A DATA CONVERTER FOR AN ADAPTIVE PROGRAMMABLE MEASUREMENT SYSTEM

Thesis for the Degree of M. S.
MICHIGAN STATE UNIVERSITY
SIGURD L. LILLEVIK
1974

THESIS

ABSTRACT

A DATA CONVERTER FOR AN ADAPTIVE
PROGRAMMABLE MEASUREMENT SYSTEM

By

Sigurd L. Lillevik

A data converter for an adaptive programmable measurement system has been specified, designed, constructed, and evaluated. Its function is to sample analog voltages and translate them into digital formats compatable with Computer Automation's Alpha-16 minicomputer.

The data converter has been designed to sample 32 analog voltage channels, upward expandable to 256 channels, at a maximum sampling rate of 10,000 samples per second. These voltages may range from -5 V to +5 V and have Thevenin series resistances of less than 1 k ohms. The input impedance of the data converter is a shunt capacitance in parallel with a resistance of 10 megohms. It is shown that as long as the shunt capacitance is less than 0.16 microfarads, the analog signals will not be significantly attenuated. Since the programmable measurement system must be adaptive there are two modes of operation (manual and automatic), each with three schemes of operation (sequential, random, and static). Resolution of the data converter is 2.5 millivolts, and it has an accuracy better than 0.1% of full-scale. The output of the data converter is a 12-bit binary word.

Several experiments were performed on the data converter for the purpose of evaluating its performance. It was demonstrated that

the data converter's control logic functioned as designed, and that the accuracy (0.09% of full-scale) exceeds the required accuracy (0.1% of full-scale). An example is provided which illustrates the basic dynamic properties of the data converter.

A DATA CONVERTER FOR AN ADAPTIVE PROGRAMMABLE MEASUREMENT SYSTEM

By

Sigurd L. Lillevik

A THESIS

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

MASTER OF SCIENCE

Department of Electrical Engineering
and
Systems Science

1974

ACKNOWLEDGMENTS

I wish to thank my major professor, Dr. P. David Fisher, for his guidance during the entire course of this research project. While the thesis was being prepared he provided encouragement and inspiration, for this I am deeply grateful.

Special appreciation is extended to Dr. L. J. Giacoletto, Dr. J. B. Kreer, and Dr. G. L. Park for their thorough review of the thesis. Others who have contributed to the completion of this research project include: Mr. Stephen M. Welch for his advise on software decisions, Mr. James W. Maine for his advise on hardware decisions, and Mr. Gary D. Bauer for his diagnostic program. Finally, I wish to thank my wife, Sandi, for her typing of the thesis and encouragement during the final stages of this research project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I INTRODUCTION

A programmable measurement system has been proposed for the purpose of monitoring and managing pest-crop ecosystems.[1] This facility must be adaptive and must be capable of providing real-time environmental information in a form which is readily available to users involved in the design and implementation of pest-management systems. One essential component in this measurement facility is the DATA CONVERTER (see Figure 1.1). It links environmental signals with the minicomputer. Specifically, its function is to sample the analog voltages on each channel, and translate these into a digital format which is acceptable to the minicomputer.

The purpose of the research project reported here was to specify, design, construct, and evaluate the data converter. The output characteristics of the programmable measurement system, its overall transfer characteristics, and the I/O requirements of the minicomputer, establish the design specifications for the data converter. These design specifications are described in Chapters II and III. The details of the prototyped circuit are presented in Chapter IV. Experimental procedures used to evaluate the data converter are described in Chapter V. The experimental results are also summarized in Chapter V.

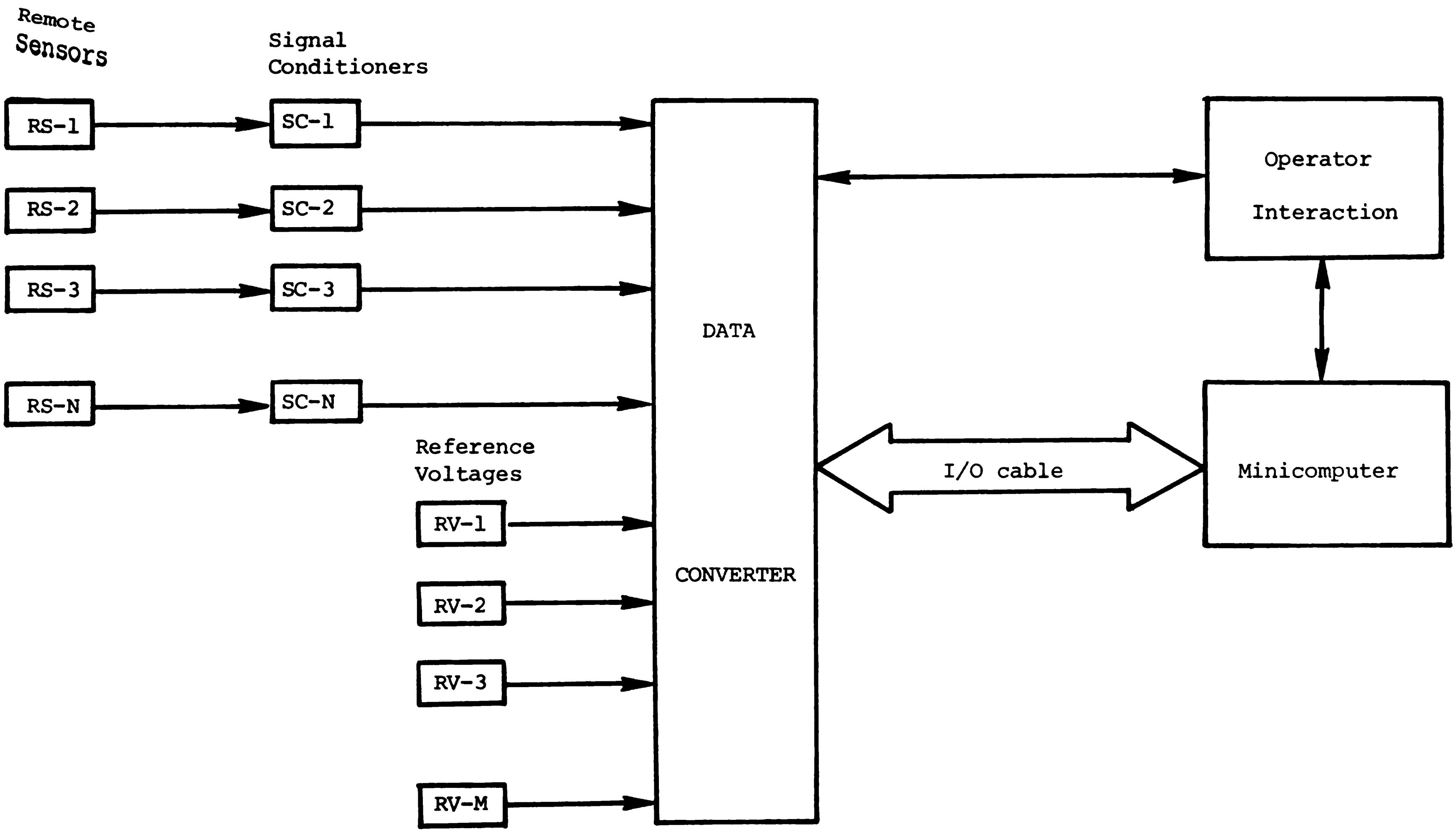

Figure 1.1 Block diagram of the programmable measurement system.

CHAPTER II DESIGN SPECIFICATIONS

The data converter is but one component in the overall programmable measurement system (see Figure 1.1) and, as such, must interact with other components in the system. Its basic functions are to select a predetermined analog signal line, sample the information on the line, code it, and present it to the minicomputer for processing. This sequence of events must be accomplished under program control without the necessity of operator interaction. The principal design specifications placed on the converter depend upon the following:

1) the output characteristics of the signal conditioners,
2) the I/O properties of the minicomputer,
3) the transfer characteristics of the programmable measurement system,
4) the environmental limitations, and
5) the calibration requirements.

These design specifications will be enumerated in this Chapter.

2.1 Output Characteristics of the Signal Conditioners

In the initial application of the programmable measurement system 13 distinct types of environmental signals will be observed (see Table 2.1).[2] From discussions with the users of this equipment it was felt that the data converter should be designed to multiplex 32 analog channels. Also, in the event that, at some future time, it becomes desirable to increase the number of channels, facilities for expansion of up to 256 channels should be provided.

Table 2.1 Representative environmental data required.

Parameter	Number of measurements required per site	Range	Accuracy	Threshold	Maximum rate of change of the parameter per unit of time
Air temperature	1	-55-120°F	$\pm 2^{\circ}$F	- -	2°F/min
Soil temperature	4 (0,1,3,5 in.)	28-130°F	$\pm 2^{\circ}$F	- -	2°F/min(surface) 2°F/hr(sub-surface)
Wind speed	1	0-50 mph	$\pm$1 mph	1 mph	30mph/sec
Wind direction	1	0-360°	$\pm 10^{\circ}$	1°	180°/min
Barometric pressure	1	27.0-31.5 in.Hg.	$\pm$0.2 in.Hg.	27 in.Hg.	.33 in.Hg./hr
Humidity	1	0-95%	$\pm$ 1%	5%	20%/hr
Precipitation	1	0-40"rainfall	$\pm$0.1"	0.1"	15-20"/hr
Soil moisture	1	0-100% field capacity	$\pm$ 17%	0%	33%/hr
Light intensity					
Low	1	0-300 cal/cm^2/d	$\pm$5 cal/cm^2/d	0	- -
High	1	0-10^4 ft-candles	20%	- -	- -

Because of the widely differing characteristics of the environmental data, two sampling modes are needed. The first mode involves a fixed sampling rate, whereby the minicomputer changes channels according to a predetermined pattern. The second mode uses a variable sampling rate, the minicomputer samples channels only when it has received an external stimulus (i.e., an interrupt).

To determine the maximum sampling rate of the data converter, Table 2.1 can be employed as a useful guide. Wind speed has the greatest rate of change of the parameter per unit time, and is found to be a change of 60% of full-scale per second. If 100 microseconds are needed for a complete conversion, then in one second 10,000 conversions are possible. As a worst case analysis, let us assume that 200 channels must be multiplexed, then each channel receives 50 conversions per second. Now if 5 readings are required to accurately determine a full-scale change, then 10 full-scale changes per second will be achieved. Since only 1 full-scale change per second is needed, a total conversion time of 100 microseconds appears adequate for 200 signals similar to wind speed. When fewer than 200 channels are multiplexed, great improvements in the sampling rate is realized.

The signal conditioners present the data converter with a Thevenin equivalent voltage source that may swing -5 V to +5 V, and a Thevenin series resistance of less than 1,000 ohms. The input impedance of the data converter is a capacitance (due to the analog signal cables and other stray capacitances) in parallel with a load resistance. This load resistance is mainly due to the input resistance of the multiplexers. Because the load resistance is much greater than the Thevenin equivalent series resistance (10 megohms compared to 1,000 ohms) the signal conditioner and multiplexer act as an RC low-pass filter. This RC low-pass filter model can be used to determine the maximum allowable capacitance.

Of the environmental signals, wind speed has the highest frequency --1 Hz (see Table 2.1). If we assume, as a worst case analysis, that the highest frequency of the RC low-pass filter is 1,000 Hz, and that the series resistance is 1,000 ohms, then the signal will attenuate 3dB when the capacitance is 0.16 microfarads. As long as the combined capacitance of the analog signal cable, and all associated stray capacitance, is less than 0.16 microfarads, the environmental signal will not be attenuated too severely.

The data converter will have single-ended inputs, thus, there will be a common signal ground for all channels. This analog signal ground will be separate from the digital circuit ground, only coming together at the system power supply ground mecca. This technique of using distinct analog and digital grounds allows separate current returns for the low-frequency components of the analog signals, and the high-frequency components of the digital pulses. (For further details on this subject, Analog Devices has published "Analog-Digital Conversion Handbook."(3))

2.2 Characteristics of the Minicomputer I/O

Computer Automation's Alpha-16 is a general purpose 16-bit word-length digital minicomputer.(4) The I/O data lines transfer information in bit-parallel, word-serial, manner. These lines conform to standard TTL logic levels and loads, and operate asynchronously to accomodate many peripherals with widely varying speeds. Because the minicomputer can perform the data transfers in the microsecond range, as compared to 100 microseconds required for the data converter to operate, time constants in the I/O module are inconsequential with respect to data transfers.

Details of the minicomputer's I/O structure will be discussed in Chapter III.

2.3 Transfer Characteristics of the Overall Measurement System

The measurement system must select a predetermined analog signal line, sample the information on the line, code it, and present it to the minicomputer for processing. This entire operation must be accomplished without the loss of required information. Representing this loss of information is the degree of accuracy required.

Accuracy is the difference between the measured value and the actual value. This difference is usually expressed in terms of percent of full-scale. As noted in Table 2.1 the required accuracy is 1% of full-scale, to provide a margin of safety the overall accuracy of the data converter will be .1% of full-scale.

The overall accuracy of the data converter can be understood in terms of the following parameters:

a) repeatability,

b) linearity,

c) hysteresis,

d) sensitivity,

e) resolution, and

f) threshold.

Each of the above parameters have established meanings for measurement systems (see "Measurement Systems: Application and Design", by Doebelin[5]), and it will be required that the conglomeration of all these errors is no greater than 1 LSB.

2.4 Environmental Limitations

The environmental limitations placed on the programmable measurement systems, will determine the construction and packaging constraints. It is envisioned that the entire system will be housed in a standard 19" rack-type cabinet. This method of mounting is desirable to compliment the rack cabinet of the minicomputer and associated peripherals. The size and weight of the entire unit must be such as to conveniently fit inside an 8 foot by 6 foot by 7 foot portable utility trailer. Because the trailer may be moved from point to point, steps must be taken to eliminate damage caused by excessive vibration.

Power will be available in the form of standard 117 V AC, 60 Hz, presenting no problems. Three power supplies will be distributed throughout the data converter, +15 V DC, -15 V DC, and +5 V DC. These three power supplies will accomodate both analog and digital circuits.

The ambient temperature within the trailer can range from 0 degrees Celsius to 60 degrees Celsius, and the humidity may vary up to 100% noncondensing. In either extreme the measurement system must be operable. These design and packaging limitations should not be overlooked, and must be included in the design considerations.

2.5 Calibration

Several aspects are involved in the measurement system's realiability, or "trust-worthiness". Calibration is an important aspect, for it guarantees that the measurement system has the required transfer characteristics. To facilitate the calibration procedure, no specialized or sophisticated test equipment should be necessary--test equipment should be "built-in" to the circuitry if possible.

Probability of failure should be lowered by providing means by which the minicomputer can evaluate the integrity of major circuits. These checks can be included in the initialization section of all software data taking routines. When an error condition has been detected, some external means of warning should be available. If these few aspects of calibration are implemented into the design, the measurement system's realiability will be increased.

2.6 Summary

At the present time, one can not purchase an "off-the-shelf" commercially manufactured programmable measurement system which meets the above requirements. In the future, it is expected, this will be possible; however, until that time important biological research will be unnecessarily delayed.

An alternative approach is to have the programmable measurement system custom designed, and a prototype built, by another firm. In terms of man-years and cost this is a poor choice. Delivery dates and price quotations can be exaggerated, causing the project time-table to be significantly altered.

By designing and prototyping the measurement system from within the project group personnel, again in terms of man-years and cost, it is usually found that results only differ slightly from an outside firm. Yet several benefits will be derived from an inter-project effort, benefits such as the experience and expertise gained in this area of instrumentation.

CHAPTER III SIXTEEN-BIT INPUT/OUTPUT MODULE

Computer Automation's "16-Bit Input/Output Module"[6] facilitates the unambiguous exchange of information between the central processing unit (CPU) and a peripheral device. The module is comprised of four principal circuits: control, interrupt, input buffer and output buffer. These four circuits are illustrated in Figure 3.1. A general description of each of these circuits will be provided below.

3.1 General Description

Control Circuits- The control circuits "select" and "sense" functions both internal and external to the I/O module. The external control circuits are of primary interest to the digital systems designer because they allow the CPU, under program control, to select unique functions within the peripheral (such as setting a flip-flop which enables the paper tape punch of a teletype), or to sense the status of peripheral defined functions (such as sensing a flip-flop output to determine whether the paper punch of a teletype is indeed in the desired state).

Interrupt Processing Circuits- The interrupt processing circuits respond to externally generated interrupt conditions and generate a vectored interrupt request. The CPU responds by branching to the vectored software address and executing the instruction at this address. The next branching of the CPU would depend upon the nature of this instruction and the priority level of the interrupt. As a specific example, the instruction may authorize the CPU to jump to an address and enter a data acquisition routine.

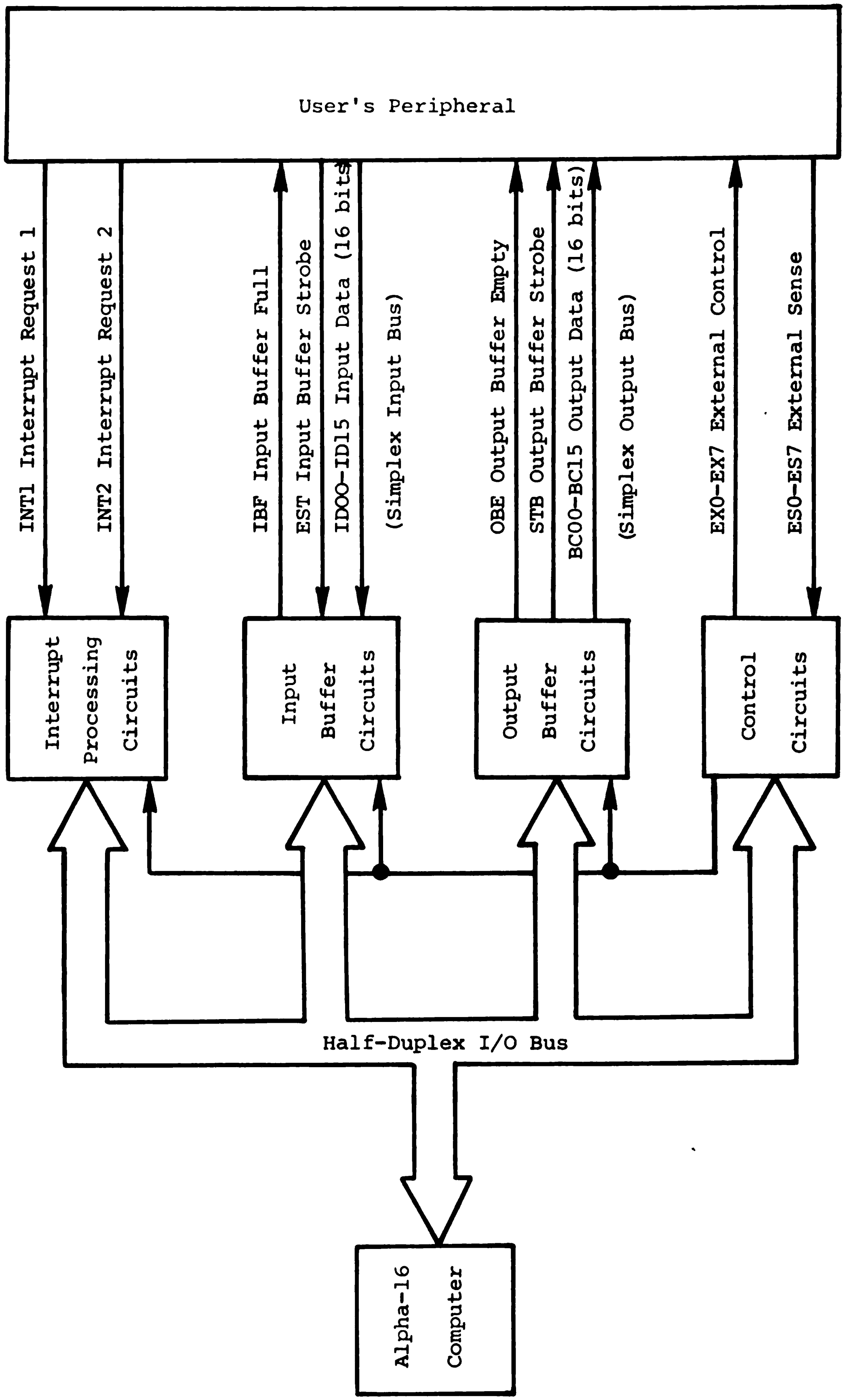

Figure 3.1 Block diagram of the 16-bit I/O module.

