MICROCOMPUTER BASED DESIGN OF A NEUROPHYSIOLOGICAL STIMULATOR

By

John A. Halter

Electrical Engineering

Submitted in Partial Fulfillment of the Requirements of the University Undergraduate Fellows Program

1977-1978

Approved by:

Barry Mattox

Dr. B. G. Mattox

April 1978

ABSTRACT

The design of a microcomputer controlled neurophysiological stimulator is described. The stimulator is a complete self contained computer system with the capability of controlling multiple completely independent pulse trains. The stimulator possesses a custom program console via which an operator can specify the following parameters: pulse width, pulse frequency, number of pulses in a train, train frequency, triggering mode (internal, external, manual), pulse delay, and a random option for any parameter with preset boundaries. Limits of the stimulator are .1 ms to 1 sec. for the pulse width and 5 khz to .5 hz for the pulse frequency. The system is designed around a Motorola 6800 microprocessor. The stimulator also has the capabilities of communicating with a larger computer, enabling the implementation of even more complex and sophisticated experimentation.

Dedicated to: Elaine,

Who handles a wire wrap gun very well.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES

INTRODUCTION

Sophisticated electronic systems are becoming increasingly integral to the modern research laboratory. One laboratory which utilizes a relatively large quantity of electronics is the neurophysiological research laboratory at the Texas Institute for Research and Rehabilitation (TIRR). Researchers at this laboratory are involved both in studying and treating neuromuscular systems.

A typical experiment at the laboratory involves stimulating the nervous system of a patient/subject with an electronic stimulator and monitoring the responses with various electronic recording equipment. It is hoped that these experiments will increase understanding of neuromuscular systems and hopefully result in some theraputic techniques. Figure 1 is an illustration of a typical neuromuscular signal. It is this type of signal that the researchers wish to artificially generate with their electronic stimulator. Figure 2 illustrates the same parameters which need to be controlled. The stimulator presently in use at the laboratory is an analog-based piece of equipment. In other words, it contains several knobs which are adjusted to determine these various parameters. The problem with this piece of equipment is that it takes

---

*Modeled after the IEEE Transactions on Biomedical Engineering.

