMENTS OR THE
C      FINAL CORRECTION MEASUREMENTS. WILL ONLY NEED TO DO A DARK
C      MEASUREMENT IF ENOUGH DATA POINTS HAVE BEEN ACQUIRED SINCE THE
C      LAST ONE. . . .
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
103   IDCNT = IDCNT + 1
       IF (IDCNT - NPPD) 115,110,110
110   ID = 1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      AND A REFERENCE MEASUREMENT IF WE ARE ALLOWED TO DO ANY AT
C      ALL (TIMER NOT <0.) AND ENOUGH DATA POINTS HAVE BEEN ACQUIRED
C      SINCE THE LAST ONE. WHEN THIS DECISION IS MADE, GO TO EXECUTE
C      THE MEASUREMENTS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
115   IF (TIMER) 142,120,120
120   IRCNT = IRCNT + 1
       IF (IRCNT - NPRR) 142,125,125
```

REC02.FT, CONT'D.

```

125      IR = 1
          IORDR = 1
          GO TO 250
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WE WIND UP HERE IF WE ARE IN THE PRELIMINARY OR FINAL CORRECTION
C      MEASUREMENT SECTION.  A DARK MEASUREMENT IS MANDATORY.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
130      ID = 1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      AND WE NEED TO DO ALL ALLOWED BACKGROUND AND REFERENCE
C      MEASUREMENTS AS WELL.  HERE'S WHERE WE DECIDE IF BOTH B AND R
C      MEASUREMENTS ARE NEEDED, OR ONLY ONE, OR NEITHER.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          WAVE = WSCNT / WSPA
          WA = ABS (WAVE - WBAK1)
          WB = ABS (WAVE - WBAK2)
          WC = ABS (WAVE - WREF)
          IF (TIMER) 135,140,140
135      IF (TIMEB) 142,150,150
140      IF (TIMEB) 145,170,170
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      NEITHER BACKGROUND OR REFERENCE IS ALLOWED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
142      IORDR = 13
          GO TO 250
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      REFERENCE ALLOWED, BACKGROUND IS NOT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
145      IORDR = 1
          IR = 1
          GO TO 250
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      BACKGROUND ALLOWED, REFERENCE IS NOT.  NOW HAVE TO DECIDE
C      WHETHER TO DO ONE OR TWO MEASUREMENTS OF BACKGROUND, AND IF
C      TWO, WHAT ORDER TO DO THEM IN.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
150      IF (WA - WB) 160,151,165
151      IF (WBAK1 - WBAK2) 160,155,160
155      IORDR = 2
          IB1 = 1
          GO TO 250
160      IORDR = 3
          IB1 = 1
          IB2 = 1
          GO TO 250
165      IORDR = 4
          IB1 = 1
          IB2 = 1
          GO TO 250
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      NEED TO DO BOTH BACKGROUND AND REFERENCE.  CHECK TO SEE IF ONLY
C      ONE BACKGROUND IS NEEDED AND IF SO, DECIDE WHETHER TO DO THE
C      BACKGROUND OR REFERENCE FIRST, DEPENDING ON WHICH IS CLOSEST TO
C      THE CURRENT WAVELENGTH.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
170      IF (WA - WB) 190,171,190
171      IF (WBAK1 - WBAK2) 190,175,190

```

[illegible]

REC0R2.FT, CONT'D.

```

250      IF (ID) 270,270,255
255      IDCNT = 0
S        SHCL
          CALL DELAY (ISTIM)
          ICODE = 4
          WAVE = WSCNT / WSPA
          CALL COUNT (IDTIM,CNTP,CNTI)
          CALL KCHEK (IDUM)
          IF (IDUM) 260,260,820
260      CONTINUE
S        SHOP
          DP = CNTP / TIMED
          DI = CNTI / TIMED
          ELAPS = ELAPS + TIMES + TIMED
          WRITE (4,2000) ICODE, WAVE, DP, DI, ELAPS
          ELAPS = ELAPS + TIMD
          ISETTL = 1
          CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C        IF THIS IS THE FIRST DARK MEASUREMENT (ISECT = 1) AND NO
C        BACKGROUNDS ARE TO BE MEASURED, RECORD A DUMMY BACKGROUND FOR
C        THE CORRECT OPTION.  LIKEWISE WITH REFERENCE.
C        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
265      IF (ISECT - 1) 270,266,270
266      IF (TIMEB) 267,268,268
267      ICODE = 8
          RT = 0.
          WRITE (4,2000) ICODE, START, RT, RT, RT
268      IF (TIMER) 269,270,270
269      ICODE = 17
          RT = 0.
          RT2 = 1.
          WRITE (4,2000) ICODE, START, RT2, RT2, RT
          CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C        THE FOLLOWING FOUR GO TO STATEMENTS CAUSE THE PROGRAM TO GO TO
C        THE SECTIONS THAT TAKE THE REFERENCE, 1ST BACKGROUND OR 2ND
C        BACKGROUND MEASUREMENTS, IN THE PROPER ORDER, BASED ON THE
C        VALUE OF THE IORDR FLAG.
C        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
270      TIME1 = 0.
          TIME2 = 0.
          TIME3 = 0.
          DO 330 I = 1,3
          GO TO (272,275,280) I
272      GO TO (335,300,300,320,335,300,300,300,335,320,320,335,352)
          1 IORDR
275      GO TO (352,352,320,300,300,335,320,335,300,300,335,320,352)
          1 IORDR
280      GO TO (352,352,352,352,352,352,335,320,320,335,300,300,352)
          1 IORDR
          CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C        FIRST BACKGROUND MEASUREMENT
C        CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
300      CALL TCALC (TEMP,SCB1)
          CALL MOVE (SCB1)
          CALL KCHEK (IDUM)
          IF (IDUM) 305,305,820
305      CALL DELAY (ISTIM)
          CALL COUNT (IDTIM, CNTP, CNTI)
          CALL KCHEK (IDUM)
          IF (IDUM) 310,310,820
310      BP1 = CNTP / TIMEB
          BI1 = CNTI / TIMEB
          ELAPS = ELAPS + TEMP + TIMES + TIMB

```

[illegible]


```

C
C      HAVE JUST COMPLETED THE PRELIMINARY MEASUREMENTS AND ARE
C      ABOUT TO START DATA ACQUISITION.  CALCULATE THE BACKGROUND
C      CORRECTION PARAMETERS TO BE USED FOR THE REAL TIME PLOT.  THEN
C      SET ISECT = 2, DRAW THE AXES ON THE RECORDER AND GO TO THE
C      STARTING WAVELENGTH.  THEN JUMP TO THE SECTION THAT
C      ACQUIRES A DATA POINT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
380      IF (TIMEB) 381,382,382
381      SLOPP = 0.
          SLOPI = 0.
          RINTP = 0.
          RINTI = 0.
          GO TO 395
382      BP1 = BP1 - DP
          BI1 = BI1 - DI
          IF (WBAK1 - WBAK2) 390,385,390
385      SLOPP = 0.
          SLOPI = 0.
          RINTP = BP1
          RINTI = BI1
          GO TO 395
390      BP2 = BP2 - DP
          BI2 = BI2 - DI
          SLOPP = (BP2 - BP1) / (WBAK2 - WBAK1)
          SLOPI = (BI2 - BI1) / (WBAK2 - WBAK1)
          RINTP = BP1 - (WBAK1 * SLOPP)
          RINTI = BI1 - (WBAK1 * SLOPI)
395      CONTINUE
S      CLA CLL
S      TAD (315          /205 DECIMAL; .10 FULL SCALE
S      RAL
S      RLOAD
S      CLA CLL
          CALL DELAY (30)
          CALL SCST (205)
          CALL SCPLT (600,205)
S      CLA CLL
S      TAD (7777
S      RLOAD
S      CLA CLL
          CALL DELAY (30)
S      CLA CLL
S      TAD (0400
S      LCTRL
S      TAD (7765
S      LPSET
S      CLA CLL
          DO 400 I = 0,2047
          J = 2047 - I
S      CLA CLL
S      TAD \J
S      RAL
S      RLOAD
S      STCLK
SA,    POSKP
S      JMP A
S      CLA CLL
400    CONTINUE
          CALL DELAY (30)
S      PEN
          CALL SCST (205)
          ISECT = 2
          CALL TCALC (TEMP, SCINIT)
          CALL MOVE (SCINIT)
          ELAPS = ELAPS + TEMP
          CALL KCHEK (IDUM)

```

RECOR2.FT, CONT'D.

```

                IF (IDUM 500,500,820
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      THE CORRECTION MEASUREMENTS MADE, IF ANY, WERE ROUTINE, SO
C      RETURN TO THE PREVIOUS (DATA) WAVELENGTH AND THEN GET MORE
C      DATA.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
405  CALL TCALC (TEMP, SCLAST)
      CALL MOVE (SCLAST)
      ELAPS = ELAPS + TEMP
      CALL KCHEK (IDUM)
      IF (IDUM 500,500,820
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      HAVE JUST FINISHED THE FINAL CORRECTION MEASUREMENTS, SO
C      CLOSE FILE, DRAW FINAL AXES AND RETURN TO PIMS8.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
410  ICODE = 99
      WRITE (4,2990) ICODE
      CALL OCLOSE
      ISCNO = ISCNO + 1
S      CLA CLL
S      TAD (7777
S      RLOAD
S      CLA CLL
      CALL DELAY (30)
S      CLA CLL
S      TAD (0401
S      LCTRL
S      TAD (7766
S      LPSET
      DO 415 I = 1,2047
      J = 2047 - I
S      CLA CLL
S      TAD \J
S      RAL
S      RLOAD
S      STCLK
SB,   POSKP
S      JMP B
S      CLA CLL
415  CONTINUE
      CALL DELAY (30)
S      PEN
      CALL SCST (205)
      CALL DELAY (40)
      CALL SCPLT (600,205)
      CALL SCEND
      CCNT = CCNT / 600.
      RT = TRUNC (CCNT)
      RT = 600. * (1.0 - (CCNT - RT))
      IL = IFIX (RT)
      CALL SCPLT (IL,0)
      CALL SCPLT (600,0)
      CALL SCPLT (600,0)
      CALL CHAIN ('PIMS8')
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WE ARE ABOUT TO COLLECT A DATA POINT, BUT FIRST WE NEED TO SEE
C      IF WE SHOULD WAIT AWHILE.  THIS WOULD BE THE CASE IF WE JUST
C      FINISHED A DARK MEASUREMENT OR REFERENCE AND NEED TO LET
C      THE ELECTROMETER SETTLE, FOR INSTANCE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
500  CONTINUE
S      SHOP          /JUST TO MAKE SURE!
      IF (ISETTL) 503,503,502

```

REC0R2.FT, CONT'D.

```

502      CALL DELAY (ISTIM)
          ELAPS = ELAPS + TIMES
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      INTEGRATE PHOTON AND ION INTENSITY FOR TMIN SECONDS AND
C      CALCULATE THE EQUIVALENT COUNTING RATES (COUNTS PER SECOND)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
503      CALL COUNT (MINT, TEMP, TEMI)
          CALL KCHEK (IDUM)
          IF (IDUM) 503,505,820
505      TIMET = TMIN
          CNTP = TEMP / TMIN
          CNTI = TEMI / TMIN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DECIDE ON THE BASIS OF IREC WHICH TYPE OF COUNTS ARE IMPORTANT
C      IN THE VARIABLE INTEGRATION TIME CALCULATIONS (PHOTON OR ION).
C      CORRECT THEM FOR THE MOST RECENT BACKGROUND AND DARK
C      MEASUREMENTS, AND CALCULATE THE TIME NECESSARY TO REACH THE
C      DESIRE NUMBER OF COUNTS, TNES.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          WAVE = WSCNT / WSPA
          BP = SLOPP * WAVE + RINTP
          BI = SLOPI * WAVE + RINTI
          GO TO (510,520,510) IREC
510      CNTS = CNTI - BI - DI
          IF (CNTS) 600,600,530
520      CNTS = CNTP - BP - DP
          IF (CNTS) 600,600,530
530      TNES = CDES / CNTS
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      IF THE NECESSARY TIME IS BETWEEN TMIN AND TMAX, PERFORM THE
C      EXTRA INTEGRATION AND ADD THE COUNTS ACQUIRED TO THOSE GOTTEN
C      IN THE FIRST INTEGRATION (FOR TMIN SECONDS).
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          IF (TNES - TMIN) 600,600,540
540      IF (TNES - TMAX) 550,550,600
550      TNES = (TNES - TMIN) * 10.
          CALL DFIX (TNES, IH, IL)
          CALL DFLOT (IH, IL, TNES)
          TNES = TNES / 10.
          CALL COUNT (IL, CNTP, CNTI)
          CALL KCHEK (IDUM)
          IF (IDUM) 560,560,820
560      TIMET = TNES + TMIN
          CNTP = (CNTP + TEMP) / TIMET
          CNTI = (CNTI + TEMI) / TIMET
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CNTP AND CNTI NOW CONTAIN THE COUNTS / SECOND AVERAGED OVER
C      THE ENTIRE INTEGRATION TIME. STEP TO THE NEXT
C      WAVELENGTH, WRITE OUT THE DATA AND CALCULATE THE DATA FOR
C      THE REAL-TIME PLOT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
600      SCLAST = WSCNT + SCSTP
          CALL TCALC (TEMI, SCLAST)
          CALL MOVE (SCLAST)
          WAVE2 = WAVE
          ELAPS = ELAPS + (TIMET / 2.0)
610      GO TO (630,615,620) IREC
615      WRITE (4,2000) IREC, WAVE, CNTP, ELAPS, TIMET
          TEMP = ((CNTP - BP - DP) * (1842. / FSPHT)) + 204.7
          GO TO 660
620      WRITE (4,2000) IREC, WAVE, CNTI, ELAPS, TIMET

```

REC0R2.FT, CONT'D.

```

      TEMP = ((CNT1 - BI - DI) * (1842. / FS10N)) + 204.7
      GO TO 660
630   WRITE (4,2000) IREC, WAVE, CNTP, CNT1, ELAPS, TIMET
      DEN = (CNTP - BP - DP) / FSPHT
      IF (DEN) 650,640,650
640   DEN = 0.0000000001
650   RNUM = (CNT1 - BI - DI) / FS10N
      TEMP = ((RNUM / DEN) * 1842.) + 204.7
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      TEMP NOW CONTAINS THE PLOT DATA, WHATEVER IT IS, HOPEFULLY
C      SCALED BETWEEN 0 AND 2047 AS THE SCSYS PROGRAMS REQUIRE.
C      HOWEVER, HE SAYS OPTIMISTICALLY, IT PROBABLY IS NOWHERE NEAR
C      THAT RANGE, SO WE'D BETTER LIMIT IT TO MAKE SURE, AND UPDATE
C      ELAPS WHILE WE'RE AT IT. THEN WE CAN CALCULATE THE NUMBER
C      OF CHART STEPS TO TAKE TO GET TO THE NEXT POINT, AND PLOT THE
C      SAME.
C
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
660   IL = LIMIT (0, TEMP, 2047)
      ELAPS = ELAPS + (TIMET / 2.0) + TEMI
700   IF (IWAVE) 710,710,705
705   TEMP = ((WAVE2 / CSTP) * 600.) - CCNT
      GO TO 715
710   TEMP = ((DCNT / CSTP) * 600.) - CCNT
715   IF (TEMP) 720,750,730
720   TEMP = 0.
      GO TO 750
730   IF (TEMP - 900.) 750,750,740
740   TEMP = 900.
750   IDUM = IFIX (TEMP)
      CCNT = CCNT + FLOAT (IDUM)
      CALL SCPLT (IDUM, IL)
      ELAPS = ELAPS + (FLOAT (IDUM) * 0.01)
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DECIDE IF ANOTHER DATA POINT WILL BE NECESSARY. IF SO, GO
C      BACK TO SEE IF A DARK MEASUREMENT SHOULD BE MADE FIRST, ETC.
C      IF NOT, SET ISECT = 3 AND GO BACK FOR THE FINAL DARK, BACKGROUND
C      AND REFERENCE MEASUREMENTS.
C
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
800   DCNT = DCNT + 1.
      IF (DCNT - DATAS) 100,810,810
810   ISECT = 3
      GO TO 100
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      HERE'S WHERE WE WIND UP IF AN OPERATOR INTERRUPT WAS FOUND
C      ANYWHERE IN THE PROGRAM. ASK IF THE FINAL THREE CORRECTION
C      MEASUREMENTS SHOULD BE MADE. IF SO, MAKE THEM, IF NOT,
C      CLOSE UP SHOP AND GO HOME.
C
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
820   WRITE (1,2001) IDUM,
      CALL YESNO (IDUM)
      IF (IDUM - 3) 825,830,825
825   IF (IDUM - 89) 410,810,410
830   CALL EXIT
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      FORMAT STATEMENTS
C
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2000   FORMAT(A2,5A6)
2001   FORMAT('DO FINAL CORRECTION MEASUREMENTS? ',10)
2002   FORMAT(5A2)
2003   FORMAT('%',10)
      END

```

RTEXT.FT

```

CCCC RTEXT.FT CCCC 4/18/76 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                     CC
CC      SUBROUTINE RTEXT.FT      4/18/76      E.J.  DARLAND      CC
CC                                     CC
CC      PURPOSE:      CC
CC      TO READ COMMENT LINES FROM PIMS8 FORMAT FILES AND      CC
CC      TRANSFER THEM IN ASCII TO AN OUTPUT DEVICE.      CC
CC      RTEXT ASSUMES THAT THE INPUT IS VIA FORTRAN II DEVICE-      CC
CC      INDEPENDENT I/O (IE., DEVICE CODE 4).  IT FURTHER ASSUMES      CC
CC      THAT THE '98' TAG AT THE BEGINNING OF THE LINE HAS BEEN      CC
CC      READ IN THE CALLING PROGRAM, WITH CARRIAGE RETURN      CC
CC      SUPPRESSED.  IN OTHER WORDS, THE NEXT CHARACTER OF INPUT IS      CC
CC      THE FIRST CHARACTER OF THE COMMENT.  RTEXT TRANSFERS ONE      CC
CC      CHARACTER AT A TIME TO THE OUTPUT DEVICE UNTIL THE '@' END      CC
CC      OF LINE CHARACTER IS ENCOUNTERED.  RTEXT ECHOES A CRLF AND      CC
CC      THEN RETURNS TO THE MAIN PROGRAM.      CC
CC                                     CC
CC      USAGE:  CALL RTEXT (IDEV)      CC
CC                                     CC
CC      PARAMETER DESCRIPTION:      CC
CC      IDEV:  FORTRAN II DEVICE NUMBER FOR OUTPUT:      CC
CC              1: TERMINAL      CC
CC              2: PAPER TAPE PUNCH      CC
CC              4: DEVICE INDEPENDENT I/O DEVICE.      CC
CC                                     CC
CC      MEMORY OCCUPIED:      1 PAGE      CC
CC                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE RTEXT (IDEV)
      ID = IDEV
S      CLA CLL      /WRITE OUT TWO BLANKS AT BEGINNING OF LINE
S      TAD (4040
S      DCA \ICAR
      WRITE(ID,1002) ICHAR,
10     READ(4,1000) ICHAR,
S      CLA CLL      /TEST FOR @ (ASCII:0000)
S      TAD \ICAR
S      AND (7700      /LOOKING AT UPPER SIX BITS ONLY
S      SNA      /IS IT AN @?
S      JMP TEND      /YES: FINISH LINE
S      NOP      /NO: ECHO
      WRITE(ID,1000) ICHAR,
      GO TO 10
STEND, CLA CLL      /ECHO CRLF
      WRITE (ID,1001) ICHAR
40     READ (4,1001) ICHAR
      RETURN
1000    FORMAT(A1)
1001    FORMAT(I0)
1002    FORMAT(A2)
      END

```

```

CCCC SCAN.FT CCCC 1/31/75 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                     CC
CC      SUBROUTINE SCAN.FT      V2      1/31/77      E.J.  DARLAND      CC
CC                                     CC
CC      PURPOSE:      CC
CC      USED TO SCAN THE MONOCHROMATOR FROM ONE WAVELENGTH TO      CC
CC      ANOTHER AT A CONSTANT RATE.      CC
CC                                     CC
CC      USAGE:  CALL SCAN      CC
CC                                     CC

```

SCAN.FT, CONT'D.

```

CC          SCAN WILL ASK THE OPERATOR TO INPUT THE STARTING AND  CC
CC          ENDING WAVELENGTHS AS WELL AS THE DESIRED SCAN RATE IN  CC
CC          ANGSTROMS PER MINUTE. IT WILL THEN CHECK TO MAKE SURE THAT CC
CC          THE DESIRED WAVELENGTHS ARE WITHIN THE PHYSICAL LIMITS OF  CC
CC          THE SYSTEM, AND THAT THE SCAN IS TOWARD HIGHER WAVELENGTH.  CC
CC          SINCE ONLY .01 MILLISECOND WAIT PERIODS ARE USED FOR TIMING  CC
CC          THE STEPPING MOTOR, NOT ALL SCAN RATES ARE POSSIBLE. SCAN  CC
CC          CALCULATES THE NEXT FASTEST SCAN RATE IF THE DESIRED RATE  CC
CC          IS NOT EXACTLY OBTAINABLE. IF THE SCAN RATE IS WITHIN  CC
CC          THE LIMITATIONS OF THE STEPPING MOTOR AND SOFTWARE, SCAN  CC
CC          WILL OUTPUT THE WAVELENGTHS AND SCAN RATE FOR OPERATOR  CC
CC          OKAY BEFORE EXECUTION OF THE SCAN.  CC
CC
CC          SUBROUTINES CALLED:      DFIX          MOVE          CC
CC                                DFLOT         POSIT         CC
CC                                EOUT          WAVCK         CC
CC                                GOSTP         YESNO         CC
CC          < FROM TEKLIB >:      CC
CC                                ERASE          CC
CC
CC          MEMORY OCCUPIED:      7 PAGES          CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          SUBROUTINE SCAN
          COMMON WSCNT,WSPA,UPLIM,MWATE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C          GET STARTING WAVELENGTH, CHECK UPPER, LOWER LIMITS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10      READ (1,1000) RLO
          IF (RLO) 500,15,15
15      CALL WAVCK (50.,RLO,UPLIM)
          IF (RLO) 10,10,40
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C          GET ENDING WAVELENGTH, CHECK LIMITS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
40      READ(1,1001) RHI
          IF (RHI) 500,45,45
45      CALL WAVCK (RLO,RHI,UPLIM)
          IF (RHI) 40,40,65
65      SLO = RLO * WSPA
          SHI = RHI * WSPA
          CALL DFIX (SLO,ISH,ISL)
          CALL DFLOT (ISH,ISL,SLO)
          CALL DFIX (SHI,ISH,ISL)
          CALL DFLOT (ISH,ISL,SHI)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C          GET DESIRED SCAN RATE, CHECK FOR LIMITS, AND CALCULATE ACTUAL
C          SCAN RATE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
          READ (1,1002) SR
          IF (SR) 500,67,67
67      WATES = 6000000. / (SR * WSPA)
          IF (WATES - FLOAT(MWATE)) 70,75,75
70      SR = 6000000. / (FLOAT(MWATE) * WSPA)
          WRITE (1,1003) SR
          GO TO 65
75      IF (WATES - 4096.) 85,85,80
80      SR = 6000000. / (4096. * WSPA)
          WRITE (1,1004) SR
          GO TO 65
85      CALL DFIX (WATES,ISH,MWATE)
          CALL DFLOT (ISH,MWATE,WATES)

```

SCAN.FT, CONT'D.