Input Buffer Circuits- The input buffer circuits temporarily store peripheral input data until the CPU is ready to accept the data. If the buffer is empty, the peripheral device places the input data on the input lines and generates a strobe. The strobe clocks the data into the input buffer register.

Output Buffer Circuits- The output buffer circuits temporarily store CPU output data until the peripheral device can accept the data. The CPU loads the data into the output buffer register. When the peripheral device is ready, it reads the output buffer register.

3.2 Loading of the Signal Lines

The artificial boundary that exists between the I/O module and the peripheral device is composed of 90 signal lines. For a more detailed explanation, see Appendix B. All of these lines are unidirectional, and each line terminates on the module at either a signal driver (for output signals from the module) or a signal receiver (for input signals to the module). The signal drivers are open-collector NPN transistors. These driving transistors may each be "pulled-up" to +15 V DC with a collector resistor, and can sink up to 40 milliamperes when driven to ground. These open-collector drivers are very flexible, and in some applications may be used as a "wired-or" gate. The signal receivers present one TTL unit load to the peripheral device. (A standard TTL load is -1.6 milliamperes into the module for a logical "0" voltage level, and 40 microamperes into the module for a logical "1" voltage level. Ideal TTL voltage levels are 4.5 V to 5.0 V for a logical "1" or high, and 0.0 V to 0.8 V for a logical "0" or low.[7])

A standard TTL inverter, a 7404, is recommended by Computer Automation as a peripheral signal driver because it has a fan-out of

10 unit loads. As a peripheral signal receiver, Computer Automation recommends any standard TTL gate with the multiple emitter input transistor. These loading requirements are very modest since the I/O module was designed to see standard 7400 series TTL circuitry.

Computer Automation also recommends twisted-wire pairs be used in the I/O cable, and that there be at least eighteen twists per foot. With every twisted-wire pair one lead is always signal ground. To terminate the I/O cable, at the peripheral, a 330 ohm 1/4 W 10% resistor is recommended. Computer Automation suggests the use of a 220 ohm 1/4 W 10% resistor for the open-collector driver "pull-up". These strict loading and matching rules will keep signal deterioration to a minimum along the I/O cable.

3.3 Software Implementation

There exist four types of I/O instructions applicable to the "16-Bit Input/Output Module": Sense, Select, Input, and Output. Sense instructions may be used to examine unique functions within the peripheral (using ES00-ES07) and to perform conditional software branches on the results of these status checks. The Select instructions may be used to control a given function within the peripheral (using EX00-EX07). Input instructions read data from the input buffer into one of two working CPU registers. The Input instructions may be combined with the Sense instructions to form one conditional Input instruction. Output instructions move data from one of two CPU-working registers to the output buffer, and may be combined with the Sense instructions to form one conditional Output instruction.

Two special features of the CPU are the ability to perform 8-bit (or byte) input and output transfers, and the ability to mask (logically "And") input data with the receiving register. These two special CPU features, along with conditional I/O, make the software instruction set extremely flexible and powerful.

For high speed data transfers directly to or from memory and the peripheral, Block I/O and Auto I/O instructions are used. Both instructions have multiple word formats (two or three words), and during execution, may be thought of as direct memory channels. Block I/O has been designed for in-line programming, whereas Auto I/O for servicing peripheral interrupts. These two high-speed instructions may be used when I/O transfers must be made every 5 machine cycles.

3.4 Summary

The CPU can easily control and sense functions within the peripheral and set up high and low-speed data transfers. The I/O module provides two very convenient digital buffers, and has been designed to see standard 7400 series TTL circuitry. Finally, the I/O module will enable the peripheral to utilize the powerful and sophisticated instruction set of the minicomputer.

Numerous trade-offs exist between the software and the hardware. The module has been skillfully designed to encompass a wide spectrum of applications, and because of this, unavoidable ambiguities exist. Let's say ITRAN was enabling the input buffer continously, why then is EST necessary? The obvious answer is that an EST strobe is required when ITRAN is off. The computer can be programmed to operate a peripheral in a sequential manner, and it may not be necessary to use the buffer status lines (IBF, OBE). If the digital systems designer takes advantage of these software-hardware trade-offs, the interface design can be greatly simplified in certain applications.

CHAPTER IV CIRCUIT REALIZATION

The design specifications require that an analog-to-digital converter (ADC) and an associated analog voltage signal multiplexer be interfaced to the minicomputer via the 16-bit I/O module. Realization of the required circuitry involves the following sequence:

a) the theory of operation must be conceived,

b) specific components must be identified,

c) the circuit must be described (schematics drawn), and

d) the prototype must be built.

This chapter describes the above process, and reveals the implementation of the design specifications delineated in Chapters II and III.

4.1 Theory of Operation

Either a minicomputer under program control or an operator must be able to interact with the data converter so as to control the ADC, select a multiplexer address, and read the converted data word. A general block diagram of such a system is depicted in Figure 4.1.

Because both the minicomputer and the operator must be able to have control of the data converter there are two fundamental modes of operation: manual and automatic. To distinguish between these modes a Gating scheme is employed.

The Multiplexer Address Generator (MAG) has been designed to scan between a lower multiplexer channel limit and an upper channel limit minus one. Upon loading new limits into the MAG, it presets the multiplexer

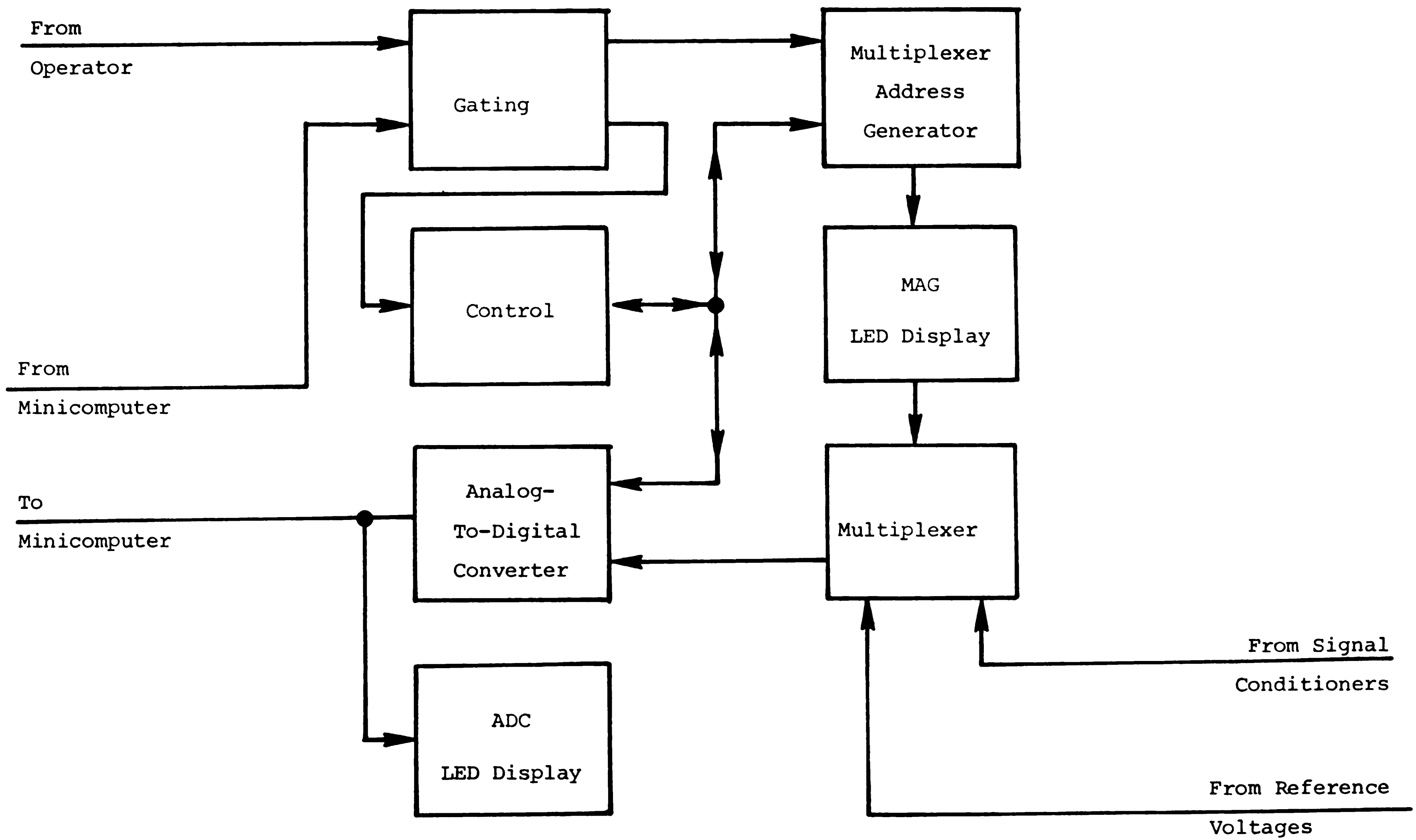

Figure 4.1 Block diagram of the data converter.

to the lower limit. After the ADC has completed each conversion, the MAG increments the multiplexer by one channel. When the upper channel minus one is reached, the MAG automatically presets the multiplexer back to the lower limit. The output of the MAG, the Multiplexer address, is available to the operator on a light emitting diode (LED) display.

The MAG may be operated using one of the following schemes; namely sequential, static, and random. When the MAG holds limits two or more channels apart and the upper limit is greater than the lower limit, sequential operation is possible. This type of operation involves scanning from the lower channel to the upper channel minus one, presetting back to the lower limit and so forth.

If the limits loaded in the MAG are one channel apart and the upper limit is greater than the lower limit, then static operation will occur During this type of operation the MAG increments the multiplexer only one channel then reaches the upper limit. It then presets back to the lower limit or original channel. Static operation enables one to take multiple readings of the same Analog Voltage Channel.

The data converter can also be operated randomly by loading the MAG with new limits (and presetting) before each ADC conversion. Random operation requires an extra step (loading the MAG) compared to sequential or static operation, but the data converter can "skip" channels, providing maximum freedom.

The Multiplexer receives an address from the MAG and decodes it to select an Analog Voltage Channel. The selected analog signal is "caught" by a sample-and-hold circuit while being converted to a digital format by the ADC. Results are available on a LED diplay, or can be read by the minicomputer.

4.2 Component Identification

After the basic theory of operation has been identified, specific components must be selected. These components are of three classes: linear, linear-digital, and digital. Justification for the selection of the components to be discussed, is focused on the requirement that a .1% of full-scale accuracy be maintained. The agglomeration of the selected components must produce this accuracy, and experimental data verifying this will be provided in Chapter V.

It was decided that the linear and linear-digital components be obtained from Datel Systems, Inc.[8] The linear components include the MM8 eight channel analog multiplexer, and the SHM-1 sample-and-hold module. The lone linear-digital component is an ADC-M12B converter, a 12-bit successive-approximation type analog-to-digital converter. Both the linear and linear-digital components are available as a modular package (the DAS-16). This package also includes additional system control logic and an LED display.

The digital components were purchased from Signetics, Corp.[9] and included the following:

a) 7400-Quad 2-Input Positive Nand Gate;

b) 7404-Hex Inverter;

c) 7475-Quad Latch;

d) 7485-4-Bit Magnitude Comparator;

e) 74121-Monostable Multivibrator;

f) 74193-Synchronous Up/Down Binary Counter.

The digital components are all standard 7400 series TTL logic, and are compatible with both the Datel product line and Computer Automation's 16-bit I/O Module.

Once the basic modules (the linear, linear-digital, and digital) have been selected, the schematic diagrams must be drawn. This process can be difficult due to the many subtleties involved, such as, maintaining logic polarities and fan-out/fan-ins, matching module properties, cascading similar components, and maintaining the proper timing characteristics.

4.3 Circuit Description

The circuit schematics are comprised of nine drawings, each representing distinct aspects of the overall data converter. This section will describe each drawing, delineating the data signal lines from the control signal lines. It is important to note that the number within each symbol corresponds to a physical location.

Lower Register- Figure 4.2 depicts the Lower Register. It acts to store the MAG lower limit, and originates the Multiplexer address. Integrated circuits (IC's) 1 to 6 are 7400 Quad Nand Gates and implement the Gating scheme represented in Figure 4.1. These Gates accept the data lines BC0-BC7 from the minicomputer (lower 8-bits) or the data lines SRLW0-SRLW7 defined by the operator. Control signals SRENBL and $\overline{\text{SRENBL}}$ select which set of data lines are to be presented to the 7475 Quad Latches (IC's 7 and 8). These control signals are originated by the operator and are changed by using a toggle switch. When the Quad Latches are strobed by control signal CLK, the 74193 Binary Counters (IC's 9 and 10) are also preset, by control signal PRESET, to the same number being strobed into the Quad Latches. The Binary Counters are cascaded to form an 8-bit counter whose output represents the output of the MAG. Control signal COUNT increments the counter, which changes the MPX0-MPX7 lines and the Multiplexer channel.

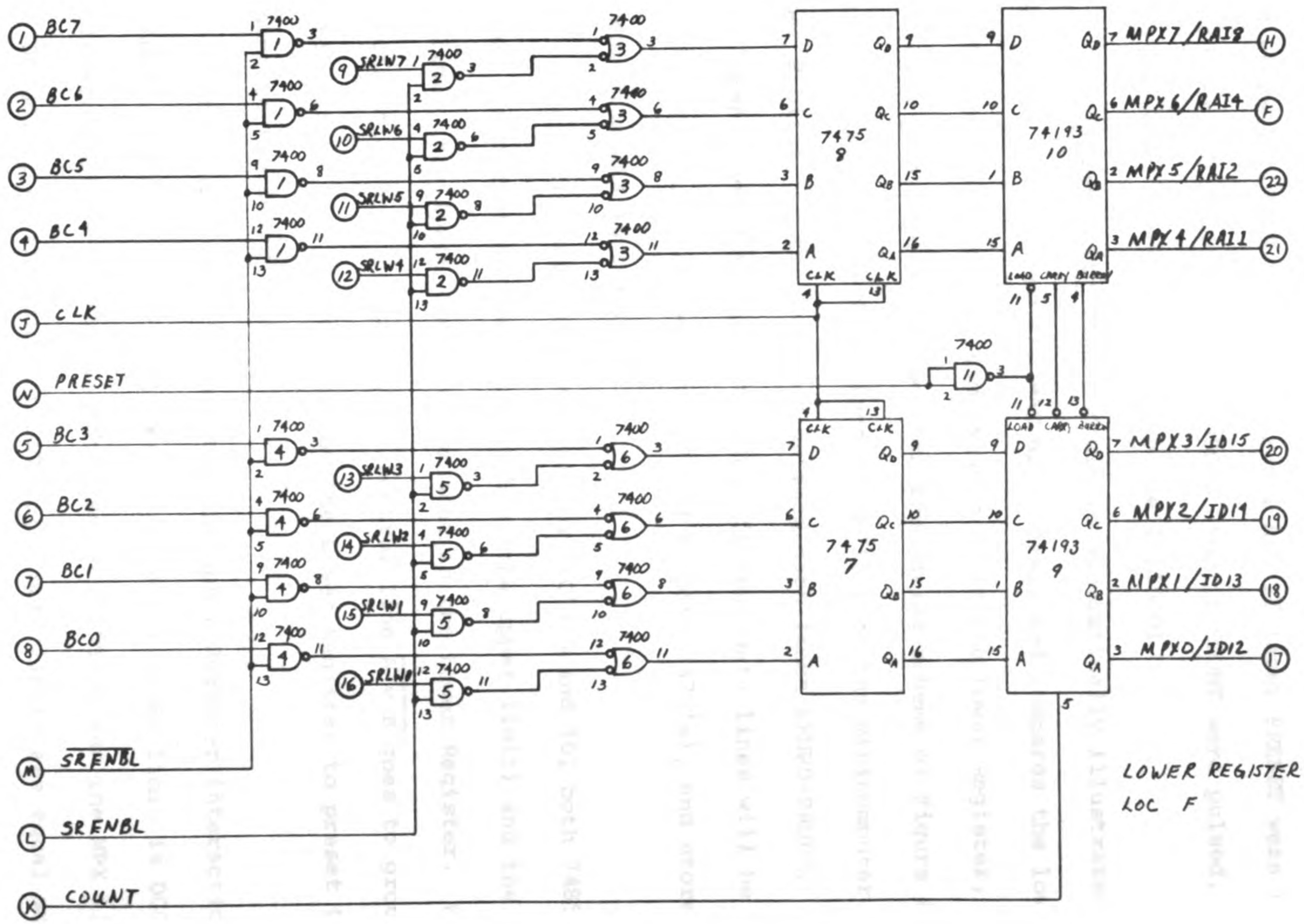

Figure 4.2 Data converter lower register circuit.

To help illustrate the above consider the following example: Suppose the minicomputer's output buffer contained 11110000 (lower 8-bits) and $\overline{\text{SRENBL}}$ were high, (SRENBL low). If CLK were strobed the Quad Latches would store 11110000 (IC 7 all zeros), and if later PRESET were strobed the counter would preset to 11110000. Now if COUNT were pulsed, then the MPX lines would increment to the value 11110001.