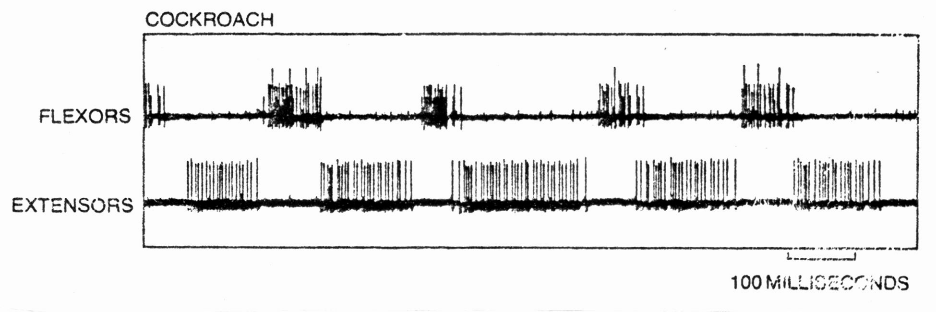


Figure 1

Typical Neuromuscular Signal

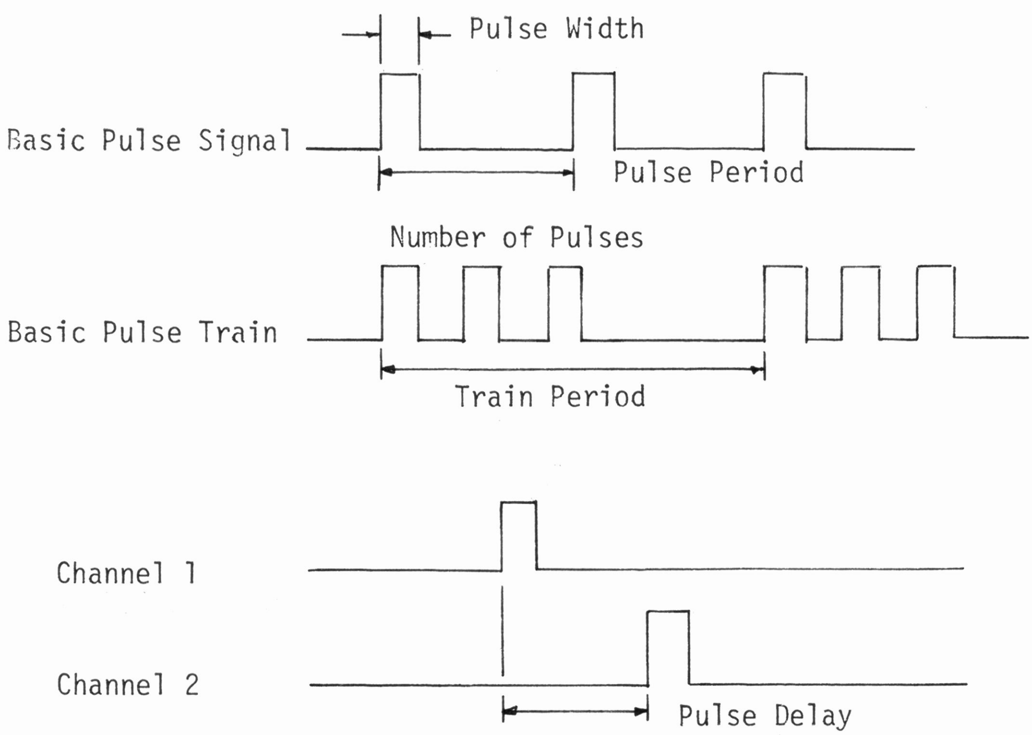


Figure 2

Basic Pulse Parameters

a long time to re-adjust between phases of an experiment. This creates difficulty in their experimentation due to the fact that the human subject can become fatigued and/or nervous during experiments, which last typically two to five hours. The researchers noted the fact that if they had a piece of equipment which was much easier to adjust, experimentation time would be significantly decreased (i.e., 20 minutes as compared to two hours).

As a result, they generated a list of desired specifications for a neurophysiological stimulator that would not only satisfy their immediate need, but would hopefully expand their potential experimental capability. The specifications are listed as follows:

- Capable of controlling the following parameters:
    1. pulse width (.1 ms to 1 sec.)
    2. pulse frequency (1 hz to 5 khz)
    3. number of pulses in train
    4. train frequency
    5. pulse delay
    6. triggering mode selection (ext., int., manual)
    7. random option (for any parameter)
- Programmable
- Multichannel capacity
- Capable of communicating with a larger computer
- "Stand-alone unit," i.e., self contained
- Easy to operate
- Modular and expandable for future experimentation

Equipped with a system specification, an appropriate design was generated.

NEUROPHYSIOLOGICAL STIMULATOR DESIGN

Preliminary System Design

Figure 3 illustrates the basic components required for the neurophysiological stimulator. The stimulator is controlled by a central control unit which communicates to the operator via an operators console. The stimulator presently in use utilizes multiposition switches and dials on its operator console. The stimulator also possesses some form of information storage to satisfy the programmability requirement. In addition, the central control unit communicates to some pulse generators, whose output signals are then utilized to stimulate the subject. Equipped with a basic system design, the specific design alternatives were then considered.