```

      SR = 6000000. / (WATES * WSPA)
      RLO = SLO / WSPA
      RHI = SHI / WSPA
      RMIN = (WATES*WSPA*(RHI-RLO)) / 6000000.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      OUTPUT ACTUAL SCAN PARAMETERS AND CHECK TO SEE IF THEY'RE OKAY.
C      IF SO, MOVE TO INITIAL WAVELENGTH
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL ERASE
      CALL POSIT (0,0)
      WRITE(1,1005) RLO,RHI,SR,RMIN
      CALL YESNO(IDUM)
      IF (IDUM - 3) 87,86,87
86      CALL EXIT
87      IF (IDUM - 7) 88,500,88
88      IF (IDUM - 89) 10,100,10
100     CALL MOVE (SLO)
      WRITE (1,1006) IDUM
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WHEN READY, A CR STARTS SCAN.  AT CONCLUSION CHECK TO SEE IF
C      AN IDENTICAL SCAN SHOULD BE REPEATED.  IF NOT, RETURN TO
C      MAIN PROGRAM.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL LETIN (IDUM)
      IF (IDUM - 3) 101,86,101
101     IF (IDUM - 7) 105,500,105
1009    FORMAT(' HIT RETURN TO START SCAN',12)
105     STPS = SHI - SLO
      CALL DFIX(STPS,ISH,ISL)
      CALL GOSTP (0,ISH,ISL,NWATE)
      WSCNT = SHI
      CALL FOUT
500     RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      FORMAT STATEMENTS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1000    FORMAT('LOWER WAVELENGTH = ',F10.4)
1001    FORMAT('UPPER WAVELENGTH = ',F10.4)
1002    FORMAT('SCAN RATE [A/MIN] = ',F10.4)
1003    FORMAT('MAXIMUM SCAN RATE IS ',F10.4,' ANGSTROMS / MINUTE')
1004    FORMAT('MINIMUM SCAN RATE IS ',F10.4,' ANGSTROMS / MINUTE')
1005    FORMAT('ACTUAL SCAN WILL BE: '/5X,'FROM 'F10.4,' A',/
17X,'TO ',F10.4,' A'/7X,'AT ',F10.5,' A/MIN'/' ELAPSED
2TIME ABOUT ',F12.4,' MINUTES'/' IS THAT OKAY?')
1006    FORMAT('<CR> TO START -> ',10)
      END
I

```

```

//// SETDV.SB //// 4/7/77 //////////////////////////////////////
////
//      SUBROUTINE SETDV.SB      4/7/77      E.J. DARLAND      //
//
//      PURPOSE:
//          TO SET A DEVICE NAME TO THE EQUIVALENT OF 'TTY'.
//
//      USAGE:  CALL SETDV (DEV)
//

```

SETDV.SB, CONT'D.

```

//      PARAMETER DESCRIPTION:      //
//      DEV =      THE FORTRAN  VARIABLE WHICH IS TO BE SET      //
//      SO THAT ITS A6 FORMAT EQUIVALENT IS 'TTY'.      //
//      //      //
////////////////////////////////////
      OPDEF ITAD 1400
      OPDEF IDCA 3400
////////////////////////////////////
/
      ENTRY BLOCK FOR ARGUMENT PASSAGE
/
////////////////////////////////////
      ENTRY SETDV
SETDV,  BLOCK 2
////////////////////////////////////
/
      GET ARGUMENT DATA FIELD AND ADDRESS
/
////////////////////////////////////
      CLA CLL
      TAD SETDV      /GET CDF MAIN
      DCA CDFM
CDFM,  HLT      /BY NOW CDF MAIN
      ITAD SETDV#    /GET CDF ARGUMENT
      DCA CDFA      /AND STORE
      INC SETDV#
      ITAD SETDV#    /GET ADDRESS OF ARGUMENT
      DCA ADDR      /AND STORE
////////////////////////////////////
/
      PASS BACK TTY@@@ INTO ARGUMENT
/
////////////////////////////////////
CDFA,  HLT      /BY NOW CDF ARGUMENT
      TAD A      /GET FIRST WORD OF NEW ARG VALUE
      IDCA ADDR  /AND DEPOSIT
      INC ADDR   /INCREMENT POINTER
      TAD B      /GET SECOND WORD
      IDCA ADDR  /DEPOSIT
      INC ADDR   /INCREMENT AGAIN
      TAD C      /GET FINAL WORD
      IDCA ADDR  /AND DEPOSIT
      AND I CDFA  /GET SABR TO CDF CURRENT
      RETRN SETDV /AND RETURN TO MAIN PROGRAM
A,      2424      /ASCII TT
B,      3100      /ASCII Y@
C,      0000      /ASCII @@
ADDR,   0
      END

```

```

CCCC STOR.FT CCCC 1/23/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      SUBROUTINE STOR.FT      1/23/77      E.J. DARLAND      CC
CC
CC      PURPOSE:      CC
CC      TO PROVIDE STORAGE AND RETRIEVAL OF DATA FOR ACQUIR.      CC
CC      THE 3300 DECIMAL STORAGE LOCATIONS NEEDED TO STORE 550      CC
CC      PHOTON AND 550 ION POINTS CAN NOT FIT IN THE SAME FIELD AS      CC
CC      THE REST OF ACQUIR. THIS PROGRAM IS CALLED TO ACQUIRE THE      CC
CC      POINTS AND STORE THEM, AND ALSO TO PASS THEM BACK TO THE      CC
CC      MAIN PROGRAM VIA COMMON.      CC
CC

```


STOR.FT, CONT'D.

```

CC      USAGE:  CALL STOR (IMODE)                                CC
CC
CC      PARAMETER DESCRIPTION:                                    CC
CC      IMODE:   DIRECTION OF TRANSFER FLAG:                     CC
CC      0: ACQUIRE DATA AND STORE IN RPHO AND RION.            CC
CC      1: STORE RPHO(IPT) AND RION(IPT) IN CNTP                 CC
CC      AND CNTI.                                                 CC
CC
CC      INTERNAL VARIABLES:                                       CC
CC      RPHO:    ARRAY FOR STORAGE OF 550 PHOTON COUNT           CC
CC      DATA.                                                 CC
CC      RION:    ARRAY FOR STORAGE OF 550 ION COUNT DATA        CC
CC
CC      SUBROUTINES CALLED:   KCHEK                               CC
CC
CC      MEMORY OCCUPIED:     33 PAGES                             CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SUBROUTINE STOR (IMODE)
COMMON I1,IPT,I2,CNTP,CNTI,TIMEP,NPTS,ITIM
DIMENSION I1(20),I2(6),RPHO(550),RION(550)
10  IF (IMODE) 20,20,30
20  DO 25 I=1,NPTS
    CALL COUNT (ITIM,CNTP,CNTI)
    RPHO(I) = CNTP
    RION(I) = CNTI
    CALL KCHEK (IDUM)
    IF (IDUM) 25,25,26
25  CONTINUE
26  RETURN
30  CNTP = RPHO (IPT)
    CNTI = RION (IPT)
    RETURN
    END

```

```

CCCC TCALC.FT CCCC 4/11/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      SUBROUTINE TCALC.FT      V2      4/11/77      E.J. DARLAND      CC
CC
CC      PURPOSE:                                                         CC
CC      TO CALCULATE THE APPROXIMATE TIME IT TAKES TO MOVE              CC
CC      FROM ONE WAVELENGTH TO ANOTHER.                                  CC
CC
CC      USAGE: CALL TCALC (TIME,DSCNT)                                    CC
CC
CC      PARAMETER DESCRIPTION:                                           CC
CC      TIME:   RETURNED AS THE CALCULATED TIME, IN SECONDS.            CC
CC      DSCNT:  THE STEP COUNT TO WHICH A MOVE IS BEING                 CC
CC      CONTEMPLATED. TCALC CALCULATES THE TIME TO GO FROM WSCNT TO    CC
CC      DSCNT.                                                            CC
CC
CC      MEMORY OCCUPIED:         3 PAGES                                  CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SUBROUTINE TCALC (TIME,DSCNT)
COMMON WSCNT,WSPA,UPLIM,MWATE,IACCL
STEPS = DSCNT - WSCNT
IF (IACCL) 10,10,50

```

TCALC.FT, CONT'D.

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      NO ACCELERATION WILL BE USED.  CHECK DIRECTION.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10      IF (STEPS) 30,20,40
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MOVE IS 0 STEPS, TIME = 0 SECONDS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
20      TIME = 0.
25      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MOVE IS TO LOWER WAVELENGTH, AND WILL GO 600 TOO FAR, DELAY,
C      AND THEN RETURN TO THE DESIRED WAVELENGTH.  TOTAL NUMBER OF
C      STEPS WILL THEN BE STEPS + 1200, AND TOTAL TIME WILL INCLUDE
C      THE TIME NECESSARY FOR THOSE STEPS PLUS THE TURNAROUND WAIT
C      TIME OF 0.2 SECONDS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
30      TIME = (((1200. - STEPS) * FLOAT(MWATE)) / 100000.) + 0.20
      GO TO 25
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MOVE IS TO HIGHER WAVELENGTHS AND IS SIMPLY THE SUM OF THE
C      WAITS FOR ALL STEPS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
40      TIME = (STEPS * FLOAT(MWATE)) / 100000.
      GO TO 25
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      ACCELERATION IS POSSIBLE.  DETERMINE DIRECTION.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
50      IF (STEPS) 60,20,70
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MOVE IS TOWARD LOWER WAVELENGTHS AND WILL DEFINITELY ACCELERATE
C      DUE TO THE EXTRA 600 STEPS.  THE BACKWARDS MOVE WILL BE
C      STEPS + 600 STEPS LONG, AND WILL TAKE (STEPS + 600)*MWATE
C      WAIT PERIODS PLUS THE ADDITIONAL 2048 WAIT PERIODS INTRODUCED BY
C      THE ACCELERATION.  THERE WILL THEN BE A 0.2 SECOND TURNAROUND
C      WAIT, AND THE 600 STEP FORWARD MOVE, MADE AT 1/5 THE MAXIMUM
C      RATE.  SO THE TOTAL NUMBER OF WAIT PERIODS IS:
C
C      WAITS = ((STEPS + 600) * MWATE) + 2048 + (600 * 5 * MWATE)
C
C      AND THE TOTAL TIME IS THE TIME FOR THOSE WAITS PLUS
C      THE 0.2 SECONDS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
60      TIME = (((3600. + STEPS) * FLOAT(MWATE)) + 2048.) / 100000.
      TIME = TIME + 0.2
      GO TO 25
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      THE MOVE IS FORWARD.  THE RATE DEPENDS ON WHETHER IT IS FAR
C      ENOUGH TO ACCELERATE (>510 STEPS) OR NOT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
70      IF (STEPS - 510.) 80,80,90
80      TIME = (STEPS * 5. * FLOAT(MWATE)) / 100000.
      GO TO 25
90      TIME = ((STEPS * FLOAT(MWATE)) + 2048.) / 100000.
      GO TO 25
      END

```

TRANS.FT

```

CCCC TRANS.FT CCCC 4/6/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      PROGRAM TRANS.FT      V2B      4/6/77      E.J. DARLAND      CC
CC      PURPOSE:      CC
CC      TO TRANSFER PIMS8 FORMAT FILES FROM ONE DEVICE OR FILE CC
CC      TO ANOTHER. THE OUTPUT FORMAT, TYPE OF INFORMATION TO BE CC
CC      TRANSFERRED, AND WAVELENGTH OR TIME LIMITS FOR TRANSFERRED CC
CC      DATA CAN BE SELECTED BY THE OPERATOR (SEE CHAPTER V). CC
CC      USAGE: TRANS IS CHAINED INTO CORE FROM PIMS8 WHENEVER CC
CC      WHENEVER THE TRANSFER OPTION IS CHOSEN. ITS USE IS CC
CC      DESCRIBED IN CHAPTER V. CC
CC      INTERNAL VARIABLES: CC
CC      DEVI: INPUT DEVICE NAME CC
CC      DEV0: OUTPUT DEVICE NAME CC
CC      FILI: INPUT FILE NAME CC
CC      FILO: OUTPUT FILE NAME CC
CC      IBACK: BACKGROUND TRANSFER FLAG: CC
CC      0: TRANSFER BACKGROUND DATA CC
CC      1: DON'T TRANSFER BACKGROUND DATA CC
CC      IDAT: PHOTON/ION INTENSITY TRANSFER FLAG CC
CC      0: TRANSFER DATA CC
CC      1: DON'T TRANSFER DATA CC
CC      IDARK: PHOTON/ION DARK MEASUREMENT TRANSFER FLAG CC
CC      0: TRANSFER DATA CC
CC      1: DON'T TRANSFER DATA CC
CC      IFLG: TEMPORARY INTEGER STORAGE CC
CC      IFORM: FORMAT FOR OUTPUT: CC
CC      1: ASCII OUTPUT CC
CC      2: PIMS8 FORMAT OUTPUT CC
CC      IHEAD: HEADINGS TRANSFER FLAG: CC
CC      0: TRANSFER HEADINGS CC
CC      1: DON'T TRANSFER HEADINGS CC
CC      IREF: REFERENCE TRANSFER FLAG: CC
CC      0: TRANSFER REFERENCE DATA CC
CC      1: DON'T TRANSFER REFERENCES CC
CC      IUNIT: UNITS OF LIMITS: CC
CC      1: ANGSTROMS (WAVELENGTH) CC
CC      2: SECONDS (TIME) CC
CC      YMAX: UPPER LIMIT CC
CC      XMIN: LOWER LIMIT CC
CC      SUBROUTINES CALLED: FILE, KCHEK, RTEXT CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
COMMON IJ, ITEK, IK
DIMENSION IJ(14), IK(6)
IUNIT = 1
IFORM = 1
IHEAD = 1
IDAT = 1
IREF = 1
IBACK = 1
IDARK = 1
IINF = 0
CALL SETDV(DEV0)
FILO = 0.0
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      WRITE OPTIONS, PROMPT. READ OPTION AND BRANCH AS APPROPRIATE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1      IF (ITEK) 10,10,5
5      CALL ERASE

```

TRANS.FT, CONT'D.

```

      CALL POSIT (0,0)
10    WRITE (1,2000)
14    IOUTF = 1
      CALL SETDV(DEVO)
      FILO = 0.0
15    CALL BELL
      WRITE (1,2001) IOPT,
      CALL LETIN (IOPT)
      IF (IOPT - 3) 25,20,25
20    CALL EXIT
25    IF (IOPT - 7) 30,10,30
30    IF (IOPT - 344) 35,500,35
35    IF (IOPT - 590) 40,100,40
40    IF (IOPT - 981) 45,150,45
45    IF (IOPT - 399) 50,200,50
50    IF (IOPT - 1305) 55,250,55
55    IF (IOPT - 1033) 60,65,60
60    IF (IOPT - 777) 1,300,1
65    CALL CHAIN ('PIMS8')
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      INPUT.  READ IN DEVICE AND FILE SPECIFICATIONS.  IF
C      SUCCESSFUL, SET THE INPUT FILE FLAG SO EXECUTE WILL KNOW THE
C      INPUT HAS BEEN PROPERLY SPECIFIED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
100    IOPT = 1
      CALL FILE (DEVI,FILO,IOPT)
      IOPT = IOPT + 3
      GO TO (105,20,100,110) IOPT
105    IINF = 0
      GO TO 15
110    IINF = 1
      GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      OUTPUT.  READ IN DEVICE AND FILE SPECIFICATIONS AND
C      SET FLAG AS APPROPRIATE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
150    IOPT = 0
      CALL FILE (DEVO,FILO,IOPT)
      IOPT = IOPT + 3
      GO TO (155,20,150,160) IOPT
155    IOUTF = 0
      GO TO 15
160    IOUTF = 1
      GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      FORMATS.  DECIDE ON ASCII OR PACKED ASCII FORMAT FOR OUTPUT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
200    WRITE (1,2002) IOPT,
      CALL YESNO (IOPT)
      IF (IOPT - 3) 210,20,210
210    IF (IOPT - 7) 215,15,215
215    IF (IOPT - 89) 220,225,220
220    IFORM = 2
      GO TO 15
225    IFORM = 1
      GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      TYPES.  DECIDE ON THE TYPES OF DATA TO BE TRANSFERRED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
250    WRITE (1,2003)
      WRITE (1,2004)

```

TRANS.FT, CONT'D.

```

      IHEAD = 0
      IDARK = 0
      IBACK = 0
      IREF = 0
      IDAT = 0
255    READ (1,2005) IOPT
      IF (IOPT) 250,15,260
260    IF (IOPT - 6) 265,265,250
265    GO TO (270,275,280,285,290,295) IOPT
270    IHEAD = 1
      GO TO 255
275    IDARK = 1
      GO TO 255
280    IBACK = 1
      GO TO 255
285    IREF = 1
      GO TO 255
290    IDAT = 1
      GO TO 255
295    IHEAD = 1
      IDARK = 1
      IBACK = 1
      IREF = 1
      IDAT = 1
      GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      LIMITS.  DECIDE ON WAVELENGTH OR TIME LIMITS, THEN INPUT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
300    READ (1,2006) IUNIT
      IF (IUNIT) 15,300,305
305    IF (IUNIT - 3) 310,310,300
310    GO TO (15,315,315) IUNIT
315    READ (1,2007) XMIN
      READ (1,2008) XMAX
      GO TO 15
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      EXECUTE.  FIRST CHECK TO MAKE SURE INPUT AND OUTPUT FILE
C      SPECIFICATIONS ARE OKAY.  THEN OPEN FILES.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
500    IF (IINF) 506,506,505
505    IF (IOUTF) 507,507,510
506    WRITE (1,2030) IOPT,
      GO TO 100
507    WRITE (1,2031) IOPT,
      GO TO 150
510    CALL IOPEN (DEVI, FILI)
      CALL OOPEN (DEVO, FILO)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CHECK FOR OPERATOR INTERRUPT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
520    CALL KCHEK (IOPT)
      IF (IOPT) 540,540,525
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      INTERRUPT.  CLOSE ANY OPEN FILES.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
525    IREC = 99
      GO TO (530,535) IFORM
530    WRITE (4,2009) IREC
      GO TO 537
535    WRITE (4,2010) IREC
537    CALL OCLOSE

```

TRANS.FT, CONT'D.

```

      GO TO 10
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      NO INTERRUPT.  READ LINE CODE AND CHECK FOR 98,99
C      (COMMENT OR EOF).
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
540      READ (4,2010) IREC,
          IF (IREC - 99) 545,525,545
545      IF (IREC - 98) 565,550,565
550      IF (IHEAD) 560,560,555
555      CALL RTEXT (4)
          GO TO 520
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SKIP REST OF THE LINE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
560      READ (4,2011) IREC
          GO TO 520
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CHECK FOR OTHER LEGAL LINE CODES.  IF FOUND, CHECK TO
C      SEE IF THAT DATA TYPE IS TO BE TRANSFERRED, AND IF SO
C      BRANCH TO THE SECTION THAT DOES THE WORK.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
565      IF (IREC - 1) 560,700,570
570      IF (IREC - 3) 705,705,575
575      IF (IREC - 4) 560,710,580
580      IF (IREC - 8) 585,720,585
585      IF (IREC - 17) 590,715,590
590      IF (IREC - 21) 600,600,560
600      IF (IHEAD) 560,560,605
605      GO TO (700,705,705,710,900,925,950,720,975,1000,1025,1050
          1,1075,1100,1125,1150,715,1175,1200,1225,1250) IREC
700      IF (IDAT) 560,560,890
705      IF (IDAT) 560,560,850
710      IF (IDARK) 560,560,850
715      IF (IREF) 560,560,850
720      IF (IBACK) 560,560,850
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      AT LAST!  THE LITTLE SECTIONS THAT ACTUALLY DO THE READING AND
C      WRITING HAVE MADE THEIR APPEARANCE.  THE NUMBERS IN THE COMMENT
C      HEADINGS FOR EACH SECTION ARE THE LINE CODES WHICH ARE
C      APPLICABLE FOR THAT SECTION.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCC
C
C      1
C
CCCCCCCCCC
800      READ (4,2013) A,B,C,D,E
          GO TO (825,805,810) IUNIT
805      X = A
          GO TO 815
810      X = D
815      IF (X - XMAX) 820,820,520
820      IF (X - XMIN) 520,825,825
825      GO TO (830,835) IFORM
830      WRITE (4,2009) IREC,A,B,C,D,E
          GO TO 520
835      WRITE (4,2012) IREC, A,B,C,D,E
          GO TO 520

```

TRANS.FT, CONT'D.

CCCCCCCCC

C

C 2,3,4,8,17

C

CCCCCCCCC

850 READ (4,2013) A,B,C,D
 GO TO (875,855,860) IUNIT

855 X = A
 GO TO 865

860 X = C
 IF (IREC - 8) 865,875,865

865 IF (X - XMAX) 870,870,520

870 IF (X - XMIN) 520,875,875

875 GO TO (880,835) IFORM

880 WRITE(4,2009) IREC,A,B,C,D
 GO TO 520

885 WRITE(4,2012) IREC,A,B,C,D
 GO TO 520

CCCCCCCCC

C

C 5

C

CCCCCCCCC

900 READ (4,2013) A
 GO TO (905,910) IFORM

905 WRITE(4,2014) A
 GO TO 520

910 WRITE(4,2012) IREC,A
 GO TO 520

CCCCCCCCC

C

C 6

C

CCCCCCCCC

925 READ (4,2010) IA,IB,IC
 GO TO (930,935) IFORM

930 WRITE(4,2015) IA,IB,IC
 GO TO 520

935 WRITE(4,2010) IREC,IA,IB,IC
 GO TO 520

CCCCCCCCC

C

C 7

C

CCCCCCCCC

950 READ (4,2010) IA
 GO TO (955,960) IFORM

955 WRITE(4,2016) IA
 GO TO 520

960 WRITE(4,2010) IREC,IA
 GO TO 520

CCCCCCCCC

C

C 9

C

CCCCCCCCC

975 READ (4,2013) START,FINSH
 GO TO (980,985) IFORM

980 WRITE(4,2017) START,FINSH
 GO TO 520

985 WRITE(4,2012) IREC,START,FINSH
 GO TO 520

CCCCCCCCC

C

C 10

C

CCCCCCCCC

1000 READ (4,2013) A
 GO TO (1005,1010) IFORM

TRANS.FT, CONT'D.

```

1005    WRITE(4,2018) A
        GO TO 520
1010    WRITE(4,2012) IREC,A
        GO TO 520
CCCCCCCCC
C
C      11
C
CCCCCCCCC
1025    READ (4,2013) A,B,C
        GO TO (1030,1035) IFORM
1030    WRITE(4,2019) ,A,B,C
        GO TO 520
1035    WRITE(4,2012) IREC,A,B,C
        GO TO 520
CCCCCCCCC
C
C      12
C
CCCCCCCCC
1050    READ (4,2013) A,B
        GO TO (1055,1060) IFORM
1055    WRITE(4,2020) A,B
        GO TO 520
1060    WRITE(4,2012) IREC,A,B
        GO TO 520
CCCCCCCCC
C
C      13
C
CCCCCCCCC
1075    READ (4,2010) IA
        GO TO (1080,1085) IFORM
1080    WRITE(4,2021) IA
        GO TO 520
1085    WRITE(4,2010) IREC,IA
        GO TO 520
CCCCCCCCC
C
C      14
C
CCCCCCCCC
1100    READ (4,2013) A,B
        GO TO (1105,1110) IFORM
1105    WRITE(4,2022) A,B
        GO TO 520
1110    WRITE(4,2012) IREC,A,B
        GO TO 520
CCCCCCCCC
C
C      15
C
CCCCCCCCC
1125    READ (4,2013) A,B
        GO TO (1130,1135) IFORM
1130    WRITE(4,2023) A,B
        GO TO 520
1135    WRITE(4,2012) IREC,A,B
        GO TO 520
CCCCCCCCC
C
C      16
C
CCCCCCCCC
1150    READ (4,2010) IA
        GO TO (1155,1160) IFORM
1155    WRITE(4,2024) IA
        GO TO 520
1160    WRITE(4,2010) IREC,IA