Upper Register- This circuit is schematically illustrated in Figure 4.3. It stores the MAG upper limit and compares the lower and upper limits. In much the same manner as in the Lower Register, IC's 1 to 6 (7400 Quad Nand Gates) perform the Gating scheme of Figure 4.1. These Gates accept the data lines BC8-BC15 from the minicomputer buffer (upper 8-bits), or the operator defined data lines SRUP0-SRUP7. Control lines SRENBL and $\overline{\text{SRENBL}}$ select which of these data lines will be presented to the Quad Latches (IC's 7 and 8, both 7475's), and stored when control line CLK is strobed.

The 4-Bit Magnitude Comparitors (IC's 9 and 10, both 7485's) monitor the output of the Quad Latches (the upper limit) and the MPX0-MPX7 lines (the lower limit) originated in the Lower Register. When these two sets of data lines are equal, control line $\overline{\text{A = B}}$ goes to ground. This control signal will be utilized by the Lower Register to preset the 8-bit counter to the lower limit.

An example that illustrates the Upper Register interaction with the Lower Register is as follows: Suppose the upper limit is 00001000 and the lower limit is 00000000. After seven COUNT pulses the MPX lines contain 00000111, on the eighth COUNT pulse both limits are equal and $\overline{\text{A = B}}$ will go low. This change will cause PRESET to be pulsed and the MAG will be preset to the lower limit, or 00000000.

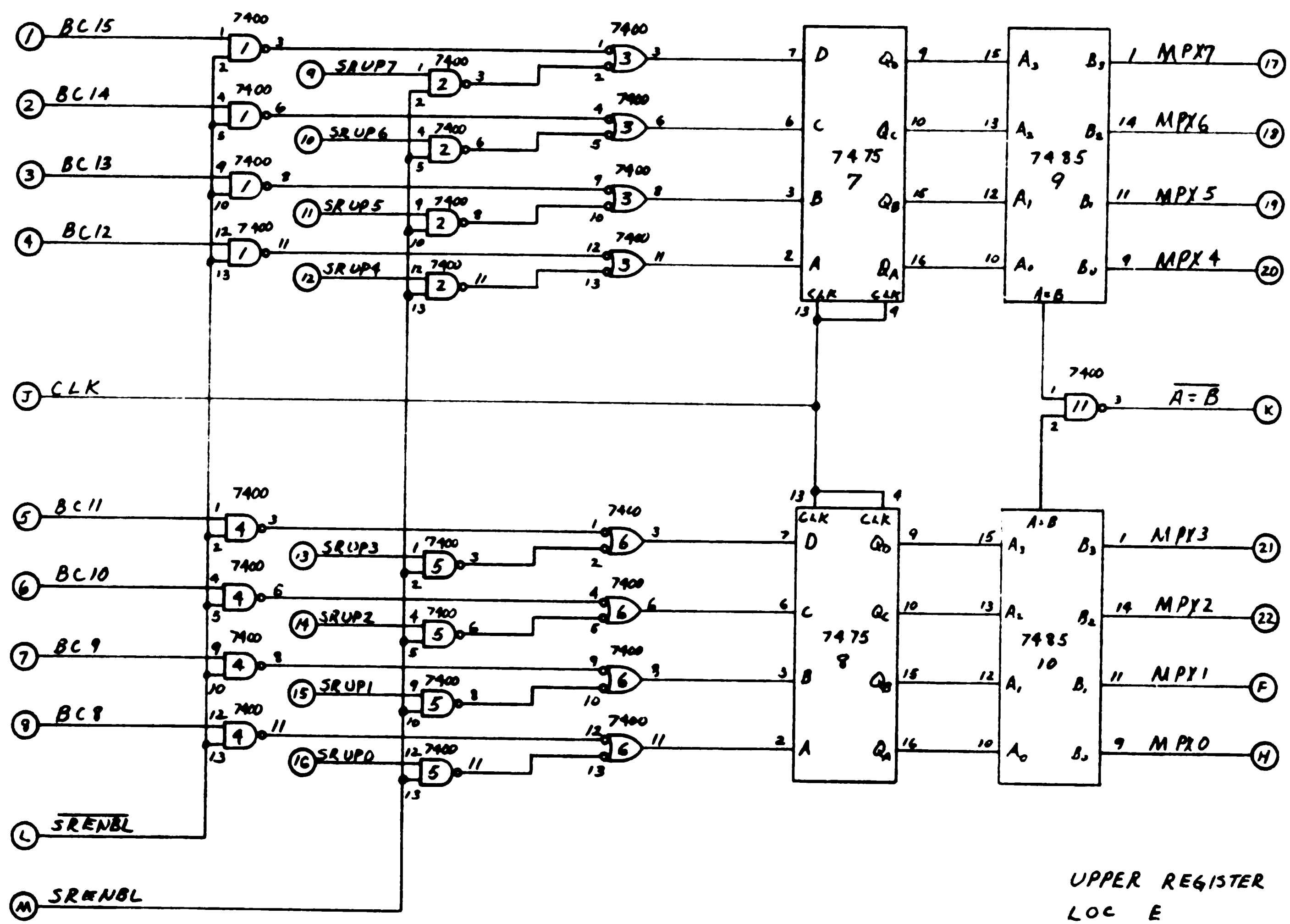

Figure 4.3 Data converter upper register circuit.

Switch Register- In the two previous diagrams (Lower Register and Upper Register) the data lines which establish the lower and upper limits were from the minicomputer, the BC0-BC7 and BC8-BC15 lines, and the operator SRLW0-SRLW7 and SRUP0-SRLW7. The Switch Register circuit (Figure 4.4) establishes the latter two data lines.

Both sets of data lines employ the same signal scheme, a SPST toggle switch either closes (for a zero) or remains open. In the open position a 4.7 K ohm 1/4 watt resistor is used as a "pull-up" to provide the 5 V "one" level. These signal lines are then used to establish the lower and upper limits of the MAG in the manual mode.

Control signals also originate in the Switch Register circuit. SRENBL and $\overline{\text{SRENBL}}$ are formed here by a DPDT switch. These control lines establish the two basic modes of the data converter operation, i.e., manual and automatic. Two other control signals, LOAD and CNVRT, are also found here. These signals are activated by depressing normally-open momentary toggle switches. Through the use of RC contact bounce elimination circuits, these toggle switches develop 1 microsecond pulses.

The last circuit in the Switch Register is a LED driver and display circuit. The LED's are to be read by the operator to determine the current Multiplexer (MPX lines) address. The drivers utilize an RCA[10] IC, the CA3081, which contains seven common-emitter NPN transistors. The transistors are biased "on" by the digital counter and are essentially no "load" on the TTL logic due to the 10 K ohm base resistor. The 150 ohm resistor in series with the LED's serve to limit the power dissipated in the LED's. This type of driver circuit is ideal for this application because it does not affect the fan-out of the digital logic.

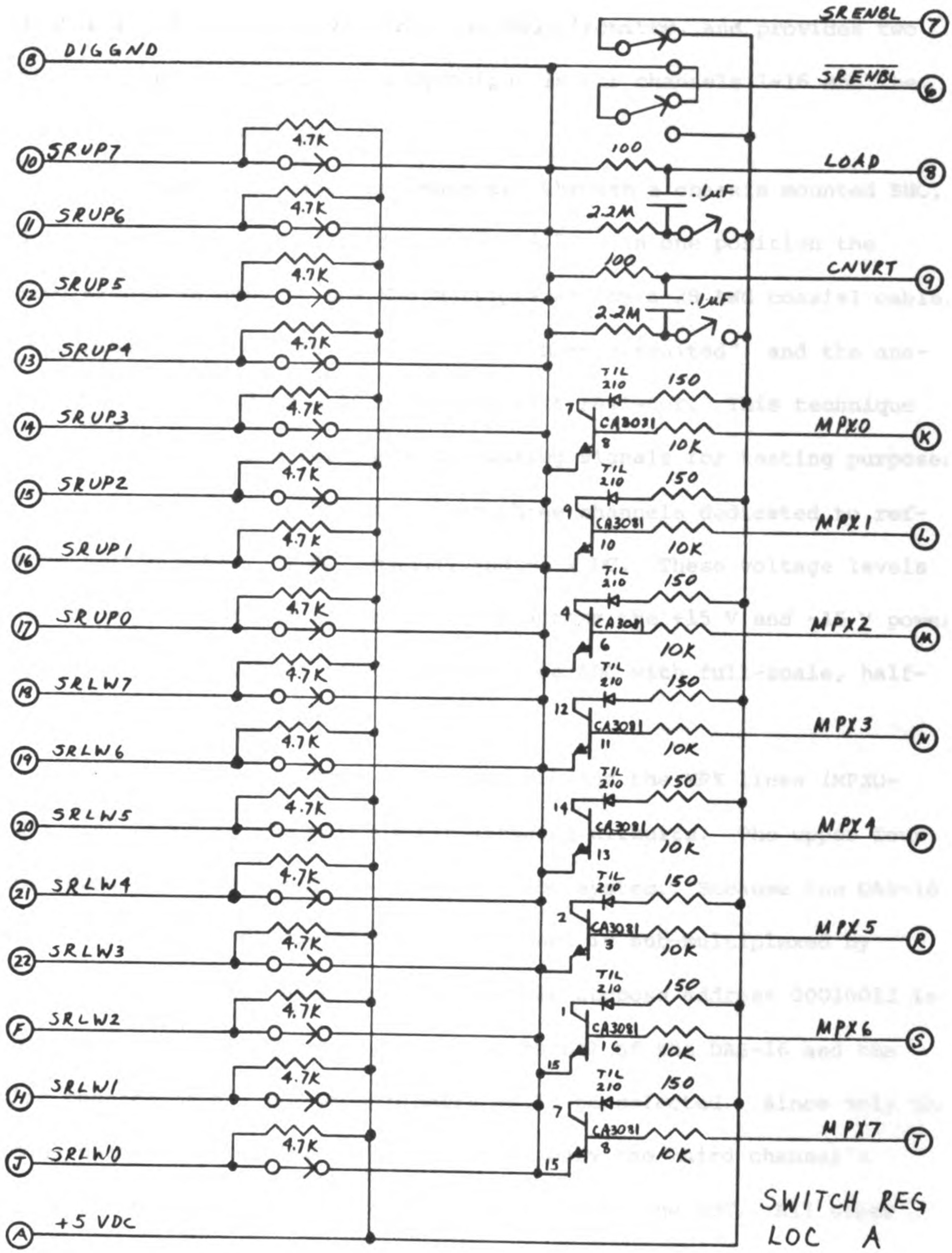

Figure 4.4 Data converter switch register circuit.

Multiplexer- This circuit accepts 32 analog voltages, decodes the digital address received from the MAG circuitry, and provides two analog outputs. The first analog output is for channels 1-16 and the second for channels 17-32.

The analog signals are connected through a chassis mounted BNC, and to a DPDT toggle switch (see Figure 4.5). In one position the analog signal is sent to the MM8 multiplexers by a 28 AWG coaxial cable. In the other position the MM8 input is "short-circuited", and the analog signal is loaded by a 10 K ohm 1/4 watt resistor. This technique is used to isolate and terminate the analog signals for testing purposes.

The Multiplexer has the first three channels dedicated to reference voltages of +5 V DC, ground, and -5 V DC. These voltage levels arise from a resistance voltage divider across the +15 V and -15 V power supplies. These voltage levels provide the ADC with full-scale, half-scale, and zero readings.

Only the lower significant four bits of the MPX lines (MPX0-MPX3) are needed in the multiplexer decoding circuits. The upper four bits (MPX4-MPX7) are used by the DAS-16 ADC system. Because the DAS-16 has 16 channels of multiplexing each channel is sub-multiplexed by another 16 channels. As a specific example suppose address 00010011 is contained in the MPX lines, the second channel of the DAS-16 and the third channel of all sub-multiplexers would be selected. Since only the second channel of the DAS-16 is selected, only the third channel's signal of the second group would by converted by the ADC. All other sub-multiplexer third channels would not pass through the DAS-16 multiplexers.

To achieve 16 channels of multiplexing from the MM8 modules

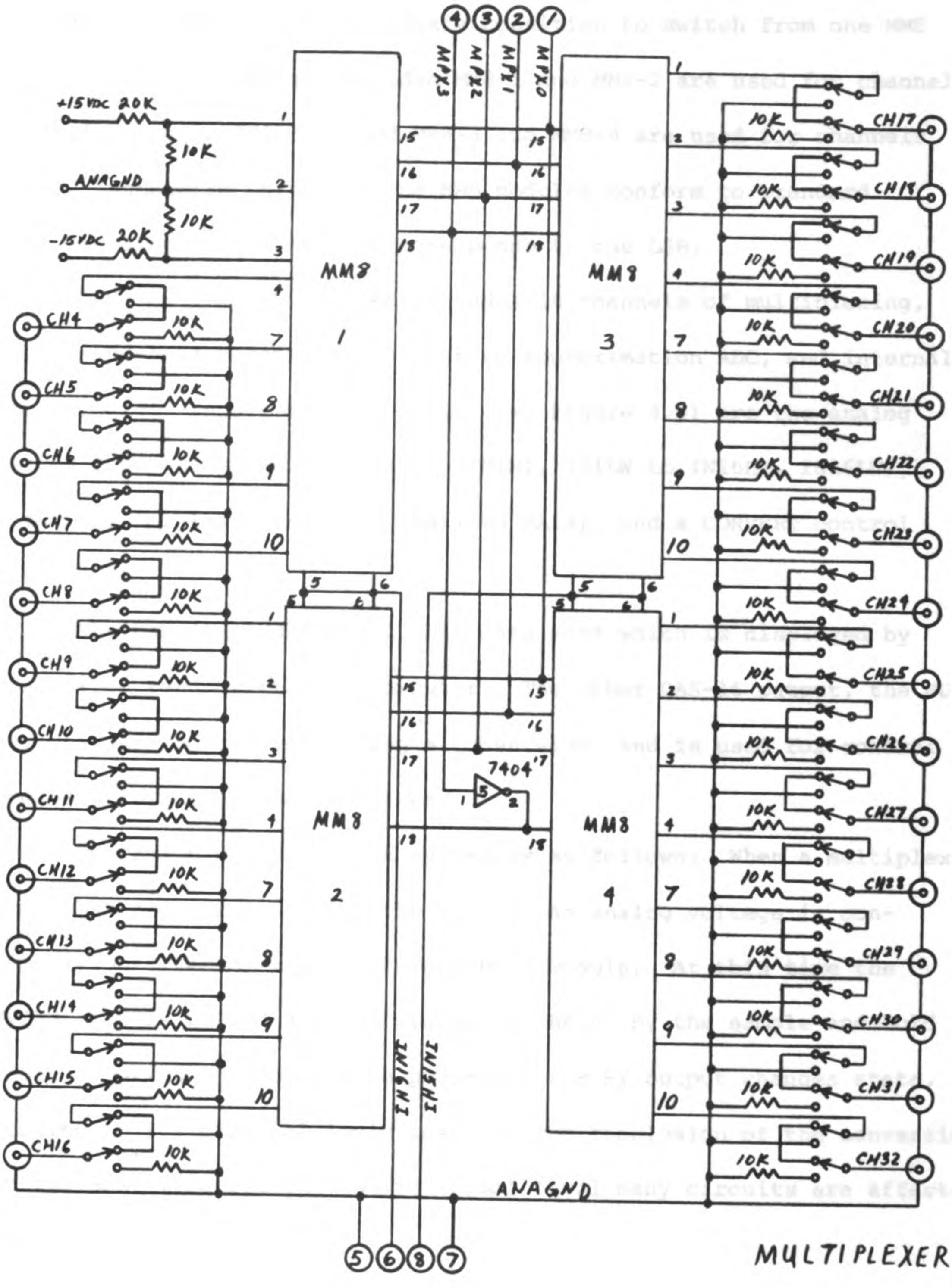

Figure 4.5 Data converter multiplexer circuit.

(8 channels of multiplexing), they are cascaded using a 7404 Inverter. This Inverter supplies the necessary condition to switch from one MM8 to the other. Following this idea MM8-1 and MM8-2 are used for channels 1 to 16 (output is IN16HI), and MM8-3 and MM8-4 are used for channels 17 to 32 (output is IN15HI). The MM8 modules conform to standard binary decoding, the MPX0 line coresponds to the LSB.

DAS-16- This ADC system includes 16 channels of multiplexing, a sample-and-hold circuit, a successive approximation ADC, and internal control logic. Inputs to the DAS-16 (see Figure 4.6) are the analog voltages from the sub-multiplexers (IN1HI, IN1LW to IN16HI, IN16LW), the address for these channels (RAI1 to RAI4), and a CONVERT control signal.

The DAS-16 produces a 12-bit data word which is displayed by LED's, or is read by the minicomputer. The other DAS-16 output, the BUSY line, changes logic level during a conversion, and is used for control purposes within the data converter.

Operation of the DAS-16 system is as follows: When a multiplexer address has been presented to the system, an analog voltage is continuously observed by the sample-and-hold module. At this time the CONVERT line is pulsed and the signal is "held" by the sample-and-hold circuit. During the converting process the BUSY output changes state, returning to its original logic level at the conclusion of the conversion. When the BUSY line returns to its normal level many circuits are affected, this is discussed next.