Design Alternatives

The use of analog type circuitry was readily disregarded since it would involve using the same controls (knobs, etc.). Therefore a digital design was considered, the abundance of components and easy implementation of programmable circuits being a strong advantage. The next design decision involved selection of the specific digital technology to be implemented. The stimulator control circuitry could be constructed out of small scale integrated circuits (TTL: Transistor-Transistor Logic), but the component

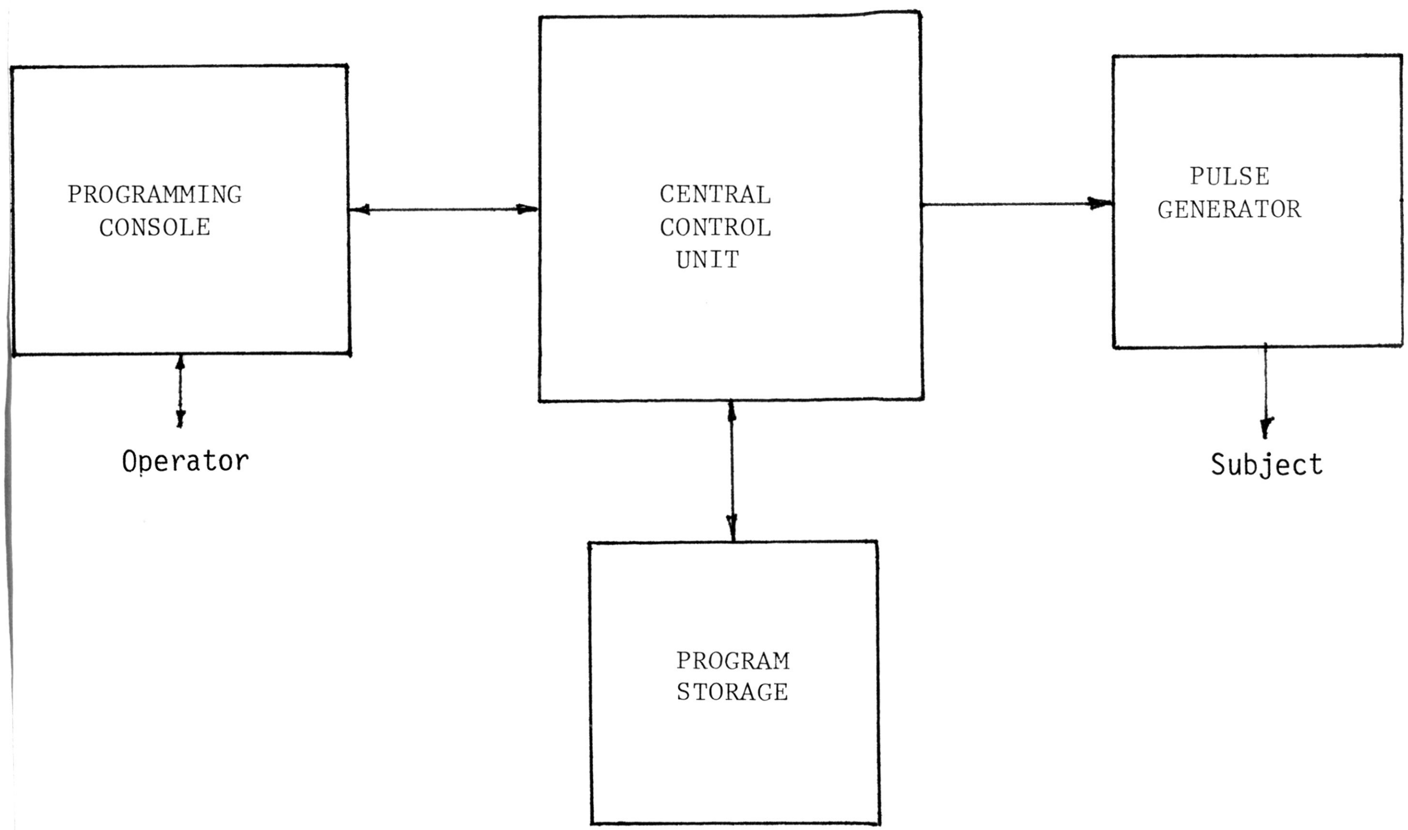


Figure 3

Basic Stimulator Configuration

quantity would be quite large and the design necessarily complex in order to satisfy the programmability requirement. In addition, the stimulator would be extremely limited, due to the inflexibility of the circuitry. A much more optimum path to take for the design of the central control unit would be to use one of the latest products generated by the electronics industry, namely the microprocessor. The microprocessor (a single electronic component) is a small computer, small in size and cost but large in capability. It is ideal for this application, due to the fact that it is by nature programmable and capable of easily controlling of other electronic devices. Since a microcomputer (a microprocessor with supportive electronics) is under consideration as a key part of the design, the software or programming of the microcomputer must be considered. The microprocessor was designed to be a bridge between hardware (electronic components) and software (programs). The nature of the device is such that the microprocessor can be programmed to perform functions which were previously done in hardware form. The problem is that the more a microcomputer has to do, the longer it takes.

Since the objective of the project is to generate precision pulse trains, this must be taken into account. It was therefore decided to build programmable pulse generators which would independently generate pulse signals with parameters defined by the microcomputer. This would tremendously relieve the workload of the microcomputer which would have had to perform all of the timing requirements. Figure 4 illustrates the final system block diagram. The microcomputer controls all of the other devices which consist of common TTL circuitry.

Operator Console Design

This was a very important aspect of the system design. The personnel at TIRR wanted the device to be straightforward and easy to use. The last thing that they desired was to make their experimentation procedure more tedious than it already was. An operator console layout was generated (figure 5) which met with their approval. Since a microcomputer was being used as the central control unit it was relatively easy to control a sophisticated operator console. Under actual operation the microcomputer would illuminate the appropriate light on the left side of the console, requesting information from the operator. The operator would then enter the appropriate information on the data keypad in the center of the calculator. The entry would then be displayed on the four digit numerical display on the top of the console. In addition, the appropriate status lights to the right of the display would be illuminated. The keys on the right portion of the console are special function keys which permit the operator to