```


TRANS.FT, CONT'D.

```

                GO TO 520
CCCCCCCCCCCC
C
C          18
C
CCCCCCCCCCCC
1175      READ (4,2013) A
          GO TO (1180,1185) IFORM
1180      IF (START - FINSH) 1181,1182,1181
1181      WRITE(4,2025) A
          GO TO 520
1182      WRITE (4,2032) A
          GO TO 520
1185      WRITE(4,2012) IREC, A
          GO TO 520
CCCCCCCCCCCC
C
C          19
C
CCCCCCCCCCCC
1200      READ (4,2013) A,B,C
          GO TO (1205,1210) IFORM
1205      WRITE (4,2026) A,B,C
          GO TO 520
1210      WRITE (4,2012) IREC,A,B,C
          GO TO 520
CCCCCCCCCCCC
C
C          20
C
CCCCCCCCCCCC
1225      READ (4,2010) IA, IB
          GO TO (1230,1235) IFORM
1230      WRITE (4,2027) IA, IB
          GO TO 520
1235      WRITE (4,2010) IREC, IA, IB
          GO TO 520
CCCCCCCCCCCC
C
C          21
C
CCCCCCCCCCCC
1250      READ (4,2013) A,B
          IF (A - XMAX) 1255,1255,520
1255      IF (X - XMIN) 520,1260,1260
1260      GO TO (1265,1270) IFORM
1265      WRITE (4,2029) A,B
          GO TO 520
1270      WRITE (4,2012) IREC,A,B
          GO TO 520
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C          FORMAT STATEMENTS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2000      FORMAT('EXECUTE FORMATS INPUT LIMITS OUTPUT PIMS TYPES')
2001      FORMAT('-> ',I0)
2002      FORMAT('ASCII FORMAT? ',I0)
2003      FORMAT(' 1      2      3      4      5      6')
2004      FORMAT('HEADING DARK BACKGRND REFERENCE DATA ALL')
2005      FORMAT(': ',I2)
2006      FORMAT('NONE [1], ANGSTROMS [2] OR SECONDS [3] : ',I2)
2007      FORMAT('XMIN = ',F15.7)
2008      FORMAT('XMAX = ',F15.7)
2009      FORMAT(I2,1X,E15.8,4(1X,E15.8))
2010      FORMAT(5A2)
2011      FORMAT(I0)
2012      FORMAT(A2,5A6)
2013      FORMAT(5A6)

```

TRANS.FT, CONT'D.

```

2014  FORMAT(10X,'ORIGINAL FILENAME: ',A6,'.DA')
2015  FORMAT(23X,'DATE: ',I2,'/',I2,'/197',I1)
2016  FORMAT(16X,'SCAN NUMBER: ',I3)
2017  FORMAT(8X,'STARTING WAVELENGTH: ',F9.3,' ANGSTROMS'/10X,'END
      1ING WAVELENGTH: ',F9.3,' ANGSTROMS')
2018  FORMAT(7X,'WAVELENGTH INCREMENT: ',F9.3,' ANGSTROMS')
2019  FORMAT('MANDANTORY INTEGRATION TIME: ',F5.1,' SECONDS'/3X,
      1'MAXIMUM INTEGRATION TIME: 'F5.1,' SECONDS'/8X,'DESIRED TOTAL
      2 COUNT: ',F9.0)
2020  FORMAT(10X'PHOTON FULL SCALE: 'F9.0,' CPS'/13X,'ION FULL
      1 SCALE: ',F9.0,' CPS')
2021  FORMAT(1X,'ONE DARK MEASUREMENT EVERY ',I4,' DATA POINTS')
2022  FORMAT('      DARK INTEGRATION TIME: ',F5.1,' SECONDS'/1X,
      1'ELECTROMETER SETTLING TIME: ',F5.1,' SECONDS')
2023  FORMAT(7X,'REFERENCE WAVELENGTH: ',F9.4,' ANGSTROMS'/1X,
      1'REFERENCE INTEGRATION TIME: ',F5.1,' SECONDS')
2024  FORMAT(2X,'ONE REFERENCE POINT EVERY ',I4,' DATA POINTS')
2025  FORMAT(6X,'REAL-TIME PLOT SCALED: ',F9.2,' ANGSTROMS/INCH'//)
2026  FORMAT('FIRST BACKGROUND WAVELENGTH: ',F9.4,'ANGSTROMS'/1X
      1,'LAST BACKGROUND WAVELENGTH: ',F9.4,'ANGSTROMS'/'BACKGROUND
      2INTEGRATION TIME: 'F5.1' SECONDS')
2027  FORMAT(7X,'PHOTONS PRESCALED BY: ',I5,'/10X,'IONS PRESCALED BY:
      1 ',I5)
2028  FORMAT(2(E15.3))
2029  FORMAT(2(1X,E15.8))
2030  FORMAT('FROM ',I0)
2031  FORMAT(' TO ',I0)
2032  FORMAT(6X,'REAL-TIME PLOT SCALED: ',F9.2,' POINTS / INCH'//)
      END

```

```

CCCC TRUNC.FT CCCC 2/21/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      FUNCTION TRUNC.FT          2/21/77          E.J. DARADND          CC
CC                                                                 CC
CC      PURPOSE:                                                                 CC
CC      TO ELIMINATE REPETITIVE CALLS TO DFIX AND FLOTT WHERE              CC
CC      THE PURPOSE IS MERELY TO TRUNCATE THE REAL VARIABLE TO THE         CC
CC      NEAREST INTEGER CONSISTENT WITH THE RANGE LIMITATIONS OF           CC
CC      DFIX AND DFLOT.                                                                 CC
CC                                                                 CC
CC      USAGE:  RVAL = TRUNC (RVAL)                                          CC
CC               WHERE RVAL IS TRUNCATED AND RETURNED                      CC
CC                                                                 CC
CC      SUBROUTINES CALLED:          DFIX          DFLOT                    CC
CC                                                                 CC
CC      MEMORY OCCUPIED:  1 PAGE                                            CC
CC                                                                 CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      FUNCTION TRUNC (RVAL)
      CALL DFIX (RVAL, IH, IL)
      CALL DFLOT (IH, IL, RTEMD)
      TRUNC = RTEMD
      RETURN
      END

```

WAVCK.FT

```

CCCC WAVCK.FT CCCC 1/31/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      SUBROUTINE WAVCK.FT      1/31/77      E.J. DARLAND      CC
CC
CC      PURPOSE:      CC
CC      THIS SUBROUTINE WILL CHECK TO MAKE SURE AN INPUT      CC
CC      WAVELENGTH IS IN A REGION THAT THE MONOCHROMATOR CAN REACH, CC
CC      AND THAT IT IS WITHIN THE LIMITS SET BY THE CALLING PROGRAM. CC
CC      IF THE WAVELENGTH IS LESS THAN 500 ANGSTROMS, WAVCK      CC
CC      WILL ASK IF THE OPERATOR IS SURE, SINCE IT IS LIKELY SUCH      CC
CC      A LOW WAVELENGTH IS A MISTAKE. IF THE WAVELENGTH IS BELOW      CC
CC      EITHER 0.0 ANGSTROMS OR THE LOWER LIMIT SET BY THE CALLING      CC
CC      PROGRAM, THE ERROR MESSAGE "X.X ANGSTROMS TOO LOW" IS      CC
CC      PRINTED. IF THE WAVELENGTH IS HIGHER THAN EITHER UPLIM OR      CC
CC      THE UPPER LIMIT SET BY THE CALLING PROGRAM, THE MESSAGE      CC
CC      "X.X ANGSTROMS TOO HIGH" IS PRINTED.      CC
CC
CC      USAGE: CALL WAVCK (WLOW,WDES,WHIGH)      CC
CC
CC      PARAMETER DESCRIPTION:      CC
CC      WLOW: THE LOWER WAVELENGTH LIMIT SET BY THE      CC
CC      CALLING PROGRAM.      CC
CC      WDES: THE DESIRED WAVELENGTH. RETURNED AS -1.0      CC
CC      IF AN ERROR IS FOUND.      CC
CC      WHIGH: THE UPPER WAVELENGTH LIMIT SET BY THE      CC
CC      CALLING PROGRAM.      CC
CC
CC      SUBROUTINES CALLED: YESNO.SB      CC
CC
CC      MEMORY OCCUPIED: 3 PAGES      CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SUBROUTINE WAVCK (WLOW,WDES,WHIGH)
COMMON R1,R2,UPLIM
IF (WDES - 500.) 10,20,20
10 WRITE (1,2000) WDES,IOPT,
CALL YESNO (IOPT)
IF (IOPT - 89) 15,20,15
15 WDES = -1.
RETURN
20 IF (WDES - WLOW) 25,30,30
25 WRITE (1,2001) WDES
GO TO 15
30 IF (WDES) 15,35,35
35 IF (WDES - UPLIM) 45,45,40
40 WRITE (1,2002) WDES
GO TO 15
45 IF (WDES - WHIGH) 50,50,40
50 RETURN
2000 FORMAT(E15.8,' ANGSTROMS?? ',I0)
2001 FORMAT(E15.8,' ANGSTROMS TOO LOW!')
2002 FORMAT(E15.8,' ANGSTROMS TOO HIGH!')
END

```

```

CCCC WTEXT.FT CCCC 4/18/76 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      SUBROUTINE WTEXT.FT      4/18/76      E.J. DARLAND      CC
CC
CC      PURPOSE:      CC
CC      TO TRANSFER ASCII CODE FROM THE KEYBOARD TO THE DEVICE CC
CC      CURRENTLY OPEN FOR OS/8 FORTRAN II DEVICE-INDEPENDENT      CC
CC      OUTPUT. THE ASCII INFORMATION IS OUTPUT IN PIMS8 FORMAT,      CC
CC

```

WTEXT.FT, CONT'D.

```

CC      IE., A 98 COMMENT CODE WRITTEN IN A2 FORMAT, UP TO 72 CHAR- CC
CC      ACTERS OF ASCII CODE, WRITTEN IN A1 FORMAT, AND AN '@' END CC
CC      OF LINE CHARACTER. CC
CC      USAGE: CALL WTEXT CC
CC      WTEXT WILL IMMEDIATLY START ACCEPTING CHARACTERS FROM CC
CC      THE KEYBOARD. IT INCORPORATES A SIMPLE LINE EDITOR WHICH CC
CC      RECOGNIZES THE FOLLOWING SPECIAL CHARACTERS: CC
CC
CC      RUBOUT: DELETE LAST CHARACTER ENTERED CC
CC      ^U:      DELETE ENTIRE LINE SO FAR ENTERED CC
CC      LF:      ECHO CRLF, TYPE OUT LINE SO FAR ENTERED CC
CC      CR:      END LINE AND WRITE TO OUTPUT DEVICE CC
CC      ^G:      END OF COMMENTS. WRITE LAST LINE AND CC
CC      RETURN TO CALLING PROGRAM. CC
CC
CC      ***** NOTE: ^C NOT RECOGNIZED IN WTEXT ***** CC
CC
CC      LINES ARE LIMITED TO 72 CHARACTERS, PLUS THE 98 CC
CC      COMMENT CODE AND THE @ END OF LINE CHARACTER. AFTER THE CC
CC      72ND CHARACTER IS TYPED, WTEXT WILL NOT RESPOND TO ANY CC
CC      FURTHER CHARACTERS, EXCEPT FOR THE ABOVE CONTROL CHARACTERS. CC
CC
CC      MEMORY OCCUPIED: 3 PAGES CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SUBROUTINE WTEXT
DIMENSION IBUF (75)
S      CLA CLL
S      TAD (0142          /98, DECIMAL
S      DCA \ICHA
IBUF (2) = ICHAR
10     ICNT = 3
SA,    CLA CLL          /READ A CHARACTER
SB,    KSF
S      JMP B
S      KRB
S      AND (0177          /STORE AS 7-BIT ASCII
S      DCA \ICHA
S      TAD \ICHA
S      TAD (7771          /-7OCTAL, CHECK FOR ^G (ASCII:007)
S      SNA
S      JMP BELL
S      TAD (7775          /-3; CHECK FOR LF (ASCII:012)
S      SNA
S      JMP LF
S      TAD (7775          /-3; CHECK FOR CR (ASCII:015)
S      SNA
S      JMP CR
S      TAD (7770          /-10; CHECK FOR ^U (ASCII:025)
S      SNA
S      JMP AGAIN
S      TAD (7626          /-152; CHECK FOR RUBOUT (ASCII:177)
S      SNA
S      JMP ERASE
S      CLA CLL          /THE CHARACTER IS NOT A SPECIAL
S      TAD \ICNT
S      TAD (7667          /-73 DECIMAL
S      SNA              /IS ICNT < 73?
S      JMP A              /NO: DO NOTHING BUT LOOK FOR A SPECIAL.
S      CLA CLL          /YES: ECHO THE CHARACTER
S      TAD \ICHA
S      JMS TYPE
S      TAD \ICHA          /CONVERT TO 6BIT, UPPER HALF OF WORD
S      AND (0077
S      RTL; RTL; RTL
S      DCA \ICHA
S      IBUF (ICNT) = ICHAR

```

WTEXT.FT, CONT'D.

```

S      INC \ICNT                / HAVE STORED IN BUFFER, INCREMENTED
S      JMP A                    /BUFFER POINTER. GET ANOTHER CHARACTER.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SECTION TO HANDLE A CONTROL C.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SBELL,  CLA CLL                /STORE ASCII @ IN LAST CHARACTER OF THE
S      DCA \ICHAR              /LINE, THEN WRITE THE BUFFER TO THE OUT-
S                                  /PUT DEVICE.

      IBUF (ICNT) = ICHAR
      WRITE(4,1000) (IBUF(1), I = 2, ICNT)
S      CLA CLL
S      TAD (0207                /ECHO BELL
S      JMS TYPE
S      JMS CRLF                /ECHO CRLF AND RETURN TO MAIN PROGRAM.
      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SECTION TO HANDLE LINE FEED
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SLF,    JMS CRLF                /ECHO CRLF
S      TAD \ICNT                /ONLY WANT TO ECHO ICNT - 1 CHARACTERS,
S      TAD (7777                /SINCE ICNT POINTS TO NEXT CHARACTER,
      NOT
S                                  /THE LAST ONE. USE ICHAR TEMPORARILY.
S      DCA \ICHAR              /WRITE TEXT PORTION OF BUFFER.
      WRITE (1,1001) (IBUF (1), I = 3, ICHAR),
S      JMP A                    /GO BACK TO GET NEXT CHARACTER.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SECTION TO HANDLE CR.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SCR,    CLA CLL                /PUT STRIPPED ASCII @ (0000) IN LAST
S      DCA \ICHAR              /CHARACTER OF BUFFER USED.
      IBUF (ICNT) = ICHAR
      WRITE(4,1000) (IBUF(1), I = 2, ICNT)
S      JMS CRLF
      GO TO 10
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SECTION TO HANCLE CONTROL U
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SAGAIN, JMS CRLF                /ECHO CRLF
S      NOP                      /THEN JUMP TO START OF SUBROUTINE
      GO TO 10
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SECTION TO HANDLE RUBOUT
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SERASE, CLA CLL
S      TAD \ICNT
S      TAD (7774                /-4 OCTAL
S      SPA                      /IS ICNT < 4?
S      JMP A                    /YES; ALREADY BACK TO BEGINNING OF LINE,
S                                  /SO IGNORE RUBOUT.
S      CLA CLL                  /NO; REDUCE ICNT BY ONE.
S      TAD \ICNT
S      TAD (7777                /-1
S      DCA \ICNT
S      TAD (0134                /ECHO '/' (ASCII: 0134)
S      JMS TYPE
S      JMP A                    / GET NEXT CHARACTER.

```

```

##### YESNO.SB 12/1/75 #####
//
// SUBROUTINE YESNO.SB 12/1/75 E.J. DARLAND //
//
// PURPOSE: //
// TO ACCEPT A SINGLE CHARACTER OF INPUT FROM THE //
// KEYBOARD. IF THE CHARACTER IS A'Y', YESNO ECHOES 'YES' //
// PLUS CR LF, OTHERWISE, IT ECHOES 'NO' PLUS CR LF. THE SEVEN //
// BIT ASCII EQUIVALENT OF THE CHARACTER TYPED IS RETURNED TO //
// THE MAIN PROGRAM. //
//
// USAGE: CALL YESNO (IVAL) //
//
// PARAMETER DESCRIPTION: //
// IVAL: CHARACTER TYPED, IN 7 BIT ASCII. //
//
// MEMORY OCCUPIED: 1 PAGE //
//
#####
//
// OPDEF ITAD 1400 /INDIRECT TAD TO FOOL SABR //
// OPDEF IDCA 3400 /INDIRECT DCA TO FOOL SABR //
// OPDEF KCC 6032 /CLEAR KEYBOARD FLAG //
//
// ENTRY BLOCK FOR ARGUMENT PASSAGE //
//
#####
//
// ENTRY YESNO //
YESNO, BLOCK 2

```

YESNO.SB, CONT'D.

```

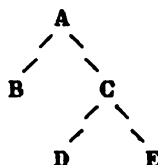
////////////////////////////////////
/
/      MAIN SUBROUTINE
/
////////////////////////////////////
      CLA CLL
      KCC
A,      KSF          /GET THE CHARACTER
      JMP A
      KRB
      AND M0177      /MASK IT TO 7-BIT ASCII
      DCA IVAL
      TAD IVAL
      TAD NY
      SZA          /IS IT A Y?
      JMP NO        /NO: ECHO NO AND CRLF
      TAD IVAL      /YES: PRINT YES AND CRLF
      JMS TYPE
      TAD PE
      JMS TYPE
      TAD PS
      JMS TYPE
CRLF,   TAD PCR
      JMS TYPE
      TAD PLF
      JMS TYPE
RETN,   CLA CLL      /RETURN IVAL TO MAIN PROGRAM
      TAD YESNO
      DCA B
B,      HLT          /THEN CDF MAIN, ADDR ARG
      ITAD YESNO#
      DCA C          /C IS CDF ART
      INC YESNO#
      ITAD YESNO#
      DCA B          /B IS NOW ADDR ARG
      INC YESNO#
C,      HLT          /THEN CDF ARG
      TAD IVAL
      IDCA B
      AND I C        /GET SABR TO CDF CURRENT
      CLA CLL
      RETRN YESNO    /RETURN TO MAIN THRU YESNO#
NO,     CLA CLL
      TAD PN
      JMS TYPE
      TAD PO
      JMS TYPE
      JMP CRLF
TYPE,   0
Z,      TSF
      JMP Z
      TLS
      CLA CLL
      JMP I TYPE
////////////////////////////////////
/
/      VARIABLE AND CONSTANT STORAGE
/
////////////////////////////////////
IVAL,   0
M0177,  0177        /SEVEN BIT MASK
NY,     7647        /-ASCII Y
PE,     0105        /E
PS,     0123        /S
PCR,    0013        /CR
PLF,    0012        /LF
PN,     0116        /N
PO,     0117        /O
      END

```

```

11
12 PIMS11.ODL JUNE 5, 1977 E.J. DARLAND
13
14 OVERLAY DESCRIPTION FILE FOR PIMS11. THIS VERSION FOR THE
15 CEMCOMGRAPH SYSTEM, USING THE VECTOR GENERAL TERMINAL AND
16 VERSATEC PLOTTER. REFER TO THE DOS/BATCH HANDBOOK (DIGITAL
17 EQUIPMENT CORP. NUMBER DEC-11-ODBHA-A-D) FOR INFORMATION
18 ON CREATING OVERLAYED FILES.

```



```

;
;
;      .ROOT      A-*(B,C-*(D,E))
A:      .FCTR      PIMS00-PIMSCO-VECLAY-OCALL-VGDD3-SAVREG-RESREG-A1
A1:     .FCTR      TWAIT-VERSOL-KBCHEK-FTNLIB/EX: VECTOR: VGDD/IN: DRAW: MODE
;
;      PIMS1A OVERLAY - THE PARAMETER INPUT SECTION
;
;
B:      .FCTR      PIMS1A-FILE11-FORPIP-PLSET-REPORT-B1
B1:     .FCTR      FTNLIB/EX: VECTOR: VGDD: DRAW: MODE
;
;      PIMS1B MAIN OVERLAY - THE EXECUTION SECTION
;
;
C:      .FCTR      PIMS1B-R3-R4-F2FORM-YCALC-OUTERF-POLCAL-C1
C1:     .FCTR      XCALC-FTNLIB/EX: VECTOR: VGDD: DRAW: MODE
;
;      EXECUTION SECOND LEVEL OVERLAY 1 - EVERYTHING NOT NECESSARY
;      FOR PLOTTING.
;
;
D:      .FCTR      FIT-DETERM-INTERP-LIST-STORE-SEARCH-D1
D1:     .FCTR      FTNLIB/EX: VECTOR: VGDD: DRAW: MODE
;
;      EXECUTION  SECOND LEVEL OVERLAY 2 - PLOTTING STUFF.
;
;
E:      .FCTR      PLOT-WINDOW-PURGE-ELEMNT-DEFINE-PUTARB-CROSS-E1
E1:     .FCTR      VGERR-VGFIX-BITTST-FTNLIB/EX: VECTOR: VGDD: DRAW: MODE
;      .END

```

```

CCCC PIMSCO.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                     CC
CC      PIMSCO.FTN                   JUNE 18, 1977                   E.J. DARLAND   CC
CC                                     CC
CC      PURPOSE:      TO INITIALIZE PIMS11 COMMON VARIABLES.         CC
CC                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
BLOCK DATA
BYTE FILOUT(24), FILIN(24), LINE(80), BELL(2)

```


PIMSCO.FTN, CONTINUED

```

LOGICAL KBCHEK
DOUBLE PRECISION XTYPE(3)
DIMENSION ICTYP (12), IDTYP(8)
DIMENSION YTYPE(12), YMOD(3), XLABL(3), CTYPE(5)
DIMENSION NDATA(8), DATA (8,2,50)
DIMENSION FITPAR (12,17)
COMMON CORR(12), RDAT(12), RIZ, RPZ, LINE, BELL, IBELL
COMMON /FILES/ FILIN,FILOUT
COMMON /FLAGS/ IFIN,ISKP,NPIE,NPHO,NION,NBAK,NDARK
1,NREF,MODE,IWF,ITYA,ITYM
COMMON /TYPES/ ICTYP,IDTYP,IXTYP,IYTYT,IYMOD
COMMON /PLOTS/ WXMIN,WXMAX,WYMIN,WYMAX,DXMIN,DXMAX,DYMIN
1,DYMAX,XDIV,YDIV,IPLT
COMMON /DATAS/ X,Y,WAVE,TIME
COMMON /LABELS/ YTYPE,YMOD,XTYPE,XLABL,CTYPE
COMMON /TERPS/ NDATA,DATA
COMMON /FITSTF/ FITPAR,NORDR
COMMON /VECTOR/ XSCALE,YSCALE,IDEV
COMMON /VTEK/ XUPPER,YUPPER,NIBS
COMMON /VOLAY/XOFF,YOFF,XLAST,YLAST,XV(10),YV(10),NI
COMMON /WINDOW/ XLDUM,YLDUM
DATA YTYPE /'UBP','UBI','DP','DI','BP','BI','RP'
1,'RI','RPIE','PHO','ION','PIE'/
DATA XTYPE /'WAVLNTH','TIME','ENERGY'/
DATA XLABL /'ANG.','SEC.','EV.'/
DATA YMOD /'DATA','FIT','RES.'/
DATA CTYPE /'N','L','P','I','-'/
DATA BELL /07,07/
DATA IBELL /0007/
END

```

```

CCCC PIMS00.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      PIMS00.FTN                JUNE 18, 1977                E.J. DARLAND      CC
CC
CC      PURPOSE:  TO COMMUNICATE BETWEEN THE PARAMETER SPECIFI-    CC
CC                  CATION AND OPTION EXECUTION SECTIONS OF PIMS11. THIS    CC
CC                  PROGRAM INITIALIZES THE PARAMETERS TO THEIR DEFAULT    CC
CC                  VALUES AND THEN CALLS THE PARAMETER SPECIFICATION    CC
CC                  SUBROUTINE, PIMS1A. PIMS1A WILL RETURN TO PIMS00 ONLY    CC
CC                  TO REINITIALIZE, QUIT, OR EXECUTE A MAJOR OPTION. IN    CC
CC                  THE LATTER CASE, PIMS00 MERELY PASSES THE INFORMATION    CC
CC                  ABOUT WHICH OPTION TO EXECUTE TO THE EXECUTION SUB-    CC
CC                  ROUTINE, PIMS1B. PIMS00 IS MADE AS SMALL AS POSSIBLE,    CC
CC                  AND AS MUCH TELETYPE INTERACTION AS POSSIBLE IS DONE IN    CC
CC                  PIMS1A, SO THAT THE AMOUNT OF CORE OCCUPIED BY PIMS00,    CC
CC                  PIMS1B, AND OVERLAYS WILL BE AS SMALL AS POSSIBLE.    CC
CC
CC      USAGE:  SEE TEXT, CHAPTER V.    CC
CC
CC      SUBROUTINES CALLED:  PIMS1A, PIMS1B    CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