Control- This circuit is responsible for distributing the control signals throughout the data converter and to the minicomputer. The control signals cause the limits to be strobed into the lower and upper

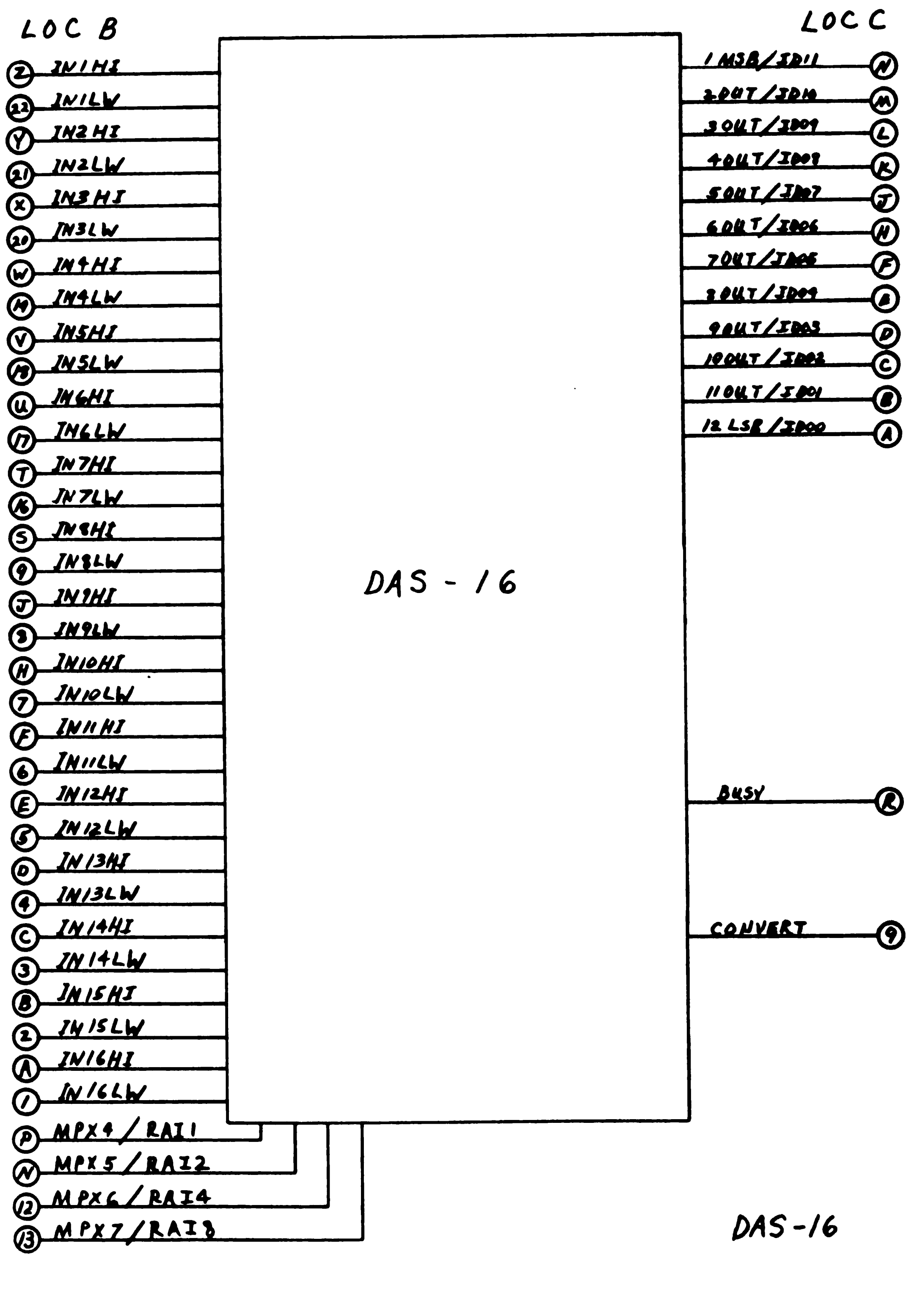

Figure 4.6 Data converter DAS-16 circuit.

buffer, provide a pulse used to increment the binary counter, pulse the DAS-16 to begin a conversion, and initiates an interrupt. The Control diagram is Figure 4.7.

A timing diagram (Figure 4.8) has been constructed to aid in understanding the control circuits. When the minicomputer executes an output instruction and loads the output buffer, a pulse is sent out on the STB line. Likewise, when the user presses the momentary LOAD toggle switch a pulse is formed on the LOAD line. In either case, one of these pulses (200 nsec) will trigger IC 3 (a 74121 Monostablè). The output from the IC is a pulse (CLK) used to strobe the lower and upper buffer, and preset the binary counter using the PRESET line.

A length of time will go by, at least one machine cycle, and the minicomputer will issue a SEL :12, 3 ($\overline{\text{EX3}}$) instruction, or the user will push the CNVRT toggle switch. At this time IC4 (a 74121 Monostable) will trigger. The result is a 5 microsecond pulse at CONVERT, on the trailing edge of this pulse the DAS-16 will begin converting. The trailing-edge triggering of the DAS-16 is intentional since it allows the sample-and-hold module 5 microseconds to settle. The conversion process requires around 10 microseconds as shown on the DAS-16 BUSY line in Figure 4.8.

When the DAS-16 BUSY line returns to ground IC 1 (a 74121 Monostable) fires and is trailing-edge triggered. This pulse is used to provide an external strobe ($\overline{\text{EST}}$) to the input buffer. When the EST pulse is received by the input buffer, it acknowledges by sending back A0. This A0 pulse is then looped back to the I/O Module to provide an interrupt stimulus at RNT1. Thus, the minicomputer interrupts and reads the input buffer which contains the ADC data word. Notice that

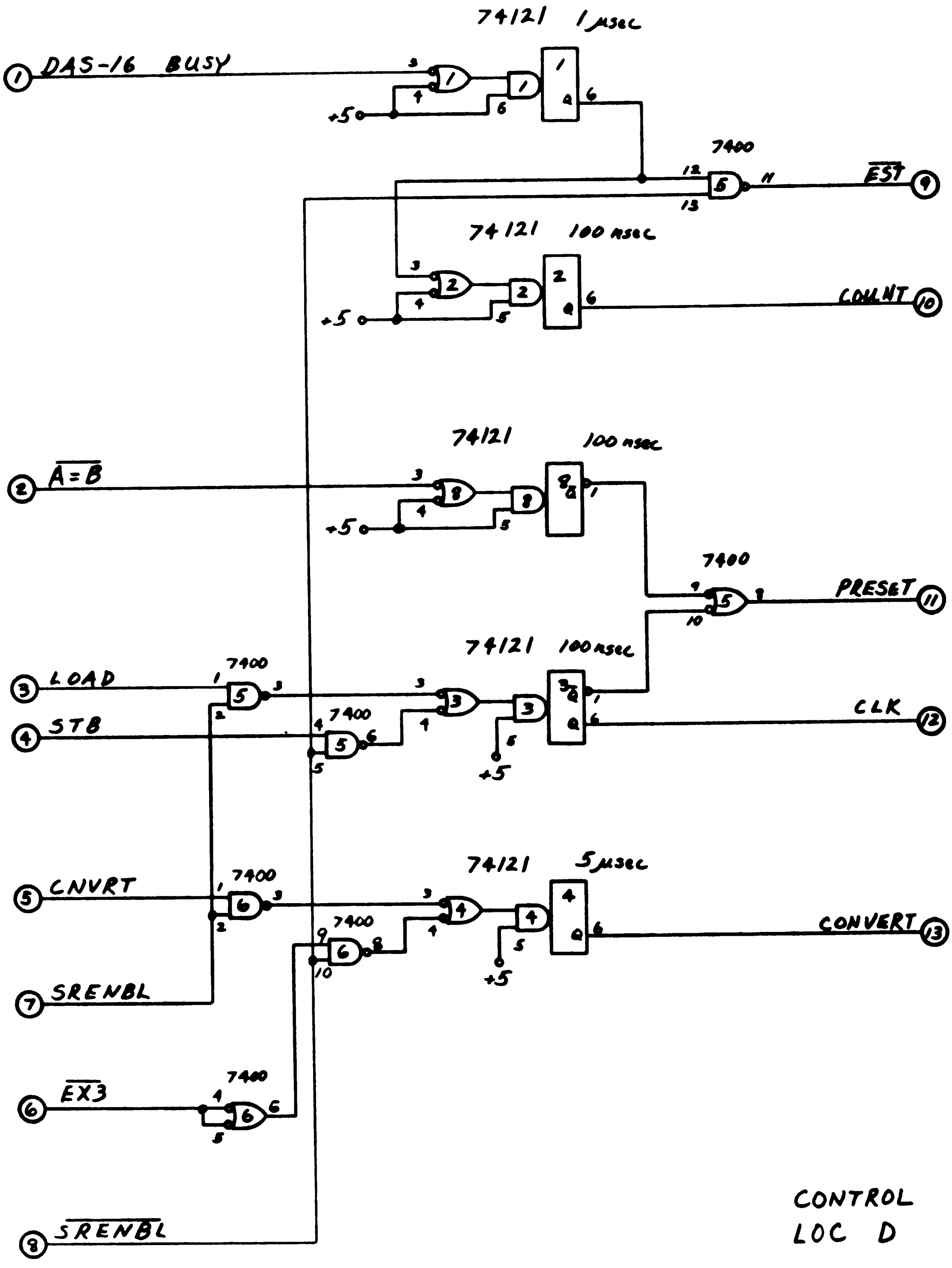

Figure 4.7 Data converter control circuit.

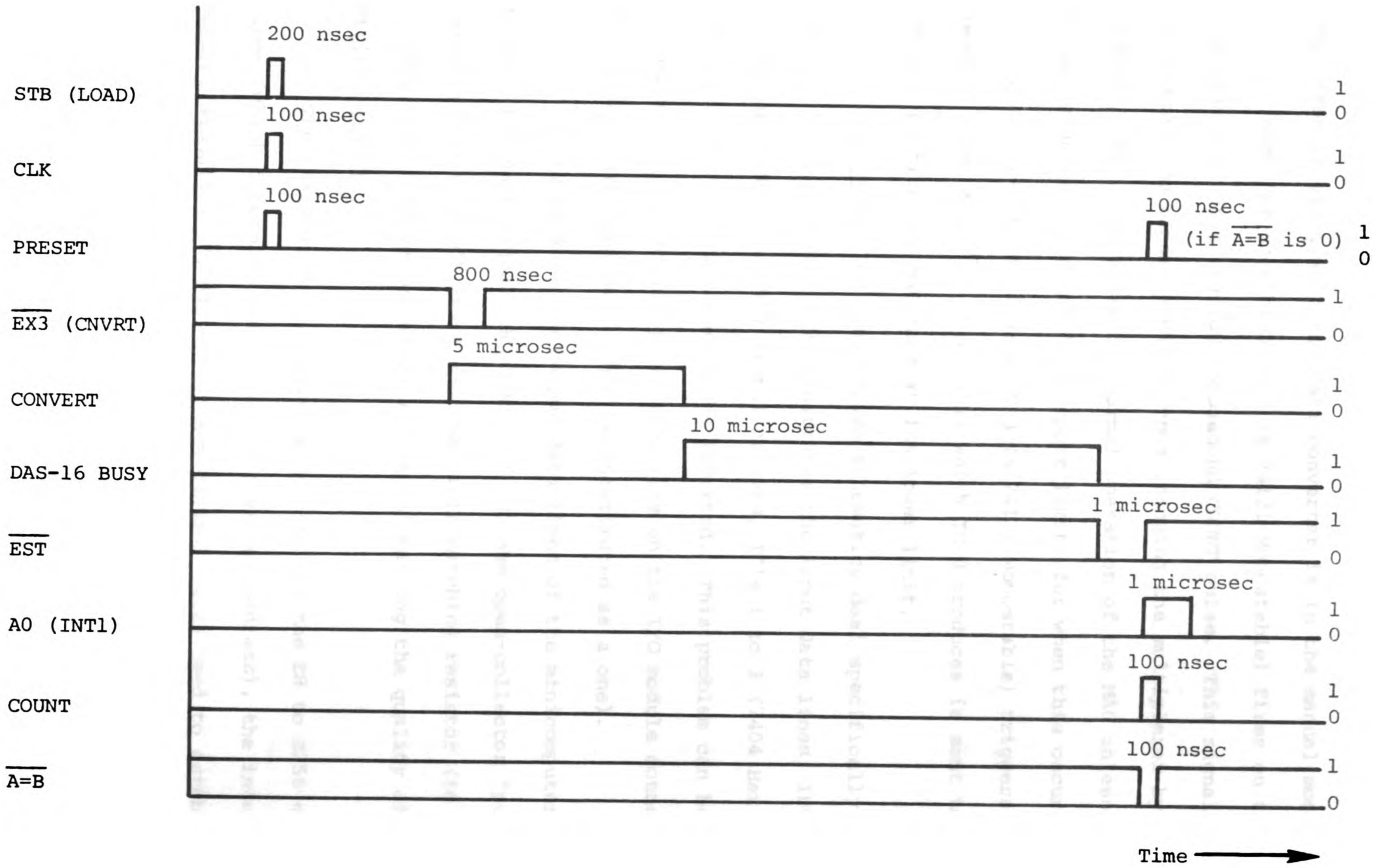

Figure 4.8 Control circuit timing diagram.

EST is only generated when $\overline{\text{SRENBL}}$ is high. This is to prevent an interrupt from occuring when the data converter is in the manual mode.

When EST goes high IC 2 (a 74121 Monostable) fires on the trailing edge producing the 100 nonosecond COUNT pulse. This signal then increments the binary counter, thus changing the multiplexer by one channel. This completes the normal operation of the MAG unless the binary counter has reached the upper limit, for when this occurs $\overline{A = B}$ goes low. When this happens IC 8 (a 74121 Monostable) triggers on the leading edge of $\overline{A = B}$. The pulse which IC 8 produces is sent to line PRESET, putting the MAG back at its lower limit.

I-O Cable- The final three schematics deal specifically with the I-O cable. Figure 4.9 illustrates the input data lines, ID00-ID15, to the minicomputer. The signal drivers, IC's 1 to 3 (7404 Hex Inverters), cause the logic level to be inverted. This problem can be corrected by simply grounding the IPOL line on the I/O module connector (a ground logic level will then be interpreted as a one).

In Figure 4.10 the output data lines of the minicomputer are shown. For each signal, BC00-BC15, a 220 ohm open-collector "pull-up" resistor (to +5 V DC) and a 330 ohm cable matching resistor (to ground) is used. Both these resistors aid in maintaining the quality of the digital signals.

The third drawing, Figure 4.11, depicts the E8 to E256 data lines (which determine the vectored interrupt address), the interface control signals ($\overline{\text{EST}}$, STB, $\overline{\text{EX3}}$), and several lines used to establish the digital ground return for the minicomputer and the data converter. If the desired vectored interrupt address is :92 (hexadecimal), as was the case, then switches E16 and E128 are left open while the remaining

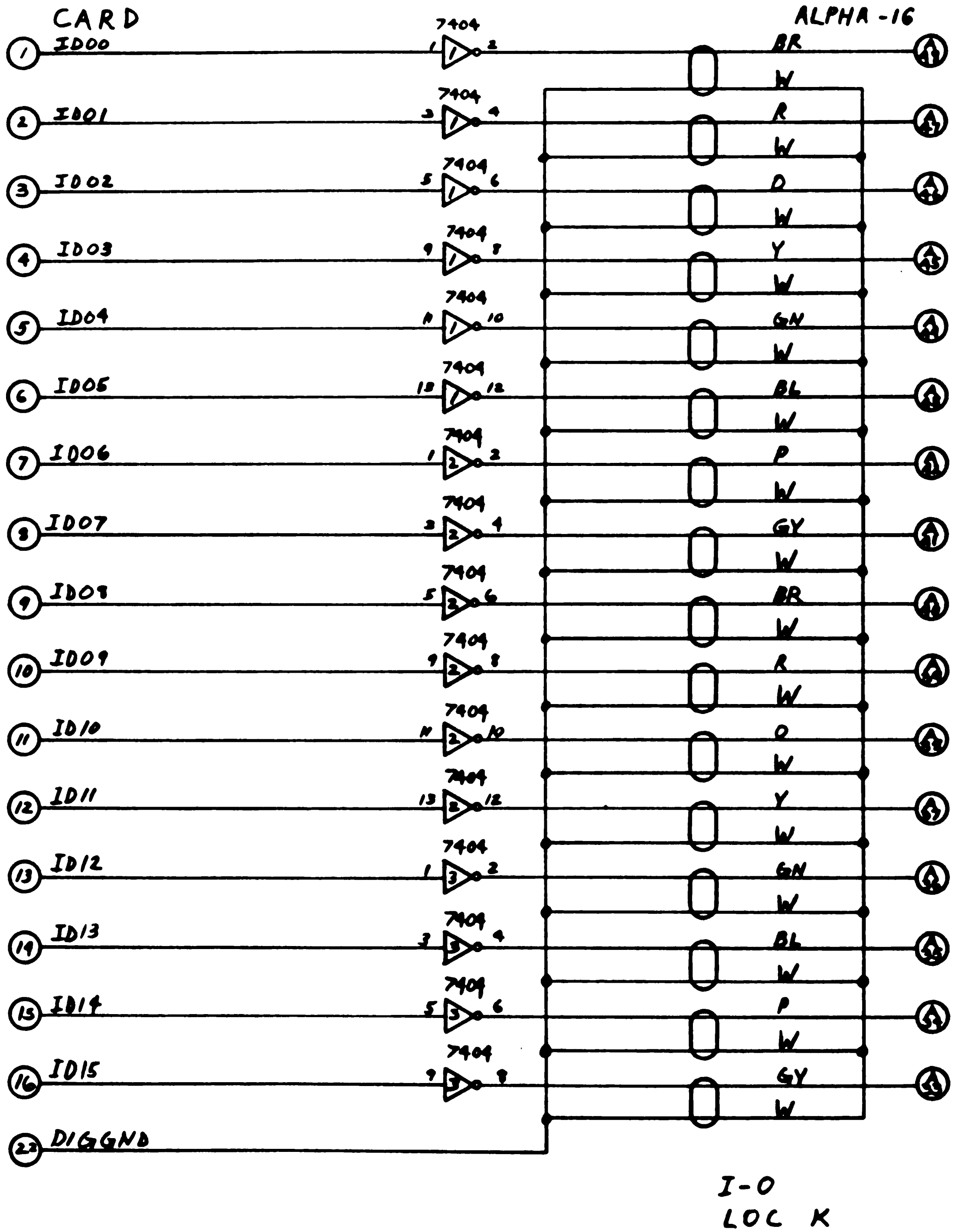

Figure 4.9 Data converter I/O(1) circuit.

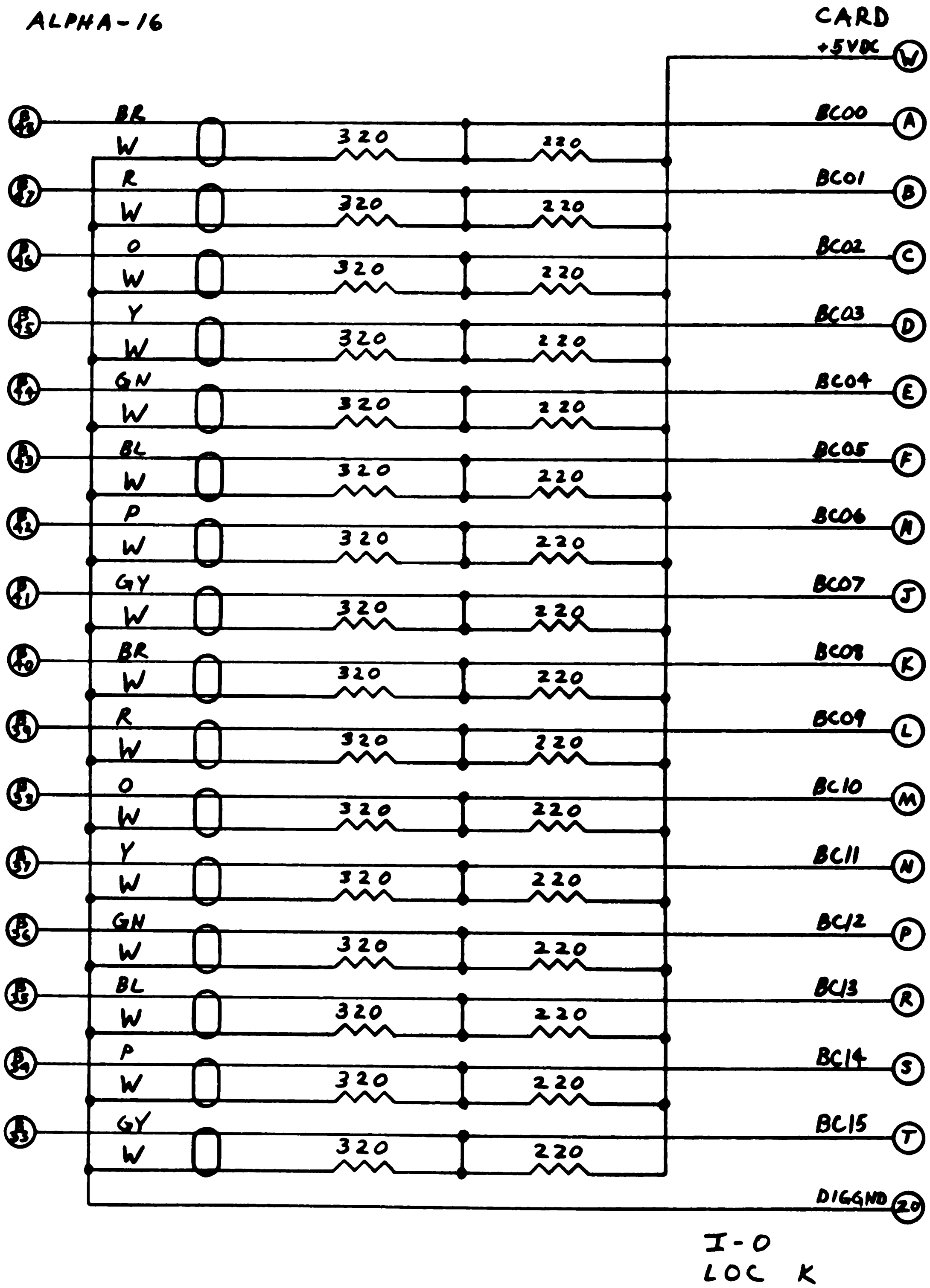

Figure 4.10 Data converter I/O(2) circuit.