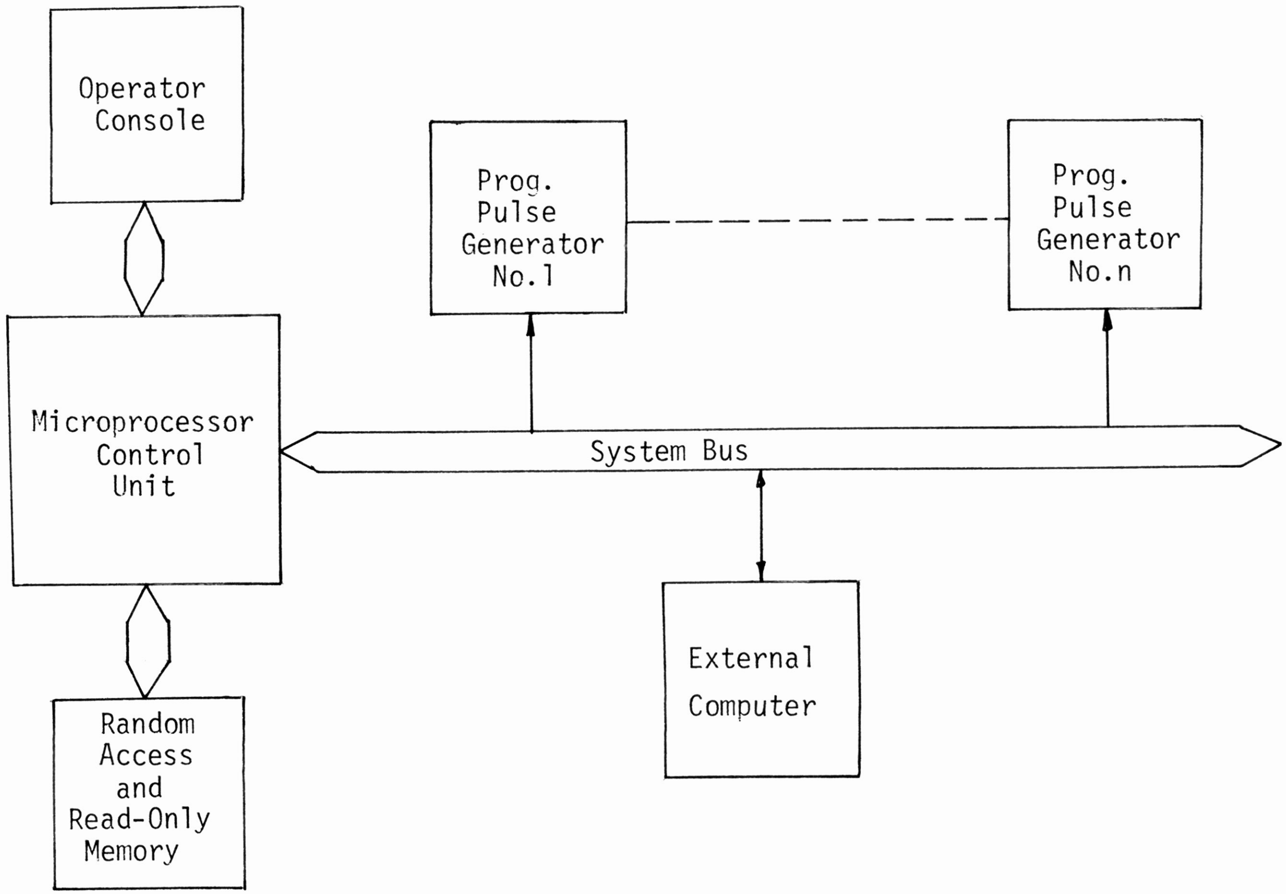


Figure 4

Stimulator System Block Diagram

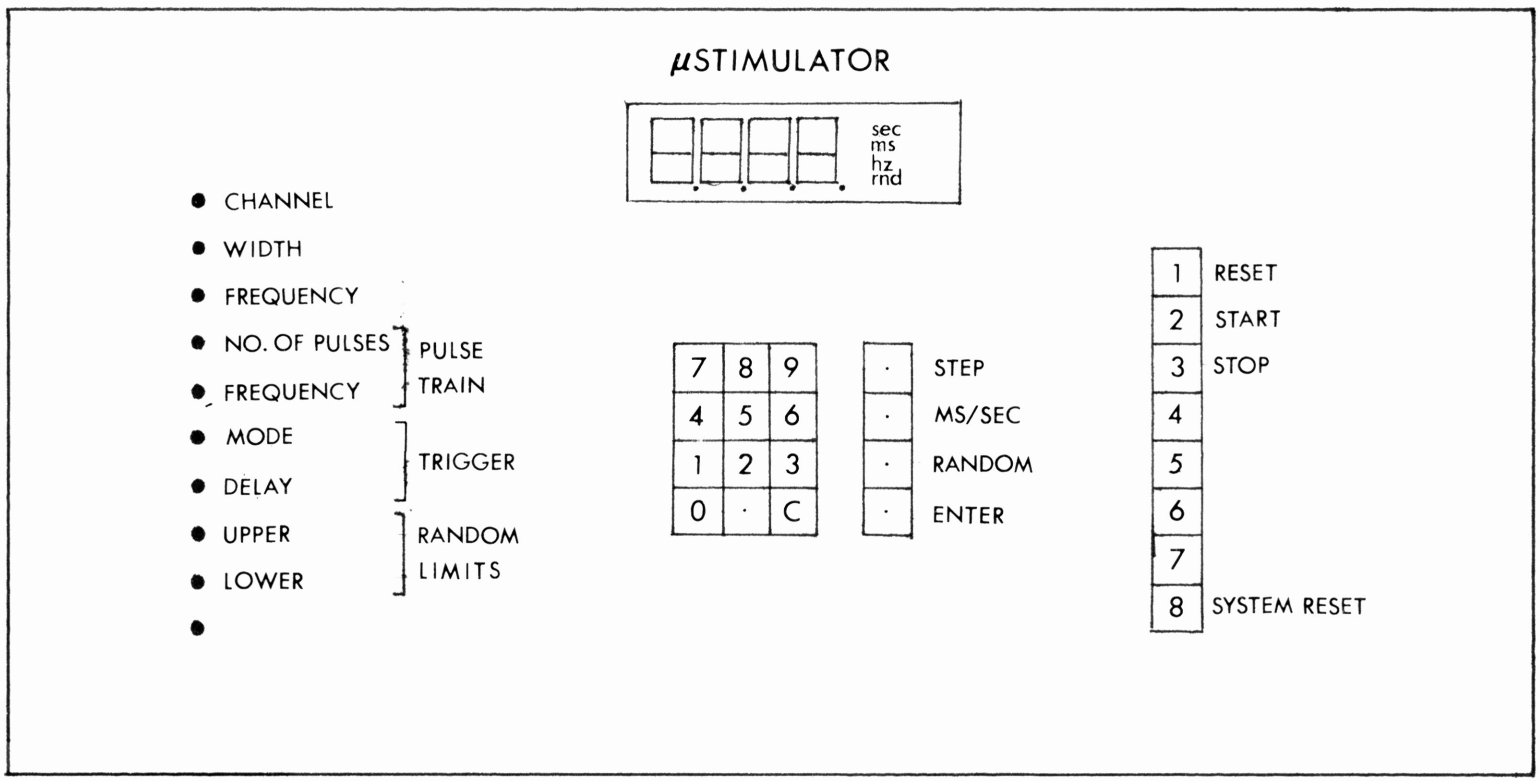


Figure 5

Operator Console Layout

reset the stimulator or start and stop a pulse generator. There are also some unused keys whose functions can be defined by the user. For instance, since four keys are available, four separate experimental configurations could be preprogrammed and stored for immediate one-key recall. The utility and flexibility of this console should be stressed. All that needs to be modified in order to change operation is some software used by the microcomputer. The design of this portion of the project involved the most negotiation. All of the required keystrokes and sequences were discussed to insure that the proposed console configuration was acceptable.

SYSTEM HARDWARE

This section deals with each hardware module in the stimulator.

Microcomputer

This is the heart of the stimulator. This is also the only portion of the project which was pre-engineered, i.e., pre-designed. The microprocessor being utilized is a Motorola 6800 microprocessor. It is supported by various TTL circuitry and circuit components which are mounted on a printed circuit board designed by Southwest Technical Products Corporation (SWTPC) (see Figure 6). This is the computer board of the SWTPC 6800 computer system. In addition to the microcomputer board a memory board is being used in the stimulator system (see Figure 7). This memory board is also a product of Southwest Technical Products and has a capacity of 4096 bytes of memory. This memory is used by the stimulator to store various pulse parameters (data). The memory board has the capacity for much more memory than will most likely be needed, providing TIRR with potential inexpensive expandability.

The other pre-engineered board by SWTPC is a communications board (see Figure 8). This board enables a terminal such as a teletype to be connected to the stimulator.