```

```

BYTE FILOUT(24), FILIN(24), LINE(80), BELL(2)
LOGICAL KBCHEK
DOUBLE PRECISION XTYPE(3)
DIMENSION ICTYP (12), IDTYP(8)
DIMENSION YTYPE(12), YMOD(3), XLABL(3), CTYPE(5)
DIMENSION NDATA(8), DATA (8,2,50)
DIMENSION FITPAR (12,17)
COMMON CORR(12), RDAT(12), RIZ, RPZ, LINE, BELL, IBELL
COMMON /FILES/ FILIN,FILOUT
COMMON /FLAGS/ IFIN,ISKP,NPIE,NPHO,NION,NBAK,NDARK

```

PIMS00.FTN, CONTINUED

```

      1,NREF,MODE,IWF,ITYA,ITYM
      COMMON /TYPES/ ICTYP, IDTYP, IXTYP, IYTYP, IYMOD
      COMMON /PLOTS/ WXMIN, WXMAX, WYMIN, WYMAX, DXMIN, DXMAX, DYMIN
      1, DYMAX, XDIV, YDIV, IPLT
      COMMON /DATAS/ X, Y, WAVE, TIME
      COMMON /LABELS/ YTYPE, YMOD, XTYPE, XLABL, CTYPE
      COMMON /TERPS/ NDATA, DATA
      COMMON /FITSTF/ FITPAR, NORDR
      COMMON /VECTOR/ XSCALE, YSCALE, IDEV
      COMMON /VTEK/ XUPPER, YUPPER, NIBS
      COMMON /VOLAY/ XOFF, YOFF, XLAST, YLAST, XV(10), YV(10), NI
      COMMON /WINDOW/ XLDUM, YLDUM
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SET PARAMETERS AND FLAGS TO DEFAULT VALUES
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10      IDEV = 1
      IFIN = 0
      IPLT = 2
      ISKP = 1
      IYMOD = 1
      XDIV = 1.
      YDIV = 1.
      ITYA = 12
      ITYM = 1
      IYTYP = 12
      IXTYP = 1
      IWF = 0
      IYMOD = 1
      DO 20 I = 1, 12
      FITPAR (I, 12) = 0.
20      RDAT (I) = 0.
      ICTYP (9) = 5
      ICTYP (10) = 2
      ICTYP (11) = 2
      ICTYP (12) = 5
      DO 22 I = 1, 8
      IDTYP (I) = 1
      ICTYP (I) = 1
22      NDATA (I) = 0
      XUPPER = 9.0
      YUPPER = 7.0
      CALL PLSET
      CALL SETERR (6, -1)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CALL PIMS1B TO INITIALIZE GRAPHICS TERMINAL
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      CALL PIMS1B (1)
      IACTON = -1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      GO TO OPTION INPUT SECTION OF PIMS11. ON RETURN EITHER
C      RE-INITIALIZE, QUIT, OR PASS IACTON ON TO PIMS1B FOR ACTION.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
30      CALL PIMS1A (IACTON)
      IF (IACTON) 10, 40, 50
40      CALL EXIT
50      CALL PIMS1B (IACTON)
      GO TO 30
      END

```

PIMS1A.FTN

```

CCCC PIMS1A.FTN CCCC 9/10/1977 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      SUBROUTINE PIMS1A.FTN      JUNE 18, 1977      E.J. DARLAND      CC
CC
CC      PURPOSE:  TO PERFORM ALL USER-COMPUTER INTERACTION VIA THE CC
CC                TERMINAL, AND SET UP ALL PARAMETERS FOR THE EXECUTION OF CC
CC                THE MAJOR PIMS11 OPTIONS.      CC
CC
CC      USAGE:  CALL PIMS1A (IACTON)      CC
CC
CC      PARAMETER DESCRIPTION:  IACTON INDICATES WHAT ACTION IS CC
CC                EXPECTED OF PIMS1A WHEN IT IS CALLED BY PIMS00, AND CC
CC                ALSO INDICATES TO PIMS00 WHAT ACTION IT SHOULD REQUIRE CC
CC                OF PIMS1B.      CC
CC
CC                ***** ON CALL FROM PIMS00 *****      CC
CC
CC                IACTON < 0:  WRITE FULL  OPTION PROMPT.      CC
CC
CC                IACTON = 0:  WRITE ONLY BRIEF PROMPT.      CC
CC
CC                IACTON > 0:  PERFORM THE FOLLOWING ACTION:      CC
CC                             1: REPORT X,Y AS LOCATION OF      CC
CC                               CROSSEHAIRS.      CC
CC                             2: WRITE PLOT DEVICE SIZE  ERROR      CC
CC                               MESSAGE.      CC
CC                             3: WRITE NO POINTS IN RANGE      CC
CC                               WARNING.      CC
CC
CC                ***** ON RETURN TO PIMS00 *****      CC
CC
CC                IACTON < 0:  RE-INITIALIZE      CC
CC
CC                IACTON = 0:  EXIT TO MONITOR      CC
CC
CC                IACTON > 0:  PERFORM THE FOLLOWING ACTION, VIA CC
CC                             THE EXECUTION SECTION (PIMS1B)      CC
CC                             1: ERASE THE TEK TERMINAL OR VG      CC
CC                               SCREEN.      CC
CC                             2: RESERVED FOR FUTURE VERSATEC      CC
CC                               OVERLAID PLOT FINISH.      CC
CC                             3: EXECUTE THE LOCATE OPTION OF      CC
CC                               TERMINAL      CC
CC                             4: LIST,STORE,PLOT OR SEARCH DTAA      CC
CC                               CALCULATED FROM POLYNOMIAL      CC
CC                               COEFFICIENTS STORED IN FITPAR.      CC
CC                             5: FIT,LIST,STORE,PLOT OR SEARCH      CC
CC                               UBP OR UBI DATA STOREE IN THE      CC
CC                               ARRAY "DATA".      CC
CC                             6: FIT,LIST,STORE,PLOT,SEARCH OR      CC
CC                               STORE IN "DATA" INFORMATION      CC
CC                               DERIVED FROM THE INPUT FILE.      CC
CC
CC      INTERNAL VARIABLES:      CC
CC      FILTEM: TEMPORARY FILE SPECIFICATION STORAGE.      CC
CC      IBELL:  ASCII CONTROL G (BELL)      CC
CC
CC      SUBROUTINES CALLED:  FILE11, FORPIP, PLSET, REPORT      CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SUBROUTINE PIMS1A (IACTON)
BYTE FILOUT(24), FILIN(24), LINE(80), BELL(2)
LOGICAL KBCHEK
DOUBLE PRECISION XTYPE(3)
DIMENSION ICTYP (12), IDTYP(8)
DIMENSION YTYPE(12), YMOD(3), XLABL(3), CTYPE(5)
DIMENSION NDATA(8), DATA (8,2,50)

```

PIMS1A.FTN, CONTINUED

```

      DIMENSION FITPAR (12,17)
      COMMON CORR(12), RDAT(12), RIZ, RPZ, LINE, BELL, IBELL
      COMMON /FILES/ FILIN,FILOUT
      COMMON /FLAGS/ IFIN,ISKP,NPIE,NPHO,NION,NBAK,NDARK
      1,NREF,MODE,IWF,ITYA,ITYM
      COMMON /TYPES/ ICTYP,IDTYP,IXTYP,IYTYP,IYMOD
      COMMON /PLOTS/ WXMIN,WXMAX,WYMIN,WYMAX,DXMIN,DXMAX,DYMIN
      1,DYMAX,XDIV,YDIV,IPLT
      COMMON /DATAS/ X,Y,WAVE,TIME
      COMMON /LABELS/ YTYPE,YMOD,XTYPE,XLABL,CTYPE
      COMMON /TERPS/ NDATA,DATA
      COMMON /FITSTF/ FITPAR,NORDR
      COMMON /VECTOR/ XSCALE,YSCALE,IDEV
      COMMON /VTEK/ XUPPER,YUPPER,NIBS
      COMMON /VOLAY/XOFF,YOFF,XLAST,YLAST,XV(10),YV(10),NI
      COMMON /WINDOW/ XLDUM,YLDUM
      BYTE FILTEM(24)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DECIDE WHERE TO GO
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10      IF (IACTON) 100,110,20
20      GO TO (25,30,35) IACTON
25      WRITE (6,9029) X,Y
      GO TO 110
30      WRITE (6,9052)
      GO TO 110
35      WRITE (6,9056)
      GO TO 110
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WRITE THE NAMES OF THE AVAILABLE OPTIONS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
100     WRITE (6,9000)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MAKE SURE WINDOW IS SET
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
110     IF (IWF) 112,112,115
112     WXMIN = DXMIN
      WXMAX = DXMAX
      WYMIN = DYMIN
      WYMAX = DYMAX
      CALL PLSET
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      RESET IYTYP AND IYMOD
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
115     IYTYP = ITYA
      IYMOD = ITYM
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WRITE THE PROMPT, READ THE INPUT OPTION NAME AND BRANCH.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WRITE (6,9001) IBELL
      READ (6,9002) IOPT,ICNT
      IF (ICNT.LT.2) GO TO 100
      IF (IOPT.EQ.'CU') GO TO 150
      IF (IOPT.EQ.'DI') GO TO 200
      IF (IOPT.EQ.'ER') GO TO 225
      IF (IOPT.EQ.'FI') GO TO 250
      IF (IOPT.EQ.'IB') GO TO 300
      IF (IOPT.EQ.'IN') GO TO 350
      IF (IOPT.EQ.'LI') GO TO 400

```

```

IF (IOPT.EQ.'LO') GO TO 450
IF (IOPT.EQ.'ME') GO TO 500
IF (IOPT.EQ.'PA') GO TO 575
IF (IOPT.EQ.'PB') GO TO 600
IF (IOPT.EQ.'PL') GO TO 650
IF (IOPT.EQ.'PR') GO TO 675
IF (IOPT.EQ.'QU') GO TO 700
IF (IOPT.EQ.'RE') GO TO 125
IF (IOPT.EQ.'SE') GO TO 750
IF (IOPT.EQ.'SI') GO TO 800
IF (IOPT.EQ.'SK') GO TO 825
IF (IOPT.EQ.'ST') GO TO 850
IF (IOPT.EQ.'WI') GO TO 900
IF (IOPT.EQ.'XA') GO TO 950
IF (IOPT.EQ.'XL') GO TO 1000
IF (IOPT.EQ.'YA') GO TO 1050
IF (IOPT.EQ.'YL') GO TO 1100
IF (IOPT.EQ.'YM') GO TO 1150
GO TO 100
125 IACTON = -1
RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CURRENT.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
150 WRITE (6,9006) YTYPE(IYTYP), YMOD(IMOD), XTYPE(IXTYP)
WRITE (6,9059) ISKP
WRITE (6,9003)
175 WRITE (6,9004) DYMAX,WYMAX
WRITE (6,9005) DXMIN,dxmax,wxmin,wymax
WRITE (6,9004) DYNIN,WYNIN
GO TO 115
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
DIVISIONS. THE BUSINESS OF READING NEW VALUES INTO RA AND RB
FIRST, IN THIS AND MANY OTHER INPUT SECTIONS IS IN CASE
THE USER TYPES A NEGATIVE NUMBER FOR ONE OF THE INPUTS. THIS
WOULD SIGNIFY "ABORT THE OPTION"; READING INTO RA, RB FIRST
ALLOWS THE OLD VALUES OF THE PARAMETERS TO BE SAVED.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
200 WRITE (6,9007) XDIV,YDIV
WRITE (6,9008)
READ (6,9009) RA, RB
IF (RA.LE.0.0 .OR. RB.LE.0.0) GO TO 115
XDIV = RA
YDIV = RB
GO TO 115
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
ERASE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
225 IACTON = 1
IDEV = 1
RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
FIT. FIRST WRITE OUT CURRENT PARAMETERS TO MAKE SURE IT'S
OKAY.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
250 IF (DXMAX - DXMIN) 252,251,252
251 WRITE (6,9057)
GO TO 1000
252 ITYA = IYTYP
ITYM = IMOD
IYMOD = 1

```

PIMS1A.FTN, CONTINUED

```

      WRITE (6,9006) YTYPE(IYTP), YMOD(IMOD), XTYPE (IXTP)
      WRITE (6,9010) ISKP,DXMIN,DXMAX,XLABL(IXTP)
      WRITE (6,9011)
      READ (6,9012) IOPT
      IF (IOPT.NE. 'Y') GO TO 115
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      IT'S OKAY, SO READ IN THE ORDER OF THE EQUATION DESIRED AND
C      PROCEED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
255      WRITE (6,9013)
      READ (6,9014) NORDR
      IF (NORDR) 115,260,260
260      IF (NORDR.LE.9) GO TO 265
      WRITE (6,9015)
      GO TO 255
265      MODE = 2
      GO TO 2000
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      IBACK. SET IYTP SO IT KNOWS IT'S DOING IONS, THEN GIVE THE
C      USER THE OPTION OF CHANGING ONE OR ALL OF THE POINTS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
300      IYTP = 2
301      WRITE (6,9016)
      READ (6,9014) IOPT
      IF (IOPT) 115,305,302
302      IF (IOPT.LE. NDATA(IYTP)) GO TO 335
      WRITE (6,9015)
      GO TO 301
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      ENTERING ALL NEW DATA. FIRST GET DESIRED NUMBER OF POINTS
C      AND UNITS OF ABSCISSA.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
305      WRITE (6,9017)
      READ (6,9014) ICNT
      IF (ICNT) 115,305,310
310      IF (ICNT.LE. 50) GO TO 315
      WRITE (6,9015)
      GO TO 305
315      WRITE (6,9018)
      READ (6,9014) IOPT
      IF (IOPT.GE.1 .AND. IOPT.LE.3) GO TO 320
      GO TO 315
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      NOW READ IN THE NEW VALUES, AFTER 'ZEROING' THE CONTENTS OF
C      THE ARRAY BY SETTING NDATA = 0.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
320      NDATA (IYTP) = 0
      DO 330 I = 1,ICNT
      WRITE (6,9019) I
      WRITE (6,9020)
      READ (6,9009) DATA (IYTP,1,I)
      IF (DATA(IYTP,1,I)) 115,325,325
325      WRITE (6,9021)
330      READ (6,9009) DATA (IYTP,2,I)
      NDATA (IYTP) = ICNT
      IDTYP (IYTP) = IOPT
      GO TO 115

```

```
C  
C ONLY CHANGING ONE VALUE. WRITE OUT CURRENT VALUE FIRST, THEN  
C READ NEW VALUES.  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
335 WRITE (6,9054) IOPT, DATA(ITYP,1,IOPT), DATA(ITYP,2,IOPT)  
WRITE (6,9008)  
READ (6,9009) RA,RB  
IF (RA) 115,340,340  
340 DATA (ITYP,1,IOPT) = RA  
DATA (ITYP,2,IOPT) = RB  
GO TO 115  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C  
C INPUT  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
350 IOPT = 1  
IFIN = 1  
CALL FILE11 (FILTEM,IOPT)  
IF (IOPT) 115,115,355  
355 DO 360 I = 1,24  
360 FILIN(I) = FILTEM(I)  
IFIN = 1  
GO TO 115  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C  
C LIST  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
400 MODE = 3  
GO TO 2000  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C  
C LOCATE  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
450 IACTON = 3  
RETURN  
455 WRITE (6,9029) X,Y  
GO TO 115  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C  
C METHOD. WRITE OUT THE METHODS THAT ARE CURRENTLY IN FORCE,  
C THEN READ THE NUMBER OF THE DATA TYPE WHOSE METHOD IS TO BE  
C CHANGED. READ THE METHOD TO BE USED, AND BRANCH TO THE  
C SECTION APPROPRIATE TO CHECKING OUT ITS LEGALITY.  
C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
500 WRITE (6,9030) (I,I=1,12)  
WRITE (6,9031) (YTYPE(I),I=1,12)  
WRITE (6,9032) (CTYPE(ICTYP(I)),I=1,12)  
WRITE (6,9033)  
READ (6,9014) ICNT  
IF (ICNT) 115,115,505  
505 IF (ICNT .LE. 12) GO TO 510  
WRITE (6,9015)  
GO TO 500  
509 WRITE (6,9053)  
GO TO 115  
510 IF (ICNT.EQ.9 .OR. ICNT.EQ.12) GO TO 509  
WRITE (6,9034) IBELL  
READ (6,9002) IOPT, IT  
IF (IT) 510,510,515  
515 IF (IOPT.EQ.'N') GO TO 520  
IF (IOPT.EQ.'L') GO TO 525  
IF (IOPT.EQ.'P') GO TO 530  
IF (IOPT.EQ.'I') GO TO 540
```

PIMS1A.FTN, CONTINUED

```

      IF (IOPT.EQ.IBELL) GO TO 115
      GO TO 510
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      NO CORRECTION, ALWAYS LEGAL.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
520      ICTYP (ICNT) = 1
      GO TO 115
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      USE LATEST POINT.  LEGAL FOR ALL DATA TYPES EXCEPT UBP OR UBI.
C      THESE ARE NOT IN THE INPUT FILE, AND SO A "LATEST VALUE FOUND"
C      CAN NEVER EXIST FOR THEM.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
525      IF (ICNT .LE. 2) GO TO 526
      ICTYP (ICNT) = 2
      GO TO 115
526      WRITE (6,9046)
      GO TO 115
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      POLYNOMIAL CORRECTION.  LEGAL FOR ALL DATA TYPES, PROVIDING
C      THAT A FIT HAS BEEN PERFORMED ON DATA OF THE DESIRED TYPE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
530      IF (FITPAR (ICNT,12) .EQ. 0.0) GO TO 535
      ICTYP (ICNT) = 3
      GO TO 115
535      WRITE (6,9035) YTYPE (ICNT)
      GO TO 115
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      INTERPOLATIVE CORRECTION.  NOT LEGAL FOR RPIE,PHO,ION OR PIE.
C      IF LEGAL, CHECK TO MAKE SURE DATA EXISTS IN THE DATA (I,J,K)
C      ARRAY TO MATCH THE DESIRED TYPE.  IF NOT, AND THE TYPE IS
C      UBP OR UBI, THE USER MUST ENTER IT.  IF NOT AND THE TYPE IS
C      ANY OTHER LEGAL TYPE, MAKE A PASS THROUGH THE INPUT FILE TO
C      GET THE INFORMATION.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
540      IF (ICNT .LE. 8) GO TO 545
542      WRITE (6,9036) YTYPE (ICNT)
      GO TO 115
545      IF (NDATA (ICNT) .LE. 0) GO TO 550
      ICTYP (ICNT) = 4
      GO TO 115
550      IF (ICNT .GT. 2) GO TO 555
552      WRITE (6,9037)
      GO TO 115
555      MODE = 1
      ITYA = IYTYP
      ITYM = IYMOD
      IYTYP = ICNT
      IYMOD = 1
      GO TO 2000
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      PARAMETERS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
575      IOPT = 0
      CALL FILE11 (FILTEM,IOPT)
      IF (IOPT) 115,115,580
580      DO 585 I = 1,24
585      FILOUT(I) = FILTEM(I)
      CALL REPORT
      GO TO 115

```


PIMS1A.FTN, CONTINUED

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      PBACK.  SET IYTP AND DO THE SAME AS FOR IBACK.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
600      IYTP = 1
          GO TO 301
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      PLOT.  READ DEVICE AND TYPE OF PLOT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
650      WRITE (6,9038)
          READ (6,9014) IDEV
          IF (IDEV) 115,650,655
655      IF (IDEV .GT. 2) GO TO 650
660      WRITE (6,9039)
          READ (6,9014) IPLT
          IF (IPLT) 115,660,665
665      IF (IPLT .GT. 2) GO TO 660
          MODE = 5
          GO TO 2000
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      PREPARE (FOR NEW INTERP DATA)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
675      ICNT = IYTP
          IF (IYTP .GT. 8) GO TO 542
          IF (IYTP .LE. 2) GO TO 552
          GO TO 555
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      QUIT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
700      CALL EXIT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SEARCH.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
750      MODE = 6
          GO TO 2000
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SIZE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
800      WRITE (6,9040)
          RA = XUPPER - 1.
          RB = YUPPER - 1.
          WRITE (6,9029) RA, RB
          WRITE (6,9008)
          READ (6,9009) RA, RB
          IF (RA) 115,806,805
805      IF (RB) 115,807,810
806      XUPPER = 9.0
          YUPPER = 7.0
          GO TO 815
807      XUPPER = RA + 1.0
          YUPPER = (0.75 * RA) + 1.0
          GO TO 815
810      XUPPER = RA + 1.0
          YUPPER = RB + 1.0
815      CALL PLSET
          GO TO 115

```