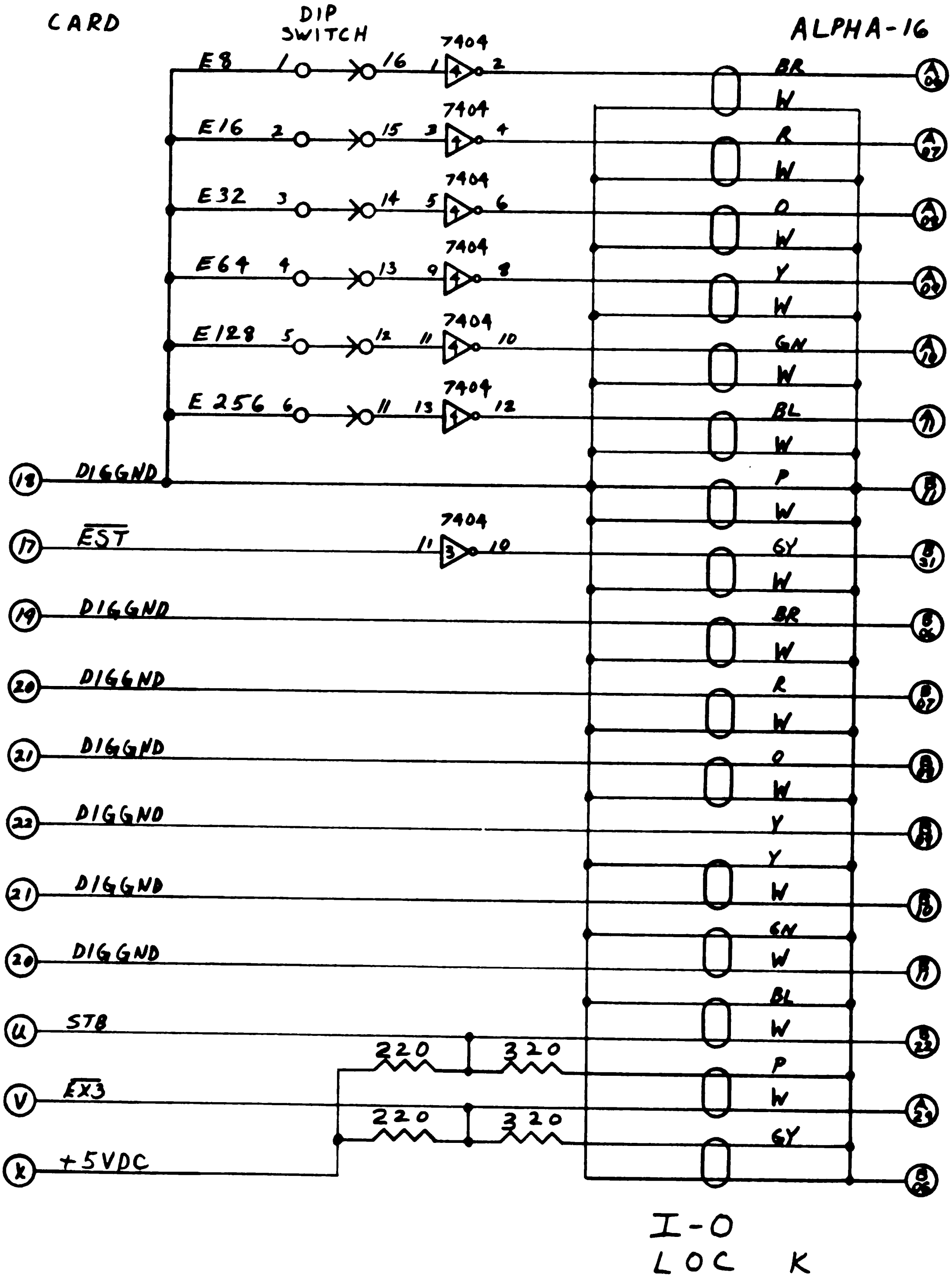

Figure 4.11 Data converter I.O(3) circuit.

switches are closed. Upon recognition of an interrupt, from the data converter, the minicomputer will then vector to address :92 (hexadecimal).

The circuit is now completely described in terms of a theory, specific components, and electrical schematics. What remains to obtain realization of the circuit, is to construct the prototype. This step is outlined in the next Section.

4.4 Construction of the Prototype

The data converter is housed in a standard 19" rack-type cabinet, four panels are mounted. Each panel can be removed from the rack by detaching a connector. One panel mounts the 32 BNC chassis connectors, another the power supplies, a third holds the switch register (for manual operation), and the fourth panel mounts a card cage assembly. It is noted thatin Appendix C, several photographs of the data converter are available.

The BNC panel also includes 32 DPDT on-off toggle switches for the analog signals. Running from each BNC is a 28 AWG coaxial cable. These cables are tied together to form a larger cable. This larger cable ends at a card inserted into the card cage assembly. Located on this card, besides the 32 coaxial cables, are four MM8 multiplexer modules.

The power supply panel has a +15 V DC and -15 V DC (at 300 milliamperes) analog supply and a 5 V DC, 3 ampere, logic supply. A toggle switch controls the AC power, and a fuse is mounted to provide overload protection. The voltage outputs are located on the back side of the panel at a terminal strip, but may also be monitored on the front side using bannana jacks.

The switch register has 19 toggle switches mounted on it. They are situated on the panel in three groups: control, lower buffer, and upper buffer. Also included on this panel are 8 LEDs, which display the current multiplexer address (the MPX lines). The LED drivers are mounted on a small board located on the back of the panel. Signals from the switch register panel are sent to other circuits on a cable which stops at a card inserted into the card cage assembly.

The fourth panel contains the card cage assembly which holds nine cards. Each card slides into its own slot and plugs into its own connector. Connections between cards are then accomplished by wiring from connector to connector.

Whenever possible wirewrapping was employed. The IC packages were inserted into wirewrap sockets, components were soldered to wirewrap stakes, and modules were plugged into wirewrap socket terminals. Characteristics typical of wirewrapping are flexibility and density. Both these characteristics are desirable when prototyping.

The I/O cable was constructed with a twisted-pair 26 AWG ribbon cable. Six lengths of this wire were necessary to send the required minicomputer signals to and from the data converter. The ribbon cables were tied together and ended at a card inserted into the cage assembly. The other end of the I/O cable was wired to a plug which mates with the 16-bit I/O Module. Located on the card are the I/O signal receivers and drivers, along with a special IC which contains miniature switches. These switches are used to determine the vectored interrupt address. This address can be easily modified by these switches.

Because the sockets into which the cage assembly cards are inserted has wirewrap terminals, the backplane connections are wirewrapped.

This differs from the card wirewrapping only in wire size, 24 AWG compared to 30 AWG.

The construction of a prototype involves numerous steps, many of them too tedious to be discussed here. It is pointed out, though, that these decisions must be made and can affect many aspects of the resulting product.

4.5 Summary

Circuit realization can be achieved only after the design specifications are known. The process begins with a theory of operation, components are then identified, next circuit schematics are drawn, and finally the prototype is built. Before the prototype is installed in the field, it must be tested to see if the design specifications have, indeed, been met. Experimental procedures, results, and a discussion of such tests are the subject of the next Chapter.

CHAPTER V PERFORMANCE

It was necessary to perform several tests on the data converter to evaluate its actual capabilities. These tests can be grouped as follows:

1) verification of control logic,

2) calibration of the ADC,

3) determination of the static accuracy, and

4) an example of the dynamic response.

The first test demonstrates that the data converter's control logic operates as designed. Calibration of the ADC, the second test, adjusts the ADC transfer characteristics for optimum accuracy. The final two tests, determination of the static accuracy and an example of the dynamic response, illustrates the DC and AC operation of the data converter. These four experiments will be outlined and discussed in the Chapter.

5.1 Verification of Control Logic

As stated in Section 4.1, there exist two fundamental control logic modes; namely, manual and automatic. Each of these modes has three schemes of operation: sequential, static, and random. Consequentially, six distinct sets of experiments are required to fully verify that the data converter's control logic operates as designed.

Verification of the manual mode follows the algorithm outlined in Figure 5.1. Limits used in the sequential operation test were 00 (hexadecimal) for the lower channel and 20 (hexadecimal) for the upper

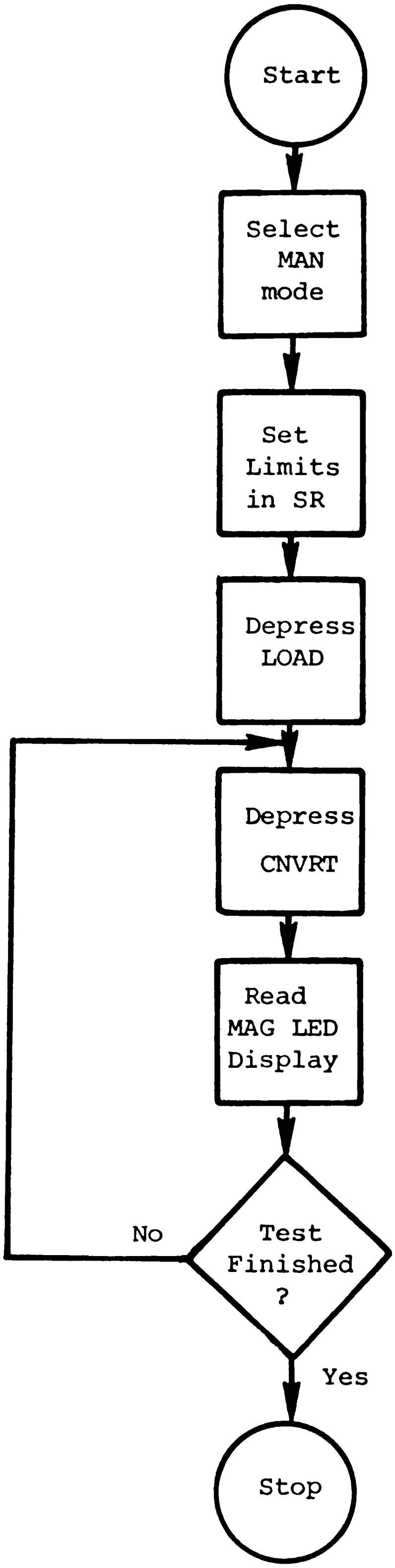

Figure 5.1 Data Converter Manual Operation Algorithm.

channel. Using these limits, and following the algorithm, the data converter sequentially stepped from channel-to-channel, and preset, as required, when the upper limit was reached.

Static operation using the manual mode, was achieved by following the algorithm and loading such sets of limits as: 0-1, 5-6, A-B, and 15-16 (all hexadecimal). In each case multiple readings of the lower channel were observed. Random operation using the manual mode, was simultaneously verified in this test since the lower channel was changed (preset) each time that new limits were loaded.

A small program, see Figure 5.2, was written and toggled into memory to verify automatic operation. The program begins at address 100 (hexadecimal) by loading into the data converter the limits found in the X register. After the ADC interrupt is reset and armed, the convert instruction is given. Following the convert instruction is an endless loop. The minicomputer will interrupt while in this endless loop and vector to address 92 (hexadecimal). Here the input buffer, containing the data word, is read into the A register and the program stops. At this time the limits in the X register may be altered and the program started again. If the limits are not to be changed the operator depresses the run switch (on the minicomputer) to continue the program. The program is continued and another conversion is made.

To verify each scheme of operation (sequential, static, and random) in the automatic mode, identical limits as used in the manual mode, were stored in the X register. In each case identical results were obtained; therefore, all six possible schemes of operating the data converter were verified.

Address	Code	Mnemonic	Comment
92	F201	JMP $+1	Jump to next word
93	5896	INA :12,6	Read input buffer
94	0800	HLT	Stop program
95	F20C	JMP $+:C	Jump to :101
100	6E92	OTX :12,2	Load Limits
101	4090	SEL :12,0	Reset ADC interrupt
102	4091	SEL :12,1	Arm ADC interrupt
103	0A00	EIN	Enable all interrupts
104	4093	SEL :12,3	Convert
105	F600	JMP $+0	Loop to self

Starting address- 100

Figure 5.2 A program to verify the data converter automatic operation.

5.2 Calibration of the ADC

This process began after all equipment had been powered for one hour. The circuit used for calibration is illustrated in Figure 5.3. A battery was used as a reference voltage to eliminate any "ripple" that might be associated with a power supply. The voltmeter used to calibrate the data converter, a Wavetek 201,[11] has an accuracy of (±) .11 millivolts on the 10 volt range, and .01 millivolts on the 1 volt range. Resolution of the voltmeter is to 5 decimal digits. The voltmeter, then, has sufficient accuracy and resolution to be used for calibration purposes.

Calibration requires the adjustment of two miniature 10-turn potentiometers located on the DAS-16. The two adjustments determine the ADC off-set and gain.

The off-set is adjusted first by selecting a channel which presents exactly -5 V DC to the ADC (this voltage is monitored by the voltmeter). The data converter is operated manually, and the off-set is adjusted until the converted code is 000 (hexadecimal). This determines the zero code of the ADC.

When the ADC off-set has been set, the gain adjustment is made. The data converter is operated manually, selecting a channel which presents exactly 4.9975 V DC (as monitored by the voltmeter). This voltage is determined by subtracting 1 LSB from the full-scale voltage, or 2.5 millivolts from 5 volts. The gain adjustment is varied until a code of FFF (hexadecimal) is displayed in the LED's of the DAS-16. This adjustment determines the full-scale code of the ADC.

When the off-set and gain adjustments have been made, the data

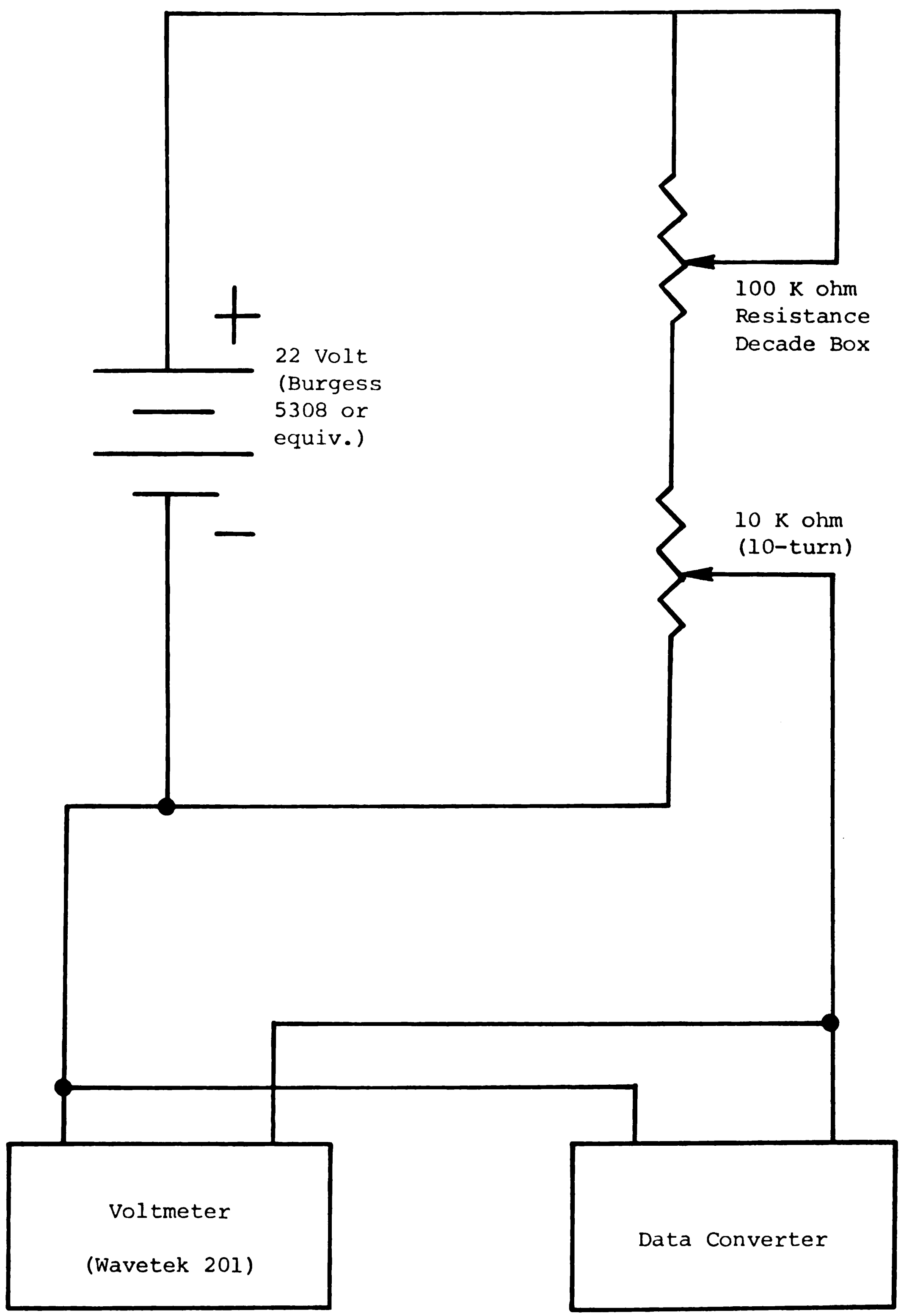

Figure 5.3 Calibration circuit for the data converter.

converter is then fully calibrated. A convenient third voltage (the half-scale value), analog ground, can now be checked and should yield a code of 800 (hexadecimal). This type of coding is "off-set binary" since the analog ground code is 800 (hexadecimal). For a detailed explanation of the calibration procedure Datel has prepared "Applications Handbook For Adjustment and Timing of Analog-To-Digital And Digital-To-Analog Converter Products".[12]

5.3 Determination of the Static Accuracy

To determine the accuracy of the data converter a circuit identical to Figure 5.3 was employed. Data points were taken every 0.5 volts and the results are tabulated in Table 5.1. The final column in the table, total error, includes both the error of the data converter and the error of the voltmeter used in the test. The largest error was found to be 9.21 millivolts or 0.0921% of full-scale. Since the required accuracy of the data converter is 0.1% of full-scale, an experimentally determined accuracy of 0.0921% of full-scale is, indeed, satisfactory.