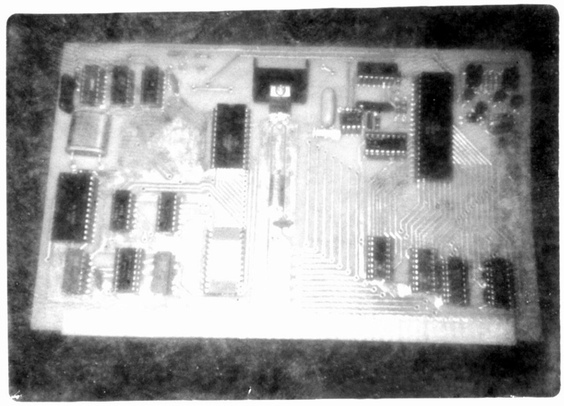


Figure 6

SWTPC 6800 Computer Board

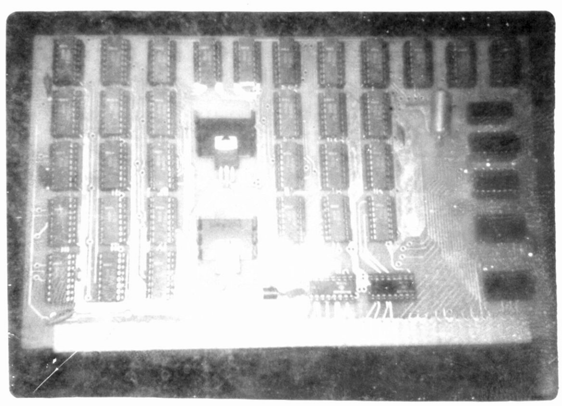


Figure 7

SWTPC 4K Memory Board

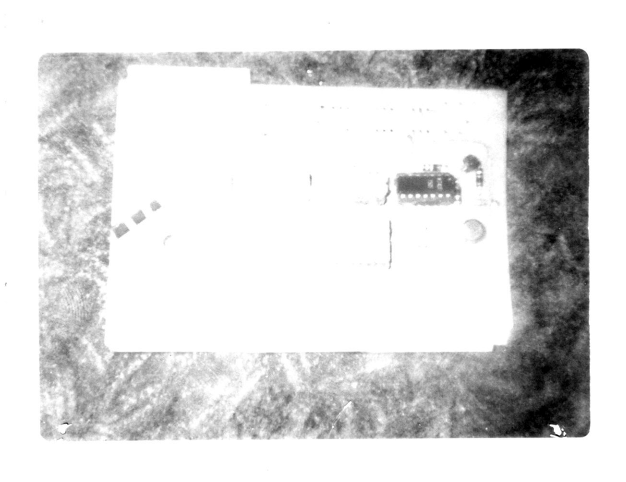


Figure 8

SWTPC Communication Board

This was installed for development and maintenance purposes since the front console is extremely limited. This enables communication directly with the microcomputer. In addition, this communication board could be used as a link to another computer, such as the Hewlett-Packard mini-computer which TIRR possesses. This is one more benefit of utilizing a microprocessor, enabling virtually unlimited expansion and complexity of operation.

ROM/Address Decode/Clock Board

This board serves several purposes (see Figure 9). First of all it contains special read-only-memory devices (ROM's) which contain the software (programming) required to make the stimulator function. The ROM's are located on the upper half of the board. The schematic for this section of the board can be seen in Figure 10. The computer communicates via some data lines and selects what it is going to communicate with by utilizing its address lines. This schematic illustrates the layout of the ROM's and some supportive circuitry.

The schematic for the address decode section of the board is shown in Figure 11. This circuitry is required in order that a unique device is activated for a unique address requested by the computer. The outputs of IC13 are used to activate different boards in the system, the