```

WRITE (6,9031) (YTYPE(I), I=1,12)
WRITE (6,9044)
READ (6,9014) IOPT
IF (IOPT) 115,1050,1055
1055 IF (IOPT.GT. 12) GO TO 1050
IYTYP = IOPT
ITYA = IYTYP
GO TO 115
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      YLIMITS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1100 WRITE (6,9042)
      READ (6,9009) DYMIN,DYMAX
      GO TO 110
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      YMOD
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1150 WRITE (6,9045)
      READ (6,9014) IOPT
      IF (IOPT) 115,1150,1155
1155 IF (IOPT.GT. 3) GO TO 1150
      IYMOD = IOPT
      ITYM = IOPT
      GO TO 115
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      ***** EXECUTION SETUP SECTION *****
C
C      THE REMAINDER OF PIMS1A IS DEVOTED TO TWO TASKS. FIRST,
C      THE PARAMETERS THAT THE USER HAS SPECIFIED ARE CHECKED FOR
C      COMPATABILITY WITH THE DESIRED ACTION AND THE CAPABILITIES
C      OF THE PROGRAM. THEN, ASSUMING ALL IS OKAY, ANY NECESSARY
C      INTERNAL PARAMETERS ARE SET UP FOR THE ACTUAL EXECUTION,
C      PERFORMED BY PIMS1B.
C
C      FIRST, MAKE SURE THAT THE X AND Y RANGES ARE NOT ZERO, UNLESS
C      EXECUTING A SEARCH OPTION THAT DOESN'T NEED TO USE
C      DXMIN , ETC.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2000 IF (DXMIN.EQ. DXMAX) GO TO 2001
      IF (DYMIN.EQ. Dymax) GO TO 2002
      GO TO 2003
2001 IF (MODE.EQ. 6 .AND. IYMOD.NE. 2) GO TO 2003
      WRITE (6,9057)
      GO TO 1000
2002 IF (MODE.NE. 5) GO TO 2003
      WRITE (6,9058)
      GO TO 1100
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      IF THE DESIRED DATA IS A FIT OR RESIDUAL, MAKE SURE
C      THAT A FIT HAS ACTUALLY BEEN DONE TO THAT DATA TYPE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2003 IF (IYMOD.LE. 1) GO TO 2005
      IF (FITPAR(IYTYP,12).GT. 0.0) GO TO 2005
      WRITE (6,9035) YTYPE(IYTYP)
      GO TO 115

```

PIMS1A.FTN, CONTINUED

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      IF DOING UBP OR UBI, WHICH ARE STORED WITH A PARTICULAR
C      ABSCISSA TYPE, MUST MAKE SURE THAT IXTYP MATCHES THE UNITS
C      OF THE DATA STORED IN THE DATA ARRAY.
C      ALSO, IF THE DESIRED DATA MUST BE CALCULATED FROM
C      POLYNOMIAL COEFFICIENTS, THE ABSCISSA USED WHEN THE
C      COEFFICIENTS WERE CALCULATED MUST MATCH THE CURRENT ABSCISSA
C      (IXTYP).
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2003      IF (IYTYP .GT. 2) GO TO 2010
          IF (IDTYP(IYTYP).EQ.IXTYP) GO TO 2010
          IF (IDTYP(IYTYP).EQ.(4-IXTYP)) GO TO 2010
          WRITE (6,9047) XTYPE (IXTYP)
          GO TO 115
2010      IF (IYMOD .LE. 1) GO TO 2015
          IF (FITPAR (IYTYP,12) .EQ. IXTYP) GO TO 2015
          WRITE (6,9048) XTYPE (IXTYP)
          GO TO 115
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DECIDE WHICH OF THE FOLLOWING MAJOR SECTIONS TO BRANCH TO.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2015      IF (IYMOD .EQ. 2) GO TO 2100
          IF (IYTYP .LE. 2) GO TO 2300
          GO TO 2500
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CASE 1. WE WILL BE LISTING, STORING, PLOTTING, OR SEARCHING
C      DATA CALCULATED FROM A POLYNOMIAL EQUATION. WE CAN'T GET
C      HERE WITH MODE = 1 OR 2, AT LEAST IN THEORY. WILL DO 100
C      POINTS BETWEEN THE CURRENT DATA ABSCISSA LIMITS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2100      IACTON = 4
          RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CASE 2. THE 'INPUT' DATA IS COMING FROM UBP OR UBI DATA
C      STORED IN THE DATA ARRAY. SUCH DATA MAY BE FIT, LISTED,
C      STORED, PLOTTED OR SEARCHED. MODE WILL NOT = 1.
C      EVERY POINT FOUND IN RANGE WILL BE USED; THERE IS NO NEED TO
C      DO ONLY EVERY ISKP POINTS, SINCE THERE CAN BE ONLY 50
C      POINTS AT MOST. (IF A SEARCH IS BEING DONE, EVERY
C      POINT IS DEFINED AS BEING 'IN RANGE').
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2300      N = NDATA (IYTYP)
          IF (N) 2305,2305,2310
2305      WRITE (6,9037)
          GO TO 115
2310      IACTON = 5
          RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CASE 3. THE INPUT FILE MUST BE READ TO GET THE NECESSARY DATA.
C      INITIALIZE EVERYTHING UNDER THE SUN.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2500      IF (IFIN .GT. 0) GO TO 2501
          WRITE (6,9055)
          GO TO 115
2501      IF (IYTYP.EQ.9 .AND. (ICTYP(7).EQ.1.OR. ICTYP(8).EQ.1))
          1 GO TO 2502
          IF (IYTYP.EQ.12 .AND. (ICTYP(10).EQ.1.OR. ICTYP(11).EQ.1))
          1 GO TO 2502
          GO TO 2504

```

PIMS1A.FTN, CONTINUED

```

2502  WRITE (6,9050)
      GO TO 115
2504  NP1E = 0
      NPHO = 0
      N1ON = 0
      N1AK = 0
      NDARK = 0
      NREF = 0
      RPZ = 1.
      RIZ = 1.
      CORR(1) = 0.
      CORR(2) = 0.
      CORR(3) = 0.
      CORR(4) = 0.
      CORR(5) = 0.
      CORR(6) = 0.
      CORR(7) = 1.
      CORR(8) = 1.
      CORR(9) = 1.
      CORR(10) = 0.
      CORR(11) = 0.
      CORR(12) = 0.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SET THE FLAGS FOR THE VARIOUS LINE TYPES WE WILL NEED TO
C      BOTHER DECODING.
C
C      1. PIE; ONLY IF DOING PHO OR ION, OR IF DOING PIE WITH
C      LATEST PHO OR ION.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF (IYTYP.EQ.10 .OR. IYTYP.EQ.11) GO TO 2505
      IF (IYTYP.EQ.12.AND.(ICTYP(10).EQ.2.OR.ICTYP(11).EQ.2.))
      1 GO TO 2505
      GO TO 2510
2505  NP1E = 1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      2. PHO; ONLY IF DOING PHOTONS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2510  IF (IYTYP.EQ.10) NPHO = 1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      3. ION; ONLY IF DOING IONS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF (IYTYP.EQ.11) N1ON = 1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      4. DARK PHOTON/ION. READ IF DOING DP OR DI, OR IF USING
C      LATEST DP OR DI FOR BP, BI, RP, RI, RP1E, PHO, ION, PIE, AS
C      APPROPRIATE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      GO TO (2535,2535,2530,2530,2515,2520,2515,2520,2525,2515
      1,2520,2525) IYTYP
2515  IF (ICTYP(3).EQ.2) GO TO 2530
      GO TO 2535
2520  IF (ICTYP(4).EQ.2) GO TO 2530
      GO TO 2535
2525  IF (ICTYP(3).EQ.2 .OR. ICTYP(4).EQ.2) GO TO 2530
2530  NDARK = 1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      8. BACKGROUNDS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2535  GO TO (2560,2560,2560,2560,2555,2555,2540,2545,2550,2540

```

PIMS1A.FTN, CONTINUED

```

      1,2545,2550) IYYP
2540   IF (ICTYP(5) .EQ. 2) GO TO 2555
      GO TO 2560
2545   IF (ICTYP(6) .EQ. 2) GO TO 2555
      GO TO 2560
2550   IF (ICTYP(5).EQ.2 .OR. ICTYP(6).EQ.2) GO TO 2555
      GO TO 2560
2555   NBACK = 1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      17. REFERENCE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2560   GO TO (2585,2585,2585,2585,2585,2585,2580,2580,2565
      1,2570,2575) IYYP
2565   IF (ICTYP(7) .EQ. 2) GO TO 2580
      GO TO 2585
2570   IF (ICTYP(8) .EQ. 2) GO TO 2580
      GO TO 2585
2575   IF (ICTYP(8).EQ.2 .OR. ICTYP(7).EQ.2) GO TO 2580
      GO TO 2585
2580   NREF = 1
2585   IACTON = 6
      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      FORMAT STATEMENTS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
9000   FORMAT(' DIVISIONS',5X,'SKIP',13X,'CURRENT',10X,'FIT'/
      1' ERASE',9X,'WINDOW',11X,'LOCATE',11X,'LIST'/
      2' IBACK',9X,'XAXIS',12X,'PARAMETERS',7X,'PLOT'/
      3' INPUT',9X,'XLIMIT',11X,'QUIT',13X,'PREPARE'/
      4' METHOD',8X,'YAXIS',12X,'RESTART',10X,'SEARCH'/
      5' PBACK',9X,'YLIMIT',28X,'STORE'/' SIZE',10X,'YMOD')
9001   FORMAT('S->',A1)
9002   FORMAT(A2,Q,ERR=100)
9003   FORMAT(16X,'DATA',31X,'WINDOW')
9004   FORMAT(9X,G15.7,22X,G15.7)
9005   FORMAT(2(G15.7,2X,G15.7,5X))
9006   FORMAT(1X,A4,1X,A4,' VS. ',A8)
9007   FORMAT(' X,Y DIVISIONS = ',G15.7,',',G15.7)
9008   FORMAT('NEW VALUES = ')
9009   FORMAT(3E15.7)
9010   FORMAT(' ONE OF EVERY ',I3,' POINTS FROM ',G15.7,' TO '
      1,G15.7,1X,A8)
9011   FORMAT('SOKAY ? ')
9012   FORMAT(A1,IERR=110)
9013   FORMAT('SORDER OF EQUATION: ')
9014   FORMAT(I6,ERR=110)
9015   FORMAT(' TOO LARGE!')
9016   FORMAT('SPPOINT NO. TO CHANGE [0=ALL]: ')
9017   FORMAT('SNO. OF POINTS: ')
9018   FORMAT('S ABSCISSA = WAVELENGTH [1], TIME [2] OR ENERGY [3]: ')
9019   FORMAT(' POINT ',I2)
9020   FORMAT('S ABSCISSA = ')
9021   FORMAT('S ORDINATE = ')
9029   FORMAT(' (',G15.7,',',G15.7,')')
9030   FORMAT(12(2X,I2,2X))
9031   FORMAT(12(1X,A4,1X))
9032   FORMAT(12(2X,A1,3X))
9033   FORMAT('S TYPE TO CHANGE: ')
9034   FORMAT('SMETHOD ->',A1)
9035   FORMAT(' NO FIT DONE ON ',A4,' YET!')
9036   FORMAT(' SORRY. CAN'T INTERPOLATE ',A4)
9037   FORMAT(' PLEASE ENTER BACKGROUND DATA!')
9038   FORMAT('STERMINAL [1] OR PLOTTER [2]: ')
9039   FORMAT('S LINE [1] OR POINT PLOT [2]: ')
9040   FORMAT('SX,Y DIMENSIONS [INCHES] = ')

```

PIMS1A.FTN, CONTINUED

```

9041  FORMAT('XMIN,XMAX = ')
9042  FORMAT('YMIN,YMAX = ')
9043  FORMAT('WAVELENGTH [1], TIME [2] OR ENERGY [3] : ')
9044  FORMAT('YOUR CHOICE: ')
9045  FORMAT('DATA [1], FIT [2], OR RESIDUALS [3] : ')
9046  FORMAT(' SORRY, CAN'T USE THIS METHOD FOR UBP OR UBI')
9047  FORMAT(' DATA DOES NOT HAVE ',A8,' ABSCISSA! ')
9048  FORMAT(' FIT NOT DONE VS. ',A8)
9050  FORMAT(' INAPPROPRIATE N CORRECTION METHOD')
9051  FORMAT(' DO ONLY ONE POINT OUT OF EVERY : ')
9052  FORMAT(' SIZE TOO LARGE FOR DEVICE! ')
9053  FORMAT(' NEVER USED AS CORRECTION!')
9054  FORMAT(' POINT ',I3,' IS NOW ',G15.7,'.',',G15.7)
9055  FORMAT(' PLEASE SPECIFY INPUT FILE!')
9056  FORMAT(' NO POINTS FOUND IN RANGE!')
9057  FORMAT(' ZERO RANGE UNACCEPTABLE. PLEASE ENTER X LIMITS.')
9058  FORMAT(' ZERO RANGE UNACCEPTABLE. PLEASE ENTER Y LIMITS.')
9059  FORMAT(' DOING ONE POINT OUT OF EVERY ',I3)
      END

```

```

CCCC PIMS1B.FTN CCCC 6/18/1977 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                     CC
CC      SUBROUTINE PIMS1B.FTN      JUNE 18,1977      E.J. DARLAND      CC
CC                                     CC
CC      PURPOSE:  TO EXECUTE SEVERAL OF THE OPTIONS CHOSEN IN      CC
CC                  PIMS1A, ACCORDING TO THE PARAMATERS PASSED VIA PIMS00.  CC
CC                  SEE THE DESCRIPTIONS AT THE BEGINNING OF PIMS00 AND      CC
CC                  PIMS1A.      CC
CC                                     CC
CC      USAGE:    CALL PIMS1B(IACON)      CC
CC                                     CC
CC      PARAMETER DESCRIPTION:  SEE PIMS1A.FTN      CC
CC                                     CC
CC      SUBROUTINES CALLED:  DETERM, FIT, F2FORM, INTERP, KBCHEK,      CC
CC                  KBCKOF, KBCKON, LIST, OUTERP, POLCAL, PLOT, R3, R4,      CC
CC                  SEARCH, STORE, WINDOW, XCALC, YCALC      CC
CC                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE PIMS1B (IACON)
      BYTE FILIN(24), FILOUT(24), LINE(80), BELL(2)
      LOGICAL KBCHEK
      DOUBLE PRECISION XTYPE(3)
      DIMENSION ICTYP (12), IDTYP(8)
      DIMENSION YTYPE(12), YMOD(3), XLABL(3), CTYPE(5)
      DIMENSION NDATA(8), DATA (8,2,50)
      DIMENSION FITPAR (12,17)
      COMMON CORR(12), RDAT(12), RIZ, RPZ, LINE, BELL, IBELL
      COMMON /FILES/ FILIN,FILOUT
      COMMON /FLAGS/ IFIN,ISKP,NPIE,NPHO,NION,NBAK,NDARK
      1,NREF,MODE,IWF,ITYA,ITYM
      COMMON /TYPES/ ICTYP, IDTYP, IXTYP, IYTYP, IYMOD
      COMMON /PLOTS/ WXMIN,WXMAX,WYMIN,WYMAX,DXMIN,DXMAX,DYMIN
      1,DYMAX,XDIV,YDIV,IPLT
      COMMON /DATAS/ X,Y,WAVE,TIME
      COMMON /LABELS/ YTYPE,YMOD,XTYPE,XLABL,CTYPE
      COMMON /TERPS/ NDATA,DATA
      COMMON /FITSTF/ FITPAR,NORDR
      COMMON /VECTOR/ XSCALE,YSCALE,IDEV
      COMMON /VTEK/ XUPPER,YUPPER,NIBS
      COMMON /VOLAY/ XOFF,YOFF,XLAST,YLAST,XV(10),YV(10),NI
      COMMON /WINDOW/ XLDUM,YLDUM

```

PIMS1B.FTN, CONTINUED

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      DECIDE WHICH SECTION TO GO TO.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      GO TO (100,200,300,400,500,600) IACTON
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      IACTON = 1.   ERASE SCREEN OF GRAPHICS TERMINAL.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
100      CALL WINDOW (XOFF,YOFF,0)
          IACTON = 0
          RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      IACTON = 2.   NO ACTION AT PRESENT
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
200      IACTON = 0
          RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      IACTON = 3.   CALL CURSORS, RETURN VALUES IN X AND Y.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
300      CALL WINDOW (X,Y,-1)
          IACTON = 1
          RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      IN THE REMAINDER OF THE PROGRAM, DATA IS CALCULATED ACCORDING
CC      TO THE PARAMETERS SPECIFIED IN PIMS1A.
CC
CC      THE DATA WHICH IS CALCULATED WILL BE TREATED IN ONE OF SIX
CC      WAYS. IT WILL EITHER BE SAVED IN MEMORY FOR USE IN LATER
CC      INTERPOLATIVE CORRECTIONS, OR IT WILL BE FIT TO A POLYNOMIAL,
CC      LISTED ON THE LINEPRINTER, STORED IN A DATA FILE, PLOTTED,
CC      OR EXAMINED FOR EXTREMA. SUBROUTINES WILL DO MOST OF
CC      THIS ACTUAL WORK, BUT THE WAY IN WHICH THEY ARE USED DEPENDS
CC      A GOOD DEAL ON THE ULTIMATE SOURCE OF THE RAW DATA INVOLVED.
CC      ACCORDINGLY, THE THREE MAJOR SECTIONS BELOW HANDLE DATA
CC      WHICH COMES FROM:
CC          1. THE EQUATION FOR A POLYNOMIAL WHICH HAS BEEN
CC              PREVIOUSLY FIT TO OTHER DATA.
CC          2. DIRECT USER INPUT (IE.,UBP OR UBI DATA TYPES)
CC          3. A FILE OF RAW PIMS DATA (DATA OR RESIDUALS OF ALL
CC              OTHER TYPES).
CC      THE NUMBERS USED IN CASE (1) ARE CALCULATED FROM COEFFICIENTS
CC      STORED IN THE FITPAR ARRAY. THE NUMBERS IN CASE (2) ARE
CC      DERIVED FROM USER INPUT STORED IN THE DATA ARRAY.
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      IACTON = 4.   CASE 1.
C
C      WE WILL BE LISTING, STORING, PLOTTING, OR SEARCHING
C      DATA CALCULATED FROM A POLYNOMIAL EQUATION. WE CAN'T GET
C      HERE WITH MODE = 1 OR 2, AT LEAST IN THEORY. WILL DO 100
C      POINTS BETWEEN THE CURRENT WINDOW ABSCISSA LIMITS, UNLESS
C      THERE ARE NONE (WXMIN = WXMAX = 0.), IN WHICH CASE THE DATA
C      LIMITS WILL BE USED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
400      IACTON = 0
          DELTAX = (WXMAX - WXMIN) / 100.
          WAVE = WXMIN
          TIME = WXMIN
          X = WXMIN - DELTAX

```


PIMS1B.FTN, CONTINUED

```

      IPIR = 0
      DO 2205 I = 1,100
      WAVE = WAVE + DELTAX
      TIME = WAVE
      X = WAVE
      Y = POLCAL (IYTP)
      IF ((X.LT.DXMIN .OR. X.GT.DXMAX) .AND. MODE .NE.6) GO TO 2200
      IF (IPIR) 2103,2103,2105
2103      IPIR = 1
           IMODE = 1
           GO TO 2107
2105      IMODE = 2
2107      GO TO (499,499,2110,2120,2130,2140) MODE
2110      CALL LIST (IMODE)
           GO TO 2200
2120      CALL STORE (IMODE)
           GO TO 2200
2130      CALL PLOT (IMODE)
           IF (IMODE) 2135,2135,2200
2135      IACTON = 2
           RETURN
2140      CALL SEARCH (IMODE)
2200      CONTINUE
2205      CONTINUE
           IF (IPIR .LE. 0) IACTON = 3
           GO TO (499,499,2210,2220,2230,2240) MODE
2210      CALL LIST (3)
           GO TO 499
2220      CALL STORE (3)
           GO TO 499
2230      CALL PLOT (3)
           GO TO 499
2240      CALL SEARCH (3)
499      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      IACTON = 5.      CASE 2.
C
C      THE 'INPUT' DATA IS COMING FROM UBP OR UBI DATA
C      STORED IN THE DATA ARRAY.  SUCH DATA MAY BE FIT, LISTED,
C      STORED, PLOTTED OR SEARCHED.  MODE WILL NOT = 1.
C      EVERY POINT FOUND IN RANGE WILL BE USED; THERE IS NO NEED TO
C      DO ONLY EVERY ISKP POINTS.  SINCE THERE CAN BE ONLY 50
C      POINTS AT MOST.  (IF A SEARCH IS BEING DONE, EVERY
C      POINT IS DEFINED AS BEING 'IN RANGE').
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
500      IACTON = 0
           NTEST = IDTYP (IYTP)
           IPIR = 0
           N = NDATA (IYTP)
           DO 2360 I = 1,N
           X = DATA (IYTP,1,I)
           IF (IXTP .NE. NTEST) X = 12398.541 / X
           Y = DATA (IYTP,2,I)
           IF ((X.LT.DXMIN .OR. X.GT.DXMAX) .AND. MODE.NE.6) GO TO 2360
           IF (IYMOD .NE. 3) GO TO 2310
           Y2 = POLCAL (IYTP)
           Y = Y - Y2
2310      IF (IPIR) 2315,2315,2320
2315      IPIR = 1
           IMODE = 1
           GO TO 2325
2320      IMODE = 2
2325      GO TO (599,2330,2335,2340,2345,2350) MODE
2330      CALL FIT (IMODE)
           GO TO 2360
2335      CALL LIST (IMODE)
           GO TO 2360

```

[illegible]

PIMS1B.FTN, CONTINUED

```

2630      IF (NPHO) 2600,2600,2635
2635      CALL R3 (WAVE,RDAT(10),TIME)
          IF (IYTYP.EQ.10) GO TO 2800
          CORR(10) = YCALC(10)
          GO TO 2600
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      LINE TYPE 3.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2640      IF (NION) 2600,2600,2645
2645      CALL R3 (WAVE, RDAT(11), TIME)
          IF (IYTYP.EQ.11) GO TO 2800
          CORR(11) = YCALC(11)
          GO TO 2600
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      LINE TYPE 4.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2650      IF (NDARK) 2600,2600,2655
2655      CALL R4 (WAVE,RDAT(3),RDAT(4),TIME)
          IF (IYTYP.EQ.3 .OR. IYTYP.EQ.4) GO TO 2800
          IF (ICTYP(3).NE.2) GO TO 2660
          CORR(3) = YCALC (3)
2660      IF (ICTYP(4).NE.2) GO TO 2600
          CORR(4) = YCALC(4)
          GO TO 2600
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      LINE TYPE 8.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2670      IF (NBAK) 2600,2600,2675
2675      CALL R4 (WAVE,RDAT(5),RDAT(6),TIME)
          IF (IYTYP.EQ.5 .OR. IYTYP.EQ.6) GO TO 2800
          IF (ICTYP(5).NE. 2) GO TO 2680
          CORR(5) = YCALC (5)
2680      IF (ICTYP(6).NE. 2) GO TO 2600
          CORR(6) = YCALC(6)
          GO TO 2600
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      LINE TYPE 17.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2690      IF (NREF) 2600,2600,2695
2695      CALL R4(WAVE,RDAT(7),RDAT(8),TIME)
          IF (IYTYP.GE.7 .AND. IYTYP.LE.9) GO TO 2710
          IF (ICTYP(7).NE. 2) GO TO 2700
          CORR (7) = YCALC (7)
          IF (IRF .EQ. 0) RPZ = CORR(7)
2700      IF (ICTYP(8).NE. 2) GO TO 2705
          CORR(8) = YCALC(8)
          IF (IRF .EQ. 0) RIZ = CORR(8)
2705      IF (IRF .EQ. 0) IRF = 1
          GO TO 2600
2710      IF (IYTYP .LE. 8) GO TO 2800
          IDUM = ICTYP(7)
          GO TO (2800,2712,2713,2713) IDUM
2712      CORR(7) = YCALC(7)
2713      IDUM = ICTYP(8)
          GO TO (2800,2714,2800,2800) IDUM
2714      CORR(8) = YCALC(8)
          GO TO 2800

```

PIMS1B.FTN, CONTINUED

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      THE DATA IS NOW AVAILIABLE.  CALCULATE THE ABCISSA TO MAKE
C      SURE IT'S IN RANGE.  IF IT IS, CHECK TO MAKE SURE ISKP POINTS
C      HAVE BEEN SKIPPED.  IF SO, CALCULATE THE ORDINATE AND
C      PERFORM THE DESIRED TASK.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2800      X = XCALC (IXTYP)
          IF ((X.LT.DXMIN.OR.X.GT.DXMAX).AND.MODE.NE.6) GO TO 2600
          IDONE = IDONE + 1
          IF (IDONE .LT. ISKP) GO TO 2600
          IDONE = 0
          IF (IPIR) 2805,2805,2810
2805      IPIR = 1
          IF (ICTYP(7) .LT. 3) GO TO 2806
          RPZ = YCALC(7)
2806      IF (ICTYP(8) .LT. 3) GO TO 2807
          RIZ = YCALC(8)
2807      IMODE = 1
          GO TO 2815
2810      IMODE = 2
2815      Y = YCALC (IYTYP)
          IF (IYMOD .NE. 3) GO TO 2820
          Y2 = POLCAL (IYTYP)
          Y = Y - Y2
2820      GO TO (2825,2830,2835,2840,2845,2850) MODE
2825      CALL INTERP (IMODE)
          GO TO 2600
2830      CALL FIT (IMODE)
          GO TO 2600
2835      CALL LIST (IMODE)
          GO TO 2600
2840      CALL STORE (IMODE)
          GO TO 2600
2845      CALL PLOT (IMODE)
          IF (IMODE) 2135,2135,2600
2850      CALL SEARCH (IMODE)
          GO TO 2600
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      HAVE REACHED THE END OF THE FILE, ONE WAY OR ANOTHER.
C      CLOSE THE INPUT FILE AND MAKE FINAL SUBROUTINE CALLS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
2900      CALL KBACKOF
          END FILE 2
          IF (IPIR .LE. 0) IACTON = 3
          GO TO (2905,2910,2915,2920,2925,2930) MODE
2905      CALL INTERP(3)
          GO TO 699
2910      CALL FIT (3)
          IF (IPIR .LE. 0) FITPAR (IYTYP,12) = 0.
          GO TO 699
2915      CALL LIST (3)
          GO TO 699
2920      CALL STORE (3)
          GO TO 699
2925      CALL PLOT (3)
          GO TO 699
2930      CALL SEARCH (3)
699      RETURN
9000      FORMAT(80A1)
          END