The input voltage was varied in both the positive and negative direction, and no hysteresis was found. To determine if all channels were identical, in terms of accuracy, a 1.2 volt mercury battery was connected (one at a time) to each analog voltage channel. For all 32 channels the DAS-16 output code was identical, thus, the calibration and accuracy tests performed on only one channel are valid for all channels. It can be deduced, therefore, that all 32 analog voltage channels maintain a .0921% of full-scale accuracy.

Table 5.1 Experimental results to determine the static accuracy.

Input voltage	Octal code	Output voltage	Error (mV)	Voltmeter error (±mV)	Total error (±mV)
-5.00	0000	-5.0000	0.0	0.61	0.61
-4.50	0312	-4.5068	6.8	0.56	7.36
-4.00	0626	-4.0087	8.7	0.51	9.21
-3.50	1144	-3.5053	5.3	0.46	5.76
-3.00	1460	-3.0078	7.8	0.41	8.21
-2.50	1776	-2.5048	4.8	0.36	5.16
-2.00	2317	-1.9946	-5.4	0.30	5.70
-1.50	2627	-1.5063	6.3	0.26	6.56
-1.00	3144	-1.0058	5.8	0.11	5.91
-0.50	3461	-0.5053	5.3	0.06	5.36
0.00	400	0.0000	0.0	0.00	0.00
0.50	4315	0.5004	0.4	0.06	0.46
1.00	4631	0.9985	-1.5	0.10	1.60
1.50	5147	1.5014	1.4	0.26	1.66
2.00	5463	1.9995	-0.5	.30	0.80
2.50	6001	2.5024	2.4	0.36	2.76
3.00	6316	3.0029	2.9	0.41	3.31
3.50	6634	3.5058	5.8	0.46	6.26
4.00	7151	4.0063	6.3	0.51	6.81
4.50	7465	4.5043	4.3	0.56	4.86
5.00	7777	4.9975	-2.5	0.60	3.10

5.4 An example of the dynamic response

The purpose of this experiment was to evaluate the basic dynamic characteristics of the data converter. Figure 5.4 depicts the circuit used in this test. The oscillator was adjusted to produce a sinusoidal signal with an amplitude of 10 V peak-to-peak. A program was then executed by the Alpha-16 (see Appendix D) which enabled the sinusoidal waveform to be sampled several thousand times. The channels to be sampled were specified by an operator via the teletype keyboard. This program was designed to record both the maximum and minimum reading, find the mean average all readings, and print out these results for each channel.

The dynamic response test was performed using the static (without multiplexing) and sequential (with 32 channels of multiplexing) schemes of operating the data converter. Results of these two tests are tabulated in Tables 5.2 and 5.3. It is noted that as the frequency of the oscillator was increased the min and max values began to decrease. When this occurred the sampling rate and the number of periods observed were not sufficient to detect the desired min (-5.000) and max (4.997) values. If the sampling rate, or the number of periods observed were increased, then both min and max values would agree with the lower frequency cases. This of course, is only true as long as the aperature time (5 microseconds) of the sample-and-hold module is much less than the period of the sinusoidal signal.

5.5 Summary

The results of the experiments described in this Chapter clearly demonstrate that the data converter's control logic is functioning as designed. Also, the experimental results verify that the accuracy of

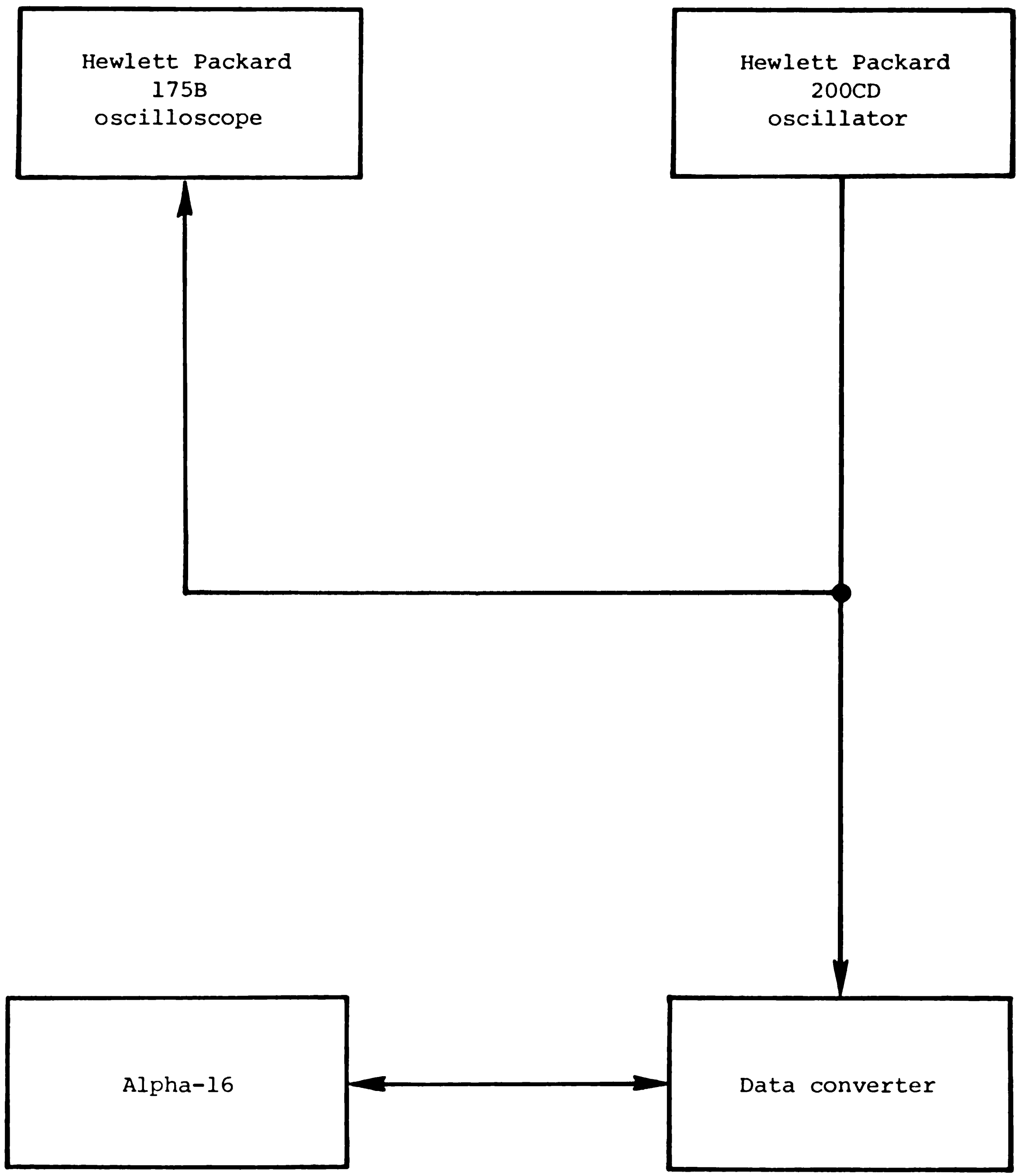

Figure 5.4 Circuit used in the dynamic response test.

Table 5.2 Results of the dynamic response test-without multiplexing.

frequency(Hz)	minimum(V)	maximum(V)	average(V)
0	-5.000	4.997	0.002
1K	-5.000	4.997	0.002
10K	-5.000	4.997	0.002
40K	-5.000	4.997	0.002
50K	-5.000	4.997	0.002
60K	-4.978	4.997	0.002
70K	-4.924	4.997	0.007
80K	-4.909	4.995	0.007
90K	-4.885	4.992	0.009
100K	-4.836	4.963	0.009

Table 5.3 Results of the dynamic response test-with multiplexing.

frequency(Hz)	minimum(V)	maximum(V)	average(V)
0	-5.000	4.997	0.002
1K	-5.000	4.997	0.002
10K	-5.000	4.997	0.002
30K	-5.000	4.997	0.002
40K	-4.975	4.997	0.004
50K	-4.963	4.997	0.004
60K	-4.909	4.975	0.007
70K	-4.904	4.956	0.007
80K	-4.839	4.941	0.009
90K	-4.819	4.904	0.009

the data converter exceeds the design requirements. A software package for the data converter is currently under development; it will enable an operator to conpletely evaluate the static and dynamic capabilities of the data converter. Some of the important features of this diagnostic package are described in the next Chapter.

CHAPTER VI SUMMARY

An adaptive programmable measurement system has been proposed for the purpose of monitoring and managing pest-crop ecosystems. One component in this facility is the data converter. The purpose of the research reported here was to specify, design, construct, and evaluate the data converter.

After carefully considering the overall programmable measurement system requirements, the prototype data converter was designed, constructed, and its performance tested. These performance tests demonstrated that the data converter's control logic functioned as designed and that its accuracy was better than 0.1% of full-scale. An example demonstrating the basic dynamic properties of the data converter was included in the experimental tests.

To further evaluate the dynamic response and to facilitate the maintainance of the data converter, a software diagnostic package is being developed. This package will be operator interactive--the desired tests may be individually selected and executed. Static performance will be evaluated in much the same manner as previously described in Chapter V. The dynamic performance, however, will involve finding the time constants and delays associated with converting a signal with a large time rate of change. This may be accomplished by synchronizing (interrupting) the minicomputer with a -5 V to +5 V ramp, and sampling it, on the order of, 50 times a period. By decreasing the period of this ramp signal these time constants and delays will become measurable. The dynamic performance tests described above will be included, along with others, in the software diagnostic package.

In retrospect several design changes might be made to suit a particular application. To reduce the cost of parts (see Appendix E) the DAS-16 ADC system could be eliminated, and separate ADC and sample-and-hold modules could be substituted. The full capabilities the DAS-16 system were not utilized in this design, nor could they be used in any application requiring greater than 16 channels of multiplexing. For this reason a saving of $300, or 15% of the total cost of parts, would be assured by implementing this substitution.

Additional cost-savings may be realized by deleting the manual mode. The Gating scheme and the complete Switch Register circuit would not be necessary, and about $100 could be shaved off the parts list.

A final recommendation to improve subsequent data converters involves a third mode of operation. This mode eliminates the need to execute a convert instruction for each conversion. Once the limits are established in the MAG, a convert instruction is issued. The data converter then provides its own CONVERT pulse (and all future pulses) for each conversion. When the upper limit is reached the MAG presets and a second interrupt is issued which signifies an end-of-block. By using an AUTO I/O instruction and two vectored interrupts, one for each data word and one for an end-of-block (or ECHO), this new mode could easily and efficiently be programmed. Since the ADC would essentially be "free-running" from lower to upper channel (once started), converted data words would be available and read into the minicomputer every 15 microseconds. Using this new mode of operation the data converter's sampling rate would be increased by nearly a factor of 10.

R E F E R E N C E S

REFERENCES

1) Dean L. Haynes, Richard K. Brandenburg, and P. David Fisher, "Environmental Monitoring Network for Pest Management Systems", Environmental Entomology, Vol. 2, no. 5 (October 1973), 889.

2) Ibid., 895.

3) Analog-Digital Conversion Handbook, prepared by Analog Devices, Inc. (1972), p. II-164.

4) Alpha 16 & Naked Mini 16 Computer Reference Manual, prepared by Computer Automation, Inc. (1971), pp. 1-1 to 1-28.

5) Ernest O. Doebelin, Measurement Systems: Application and Design (New York, 1966), pp. 38-203.

6) 16-Bit Input/Output Module, prepared by Computer Automation, Inc. (1972), pp. 1-1 to 4-16.

7) Signetics, prepared by Signetics Corp. (1972), p. 2-1.

8) A/D-D/A Converters And Accessories, prepared by Datel Systems, Inc. (1973) pp. 1-33.

9) Signetics, prepared by Signetics Corp. (1972), pp. 2-1 to 2-173.

10) RCA Solid State Databook Series (SSD-201A), prepared by RCA Corp. (1972), p. 170.

11) Voltmeter Model 201 Instruction Manual, prepared by Wavetek, Inc. (1970), pp. 30-32.

12) Applications Handbook For Timing And Adjustment Of Analog-To-Digital And Digital-To-Analog Converter Products, prepared by Datel Systems, Inc. (1973), pp. 1-29.

A P P E N D I C E S

APPENDIX A

GLOSSARY

APPENDIX A

Glossary

Note- Definitions are from "IEEE Standard Dictionary of Electrical and Electronics Terms", Std 100-1972.

asynchronous computer- A computer in which each event or the performance of each operation starts as a result of a signal generated by the completion of the previous event or operation, or by the availability of the parts of the computer required for the next event or operation.

bit- A binary digit.

buffer- A storage device used to compensate for a difference in the rate of flow of information or time of occurance of events when transmitting information from one device to another.

bus- One or more conductors used for transmitting signals or power from one or more sources to one or more destinations.

central processing unit- The unit of a computing system that includes the circuits controlling the interpretation and execution of instructions.

conditional jump- To cause, or an instruction which causes, the proper one of two (or more) addresses to be used in obtaining the next instruction, depending upon some property of one or more numerical expressions or other conditions.

decode- To produce a single output signal from each combination of a group of input signals.

diagnostic routine- A routine designed to locate either a

malfunction in the computer or a peripheral device.

half duplex bus- A bus arranged to permit signal flow in either direction but not in both directions simultaneously.

hardware- Physical entities such as computers, circuits, tape readers, et cetera.

interrupt- To stop a process in such a way that it can be resumed.

interface- A shared boundary.

mask- A pattern of characters that is used to control the retention or elimination of portions of another pattern of characters.

peripheral device- A mechanical or an electric contrivance to serve a useful purpose and is outside the computer.

real time- Pertaining to the actual time during which a physical process transpires.

register- A device capable of retaining information, often that contained in a small subset (for example, one word), of the aggregate information in a digital computer.

routine- A set of instructions arranged in proper sequence to cause a computer to perform a desired operation.

simplex bus- A bus arranged to permit signal flow in one direction only.

software- Computer programs, routines, programming languages and systems.

system- An integrated whole even though composed of diverse, interacting, specialized structures and subjunctions.

transfer- To transmit, or copy, information from one device to another.

APPENDIX B

16-BIT I/O MODULE SIGNAL DEFINITIONS

APPENDIX B

16-bit I/O module signal definitions

Note- Definitions are from "16-Bit Input/Output Module", prepared by Computer Automation, Inc. (1972), pp. A-4 to A-8.

A0- Input buffer regenerated clock pulse.

BC00-BC15- Output data lines to peripheral device. These lines can be either positive or negative polarity. Polarity is defined by use of the OPOL signal.

E8-E256- Interrupt address select lines. These lines are user defined and ground-true.

ES0-ES7- External sense lines. These lines are positive-true.

EST- External strobe. Generated by the peripheral to strobe input data into the input buffer register.

EX0-EX7- External control lines. These lines produce an 800 nanosecond ground-true pulse when activated to control user defined functions in the peripheral device.

IBF- Input buffer full. Conveys input register status to the computer and the peripheral device. A low output indicates that the computer has accepted the last input data transfer. A high ouput indicates that input data is currently stored in the input register but the computer has not accepted it.

ID00-ID15- Input data lines from a peripheral device. These lines can be either positive or negative polarity. Polarity is defined by use of the IPOL signal.

IPOL- Input register polarity control. If the data lines are ground-true logic, IPOL is strapped to ground. If positive logic is used, IPOL is left open.

ITRAN- Input transparent. Causes the input register to be transparent, permitting peripheral input data to be applied directly to the data bus.

OBE- Output buffer empty. Used to convey ouput buffer register status to the computer and peripheral device. A low ouput indicates that new output data is in the ouput register and that a transfer is imminent. A high ouput indicates that the output data from the last transfer has been received by the peripheral device.

OBRS- Output buffer ready strobe. Generated by the peripheral device to acknowledge receipt of output data. Causes the output empty signal OBE to go high.

OPOL- Output register polarity control. Grounding this line causes the BC00-BC15 outputs to be in inverted logic form (ground-true). Leaving the line open causes the output data to be noninverted (positive-true).

PS0-PS4- Address select lines. These five lines determine the 16-Bit I/O Module user defined device address. These lines are ground-true signals.

RNT 1- Interrupt request number 1

RNT 2- Interrupt request number 2.

RPOL- Interrupt request polarity control. A ground-true RPOL signal causes a ground-true external interrupt to be recognized. Likewise, a positive-true polarity signal causes a positive-true external interrupt to be recognized.

SPOL- Peripheral strobe polarity control. A ground-true polarity signal causes the EST and OBRS signals to be recognized if they are ground-true. Likewise, a high polarity signal causes the EST and OBRS signals to be recognized if they are positive-true.

STB- Output data strobe. Goes high for approximately 200 nanoseconds.

APPENDIX C

Data converter photographs

Figure C.1 Data converter, teletype, and Alpha-16 minicomputer.