```

FILE11.FTN

```

CCCC FILE11.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                     CC
CC      SUBROUTINE FILE11.FTN      JUNE 18,1977      E.J. DARLAND      CC
CC                                     CC
CC      PURPOSE:  TO INPUT I/O FILE SPECIFICATIONS AND CHECK THEM      CC
CC                TO MAKE SURE THE DEVICES ARE REASONABLE (IE, THAT THEY  CC
CC                EXIST), AND THAT THE SPECIFIED FILE EXISTS OR NOT, AS  CC
CC                APPROPRIATE (SIMILAR TO SUBROUTINE FILE. FT IN PIMS8).  CC
CC                                     CC
CC      USAGE:  CALL FILE11 (FILE,IMODE)      CC
CC                                     CC
CC      PARAMETER DESCRIPTION:      CC
CC      FILE:      A 24-BYTE ARRAY WHICH HOLDS THE FILE      CC
CC                  SPECIFICATION (EG. 'DK0:FILE.FTNK(50,52)')      CC
CC      IMODE:      ON INPUT TO FILE11:      CC
CC                  0:  OUTPUT INTENDED TO FILE      CC
CC                  1:  INPUT INTENDED FROM FILE      CC
CC                  ON RETURN FROM FILE11:      CC
CC                  -1:  A ^G WAS STRUCK IN INPUT OF      CC
CC                      FILE.      CC
CC                  0:  ERROR.      CC
CC                  1:  FILE OKAY TO USE.      CC
CC                                     CC
CC      INTERNAL VARIABLES:      CC
CC      NCHAR:      NO. OF CHARACTERS ENTERED FOR FILE.      CC
CC      ICHAR:      NO. OF CHARACTER THAT IS A COLON, IF ANY.      CC
CC      DEV:      THE CHARACTERS OF FILE UP TO BUT NOT      CC
CC                                     CC
CC      SUBROUTINES CALLED:      FORPIP      CC
CC                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE FILE11 (FILE,IMODE)
      BYTE FILE(24),DEV(4)
      EQUIVALENCE (DEVICE,DEV)
      DIMENSION TEST(11)
      DATA TEST/'LP','KB','FL0','FL1','DK0','DK1','DT0','SY'
      1,'DK','FL','DT'/
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WRITE PROMPT, READ FILE, MAKE SURE ITS MORE THAN ONE
C      CHARACTER LONG.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 5 I = 1,4
      DEV(I) = 32
      WRITE (6,1000)
      READ (6,1001) FILE,NCHAR
      IF (NCHAR .LT. 1) GO TO 40
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CHECK TO SEE IF ANY OF THE CHARACTERS ARE A ^G.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 10 I = 1,NCHAR
      IF (FILE(I) .EQ. 7) GO TO 15
      GO TO 20
      15      IMODE = -1
      DO 17 I = 1,24
      FILE(I) = 32
      17      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      MAKE SURE THE DEVICE NAME IS ONE OF THOSE TO BE ALLOWED.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      20      ICHAR = 0
      DO 25 I = 1,NCHAR

```

FILE11.FTN, CONTINUED

```

25      IF (FILE(1) .EQ. 58) ICHAR = 1
27      IF (ICHR) 40,50,30
30      IF (ICHR .LE. 2 .OR. ICHAR .GE. 5) GO TO 40
        IDUM = ICHAR - 1
        DO 35 I = 1, IDUM
35      DEV(I) = FILE(I)
        DO 37 I = 1,11
37      IF (DEVICE .EQ. TEST(I)) GO TO 50
40      WRITE (6,1002)
45      IMODE = 0
        RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      THE DEVICE NAME IS OKAY.  MAKE SURE INPUT IS NOT EXPECTED
C      FROM THE LP:, THEN CALL FORPIP TO MAKE SURE THE FILE
C      IS / IS NOT THERE, DEPENDING ON WHETHER DOING INPUT OR
C      OUTPUT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
50      IF (DEVICE .NE. TEST(1) .OR. IMODE .EQ. 0) GO TO 55
        WRITE (6,1003)
        GO TO 45
55      IOPT = 0
        CALL FORPIP (FILE,IOPT)
        IF (IOPT .EQ. -2) GO TO 65
        IF (IOPT .EQ. -1) GO TO 70
        IF (IOPT .EQ. 0 .AND. IMODE .EQ. 1) GO TO 75
        IF (IOPT .EQ. 1 .AND. IMODE .EQ. 0) GO TO 80
        IMODE = 1
        RETURN
65      WRITE (6,1004)
        GO TO 45
70      WRITE (6,1005)
        GO TO 45
75      WRITE (6,1006)
        GO TO 45
80      WRITE (6,1007)
        GO TO 45
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      FORMAT STATEMENTS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
1000     FORMAT('S ')
1001     FORMAT(24A1,Q,ERR=40)
1002     FORMAT(' BAD DEVICE!')
1003     FORMAT(' OUTPUT ONLY DEVICE!')
1004     FORMAT(' SYNTAX ERROR')
1005     FORMAT(' FILE ERROR')
1006     FORMAT(' FILE NOT FOUND!')
1007     FORMAT(' FILE ALREADY EXISTS!')
END

```

FIT.FTN

```

CCCC FIT.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC
CC      SUBROUTINE FIT.FTN      JUNE 18,1977      E.J. DARLAND      CC
CC
CC      PURPOSE:  TO PERFORM A LEAST SQUARES FIT TO A POWER      CC
CC                  SERIES POLYNOMIAL EQUATION.  THE METHOD IS TAKEN FROM      CC
CC                  "DATA REDUCTION AND ERROR ANALYSIS FOR THE PHYSICAL      CC
CC                  SCIENCES" BY PHILIP R. BEVINGTON, BUT IS MODIFIED SO      CC
CC                  THAT THE DATA TO BE FIT NEED NOT ALL RESIDE IN MEMORY      CC
CC                  SIMULTANEOUSLY.      CC
CC
CC      USAGE:  CALL FIT (IMODE)      CC
CC                  FIT USES THE CURRENT VALUES OF X AND Y IN COMMON      CC
CC                  STORAGE AS ITS INPUT DATA, AND THUS CAN INPUT ONE POINT      CC
CC                  PER CALL.  THE CALCULATED COEFFICIENTS AND RELATED      CC
CC                  INFORMATION ARE STORED DIRECTLY IN FITPAR.  THE DATA ARE      CC
CC                  SCALED SO THAT DXMIN, DXMAX AT THE TIME OF THE CALL ARE      CC
CC                  REDUCED TO -1.0 AND +1.0, RESPECTIVELY, IN AN ATTEMPT TO      CC
CC                  KEEP THE NUMBERS SMALL ENOUGH TO AVOID OVERFLOW ERRORS.      CC
CC
CC      PARAMETER DESCRIPTION:      CC
CC          IMODE:  DETERMINES THE ACTION TAKEN BY FIT      CC
CC                  1:  INITIALIZE INTERNAL VARIABLES AND      CC
CC                      ARRAYS AND START TO ACCUMULATE THE      CC
CC                      NECESSARY SUMS WITH THE FIRST POINT.      CC
CC                  2:  CONTINUE ACCUMULATING SUMS WITH THE      CC
CC                      CURRENT POINT.      CC
CC                  3:  CALCULATE COEFFICIENTS AFTER ALL SUMS      CC
CC                      HAVE BEEN ACCUMULATED.      CC
CC
CC      INTERNAL VARIABLES:      SEE P.R. BEVINGTON      CC
CC
CC      SUBROUTINES CALLED:      DETERM      CC
CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SUBROUTINE FIT (IMODE)
  DIMENSION ICTYP(12), IDTYP(8)
  DIMENSION FITPAR(12,17)
  COMMON /TYPES/ ICTYP, IDTYP, IXTYP, IYTYP, IYMOD
  COMMON /DATAS/ X,Y,WAVE,TIME
  COMMON /FITSTF/ FITPAR,NORDR
  COMMON /PLOTS/ WXMIN,WXMAX,WYMIN,WYMAX,DXMIN,DXMAX,DYMIN
  1,DYMAX,XDIV,YDIV,IPLT
  DOUBLE PRECISION SUMX(19),SUMY(10),ARRAY(10,10),XTERM,YTERM
  GO TO (10,30,50) IMODE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      INITIALIZE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10      NTERMS = NORDR + 1
      XMEAN = (DXMAX + DXMIN) / 2.0
      XDELTA = (DXMAX - DXMIN) / 2.0
      YMEAN = (DYMAX + DYMIN) / 2.0
      YDELTA = (DYMAX - DYMIN) / 2.0
      FITPAR (IYTYP,17) = YDELTA
      FITPAR (IYTYP,16) = YMEAN
      FITPAR (IYTYP,15) = XDELTA
      FITPAR (IYTYP,14) = XMEAN
      FITPAR (IYTYP,13) = FLOAT (NTERMS)
      FITPAR (IYTYP,12) = 0.
      NMAX = 2 * NTERMS - 1
      DO 15 N=1, NMAX
15          SUMX (N) = 0.
          DO 20 N = 1,NTERMS
20          SUMY(N) = 0.

```

FIT.FTN, CONTINUED

```

      NPTS = 0
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      ACCUMULATE NECESSARY SUMS WITH CURRENT X AND Y, ASSUMING
C      WEIGHT = 1.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
30      NPTS = NPTS + 1
      XT = (X - XMEAN) / XDELTA
      YT = (Y - YMEAN) / YDELTA
      XTERM = 1.0
      DO 35 N = 1, NMAX
      SUMX(N) = SUMX(N) + XTERM
35      XTERM = XTERM * XT
      YTERM = YT
      DO 40 N = 1, NTERMS
      SUMY(N) = SUMY(N) + YTERM
40      YTERM = YTERM * XT
      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CALCULATE COEFFICIENTS.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
50      IF (NPTS .GE. NTERMS) GO TO 54
      WRITE (6,1001)
      GO TO 90
54      DO 55 J = 1, NTERMS
      DO 55 K = 1, NTERMS
      N = J + K - 1
55      ARRAY (J,K) = SUMX(N)
      DELTA = DETERM (ARRAY, NTERMS)
      IF (DELTA) 70, 60, 70
60      DO 65 J = 1, NTERMS
65      FITPAR (IYTYP, J) = 0.
      WRITE (6,1000)
      RETURN
70      DO 85 L = 1, NTERMS
      DO 80 J = 1, NTERMS
      DO 75 K = 1, NTERMS
      N = J + K - 1
75      ARRAY (J,K) = SUMX(N)
80      ARRAY (J,L) = SUMY(J)
      FITPAR (IYTYP, L) = DETERM (ARRAY, NTERMS) / DELTA
      WRITE (6,1999) FITPAR (IYTYP, L)
1999     FORMAT (' 'E15.7)
85      CONTINUE
      FITPAR (IYTYP, 12) = IXYP
90      RETURN
1000     FORMAT (' FIT ERROR.  FIT NOT DONE. ')
1001     FORMAT (' TOO FEW POINTS FOR SPECIFIED ORDER. ')
      END

```

```

CCCC INTERP.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      SUBROUTINE INTERP.FTN      JUNE 18, 1977      E.J. DARLAND      CC
CC
CC      PURPOSE:  TO STORE THE DATA PASSED TO INTERP IN THE ARRAY CC
CC      "DATA" SO THAT PIMS11 CAN USE IT FOR INTERPOLATIVE CC
CC      CORRECTIONS. CC
CC
CC      USAGE:  CALL INTERP (IMODE) CC
CC

```


[illegible]

LIST.FTN, CONTINUED

```

SUBROUTINE LIST (IMODE)
  BYTE FILIN(24), FILOUT(24)
  DOUBLE PRECISION XTYPE(3)
  DIMENSION YTYPE(12), YMOD(3), XLABL(3)
  COMMON /FILES/ FILIN,FILOUT
  COMMON /TYPES/ ICTYP, IDTYP, IXTYP, IYTYP, IYMOD
  COMMON /DATAS/ X,Y,WAVE,TIME
  COMMON /LABELS/ YTYPE, YMOD, XTYPE, XLABL, CTYPE
  GO TO (10,20,30) IMODE
10  CALL ASSIGN (5, 'LP:', 3, IERR)
    WRITE (5,1000) YTYPE(IYTYP), YMOD(IYMOD), XTYPE(IXTYP)
20  WRITE (5,1001) X,Y
    RETURN
30  WRITE (5,1000)
    END FILE 5
    RETURN
1000 FORMAT('1',A4,1X,A4,' VS. ',A8/)
1001 FORMAT(' ',E16.8, 2X, E16.8)
END

```

```

CCCC OUTERP.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC                                     CC
CC      FUNCTION OUTERP.FTN      JUNE 18, 1977      E.J. DARLAND      CC
CC                                     CC
CC      PURPOSE:  TO INTERPOLATE BETWEEN VALUES STORED IN THE      CC
CC                ARRAY "DATA".      CC
CC                                     CC
CC      USAGE:    Y = OUTERP (ITYPE)      CC
CC                OUTERP WILL USE THE CURRENT VALUE OF WAVE, OR TIME, CC
CC                OR OF ENERGY CALCULATED FROM WAVE, AS THE X-VALUE, CC
CC                DEPENDING ON THE UNITS OF THE ABSCISSA DATA FROM WHICH CC
CC                IT IS INTERPOLATING.      CC
CC                                     CC
CC      PARAMETER DESCRIPTION:      CC
CC                ITYPE:  IF POSITIVE, CORRESPONDS TO ONE OF THE      CC
CC                        EIGHT DATA TYPES FOR WHICH INTERPOLATION IS CC
CC                        POSSIBLE (TYPES 1 - 8, SEE COMMON.DOC).      CC
CC                        IF NEGATIVE, OUTERP IS INITIALIZED AND THE CC
CC                        VALUE RETURNED IS 0.0      CC
CC                                     CC
CC      INTERNAL VARIABLES:      CC
CC                XLO(1) =      ONE OF THE (X) VALUES WHICH WERE      CC
CC                        USED IN THE CALCULATION THE LAST TIME THAT CC
CC                        OUTERP WAS REQUESTED TO DO AN INTERPOLATION CC
CC                        ON DATA TYPE 1.      CC
CC                XHI(1) =      THE OTHER OF THE (X) VALUES      CC
CC                        DESCRIBED IN XLO.      CC
CC                XP =      THE ABSCISSA VALUE AT WHICH AN INTERPOLATED CC
CC                        ORDINATE VALUE IS DESIRED.      CC
CC                NPTS =      THE NUMBER OF POINTS EXTANT FOR THE DATA CC
CC                        TYPE OF INTEREST IN THE ARRAY "DATA".      CC
CC                OTHERS:  SELF EXPLANATORY.      CC
CC                                     CC
CC      SUBROUTINES CALLED:  XCALC      CC
CC                                     CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
FUNCTION OUTERP (ITYPE)
  DIMENSION ICTYP(12), IDTYP(8)
  DIMENSION NDATA(8), DATA(8,2,50)
  COMMON /TYPES/ ICTYP, IDTYP, IXTYP, IYTYP, IYMOD
  COMMON /TERPS/ NDATA, DATA
  DIMENSION XLO(8), XHI(8), A(8), B(8)
  IF (ITYPE) 10,10,20

```

OUTERP.FTN, CONTINUED

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      INITIALIZE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10      DO 15 I=1,8
        XLO(I) = 0.
15      XHI(I) = 0.
        OUTERP = 0.
        RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      CALCULATE VALUE OF X TO USE.  IF IT STILL FALLS BETWEEN THE
C      CURRENT VALUES OF XLO AND XHI FOR THAT DATA TYPE, THEN WE HAVE
C      THE PARAMETERS NECESSARY FOR THE INTERPOLATION AND DON'T NEED
C      TO FIND THEM.  OTHERWISE, WE'LL HAVE TO SEARCH THE DATA ARRAY
C      TO FIND WHICH POINTS TO USE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
20      XP = XCALC (IDTYP(ITYPE))
        IF (XLO(ITYPE).LT.XP.AND.XHI(ITYPE).GT.XP) GO TO 60
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      SEARCH DATA ARRAY.  IF IT CONTAINS ONLY ONE POINT, USE IT.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
        NPTS = NDATA (ITYPE)
        IF (NPTS .GT. 1) GO TO 25
        OUTERP = DATA (ITYPE,2,1)
        RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      THERE ARE TWO OR MORE POINTS IN THE ARRAY.  NEED TO FIND TWO
C      WHICH BRACKET THE VALUE OF XP, IF POSSIBLE.  IF NOT POSSIBLE,
C      NEED TO FIND THE TWO THAT ARE CLOSEST.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
25      DO 30 I = 1,NPTS
        IF (DATA (ITYPE,1,I) - XP) 30,35,40
30      CONTINUE
        XHI(ITYPE) = DATA (ITYPE,1,NPTS)
        I = NPTS - 1
        XLO(ITYPE) = DATA (ITYPE,1,I)
        GO TO 50
35      OUTERP = DATA (ITYPE,2,I)
        RETURN
40      I = I - 1
        IF (I .LE. 0) I = 1
        XLO (ITYPE) = DATA (ITYPE,1,I)
        XHI (ITYPE) = DATA (ITYPE,1, (I+1))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      XLO AND XHI ARE NOW SET FOR THE NEXT CALL TO OUTERP, AND WE
C      NOW KNOW THAT WE SHOULD INTERPOLATE BETWEEN OR EXTRAPOLATE FROM
C      THE I-TH AND (I+1)-TH POINTS.  CALCULATE PARAMETERS FOR USE
C      IN THE LINEAR EQUATION  $Y = A + B*DEL$ .
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
50      A(ITYPE) = DATA (ITYPE,2,I)
        B(ITYPE) = DATA (ITYPE,2,(I+1)) - A(ITYPE)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      A AND B ARE SET FOR THE CALCULATION, SO DO IT AND RETURN.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
60      DEL = (XP - XLO(ITYPE)) / (XHI(ITYPE) - XLO(ITYPE))
        OUTERP = A(ITYPE) + B(ITYPE) * DEL
        RETURN
END

```

PLOT.FTN

```

CCCC PLOT.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      SUBROUTINE PLOT.FTN      JUNE 18, 1977      E.J. DARLAND      CC
CC      PURPOSE:  TO PLOT DATA CALCULATED BY PIMS11.      CC
CC      USAGE:    CALL PLOT (IMODE)      CC
CC      PARAMETER DESCRIPTION:      CC
CC          IMODE:  DETERMINES ACTION TAKEN BY PLOT:      CC
CC                  1: INITIALIZE.  DRAW AXES AND TICK MARKS,      CC
CC                      THEN PLOT THE CURRENT VALUE OF X AND Y      CC
CC                      (AS A POINT).      CC
CC                  2: PLOT THE CURRENT VALUE OF X AND Y,      CC
CC                      EITHER AS A VECTOR OR POINT, DEPENDING      CC
CC                      ON IPLT.      CC
CC                  3: FINISH THE PLOT.      CC
CC      NOTE THAT IF AN ATTEMPT HAS BEEN MADE TO SPECIFY A      CC
CC      PLOT SIZE LARGER THAN THE DEVICE CAN HANDLE, IMODE WILL      CC
CC      BE RETURNED AS -1 AFTER AN IMODE=1 CALL TO PLOT.      CC
CC      SUBROUTINES CALLED:  WINDOW      CC
CC      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
SUBROUTINE PLOT (IMODE)
COMMON /VECTOR/ XSCALE,YSCALE,IDEV
COMMON /VTEK/ XUPPER,YUPPER,NIBS
COMMON /VOLAY/ XOFF,YOFF,XLAST,YLAST,XV(10),YV(10),NI
COMMON /PLOTS/ WXMIN,WXMAX,WYMIN,WYMAX,DXMIN,DXMAX,DYMIN
1,DYMAX,XDIV,YDIV,IPLT
COMMON /DATAS/ X,Y,WAVE,TIME
GO TO (5,20,30) IMODE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      INITIALIZE.  FIRST CHECK SIZE OF DEVICE, AND RETURN IMODE =
C      -1 IF TOO SMALL FOR PLOT.
C      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
5      CALL WINDOW (RX,RY,-3)
      IF (RX.GE. XUPPER .AND. RY.GE. YUPPER) GO TO 10
      IMODE = -1
      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      SET IPLT AND IF DOING A PLOT TO THE VERSATEC, INITIALIZE IT.
C      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10     IT = IPLT + 1
      IF (IDEV.NE. 2) GO TO 11
      CALL WINDOW (XOFF,YOFF,0)
      CALL WINDOW (3.,-1.,5)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      DRAW THE AXES AND TICK MARKS
C      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
11     CALL WINDOW (WXMIN,WYMIN,1)
      CALL WINDOW (WXMIN,WYMAX,2)
      CALL WINDOW (WXMAX,WYMAX,2)
      CALL WINDOW (WXMAX,WYMIN,2)
      CALL WINDOW (WXMIN,WYMIN,2)
      DX = WXMAX - WXMIN
      DY = WYMAX - WYMIN
      NX = IFIX (XDIV)
      NY = IFIX (YDIV)
      X1 = (0.02 * DX) + WXMIN

```

PLOT.FTN, CONTINUED

```

      X2 = WXMAX - (0.02 * DX)
      Y1 = WYMIN + (0.75 * 0.02 * DY)
      Y2 = WYMAX - (0.75 * 0.02 * DY)
      DO 14 I = 1, NX
      XT = ((FLOAT(I) * DX) / XDIV) + WXMIN
      CALL WINDOW (XT, WYMIN, 1)
      CALL WINDOW (XT, Y1, 2)
      CALL WINDOW (XT, Y2, 1)
14     CALL WINDOW (XT, WYMAX, 2)
      DO 15 I = 1, NY
      YT = ((FLOAT(I) * DY) / YDIV) + WYMIN
      CALL WINDOW (WXMIN, YT, 1)
      CALL WINDOW (X1, YT, 2)
      CALL WINDOW (X2, YT, 1)
15     CALL WINDOW (WXMAX, YT, 2)
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      C      DO A POINT PLOT OF THE FIRST POINT.
      C
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IF (X .LT. DXMIN) X = DXMIN
      IF (X .GT. DXMAX) X = DXMAX
      IF (Y .LT. DYMIN) Y = DYMIN
      IF (Y .GT. DYMAX) Y = DYMAX
      CALL WINDOW (X, Y, 1)
      CALL WINDOW (X, Y, 2)
      RETURN
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      C      PLOT ANOTHER POINT
      C
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
20     IF (X .LT. DXMIN) X = DXMIN
      IF (X .GT. DXMAX) X = DXMAX
      IF (Y .LT. DYMIN) Y = DYMIN
      IF (Y .GT. DYMAX) Y = DYMAX
      GO TO (25, 22) IPLT
22     CALL WINDOW (X, Y, 1)
25     CALL WINDOW (X, Y, 2)
      RETURN
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      C      FINISH THE PLOT, AND SET IDEV TO 1.
      C
      CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
30     CALL WINDOW (0., 0., 4)
      IDEV = 1
      RETURN
      END

```

```

CCCC POLCAL.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      FUNCTION POLCAL.FRN      JUNE 18, 1977      E.J. DARLAND      CC
CC
CC      PURPOSE:  TO CALCULATE THE ORDINATE VALUE CORRESPONDING      CC
CC      TO THE CURRENT ABSCISSA VALUE, FROM COEFFICIENTS STORED      CC
CC      IN ARRAY "FITPAR".      CC
CC
CC      USAGE:  Y = POLCAL (ITYPE)      CC
CC
CC      DESCRIPTION:  THE POWER SERIES POLYNOMIAL WHOSE VALUE IS      CC
CC      THE DESIRED OUTPUT IS USUALLY WRITTEN AS FOLLOWS:      CC
CC      Y = A0 + A1X + A2X2 + A3X3 + . . . + ANXN      CC
CC
CC

```

POLCAL.FTN, CONTINUED

```

CC          HOWEVER, POLCAL CALCULATES THE RESULTS WITH A MORE COM- CC
CC          PACT METHOD WHICH HAS THE ADVANTAGE OF TENDING TO AVOID CC
CC          ROUND OFF ERRORS AND WHICH IS THEREFORE SOMETIMES MORE CC
CC          ACCURATE. THE EQUATION USED, FOR THE CASE N=3, CAN BE CC
CC          WRITTEN: CC
CC          
$$Y = (((A X)^3 + A^2 X + A) X + A) X + A$$
 CC
CC          BEFORE THE CALCULATION IS PERFORMED, THE ABSCISSA VALUE CC
CC          IS SCALED TO CORRESPOND TO THE SCALING USED WHEN THE FIT CC
CC          WAS PERFORMED. THE SCALED VALUE (X') IS RELATED TO THE CC
CC          ACTUAL VALUE (X) BY: CC
CC          
$$X' = (X - XMEAN) * \frac{1}{DELTX}$$
 CC
CC          WHERE XMEAN IS THE MEAN OF DXMIN AND DXMAX AT THE TIME CC
CC          OF THE FIT, AND DELTX IS THEIR DIFFERENCE. CC
CC          PARAMETER DESCRIPTION: CC
CC          ITYPE: DETERMINES THE DATA TYPE FOR WHICH THE CC
CC                  CALCULATION IS TO BE DONE. MAY HAVE THE CC
CC                  SAME VALUES AS IYTP (SEE COMMON.DOC) CC
CC          CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC CC
CC          FUNCTION POLCAL (ITYPE) CC
CC          DIMENSION ICTYP(12), IDTYP(8) CC
CC          DIMENSION FITPAR(12,17) CC
CC          COMMON /TYPES/ ICTYP, IDTYP, IXTYP, IYTP, IYMOD CC
CC          COMMON /FITSTF/ FITPAR, NORDR CC
CC          XP = XCALC (IDTYP(ITYPE)) CC
CC          XP = (XP - FITPAR (ITYPE,14)) / FITPAR (ITYPE,15) CC
CC          NTERMS = IFIX (FITPAR (ITYPE,13)) CC
CC          YP = 0. CC
CC          DO 10 I=1,NTERMS CC
CC          J = NTERMS - I + 1 CC
10      YP = (YP * XP) + FITPAR (ITYPE,J) CC
CC          POLCAL = (YP * FITPAR (ITYPE,17)) + FITPAR (ITYPE,16) CC
CC          RETURN CC
CC          END CC
CC          CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC CC
CC          CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC CC
CC          SUBROUTINE PLSET.FTN      JUNE 18, 1977      E.J. DARLAND      CC
CC          PURPOSE:  TO INITIALIZE VARIABLES FOR WINDOW/VECTOR USING CC
CC                  THE CURRENT PARAMETERS IN PIMS11. CC
CC          USAGE:  CALL PLSET CC
CC          CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC CC
CC          CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC CC
CC          SUBROUTINE PLSET CC
CC          COMMON /VECTOR/ XSCALE, YSCALE, IDEV CC
CC          COMMON /VTEK/ XUPPER, YUPPER, NIBS CC
CC          COMMON /VOLAY/ XOFF, YOFF, XLAST, YLAST, XV(10), YV(10), NI CC
CC          COMMON /PLOTS/ WXMIN, WXMAX, WYMIN, WYMAX, DXMIN, DXMAX, DYMIN CC
CC          1, DYMAX, XDIV, YDIV, IPLT CC
CC          XSCALE = (WXMAX - WXMIN) / (XUPPER - 1.) CC
CC          YSCALE = (WYMAX - WYMIN) / (YUPPER - 1.) CC
CC          IF (XSCALE .EQ. 0.) XSCALE = .01 CC
CC          IF (YSCALE .EQ. 0.) YSCALE = .01 CC
CC          XOFF = (-WXMIN / XSCALE) + .50 CC
CC          YOFF = (-WYMIN / YSCALE) + .50 CC
CC          RETURN CC
CC          END CC

```

[illegible]

REPORT.FTN, CONTINUED

```

      CALL ASSIGN (5,FILOUT,24,IERR)
      WRITE (5,1000)
      IF (IFIN) 5,5,1
1      WRITE (5,1001) FILIN
5      IF (IYTYP .LE. 2 .OR. IYMOD .EQ. 2) GO TO 10
      WRITE (5,1003) ISKP
10     WRITE (5,1004) YTYPE(IYTYP), YMOD(IYMOD), XTYPE(IXTYP)
      WRITE (5,1005)
      WRITE (5,1006) DYMAX,WYMAX
      WRITE (5,1007) DXMIN,DXMAX,WXMIN,WXMAX
      WRITE (5,1006) DYMIN,WYMIN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      FOR EACH OF THE OTHER DATA TYPES WHICH COULD CONCEIVABLY BE
C      USED AS CORRECTIONS, WRITE THE DATA TYPE AND THE CORRECTION
C      METHOD (N,L,P OR I). IF THE CORRECTION METHOD IS P, THEN
C      ALSO WRITE OUT THE RELEVANT INFORMATION FROM FITPAR. IF THE
C      CORRECTION METHOD IS I, WRITE OUT THE DATA STORED IN ARRAY
C      DATA.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DO 80 I = 1,11
      GO TO (11,12,13,14,15,16,17,18,80,20,20) I
11     GO TO (80,80,30,80,30,80,30,80,30,30,80,30) IYTYP
12     GO TO (80,80,80,30,80,30,80,30,80,30,80,30) IYTYP
13     GO TO (80,80,80,80,30,80,30,80,30,30,80,30) IYTYP
14     GO TO (80,80,80,80,80,30,80,30,30,80,30,30) IYTYP
15     GO TO (80,80,80,80,80,80,30,80,30,30,80,30) IYTYP
16     GO TO (80,80,80,80,80,80,80,30,30,80,30,30) IYTYP
17     GO TO (80,80,80,80,80,80,80,80,30,30,80,30) IYTYP
18     GO TO (80,80,80,80,80,80,80,80,80,30,80,30) IYTYP
20     IF (IYTYP .NE. 12) GO TO 80
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WRITE DATA TYPE AND CORRECTION METHOD
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
30     IDUM = ICTYP(I)
      WRITE (5,1008) YTYPE(I), CTYPE(IDUM)
      GO TO (80,80,40,60) IDUM
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WRITE OUT POLYNOMIAL FIT PARAMETERS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
40     NORDR = IFIX(FITPAR(I,13) - 1.)
      IF (FITPAR(I,12) .EQ. 0.) GO TO 45
      WRITE (5,1009) NORDR, XTYPE(FITPAR(I,12))
      GO TO 50
45     WRITE (5,1010) NORDR
      GO TO 80
50     XM = FITPAR(I,14)
      XD = FITPAR(I,15)
      YM = FITPAR(I,16)
      YD = FITPAR(I,17)
      WRITE (5,1011) XM,XD,YM,YD
      NTERMS = NORDR + 1
      DO 55 N = 1,NTERMS
      J = N - 1
55     WRITE (5,1012) J, FITPAR(I,N)
      GO TO 80
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      WRITE OUT DATA USED FOR INTERPOLATION
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
60     WRITE (5,1013) XTYPE(IDTYP(I))
      N = NDATA(I)
      IF (N) 65,65,70

```


REPORT.FTN, CONTINUED

```

65      WRITE (5,1014)
        GO TO 80
70      DO 75 K = 1,N
75      WRITE (5,1015) DATA (1,1,K), DATA(1,2,K)
80      CONTINUE
        WRITE (5,1000)
        END FILE 5
        RETURN
1000     FORMAT('1')
1001     FORMAT(' FROM FILE ',24A1)
1003     FORMAT(' ONE OUT OF EVERY ', 14,' POINTS')
1004     FORMAT(' ',A4,1X,A4,1X,'VS. ',AB)
1005     FORMAT(16X,'DATA',31X,'WINDOW')
1006     FORMAT(9X,G15.7,22X,G15.7)
1007     FORMAT(1X,2(G15.7,2X,G15.7,5X))
1008     FORMAT('/',',A4,3X,A1)
1009     FORMAT(10X,'ORDER = ',13,', FIT VS. ',AB)
1010     FORMAT(10X,' ORDER = ',13,', BUT NO FIT DONE YET!!!!!!')
1011     FORMAT(10X,'XM = 'G15.7,' XD = ',G15.7/10X,'YM = ',G15.7
1012     1,' YD = ',G15.7)
1012     FORMAT(14X,'A(',11,') = ',G15.7)
1013     FORMAT(10X,'ABSCISSA = ',AB)
1014     FORMAT(10X,'BUT NO POINTS IN ARRAY!!!!')
1015     FORMAT(14X,G15.7,2X,G15.7)
        END

```

```

CCCC SEARCH.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      SUBROUTINE SEARCH.FTN      JUNE 18, 1977      E.J. DARLAND      CC
CC      PURPOSE:  TO FIND THE EXTREMA OF THE ORDINATE AND ABSCISSA      CC
CC      VALUES PASSED TO SEARCH.FTN, AND STORE THEM IN DXMIN,      CC
CC      DXMAX, DYMIN AND DYMAX FOR USE IN PIMS11.      CC
CC      USAGE:  CALL SEARCH (IMODE)      CC
CC      PARAMETER DESCRIPTION:      CC
CC      IMODE:  DETERMINES THE ACTION TAKEN BY SEARCH:      CC
CC      1:  INITIALIZE BY SETTING DXMIN AND DXMAX      CC
CC      TO THE CURRENT VALUE OF X, AND DYMIN      CC
CC      AND DYMAX TO THE CURRENT VALUE OF Y.      CC
CC      2:  COMPARE EXTREMA WITH CURRENT X AND Y,      CC
CC      AND UPDATE ANY EXTREMA WHICH ARE      CC
CC      EXCEEDED BY X OR Y, AS APPROPRIATE.      CC
CC      3:  STORE THE CURRENT VALUES OF THE EXTREMA      CC
CC      IN DXMIN, DXMAX, DYMIN AND DYMAX.      CC
CC      INTERNAL VARIABLES:      CC
CC      XMIN, XMAX,      USED TO HOLD THE CURRENT VALUES OF      CC
CC      YMIN, YMAX      THE EXTREMA.      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE SEARCH (IMODE)
      COMMON /PLOTS/ WXMIN, WXMAX, WYMIN, WYMAX, DXMIN, DXMAX, DYMIN
      1, DYMAX, XDIV, YDIV, IPLT
      COMMON /DATAS/ X, Y, WAVE, TIME
      GO TO (10,20,30) IMODE
10      XMIN = X
      XMAX = X
      YMIN = Y
      YMAX = Y
      RETURN
20      IF (X.LT.XMIN) XMIN = X

```

SEARCH.FTN, CONTINUED

```

      IF (X.GT.XMAX) XMAX = X
      IF (Y.LT.YMIN) YMIN = Y
      IF (Y.GT.YMAX) YMAX = Y
      RETURN
30    DXMIN = XMIN
      DXMAX = XMAX
      DYMIN = YMIN
      DYMAX = YMAX
      WRITE (6,1000) YMAX,XMIN,XMAX,YMIN
      RETURN
1000  FORMAT(8X,E15.7/E15.7,1X,E15.7/8X,E15.7)
      END

```

```

CCCC STORE.FTN CCCC 6/18/77 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CC      SUBROUTINE STORE.FTN      JUNE 18, 1977      E.J. DARLAND      CC
CC      PURPOSE:  TO STORE OUTPUT DATA FROM PIMS11 IN THE OUTPUT      CC
CC      FILE IN ENKE GROUP STANDARE FORMAT (SUITABLE FOR INPUT      CC
CC      TO MULPLT, THE CEMCOMGRAPH FILE PLOTTING PROGRAM.      CC
CC      USAGE:  CALL STORE (IMODE)      CC
CC      PARAMETER DESCRIPTION:      CC
CC      IMODE:  CONTROLS THE ACTION TAKEN BY STORE.      CC
CC      1:  OPENS THE OUTPUT FILE AND WRITES      CC
CC      OUT THE FIRST POINT (THE CURRENT X,Y).      CC
CC      2:  WRITES OUT THE CURRENT X,Y.      CC
CC      3:  CLOSSES THE OUTPUT FILE AFTER WRITING AN      CC
CC      END OF FILE TAG.      CC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE STORE (IMODE)
      BYTE FILOUT (24), FILIN (24)
      COMMON /FILES/ FILIN,FILOUT
      COMMON /DATAS/ X, Y, WAVE, TIME
      GO TO (10,20,30) IMODE
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      OPEN OUTPUT FILE AND WRITE OUT FILE NAME
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
10    CALL ASSIGN (3,FILOUT,24,IERR)
      WRITE (3,1000) FILOUT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      WRITE OUT CURRENT X,Y
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
20    WRITE (3,1001) X,Y
      RETURN
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C      WRITE END OF FILE TAG, CLOSE FILE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
30    WRITE (3,1002)
      END FILE 3
      RETURN
1000  FORMAT ('HD',24A1)
1001  FORMAT ('RD',2E15.7)
1002  FORMAT ('ED')
      END

```

[illegible]

YCALC.FTN, CONTINUED

```

C      DECIDE WHERE TO GO.
C
C      GO TO (20,20,30,40,50,70,90,140,190,230,300,370) ITYPE
20      YCALC = 0.
25      RETURN
C
C      DP. Y = LATEST DP.
C
C      GO TO (30,30,40,50,70,90,140,190,230,300,370) ITYPE
30      YCALC = RDAT(3)
      GO TO 25
C
C      DI. Y = LATEST DI.
C
C      GO TO (40,40,50,70,90,140,190,230,300,370) ITYPE
40      YCALC = RDAT(4)
      GO TO 25
C
C      BP. Y = LATEST BP - DP', WHERE A PRIME INDICATES THE "BEST
C      CURRENT ESTIMATE", OBTAINED IN ONE OF THE FOUR STANDARD
C      METHODS (NONE, LATEST, POLYNOMIAL OR INTERPOLATED).
C
C      BEFORE DOING THE SUBTRACTION, NEED TO MAKE SURE DP' IS SET,
C      AND CALCULATE IT IF NOT.
C
C      GO TO (50,50,60,70,80,90,100,110,120,130,140,150,160,170,180,190,200,210,220,230,240,250,260,270,280,290,300,310,320,330,340,350,360,370) ITYPE
50      IDUM = ICTYP(3)
      GO TO (65,65,55,60) IDUM
55      CORR(3) = POLCAL(3)
      GO TO 65
60      CORR(3) = OUTERP(3)
65      YCALC = RDAT(5) - CORR(3)
      GO TO 25
C
C      BI. Y = LATEST BI - DI'. SET DI' IF NECESSARY, ANALOGOUSLY
C      TO DP' ABOVE.
C
C      GO TO (70,70,80,90,100,110,120,130,140,150,160,170,180,190,200,210,220,230,240,250,260,270,280,290,300,310,320,330,340,350,360,370) ITYPE
70      IDUM = ICTYP(4)
      GO TO (85,85,75,80) IDUM
75      CORR(4) = POLCAL(4)
      GO TO 85
80      CORR(4) = OUTERP(4)
85      YCALC = RDAT(6) - CORR(4)
      GO TO 25
C
C      RP. Y = RP - BI' - DI' - UBP'. THE CALCULATIONS ARE SIMILAR
C      TO THOSE ABOVE, BUT THERE ARE NOW THREE CORRECTION FACTORS TO
C      SET.
C
C      GO TO (90,90,100,110,120,130,140,150,160,170,180,190,200,210,220,230,240,250,260,270,280,290,300,310,320,330,340,350,360,370) ITYPE
90      IDUM = ICTYP(1)
      GO TO (105,20,95,100) IDUM
95      CORR(1) = POLCAL(1)
      GO TO 105
100     CORR(1) = OUTERP(1)
105     IDUM = ICTYP(3)
      GO TO (120,120,110,115) IDUM
110     CORR(3) = POLCAL(3)
      GO TO 120
115     CORR(3) = OUTERP(3)

```

YCALC.FTN, CONTINUED

```

120      IDUM = ICTYP(5)
        GO TO (135,135,125,130) IDUM
125      CORR(5) = POLCAL(5)
        GO TO 135
130      CORR(5) = OUTERP(5)
135      YCALC = RDAT(7) - CORR(5) - CORR(3) - CORR(1)
        GO TO 25
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      RI. Y = LATEST RI - BI' - DI' - UBI'
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
140      IDUM = ICTYP(2)
        GO TO (155,20,145,150) IDUM
145      CORR(2) = POLCAL(2)
        GO TO 155
150      CORR(2) = OUTERP(2)
155      IDUM = ICTYP(4)
        GO TO (170,170,160,165) IDUM
160      CORR(4) = POLCAL(4)
        GO TO 170
165      CORR(4) = OUTERP(4)
170      IDUM = ICTYP(6)
        GO TO (185,185,175,180) IDUM
175      CORR(6) = POLCAL(6)
        GO TO 185
180      CORR(6) = OUTERP(6)
185      YCALC = RDAT(8) - CORR(6) - CORR(4) - CORR(2)
        GO TO 25
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      RPIE. Y = RI' / RP', SO PIMS11 MUST SET RI AND RP BEFORE
C      CALCULATING RPIE.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
190      IDUM = ICTYP(7)
        GO TO (205,205,195,200) IDUM
195      CORR(7) = POLCAL(7)
        GO TO 205
200      CORR(7) = OUTERP(7)
205      IDUM = ICTYP(8)
        GO TO (220,220,210,215) IDUM
210      CORR(8) = POLCAL(8)
        GO TO 220
215      CORR(8) = OUTERP(8)
220      IF (CORR(7)) 221,222,221
221      YCALC = CORR(8) / CORR(7)
        GO TO 25
222      YCALC = .10 E38
        GO TO 25
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C      PHO. Y = (LATEST PHO - DP' - BP' - UBP') * (RPZ / RP')
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
230      IDUM = ICTYP(1)
        GO TO (245,20,235,240) IDUM
235      CORR(1) = POLCAL(1)
        GO TO 245
240      CORR(1) = OUTERP(1)
245      IDUM = ICTYP(3)
        GO TO (260,260,250,255) IDUM
250      CORR(3) = POLCAL(3)
        GO TO 260
255      CORR(3) = OUTERP(3)
260      IDUM = ICTYP(5)
        GO TO (275,275,265,270) IDUM
265      CORR(5) = POLCAL(5)
        GO TO 275

```

YCALC.FTN, CONTINUED