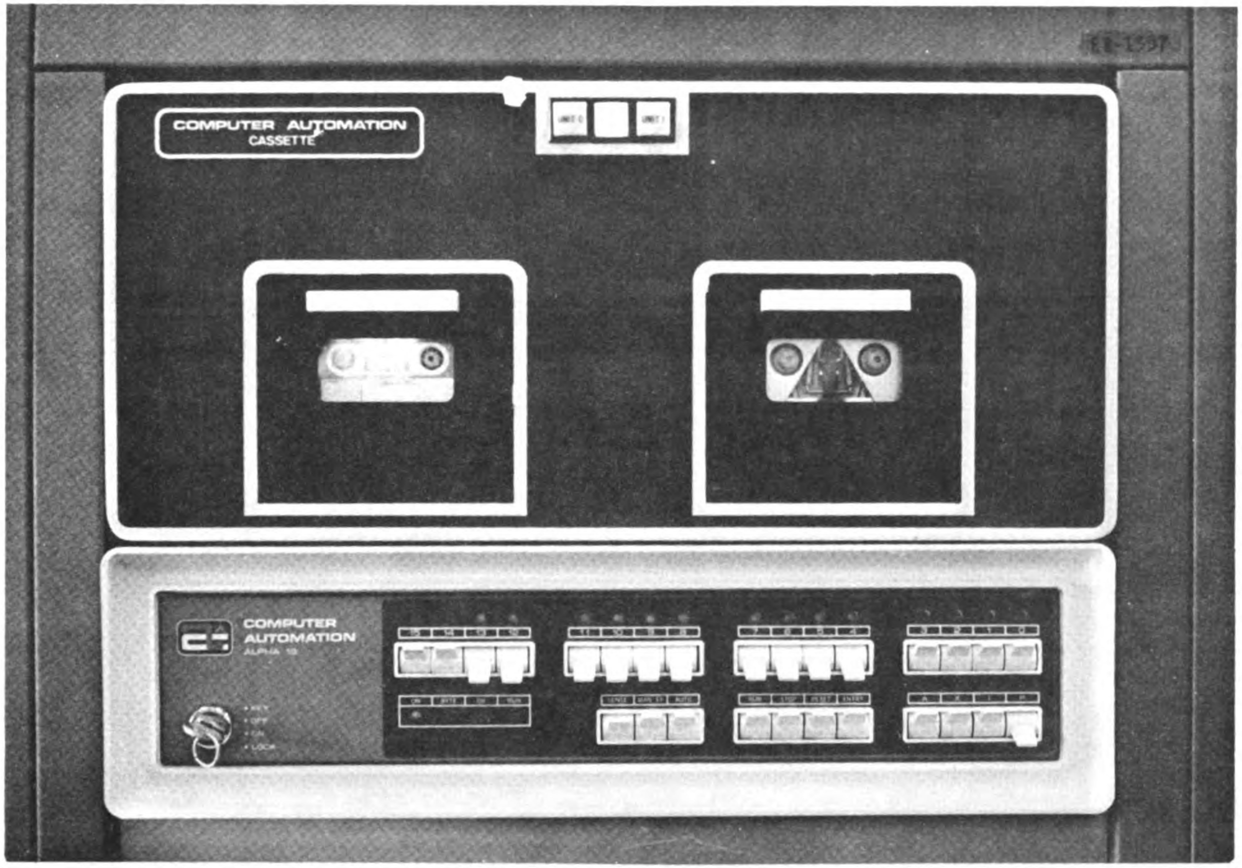

Figure C.2 Computer Automation's Alpha-16 and dual cassette tape transport.

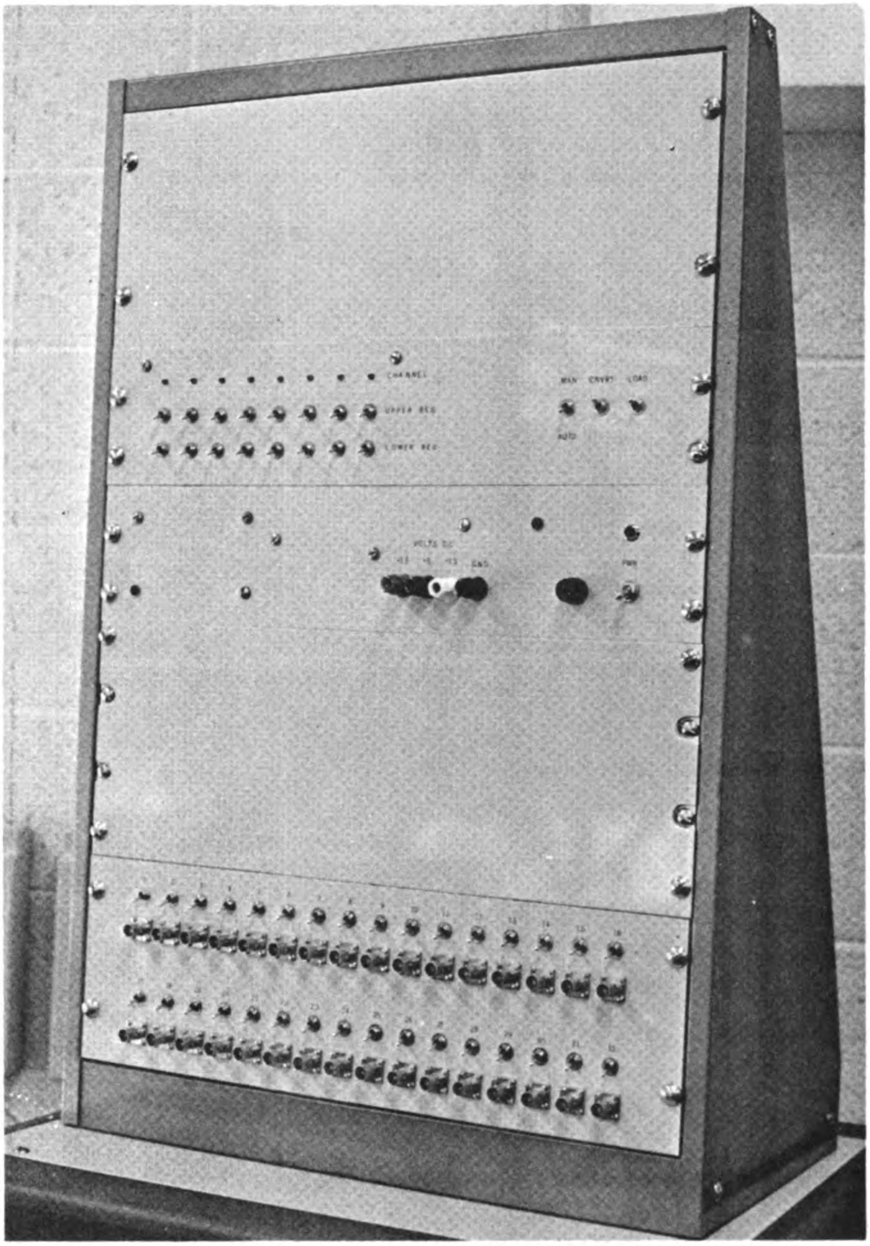

Figure C.3 Data converter front view.

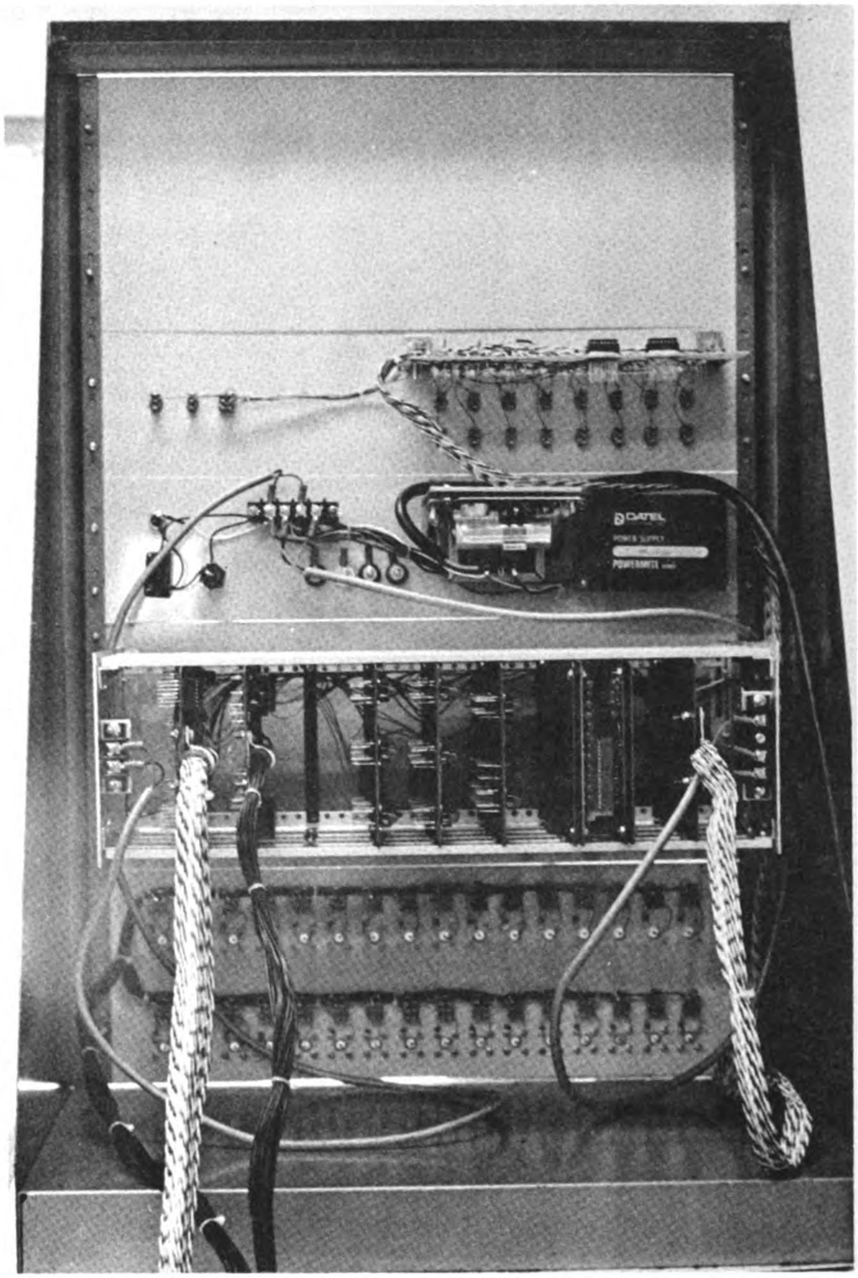

Figure C.4 Data converter rear view.

APPENDIX D

DATA CONVERTER
DYNAMIC RESPONSE PROGRAM

XA.
PAGE 0001

```
                    *                    APPENDIX D
                    *
                    *
                    *  DATA CONVERTER DYNAMIC RESPONSE PROGRAM
                    *
                    *
                    ***************************************
                    *                                     *
                    *       DATA CONVERTER DIAGNOSTIC     *
                    *        WRITTEN BY GARY D BAUER      *
                    *               4/18/1974             *
                    *                                     *
                    ***************************************
           0012     BIT    EQU  :12
           0008     RTC    EQU  8
           0007     TTY    EQU  7
                           EXTR CRID:
                           EXTR MPYS:
                    *
0000                       ABS  0
0000                LINE   RES  1         CURRENT CHANNEL NUMBER
0001                COUNT  RES  1
0002                TTYWRD RES  1         TTY WORD INTERRUPT
0003       0000     TTYWCT DATA 0         TTY WORD COUNT
0004                TTYBAD RES  1         TTY BYTE ADDRESS
0005       8006     NEWAD  DATA *LIST-1   MESSAGE ROUTINE ADDRESS
0006                AP     RES  1         ADDRESS POINTER TO PROCESSING
0007       01DF     LIST   DATA MAXMES    MESSAGE ROUTINES
0008       01E3            DATA AVGMES
0009       0162            DATA DONE
000A       03BF     BUFAD  DATA BUF-1
000B       06C0     MAXMAX DATA MAX
000C       02C0     MINMIN DATA MIN
000D       09C0     TOTAD  DATA TOT
000E       07C0     AVGAAD DATA AVGA
000F       08C0     AVGXAD DATA AVGX
0010       FD00     M768   DATA -768
0011       B007     ZERO   DATA :B007
0012                COUNT3 RES  1
0013                BAD    RES  1
0014                COUNT2 RES  1
0015       000A     TEN    DATA 10
0016                LNUM   RES  1
0017       0FFF     FFF    DATA :FFF
                    *
0018       D819            IMS  RTCONT    INCREMENT
0019                RTCONT RES  1         TIME KEEPER
001A       F91B            JST  *$+1      SYNC INTERRUPT
001B       01ED            DATA TIMERR
                    *
0092                       ABS  :92
0092       F893            JST  AIN       WORD INTERRUPT LOCATION
0093       0800     AIN    ENT
0094       5096            AIN  BIT,6     TO READ SIGNAL
```

```
PAGE 0002
0095        BITCNT RES  1         WORD COUNTER
0096        BITAD  RES  1         WORD ADDRESS
0097  F99B         JST  *CHAD     END OF BLOCK
0098  4093         SEL  BIT,3     DO CONVERT
0099  0A00  EXIT   EIN            ENABLE INTERRUPTS
009A  F193         RTN  AIN       RETURN
009B  01F2  CHAD   DATA CHNDNE
009C  04CD  AVGAD  BAC  MEND-1    MESSAGE ADDRESS
009D  04BB  MAXAD  BAC  MAVG-1    MESSAGE ADDRESS
009E  04A3  MESAD  BAC  MES+5     TO PUT IN CHANNEL NUMBER
009F  04A9  MINAD  BAC  MNMES-1   MESSAGE ADDRESS
00A0        SIGN   RES  1
00A1  2710  TTHOU  DATA 10000
00A2  0009  NINE   DATA 9
00A3  5F5F  MULT   DATA 24415     SIGNAL CONVERSION VALUE
00A4  0800  COR    DATA :800      CORRECTIVE AMOUNT
            *
0100               ABS  :100
0100  0C00  START  DIN            DISABLE INTERRUPTS
0101  4044         SEL  RTC,4     CLEAR RTC
0102  4090         SEL  BIT,0     CLEAR 16-BIT
0103  0110         ZAR            TO
0104  9803         STA  TTYWCT    CLEAR TTY
0105  FA63         JST  TTYMES    PRINT MESSAGE
0106  FFF8         DATA -8        MESSAGE LENGTH
0107  0491         BAC  SMES-1    MESSAGE ADDRESS
0108  4802         SSN  0,2       CHECK FOR RTC
0109  F204         JMP  GO        CONTINUE
010A  FA5E         JST  TTYMES    PRINT MESSAGE
010B  FFFA         DATA -6        MESSAGE LENGTH
010C  048B         BAC  RTMES-1   MESSAGE ADDRESS
010D  0800         HLT            WAIT FOR CORRECTION
            *
010E  0110  GO     ZAR            CLEAR A
010F  9800         STA  LINE      AND LINE NUMBER
0110  B005         LDA  NEWAD     TO RESET
0111  9806         STA  AP        ADDRESS POINTER
0112  FA56         JST  TTYMES    PRINT MESSAGE
0113  FFFD         DATA -3        MESSAGE LENGTH
0114  0499         BAC  READY-1   MESSAGE ADDRESS
0115  4038         SEL  TTY,0     SELECT AUTO ECHO
0116  5B39  READ   RDX  TTY,1     READ INPUT
0117  C1AE         CXI  '.'       SEE IF .
0118  F222         JMP  LDONE     YES IT IS
0119  C1AD         CXI  '-'       SEE IF '-'
011A  F20E         JMP  FX-2      YES IT IS
011B  C3B0         SXI  :B0       CONVERT TO BINARY DIGIT
011C  0030         TXA            PUT IN A
011D  20CF         JAM  GO        ERROR
011E  D0A2         CMS  NINE      COMPARE WITH 9
011F  F201         JMP  $+2       OK IF <
0120  F612         JMP  GO        ERROR IF > 9
0121  8800         ADD  LINE
0122  8800         ADD  LINE
```

```
PAGE  0003
0123  B800           EMA  LINE
0124  1352           LLA  3         * BY 8
0125  8800           ADD  LINE      ENDS UP WITH 10LINE + A
0126  9800           STA  LINE      STORE NEW LINE TOTAL
0127  9801           STA  COUNT
0128  F612           JMP  READ      CONTINUE READING
                *
0129  0110           ZAR            CLEAR A
012A  9801           STA  COUNT
012B  5B39  FX       RDX  TTY,1     GET BYTE
012C  C1AE           CXI  '.'       SEE IF '.'
012D  F20D           JMP  LDONE     YES IT IS
012E  C3B0           SXI  :B0       CONVERT TO BINARY
012F  0030           TXA            PUT IN A
0130  20E2           JAM  GO        ERROR
0131  D0A2           CMS  NINE      COMPARE WITH 9
0132  F201           JMP  $+2       OK IF <
0133  F625           JMP  GO        ERROR IF > 9
0134  8801           ADD  COUNT
0135  8801           ADD  COUNT
0136  B801           EMA  COUNT
0137  1352           LLA  3         * BY 8
0138  8801           ADD  COUNT     ENDS UP WITH 10COUNT + A
0139  9801           STA  COUNT
013A  F60F           JMP  FX        CONTINUE READING
013B  B001  LDONE    LDA  COUNT     GET UPPER CHN
013C  D000           CMS  LINE      COMPARE WITH LOWER CHN
013D  F62F           JMP  GO        ERROR
013E  0000           NOP
013F  1357           LLA  8         SHIFT
0140  A000           IOR  LINE      CHANNEL WORD FOR CONVERSION
0141  00D0           DAR            ADJUST TO CORRECT CHANNEL
0142  9800           STA  LINE      STORE CHANNEL SELECTION
0143  0110           ZAR            CLEAR A
0144  E0A4           LDX  COR       GET TABLE SIZE
0145  00A8           DXR            DECREMENT INDEX
0146  9D0A           STA  @*BUFAD   CLEAR TABLE
0147  3842           JXN  $-2       LOOP UNTIL DONE
0148  B233           LDA  TTYAOB    TO RESET MIN
0149  C5FF           LXM  255       RESET INDEX
014A  9D0A           STA  @*BUFAD
014B  0128           IXR            INCRMENT INDEX
014C  3842           JXN  $-2       LOOP UNTIL DONE
014D  C72A           LAM  42        GET RANGE LIMIT
014E  9812           STA  COUNT3    AND STORE IT
014F  B000           LDA  LINE      GET CHANNEL RANGE
0150  4040           SEL  RTC,0     ENABLE RTC
0151  4042           SEL  RTC,2     ENABLE SYNC
0152  FA02           JST  STRTCH    SELECT CHANNELS
0153  0A00           EIN            ENABLE INTERRUPTS
0154  F600           JMP  $         LOOP HERE
                *
0155  0800  STRTCH   ENT            16-BIT CHANNEL SELECTOR
0156  4090           SEL  BIT,0     CLEAR INTERFACE
```

```
PAGE   0004
0157  6C92           OTA   BIT,2      SELECT CHANNELS
0158  4091           SEL   BIT,1      ENABLE WORD INTERRUPTS
0159  C705           LAM   5          TO RESET RTC
015A  9819           STA   RTCONT     COUNTER
015B  4092           SEL   BIT,2      ENABLE END OF BLOCK INTERRUPT
015C  B00A           LDA   BUFAD      GET BUFFER ADDRESS
015D  9896           STA   BITAD      STORE FOR AUTO-I-O
015E  B010           LDA   M768       GET -768 FOR 3 CHANNEL LOOPS
015F  9895           STA   BITCNT
0160  4093           SEL   BIT,3      ISSUE CONVERT INSTRUCTION
0161  F70C           RTN   STRTCH     RETURN
             *
0162  FA06   DONE    JST   TTYMES     PRINT MESSAGE
0163  FFC4           DATA  -60        MESSAGE LENGTH
0164  049D           BAC   MES-1      MESSAGE ADDRESS
0165  D814           IMS   COUNT2     INCREMENT CHANNEL NUMBER
0166  D812           IMS   COUNT3     INCREMENT NUMBER OF CHN LEFT
0167  F221           JMP   MINMES     CONTINUE UNTIL FINISHED
0168  F65A           JMP   GO         START AGAIN
             *
0169  0800   TTYMES  ENT              PRINTS TTY MESSAGES
016A  4090           SEL   BIT,0      CLEAR INTERFACE
016B  4044           SEL   RTC,4      DISABLE RTC
016C  B003           LDA   TTYWCT     GET BYTE COUNT
016D  3141           JAN   $-1        CHECK FOR NOT BUSY
016E  B20D           LDA   TTYAOB     GET TTY AUTO I-O INSTRUCTION
016F  9802           STA   TTYWRD     STORE IN INTERRUPT
0170  E607           LDX   TTYMES     GET RETURN ADDRESS
0171  B400           LDA   00         GET BYTE COUNT
0172  9A0A           STA   TTYAOB+1   AND STORE
0173  0150           IAR
0174  9803           STA   TTYWCT     PUT FOR INTERRUPT
0175  B401           LDA   01         GET MESSAGE ADDRESS
0176  9A07           STA   TTYAOB+2   STORE
0177  0150           IAR
0178  9804           STA   TTYBAD     AND FOR INTERRUPT
0179  493B           SEN   TTY,3      SEE IF TTY BUSY
017A  F601           JMP   $-1        YES IT IS
017B  403C           SEL   TTY,4      CLEAR TTY
017C  6439   TTYAOB  AOB   TTY,1      START OUTPUT
017D                 RES   3
0180  403D           SEL   TTY,5      SELECT WORD INTERRUPT
0181  0A00           EIN              ENABLE INTERRUPTS
0182  B003           LDA   TTYWCT     GET BYTE COUNTER
0183  3141           JAN   $-1        LOOP UNITL DONE
0184  493B           SEN   TTY,3      SEE IF TTY BUSY
0185  F601           JMP   $-1        LOOP UNTIL NOT
0186  403C           SEL   TTY,4      CLEAR TTY
0187  0C00           DIN              DISABLE INTERRUPTS
0188  F402           JMP   02         RETURN
             *
0189  C703   MINMES  LAM   3          LOOP COUNTER
018A  9801           STA   COUNT      STORE LOOP COUNTER
018B  B005           LDA   NEWAD      GET ADDRESS POINTER
```

```
PAGE 0005
018C 9806         STA  AP        AND RESET POINTER
018D B09E         LDA  MESAD     GET BAD ADDRESS
018E 9813         STA  BAD
018F E014         LDX  COUNT2    GET CHANNEL NUMBER
0190 0128         IXR            INCREMENT TO NEXT CHANNEL
0191 0110  DLOOP  ZAR            CLEAR A
0192 0F00         SWM            SET WORD MODE
0193 F900         JST  CRID:     /10
0194 0015         DATA TEN
0195 0E00         SBM            SET BYTE MODE
0196 8822         ADDB ZERO      +:B0
0197 9913         STAB *BAD      STORE BYTE IN MESSAGE
0198 C701         LAM  1         TO DECREMENT ADDRESS
0199 6803         SIN  2         INHIBIT STATUS
019A 8813         ADD  BAD
019B 9813         STA  BAD
019C D801         IMS  COUNT     INCREMENT LOOP COUNTER
019D F60C         JMP  DLOOP     CONTINUE
019E 0F00         SWM            SET WORD MODE
019F E014         LDX  COUNT2    GET CHANNEL NUMBER
01A0 E50C         LDX  0*MINMIN  GET CHANNEL MINIMUM
01A1 B09F         LDA  MINAD     GET BUFFER ADDRESS
           *
01A2 E8A0  FIN    STX  SIGN      SAVE SIGNAL
01A3 9813         STA  BAD       RESET BYTE ADDRESS
01A4 C704         LAM  4
01A5 9801         STA  COUNT     STORE BYTE COUNTER
01A6 0E00         SBM            SET BYTE MODE
01A7 0110  MINLOP ZAR            CLEAR A
01A8 1903         LRL  4         MOVE DIGIT INTO A
01A9 D02B         CMSB TEN+1     COMPARE WITH 10
01AA F202         JMP  MINA      OK
01AB 0000         NOP
01AC 8823         ADDB ZERO+1    +7
01AD 8822  MINA   ADDB ZERO      +:B0
01AE D813         IMS  BAD       INCREMENT POINTER
01AF 9913         STAB *BAD      STORE IN MESSAGE
01B0 D801         IMS  COUNT     INCREMENT COUNTER
01B1 F60A         JMP  MINLOP    KEEP GOING
01B2 0F00         SWM            SET WORD MODE
01B3 B0A0         LDA  SIGN      GET SIGNAL
01B4 D813         IMS  BAD       INCREMENT BYTE POINTER
01B5 D813         IMS  BAD
01B6 C503         LXM  3         THE NUMBER OF DECIMAL
01B7 E801         STX  COUNT     DIGITS
01B8 90A4         SUB  COR       CORRECT SIGNAL
01B9 C4AB         LXP  '+'       SET TO THIS SIGN INITIALLY
01BA 3082         JAP  $+3       SKIP IF VOLTAGE IS +
01BB C4AD         LXP  '-'       ELSE SET AS -
01BC 0310         NAR            AND CONVERT TO ABS VAL
01BD 0E00         SBM            SET BYTE MODE
01BE E913         STXB *BAD      STORE SIGN
01BF 0F00         SWM            SET WORD MODE
01C0 E0A3         LDX  MULT      GET MULTIPLIER
```

```
PAGE   0006
01C1   F900           JST   MPYS:     A*X
01C2   F900           JST   CRID:     /10000
01C3   00A1           DATA  TTHOU
01C4   C606           LAP   6         TO ADJUST ADDRESS
01C5   8813           ADD   BAD       CALCULATE NEW
01C6   9813           STA   BAD       ADDRESS
01C7   B013   MINB    LDA   BAD       GET BUFFER ADDRESS
01C8   00D0           DAR             DECREMENT POINTER
01C9   9813           STA   BAD       STORE NEW BUFFER POINTER
01CA   0110           ZAR             CLEAR A
01CB   F900           JST   CRID:     /10
01CC   0015           DATA  TEN
01CD   0E00           SBM             SET BYTE MODE
01CE   8822           ADDB  ZERO      +:B0
01CF   9913           STAB  *BAD      STORE REMAINDER IN MESSAGE
01D0   0F00           SWM             SET WORD MODE
01D1   D801           IMS   COUNT     INCREMENT LOOP COUNTER
01D2   F60B           JMP   MINB
01D3   C702           LAM   2
01D4   8813           ADD   BAD       TO RESET
01D5   9813           STA   BAD       ADDRESS POINTER
01D6   0110           ZAR             CLEAR A
01D7   F900           JST   CRID:     /10
01D8   0015           DATA  TEN
01D9   0E00           SBM             SET BYTE MODE
01DA   8822           ADDB  ZERO      +:B0
01DB   9913           STAB  *BAD      STORE IN BUFFER
01DC   0F00           SWM             SET WORD MODE
01DD   D806           IMS   AP        INCREMENT ADDRESS POINTER
01DE   F106           JMP   *AP       GO TO ROUTINE
              *
01DF   B09D   MAXMES  LDA   MAXAD     GET MESSAGE ADDRESS
01E0   E014           LDX   COUNT2    GET CURRENT CHANNEL
01E1   E50B           LDX   @*MAXMAX  GET CHANNEL MAXIMUM
01E2   F640           JMP   FIN       FILL BUFFER
              *
01E3   B014   AVGMES  LDA   COUNT2    GET CURRENT CHANNEL
01E4   0048           TAX             GET AS INDEX
01E5   880D           ADD   TOTAD     GET ADDRESS OF TOTAL
01E6   9A03           STA   TOTAL
01E7   B50E           LDA   @*AVGAAD  GET TOP TO AVG
01E8   E50F           LDX   @*AVGXAD  GET BOTTOM OF AVG
01E9   F900           JST   CRID:     GET AVG
01EA          TOTAL   RES   1         NUMBER OF ELEMENTS
01EB   B09C           LDA   AVGAD     GET BUFFER ADDRESS
01EC   F64A           JMP   FIN       FILL MESSAGE
              *
01ED   0800   TIMERR  ENT             PRINT INTERRUPT MESSAGE
01EE   FE85           JST   TTYMES    PRINT MESSAGE
01EF   FFF4           DATA  -12       MESSAGE LENGTH
01F0   046B           BAC   BMES-1    MESSAGE ADDRESS
01F1   F6E3           JMP   GO        RETURN
              *
01F2   0800   CHNDNE  ENT             CHANNEL COMPLETION ROUTINE
```

```
PAGE 0007
01F3  B010         LDA  M768      GET -768
01F4  9814         STA  COUNT2    STORE WORD COUNTER
01F5  E00A         LDX  BUFAD     GET WORD INDEX
01F6  0E00         SBM            SET BYTE MODE
01F7  B001         LDAB LINE+1    GET LOWER LIMIT
01F8  982D         STAB LNUM+1    INITIALIZE CHECKER
01F9  1328         LLX  1         CONVERT INDEX TO BYTE ADDRESS
01FA  6802         SIN  1         INHIBIT STATUS
01FB  E8A0         STX  SIGN      STORE INDEX
01FC  6804  CLOOP  SIN  3         INHIBIT STATUS
01FD  E0A0         LDX  SIGN      GET INDEX
01FE  C202         AXI  2         INCREMENT TO PROPER BYTE
01FF  E8A0         STX  SIGN      STORE NEW INDEX
0200  B400         LDAB 00        GET UPPER BYTE
0201  13D3         LRA  4         SHIFT DOWN TO CHANNEL MOD
0202  A82D         XORB LNUM+1    TEST WITH PROPER RESULT
0203  802E         ANDB FFF       MASK OFF LOWER 4 BITS
0204  2105         JAZ  CGOOD     CHANNELS MATCH
0205  0F00  CBAD   SWM            SET WORD MODE
0206  FE9D         JST  TTYMES    PRINT MESSAGE
0207  FFEC         DATA -20       MESSAGE LENGTH
0208  0477         BAC  CMES-1    MESSAGE ADDDRESS
0209  F6FB         JMP  GO        RETURN
020A  6803  CGOOD  SIN  2         INHIBIT STATUS
020B  13A8         LRX  1         CONVERT BYTE TO WORD ADDRESS
020C  B400         LDA  00        GET INPUT WORD
020D  E02D         LDXB LNUM+1    GET CURRENT CHANNEL NUMBER
020E  0F00         SWM            SET WORD MODE
020F  1353         LLA  4         SHIFT OFF CHANNEL MOD
0210  13D3         LRA  4         RESTORE TO POSITION
0211  D50C         CMS  0*MINMIN  COMPARE WITH CHANNEL MINIMUM
0212  9D0C         STA  0*MINMIN  REPLACE IF LESS
0213  0000         NOP
0214  D50B         CMS  0*MAXMAX  COMPARE WITH CHANNEL MAXIMUM
0215  F201         JMP  $+2       SKIP REPLACING
0216  9D0B         STA  0*MAXMAX  REPLACE IF GREATER
0217  1200         ROV            RESET OVERFLOW
0218  8D0F         ADD  0*AVGXAD  ADD TO AVG
0219  3203         JOR  NOV       SKIP IF NO OVERFLOW
021A  1350         LLA  1         SHIFT OFF SIGN BIT
021B  13D0         LRA  1         RESTORE TO POSITION
021C  DD0E         IMS  0*AVGAAD  INCREMENT SIGNICANT PART
021D  9D0F  NOV    STA  0*AVGXAD  STORE LEAST SIGNICANT PART
021E  0E00         SBM            SET BYTE MODE
021F  0130         IXA            INCREMENT TO NEXT CHANNEL
0220  DD0D         IMS  0*TOTAD   INCREMENT ELEMENT COUNTER
0221  D800         CMSB LINE      COMPARE WITH LIMIT
0222  F202         JMP  OUT       OK
0223  F61E         JMP  CBAD      ERROR
0224  B001         LDAB LINE+1    RESET TO 1ST CHANNEL
0225  982D  OUT    STAB LNUM+1    STORE NEXT CHANNEL
0226  D814         IMS  COUNT2    INNER LOOP COUNTER
0227  F62B         JMP  CLOOP     KEEP GOING
0228  0F00         SWM            SET WORD MODE
```

```
PAGE  0008
0379  0229  B000         LDA  LINE      GET CHANNEL SELECT
0380  022A  FED5         JST  STRTCH    SELECT CHANNELS
0381  022B  D812         IMS  COUNT3    INCREMENT MASTER COUNTER
0382  022C  F099         JMP  EXIT      RETURN
0383  022D  4090         SEL  BIT,0     STOP READINGS
0384  022E  0E00         SBM            SET BYTE MODE
0385  022F  E001         LDXB LINE+1    GET 1ST CHANNEL NUMBER
0386  0230  0030         TXA            PUT IN A
0387  0231  9000         SUBB LINE      - LAST CHANNEL
0388  0232  0F00         SWM            SET WORD MODE
0389  0233  9812         STA  COUNT3   NUMBER OF CHANNELS TO LIST
0390  0234  E814         STX  COUNT2    1ST CHANNEL TO LIST
0391  0235  F6AC         JMP  MINMES    GO DO MINIMUM
0392  0236  CECF  BMES   TEXT 'NO INTERRUPT'
      0237  A0C9
      0238  CED4
      0239  C5D2
      023A  D2D5
      023B  D0D4
0393  023C  C3C8  CMES   TEXT 'CHANNEL NUMBER ERROR'
      023D  C1CE
      023E  CEC5
      023F  CCA0
      0240  CED5
      0241  CDC2
      0242  C5D2
      0243  A0C5
      0244  D2D2
      0245  CFD2
0394  0246  CECF  RTMES  TEXT 'NO RTC'
      0247  A0D2
      0248  D4C3
0395  0249  8D8A  SMES   DATA :8D8A     CRLF
0396  024A  C3C8         TEXT 'CHNSCN'
      024B  CED3
      024C  C3CE
0397  024D  8D8A  READY  DATA :8D8A     CRLF
0398  024E  BE00         DATA :BE00     '>'
0399  024F  8D8A  MES    DATA :8D8A     CRLF
0400  0250  A0A0         TEXT '      MIN= '
      0251  A0A0
      0252  A0CD
      0253  C9CE
      0254  BDA0
0401  0255        MNMES  RES  2
0402  0257  A0A0         TEXT '   .      MAX= '
      0258  A0AE
      0259  A0A0
      025A  A0A0
      025B  A0CD
      025C  C1D8
      025D  BDA0
0403  025E        MAVG   RES  2
0404  0260  A0A0         TEXT '   .      AVG= '
```

```
PAGE  0009
      0261  A0AE
      0262  A0A0
      0263  A0A0
      0264  A0C1
      0265  D6C7
      0266  BDA0
0405  0267              MEND   RES   2
0406  0269  A0A0               TEXT  '    .      '
      026A  A0AE
      026B  A0A0
      026C  A0A0
0407        026D        END    EQU   $            SAVE ORIGIN
0408  02C0                     ORG   :2C0         SET ORIGIN FOR STORAGE
0409  02C0              MIN    RES   256
0410  03C0              BUF    RES   768
0411  06C0              MAX    RES   256
0412  07C0              AVGA   RES   256
0413  08C0              AVGX   RES   256
0414  09C0              TOT    RES   256
0415  026D                     ORG   END          RESET ORIGIN
0416        0100               END   START
0000  ERRORS
```

APPENDIX E

DATA CONVERTER PARTS LIST

APPENDIX E

Data converter parts list

Item	Quantity	Description	Unit cost ($)	Total cost ($)
1.	15	7400 Quad 2-input Nand gates	0.55	8.25
2.	5	7475 Quad latches	1.40	7.00
3.	3	74193 Binary counter	3.90	11.70
4.	3	7485 4-bit magnitude comparator	7.00	21.00
5.	5	74121 Monostable multi-vibrator	1.10	5.50
6.	6	7404 Hex inverter	0.70	4.20
7.	4	MM8 multiplexer	99.00	396.00
8.	1	DAS-16 ADC system	695.00	695.00
9.	1	DPS-1 5V@3A power supply	32.06	32.06
10.	1	BPM-15/300 $\pm$15V@300 mA power supply	89.00	89.00
11.	1	MS-7 BPM socket	3.50	3.50
12.	40	10k ohm resistors	0.10	4.00
13.	20	220 ohm resistors	0.10	2.00
14.	20	330 ohm resistors	0.10	2.00
15.	10	2.2m ohm resistors	0.10	1.00
16.	10	150 ohm resistors	0.10	1.00
17.	10	1k ohm resistors	0.10	1.00
18.	10	10k film resistors	1.34	13.40
19.	50	0.01 mF capacitors 10WV	0.15	7.50
20.	20	TIL 210 LED	1.40	28.00
21.	32	BNC connectors	0.76	24.32
22.	3	3-terminal connectors	0.55	1.65
23.	3	4-terminal connectors	1.05	3.15
24.	1	AC line chord	1.50	1.50
25.	1	Pilot light	1.00	1.00
26.	1	Fuse holder/fuse	2.25	2.25
27.	1	Can of paint	1.75	1.75
28.	1	Bud relay rack (table)	18.00	18.00
29.	2	AMP DIP switches	3.73	7.46
30.	16	JBT SPST toggle switches	1.60	25.60
31.	33	JBT DPDT toggle switches	1.58	52.14
32.	2	JBT SPST momentary toggle switches	1.90	3.80

Item	Quantity	Description	Unit cost ($)	Total cost ($)
33.	2	Bannana jacks-green	0.95	1.90
34.	2	Bannana jacks-white	0.95	1.90
35.	2	Bannana jacks-black	0.95	1.90
36.	2	Bannana jacks-red	0.95	1.90
37.	2	RCA CA3081 NPN transistor array	6.24	12.48
38.	1	Beldon 8700 28 AWG coaxial cable-50 feet	14.40	14.40
39.	1	Wirewrap wire-30 AWG	59.25	59.25
40.	1	Wirewrap wire-24 AWG	25.34	25.34
41.	1	Bag of 100 wirewrap stakes	26.85	26.85
42.	10	Printed circuit card connectors	3.90	39.00
43.	1	Bag of wirewrap sockets	12.30	12.30
44.	1	Universal card cage assembly	31.44	31.44
45.	1	Bag of card guides	6.75	6.75
46.	10	Universal cards	6.18	61.80
47.	2	Boxes of 2-56 machine screws	2.00	4.00
48.	2	Boxes of 2-56 nuts	2.00	4.00
49.	2	Boxes of #2 internal tooth lockwashers	2.00	4.00
50.	1	100 foot roll of twisted-pairs ribbon cable	48.00	4a.00
51.	50	Augat 16-pin wirewrap DIP sockets	0.89	44.50
52.	30	Augat 14-pin wirewrap DIP sockets	0.79	23.70
53.	1	8"x19" vector board	11.00	11.00
54.	1	Bag of 100 clear cable ties	1.10	1.10
55.	1	Misc. aluminum & machining	50.00	50.00

TOTAL COST OF ALL PARTS- $1976.24