```

270     CORR(5) = OUTERP(5)
275     IDUM = ICTYP(7)
        GO TO (290,290,280,285) IDUM
280     CORR(7) = POLCAL(7)
        GO TO 290
285     CORR(7) = OUTERP(7)
290     YCALC = (RDAT(10)-CORR(1)-CORR(3)-CORR(5))*(RPZ/CORR(7))
        GO TO 25
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     ION.   Y = (LATEST ION - DI' - BI' - UBI') * (RIZ / RI')
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
300     IDUM = ICTYP(2)
        GO TO (315,20,305,310) IDUM
305     CORR(2) = POLCAL(2)
        GO TO 315
310     CORR(2) = OUTERP(2)
315     IDUM = ICTYP(4)
        GO TO (330,330,320,325) IDUM
320     CORR(4) = POLCAL(4)
        GO TO 330
325     CORR(4) = OUTERP(4)
330     IDUM = ICTYP(6)
        GO TO (345,345,335,340) IDUM
335     CORR(6) = POLCAL(6)
        GO TO 345
340     CORR(6) = OUTERP(6)
345     IDUM = ICTYP(8)
        GO TO (360,360,350,355) IDUM
350     CORR(8) = POLCAL(8)
        GO TO 360
355     CORR(8) = OUTERP(8)
360     YCALC = (RDAT(11)-CORR(2)-CORR(4)-CORR(6))*(RIZ/CORR(8))
        GO TO 25
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     PIE.   Y = ION' / PHO'
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
370     IDUM = ICTYP(10)
        GO TO (380,380,375,20) IDUM
375     CORR(10) = POLCAL(10)
380     IDUM = ICTYP(11)
        GO TO (390,390,385,20) IDUM
385     CORR(11) = POLCAL(11)
390     IF (CORR(10) .EQ. 0.0) GO TO 392
391     YCALC = CORR(11) / CORR(10)
        GO TO 25
392     YCALC = .10E38
        GO TO 25
END

```

D. Variable Definitions

1. PIMS8 Variables

This first block of variables is stored in common storage where it can be accessed by any of the PIMS8 programs that are chained into core. The variables are listed in the order in which they appear in the COMMON statement in PIMS8.FT.

- | | | |
|----|-------|--|
| 1 | WSCNT | Step count for the monochromator stepping motor drive, equal to the wavelength times the number of steps per Angstrom. |
| 2 | WSPA | The number of steps per Angstrom. |
| 3 | UPLIM | The upper wavelength limit of the monochromator, in Å. |
| 4 | MWATE | The minimum number of 0.100 mS wait periods per step of the wavelength drive stepping motor. This variable sets the maximum permissible scan rate. |
| 5 | IACCL | An acceleration flag for stepping motor programs:
0: No acceleration to be used.
1: Acceleration is to be used. |
| 6 | INITF | "PIMS8 initialized" flag:
0: PIMS8 has not yet been initialized.
1: PIMS8 has been initialized. |
| 7 | IPREA | Prescaling factor for channel A counter. |
| 8 | IPREB | Prescaling factor for channel B counter. |
| 9 | ITEK | Tektronix terminal flag:
0: Fast terminal (T4010) is not being used, so use short prompts and do not try to plot on the terminal!
1: The Tektronix terminal is being used. |
| 10 | ISCNO | A sequential scan number. It is set when PIMS8 is first initialized, and is stored in each recorded data file and then incremented. |
| 11 | IGOBK | A function switch for CORECT.FT. Set to 0 to cause CORECT to be initialized when it is chained to from PIMS8. |
| 12 | IMO | The number corresponding to the month of the year in the system date word. |

- | | | |
|----|------|---|
| 13 | IDAY | The day of the month from the system data word. |
| 14 | IYR | The number corresponding to the last digit of the year in the system date word. |

The following group of variables is also stored in common, but is used only by PIMS8 and ACQUIR. Since these variables do not need to be saved from one call to ACQUIR to the next, no harm is done if they are overwritten, and these same storage locations are used differently by other system programs such as RECORD and CORECT.

- | | | |
|----|-------|--|
| 15 | IPT | The point number under consideration (being stored, calculated, etc.). |
| 16 | BAKP | Photon background, counts per second. |
| 17 | BAKI | Ion background, counts per second. |
| 18 | CNTP | Integrated photon signal, counts. |
| 19 | CNTI | Integrated ion signal, counts. |
| 20 | TIMEP | Integration time per point, seconds. |
| 21 | NPTS | The number of points to acquire. |
| 22 | ITIM | The number of 0.10 mS periods to integrate each point. |

The following group of common variables is used only by RECORD and PIMS8.

- | | | |
|----|--------|---|
| 15 | DEV | The name of the data storage device. |
| 16 | FILNAM | The name of the data storage file on DEV. |
| 17 | START | Starting wavelength, Angstroms. |
| 18 | FINSH | Ending wavelength, Angstroms. |
| 19 | STEPS | Wavelength increment, Angstroms. |
| 20 | IWAVE | Wavelength scan flag:
0: Wavelength need not be changed during data acquisition.
1: The wavelength will be changed. |

21	DATAS	The number of data points to acquire, and pardon the grammar.
22	TMIN	Mandatory integration time, seconds.
23	TMAX	Maximum permissible integration time, seconds.
24	CDES	Desired number of total integrated counts per point.
25	IREC	Data type flag: 1: Record both photons and ions. 2: Record photons only. 3: Record ions only.
26	TIMES	Electrometer settling time, seconds.
27	WBAK1	First background (stray light) measurement wavelength, Angstroms.
28	TIMEB	Background integration time, seconds.
29	WREF	Reference wavelength, Angstroms.
30	TIMER	Reference integration time, seconds.
31	NPPR	Number of data points to acquire between each reference.
32	TIMED	Dark measurement integration time, seconds.
33	NPPD	Number of points to acquire between dark measurements.
34	FSPHT	Photon scaling factor for real-time plot.
35	FSION	Ion scaling factor for real time plot.
36	CSTP	Number of Angstroms or points per inch for real-time plot.
37	WBAK2	Second background (stray light) measurement wavelength, Angstroms.

2. PIMS11 Variables

The following variables, used in various PIMS11 programs and subroutines, are not all stored in common. They are listed alphabetically.

BELL	00007, or ASCII character "bell".
CORR(I)	Best Current Estimate of data type I, used in correcting other data types.

CTYPE An array of labels for the various correction methods available, i.e., N, L, P, I, and -.

DATA(I,J,K) An array of data to be used in interpolative corrections, where

I = data type (see ITYPE).
J = ordinate/abscissa flag:
 1: abscissa data.
 2: ordinate data.
K = point number, 1-50.
so that DATA(2,1,25) is the 25th UBI abscissa value.

DXMIN
DXMAX
DYMIN
Dymax Data extrema (see Chapter V).

FILIN File specification (device, filename, etc.) for the input file.

FILOUT File specification for the output file.

FITPAR(I,J) An array of parameters relating to polynomial fits for data type I, where the Jth entry is defined as:

J=1: A(0) (coefficient of X^0)
J=2: A(1) (coefficient of X^1)
 .
 .
 .
J=11: A(10) (coefficient of X^{10})
J=12: Type-of-fit flag:
 0: no fit performed yet
 1: the fit is vs. wavelength
 2: the fit is vs. time
 3: the fit is vs. energy
J=13: The number of terms in the fit polynomial equation.
J=14: (DXMAX+DXMIN)/2 at time of the fit
J=15: (DXMAX-DXMIN)/2 at time of the fit
J=16: (DYMAX+DYMIN)/2 at time of the fit
J=17: (DYMAX-DYMIN)/2 at time of the fit

ICTYP(I) Specifies the method of BCE calculation for data type I:

1: no correction
2: latest value
3: calculate from polynomial
4: interpolate from DATA
5: no BCE ever needed.

IDTYPE(I) Specifies the units of abscissa data stored in DATA(I,1,x):

- 1: Angstroms
- 2: Seconds
- 3: Electron Volts

IFIN Input file specified flag:

- 0: no file specified yet
- 1: FILIN is okay to use

ISKP Specifies that, of the output points found between DXMIN and DXMAX, only every ISKPth point will be acted on.

ITYA Temporary storage for current value of IYTYP.

ITYM Temporary storage for current value of IMODE.

IWF Window limit default flag:

- 0: use data limits (DXMIN, ...)
- 1: use window limits (WXMIN, ...)

IXTYP Specifies the units for the abscissa of any output data:

- 1: Wavelength, Angstroms
- 2: Time, Seconds
- 3: Energy, Electron Volts

IYMOD Y axis modifier:

- 1: Output data
- 2: Output data calculated from polynomial equation
- 3: Output residuals (1-2).

IYTYP Specifies the type of data to be output:

- 1: UBP 7: RP
- 2: UBI 8: RI
- 3: DP 9: RPIE
- 4: DI 10: PHO
- 5: SP 11: ION
- 6: SI 12: PIE

LINE Input storage buffer.

MODE Determines which subroutine will be entered once a datum of the desired type is found:

- 1: INTERP 4: STORE
- 2: FIT 5: PLOT
- 3: LIST 6: SEARCH

NPIE	
NPHO	
NION	Line read flags which indicate whether, when reading through an input file, lines with line codes of 1, 2, 3, 4, 8, or 17 (respectively) need be read: 0: skip 1: read
NDARK	
NBAK	
NREF	
NDATA(I)	The number of valid data points in DATA(I,x,x)
NORDR	The order of the polynomial equation to which the data is being fit.
RDAT(I)	The most recently encountered (uncorrected) value for data type I.
RIZ	Equivalent to "Initial RI" in Table V-8.
RPZ	Equivalent to "Initial RP" in Table V-8.
TIME	Most recently encountered value of elapsed time from the input file.
WAVE	Most recently encountered value of wavelength from the input file.
WXMIN	Window extrema. (see Chapter V.)
WXMAX	
WYMIN	
WYMAX	
X	Abscissa value to be output.
XDIV	Number of divisions into which the X-axis will be divided by tick marks.
XLABL	Array of labels for X axis units (ANG., SEC., EV)
XTYPE	Array of labels for X axis types (WAVLNTH, TIME, ENERGY).
Y	Ordinate of value to be output.
YDIV	Similar to XDIV, but for Y axis.
YMOD	Array of labels for Y axis modifiers (DATA, FIT, RES.).
YTYPE	Array of labels for Y axis type (UBP, UBI, DP, BI, ...).

In addition to the variables defined above, PIMS11 uses several common variables to define various features of the plots to be output and to control the output graphics devices. These variables are used by

VECTOR and WINDOW (written by Y.A. Yost and T.V. Atkinson), and are contained in labelled common blocks as defined below.

/VECTOR/

XSCALE	Scales the X value from "data space" into "plotting space" (inches).
YSCALE	Scales the Y value from "data space" into "plotting space" (inches).
IDEV	Specifies which of the available graphics output devices to use.

/VOLAY/

XOFF	X offset in inches, which specifies the distance that the data origin will appear from the lower left corner of the device.
YOFF	Y offset in inches, analogous to XOFF.
XLAST	Variables used by VECTOR, but not by PIMS11.
YLAST	
XV	
YV	
NI	

/VTEK/

XUPPER	Upper X limit of the plot in inches.
YUPPER	Upper Y limit of the plot in inches.
NIBS	Line width, in "number of nibs", where one nib is equivalent to 1/200 inch on the Versatec plotter.

REFERENCES

REFERENCES

1. N.W. Reid, *Int. J. Mass Spectrom. Ion Phys.* 6, 1 (1971).
2. W.A. Chupka, "Photoionization and Fragmentation of Polyatomic Molecules", in Chemical Spectroscopy and Photochemistry in the Vacuum Ultraviolet, ed. by C. Sandorfy, P.J. Ausloos, and M.B. Robin (D. Reidel, Boston, 1974) pp 431-463.
3. H.M. Rosenstock, *Int. J. Mass Spectrom. Ion Phys.* 20, 139 (1976).
4. A.S. Werner and T. Baer, *J. Chem. Phys.* 62, 2900 (1975).
5. W.A. Chupka, "Ion-Molecule Reactions by Photoionization Techniques", in Ion-Molecule Reactions, vol. 1, ed. by J.L. Franklin (Plenum Press, New York, 1972) pp 33-76.
6. A.D. Williamson, PH.D. Thesis, California Institute of Technology, 1976.
7. J.A.R. Samson, Techniques of Vacuum Ultraviolet Spectroscopy (John Wiley and Sons, New York, 1967).
8. A.N. Zaidel' and E.Ya. Shreider, Vacuum Ultraviolet Spectroscopy, trans. by Z. Lerman (Ann Arbor Humphrey Science Publishers, Ann Arbor, 1970).
9. G. Marr, Photoionization Processes in Gases (Academic Press, New York, 1967).
10. P.M. Guyon and J. Berkowitz, *J. Chem. Phys.* 54, 1814 (1971).
11. W.A. Chupka, *J. Chem. Phys.* 30, 191 (1959).
12. W.A. Chupka, *J. Chem. Phys.* 48, 2337 (1968).
13. J. Berkowitz, "Photoionization Mass Spectrometry and Photoelectron Spectroscopy of High Temperature Vapors", in Advances in High Temperature Chemistry, vol. 3, ed. by L. Eyring (Academic Press, New York, 1971) pp 123-176.
14. J.H.D. Eland, Photoelectron Spectroscopy (John Wiley and Sons, New York, 1974).
15. J.A.R. Samson, *Phys. Rept.* 28, 303 (1976).

16. R.W. Ditchburn and F.L. Arnot, Proc. Roy. Soc. A123, 516 (1929).
17. A. Terenin and B. Popov, Phys. Z. Sowjetunion 2, 299 (1932).
18. F.P. Lossing and I. Tanaka, J. Chem. Phys. 25, 1031 (1956).
19. H. Hurzeler, M.G. Inghram, and J.D. Morrison, J. Chem. Phys. 28, 76 (1958).
20. A.J.C. Nicholson, J. Chem. Phys. 39, 954 (1963).
21. E.P. Wigner, Phys. Rev. 73, 1002 (1948).
22. S. Geltman, Phys. Rev. 102, 171 (1956).
23. H.M. Rosenstock, Advan. Mass Spectrom. 4, 523 (1968).
24. K. Watanabe, J. Chem. Phys. 26, 542 (1957).
25. C. Goffart, J. Momigny, and P. Natalis, Int. J. Mass Spectrom. Ion Phys. 3, 371 (1969).
26. D.W. Turner, C. Baker, A.D. Baker, and C.R. Brundle, Molecular Photoelectron Spectroscopy (Wiley-Interscience, New York, 1970).
27. J.W. Rabalais, Principles of Ultraviolet Photoelectron Spectroscopy (John Wiley and Sons, New York, 1977).
28. C.A. McDowell, Mass Spectrometry (McGraw-Hill, New York, 1963).
29. J.H.D. Eland, Int. J. Mass Spectrom. Ion Phys. 8, 143 (1972).
30. B. Brehm, J.H.D. Eland, R. Frey, and A. Küstler, Int. J. Mass Spectrom. Ion Phys. 12, 197 (1973).
31. R. Stockbauer, J. Chem. Phys. 58, 3800 (1973).
32. A.S. Werner, B.P. Tsai, and T. Baer, J. Chem. Phys. 60, 3650 (1974).
33. W.A. Chupka, P.M. Dehmer, and W.T. Jivory, J. Chem. Phys. 63, 3929 (1975).
34. J.H. Beynon, G.R. Cooks, K.R. Jennings, and A.J. Ferrer-Correia, Int. J. Mass Spectrom. Ion Phys. 18, 87 (1975).
35. W.A. Chupka and J. Berkowitz, J. Chem. Phys. 48, 5726 (1968).
36. Diffraction Grating Handbook, Bausch and Lomb, Inc., Rochester, New York.
37. W.L. Stebbings and J.W. Taylor, Int. J. Mass Spectrom. Ion Phys. 9, 471 (1972).

38. G.R. Parr and J.W. Taylor, Rev. Sci. Instrum. 44, 1578 (1973).
39. D. Reinke, R. Kraessig, and H. Bäumgartel, Z. Naturforsch. 28A, 1021, (1973).
40. V.H. Dibeler and R.M. Reese, J. Res. Natl. Bur. Std. 68A, 409 (1964).
41. A.J.C. Nicholson, Vacuum 18, 675 (1968).
42. W.A. Chupka, J. Berkowitz, and K.M.A. Refaey, J. Chem. Phys. 50, 1938 (1969).
43. W.A. Chupka and J. Berkowitz, J. Chem. Phys. 51, 4244 (1969).
44. J. Smets, P. Coppens, and J. Drowart, Chem. Phys. 20, 243 (1977).
45. C.Y. Ng, P.W. Tiedemann, B.H. Mahan, and Y.T. Lee, J. Chem. Phys. 66, 3985 (1977).
46. K.E. McCulloh, J. Chem. Phys. 59, 4250 (1973).
47. S.E. Buttrill, Jr., J.K. Kim, W.T. Huntress, Jr., P. LeBreton, and A.D. Williamson, J. Chem. Phys. 61, 2122 (1974).
48. J.J. Hopfield, Phys. Rev. 35, 1133 (1930).
49. J.J. Hopfield, Phys. Rev. 36, 784 (1930).
50. J.J. Hopfield, J. Astrophys. 72, 133 (1930).
51. T. Takamine, T. Suga, Y. Tanaka, and G. Imotani, Sci. Papers Inst. Phys. Chem. Research (Tokyo) 39, 465 (1942).
52. Y. Tanaka, A.S. Jursa, and F.J. LeBlanc, J. Opt. Soc. Am. 47, 105 (1957).
53. Y. Tanaka and M. Zelikoff, Phys. Rev. 93, 933 (1954).
54. Y. Tanaka and M. Zelikoff, J. Opt. Soc. Am. 44, 254 (1954).
55. Y. Tanaka, J. Opt. Soc. Am. 45, 1044 (1955).
56. R. Huffman, Y. Tanaka, and J.C. Larrabee, Appl. Opt. 2, 617 (1963).
57. H.E. Hinterregger, in Vistas in Astronautics, ed. by M. Aperin and M. Sterns (Pergamon Press, New York, 1958) p 146.
58. B.J. Mulder, J. Phys. E. 10, 490 (1977).
59. model no. 20-800, Megavolt Corp., 150 W. Edsall Blvd., Palisades Park, NJ 07650.

60. R.E. Huffman, J.C. Larrabee, and D. Chambers, Appl. Opt. 4, 1145 (1965).
61. model 605P, Cober Electronics, Inc., 7 Gleason Ave., Stamford, Conn., 06902.
62. designed by M. Rabb, Electronics Designer, Chemistry Dept., Michigan State University.
63. J.T. Cox, G. Hass, J.B. Ramsey, and W.R. Hunter, J. Opt. Soc. Am. 63, 435 (1973).
64. M. Seya, Sci. Light 2, 8 (1952).
65. T. Namioka, Sci. Light 3, 15 (1954).
66. T. Namioka, J. Opt. Soc. Am. 49, 951 (1959).
67. GCA/McPherson Instrument Co., 530 Main St., Acton, Mass., 01720.
68. EMR Photoelectric, Box 44, Princeton, NJ, 08540.
69. RCA Electronic Components, Harrison, NJ, 07029.
70. R.B. Cairns and J.A.R. Samson, J. Opt. Soc. Am. 56, 1568 (1966).
71. model 57, Pacific Photometric Instruments, 5675 Landregan St., Emeryville, CA, 94608.
72. T20 Suprasil fused quartz, Amersil, Inc., 650 Jernees Mill Rd., Sayreville, NJ, 08872.
73. V.H. Dibeler, J.A. Walker, K.E. McCulloh, and H.M. Rosenstock, Int. J. Mass Spectrom. Ion Phys. 7, 209 (1971).
74. W.A. Chupka, J. Chem. Phys. 54, 1936 (1971).
75. C.Y. Ng, B.H. Mahan, and Y.T. Lee, J. Chem. Phys. 65, 1956 (1976).
76. A. Kantrowitz and J. Grey, Rev. Sci. Instrum. 22, 328 (1951).
77. P.C. Killgoar, Jr., G.E. Leroi, J. Berkowitz, and W.A. Chupka, J. Chem. Phys. 58, 803 (1973).
78. G.W. Ray, PH.D. Thesis, Michigan State University, 1978.
79. N.G. Utterback and T. Griffith, Jr., Rev. Sci. Instrum. 37, 866 (1966).
80. C.A. McDowell, Mass Spectrometry (McGraw-Hill, New York, 1963).
81. P.H. Dawson, ed., Quadrupole Mass Spectrometry (Elsevier, New York, 1976).

82. Extranuclear Laboratories, Inc., PO Box 11512, Pittsburgh, PA, 15238.
83. H.F. Dylla and J.A. Jarrell, Rev. Sci. Instrum. 47, 331, (1976).
84. M. VanGorkom and R.E. Glick, Int. J. Mass Spectrom. Ion Phys. 4, 203 (1970).
85. M. VanGorkom, D.P. Beggs, and R.E. Glick, Int. J. Mass Spectrom. Ion Phys. 4, 441 (1971).
86. W.E. Baumgartner and W.K. Huber, J. Phys. E, 9, 321 (1976).
87. Application Note JLI-605, Johnston Laboratories, Inc. [91].
88. N.R. Daly, Rev. Sci. Instrum. 31, 264 (1972).
89. H.W. Werner, H.A.M. DeGrefte, and J.V.D. Berg, Int. J. Mass Spectrom. Ion Phys. 8, 459 (1972).
90. C.Y. Ng, PH.D. Thesis, University of California, 1976.
91. Johnston Laboratories Inc., 3 Industry Lane, Cockeysville, MD, 21030.
92. J. Berkowitz and W.A. Chupka, J. Chem. Phys. 45, 1287 (1966).
93. W.A. Chupka and J. Berkowitz, J. Chem. Phys. 47, 2921 (1967).
94. F. Robben, Appl. Opt. 10, 776 (1971).
95. S.N. Deming and S.L. Morgan, Anal. Chem. 45, 278A (1973).
96. Digital Equipment Corp., Maynard, Mass., 01754.
97. Sykes Datatronics Inc., Rochester, NY, 14606.
98. Tektronix Inc., PO Box 500, Beaverton, OR, 97005.
99. Teletype Corp., 5555 Touhy Ave., Skokie, Ill.
100. Heath Co., Benton Harbor, MI, 49022.
101. J.D. Ingle, Jr., and S.R. Crouch, Anal. Chem. 44, 785 (1972).
102. USM Corp., Gear Systems Division, Wakefield, Mass., 01880.
103. V. Kaufmann and B. Edlén, J. Phys. Chem. Ref. Data 3, 825 (1974).
104. Datametrics Inc., 340 Fordham Rd., Wilmington, Mass., 01887.
105. Keithley Instruments, Inc., 28775 Aurora Rd., Cleveland, OH, 44139.

106. Datel Systems Inc., 1020G Turnpike St., Canton, Mass., 02021.
107. Analog Devices Inc., Norwood, Mass., 02062.
108. model 300 PVB, Electro Scientific Industries, 13900 NW Science Park Dr., Portland, OR, 97229.
109. Linear and Interface Circuits Applications, ed. by D.E. Pippenger and C.L. McCollom (Texas Instruments Inc., Dallas, 1974).
110. D. Gannage, Electroquipment Sales, Southfield, MI, private communication, 1977.
111. C.G. Enke, *Anal. Chem.* 43, 69A (1971).
112.
 - a) H.G. Jackson, *Nucl. Inst. Meth.* 33, 161 (1965).
 - b) E.H. Piepmeier, D.E. Braun, and R.R. Rhodes, *Anal. Chem.* 40, 1667 (1968).
 - c) M.L. Franklin, G. Horlick, and H.V. Malmstadt, *Anal. Chem.* 41, 2 (1969).
 - d) R.H. Eather and D.L. Reasoner, *Appl. Opt.* 8, 227 (1969).
 - e) D. Alger, R.M. Dagnall, B.L. Sharp, and T.S. West, *Anal. Chim. Acta* 57, 1 (1971).
 - f) J.B. Smith, *Rev. Sci. Instrum.* 43, 488 (1972).
 - g) J.F. Skene, D.C. Stuart, K. Fritze, and T.J. Kennett, *Spectrochim. Acta* 29B, 339 (1974).
 - h) H. Glüttinger, M. Gantschi, and E. Serrallach, *J. Phys. E* 9, 936 (1976).
 - i) R.W. Shideler, *Int. J. Mass Spectrom. Ion Phys.* 21, 213 (1976).
 - j) M.K. Kopp and J.A. Williams, *Rev. Sci. Instrum.* 48, 383 (1977).
113.
 - a) Ortec Inc., 100 Midland Rd., Oak Ridge, Tenn., 37830.
 - b) Princeton Applied Research, PO Box 2565, Princeton, NJ, 08540.
 - c) Mech-Tronix Nuclear, 430A Kay Ave., Addison, Ill., 60101.
 - d) Johnston Laboratories Inc., 3 Industry Lane, Cockeysville, MD, 21030.
 - e) Pacific Photometric Instruments, 5675 Landregan St., Emeryville, CA, 94608.
 - f) Jarrell-Ash Div., Fisher Scientific Co., 590 Lincoln St., Waltham, Mass, 02154.
114. K.C. Ash and E.H. Piepmeier, *Anal. Chem.* 43, 26 (1971).
115. D.A. Schoeller and J.M. Hayes, *Anal. Chem.* 47, 408 (1975).
116. J.M. Hayes, D.E. Matthews, and D.A. Schoeller, *Anal. Chem.* 50, 25 (1978).
117. RCA Photomultiplier Manual, 1970 [69].
118. Advanced Micro Devices Inc., 901 Thompson Place, Sunnyvale, CA, 94086.

119. Texas Instruments Inc., Semiconductor Group, Dallas, TX, 75222.
120. Motorola Semiconductor Products Inc., Phoenix, AZ, 85001.
121. M.L. Franklin, G. Horlick, and H.V. Malmstadt, *Anal. Chem.* 41, 2 (1969).
122. L.B. Robinson, *Rev. Sci. Instrum.* 32, 1057 (1961).
123. "Designing with High-Speed Comparators", AMD application note December 1975 [118].
124. The Linear and Interface Circuits Data Book, 1973 [119].
125. Cambion 3265-37-2.
126. Corcom 1K4.
127. General Electric "GE-MOV" metal oxide varistors.
128. Rotron SU2A1 "sprite" fan.
129. Poly Paks, PO Box 942, South Lynnfield, Mass., 01940.
130. See reference [113b].
131. L. Birenbaum and D.B. Scarl, *Appl. Opt.* 12, 519 (1973).
132. L. Heroux, *Appl. Opt.* 7, 2351 (1968).
133. F.T. Arecchi, E. Gatti, and A. Sona, *Rev. Sci. Instrum.* 37, 942 (1966).
134. J.D. Ingle, Jr., and S.R. Crouch, *Anal. Chem.* 44, 777 (1972).
135. Signetics Integrated Circuit Data Book (Signetics Corp., Sunnyvale, CA, 1972).
136. Hewlett-Packard, 1820 Embarcadero Rd., Palo Alto, CA, 94303.
137. Power Designs Inc., 1700 Shames Dr., Westbury, NY, 11590.
138. OS/8 Handbook, 1974, DEC-S8-OSHBA-A-D [96].
139. OS/8 Software Support Manual, 1974, DEC-S8-OSSMB-A-D [96].
140. H.A. Laitinen, Chemical Analysis (McGraw-Hill, New York, 1960) p 545.
141. D.C. Wenke, PH.D. Thesis, Michigan State University, 1972.
142. F. Robben, *Appl. Opt.* 10, 776 (1971).

143. See reference [133].
144. DOS/BATCH Handbook, 1974, DEC-11-ODBHA-A-D [96].
145. F.J. Holler, T.V. Atkinson, T.G. Kelly, and C.G. Enke, American Laboratory, September, 1976, p 9.
146. California Data Processors, 2019 S. Ritchey St., Santa Ana, CA, 92705.
147. Versatec Inc., 2805 Bowers Ave., Santa Clara, CA, 95051.
148. Vector General Inc., 21300 Oxnard St., Woodland Hills, CA, 91364.
149. M.J. Molina and F.S. Rowland, *Nature* 249, 810 (1974).
150. C.J. Noutary, *J. Res. Natl. Bur. Std.* 72A, 479 (1968).
151. T.A. Walter, C. Lifshitz, W.A. Chupka, and J. Berkowitz, *J. Chem. Phys.* 51, 3531 (1969).
152. A.S. Werner, B.P. Tsai, and T. Baer, *J. Chem. Phys.* 60, 3650 (1974).
153. J.M. Ajello, W.T. Huntress, and P. Rayermann, *J. Chem. Phys.* 64, 4746 (1976).
154. H.W. Jochims, W. Lohr, and H. Baumgärtel, *Ber. Bunsenges. physik. Chem.* 80, 130 (1976).
155. R. Gilbert, P. Sauvageau, and C. Sandorfy, *J. Chem. Phys.* 60, 4820 (1974).
156. W.C. Price, T.R. Passmore, and D.M. Roessler, *Discuss. Farad. Soc.* 35, 201 (1963).
157. C. Lifshitz and W.A. Chupka, *J. Chem. Phys.* 47, 3439 (1967).
158. F.H. Dorman, *J. Chem. Phys.* 41, 2857 (1968).
159. H.M. Rosenstock, K. Draxl, B.W. Steiner, and J.T. Herron, *J. Phys. Chem. Ref. Data* 6, supplement no. 1, 1977.
160. JANAF Thermochemical Tables (Dow Chemical Co., Clearinghouse for Federal Scientific and Technical Information, Springfield, VA, 1967).
161. B.K. Hahn and C.G. Enke, *Anal. Chem.* 45, 651A (1973).
162. Mostek Corp., 1215 W. Crosby Rd., Dallas, TX, 75006.
163. Digital Equipment Computer Users Society, Maynard, Mass., 01754, program no. 8-698.