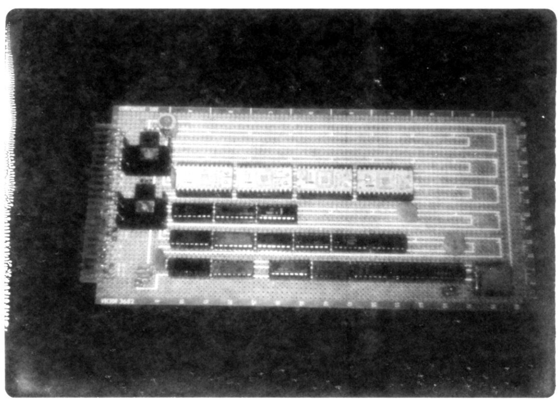


Figure 9

ROM/Address Decode/Clock Board

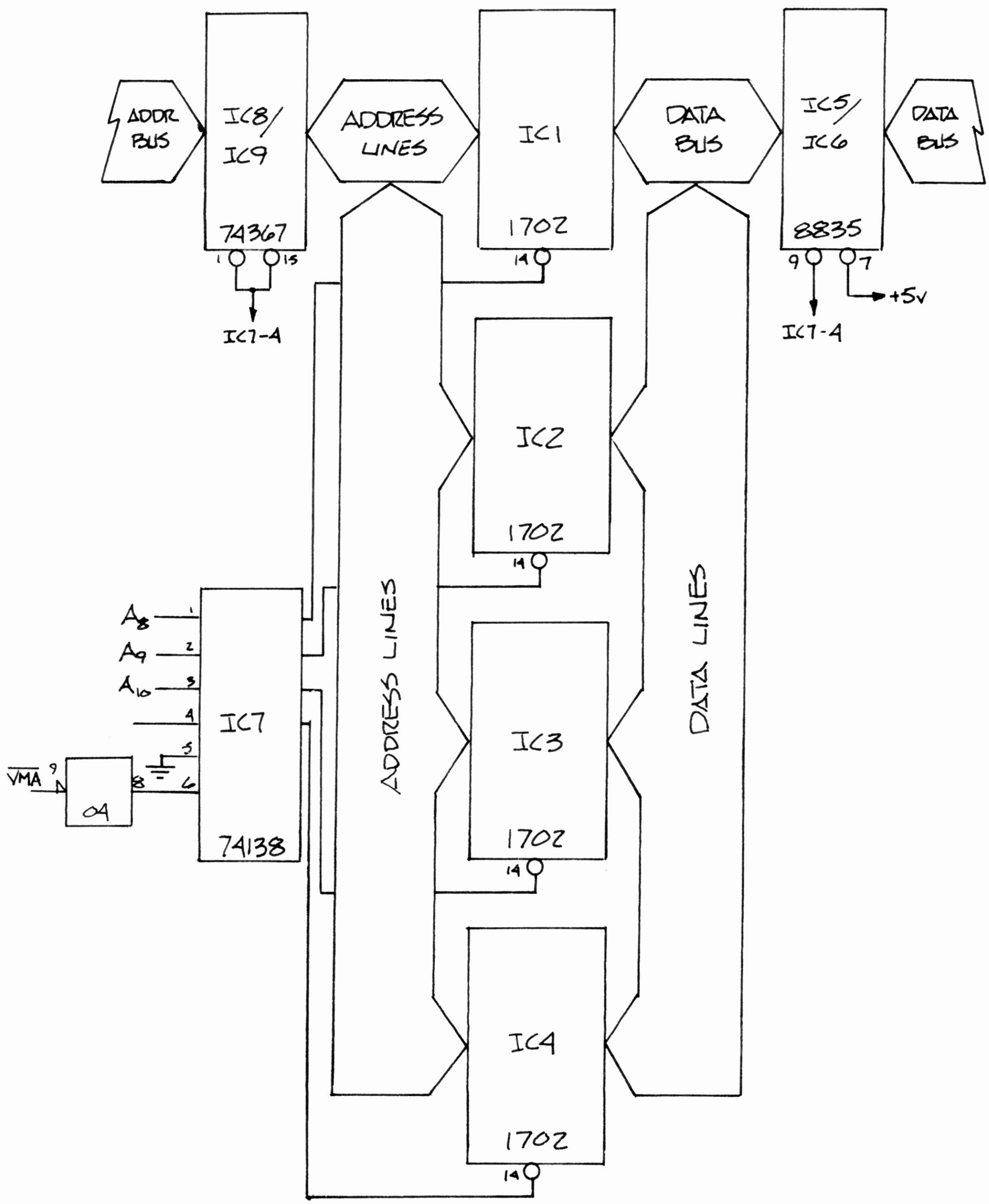


Figure 10

ROM Schematic

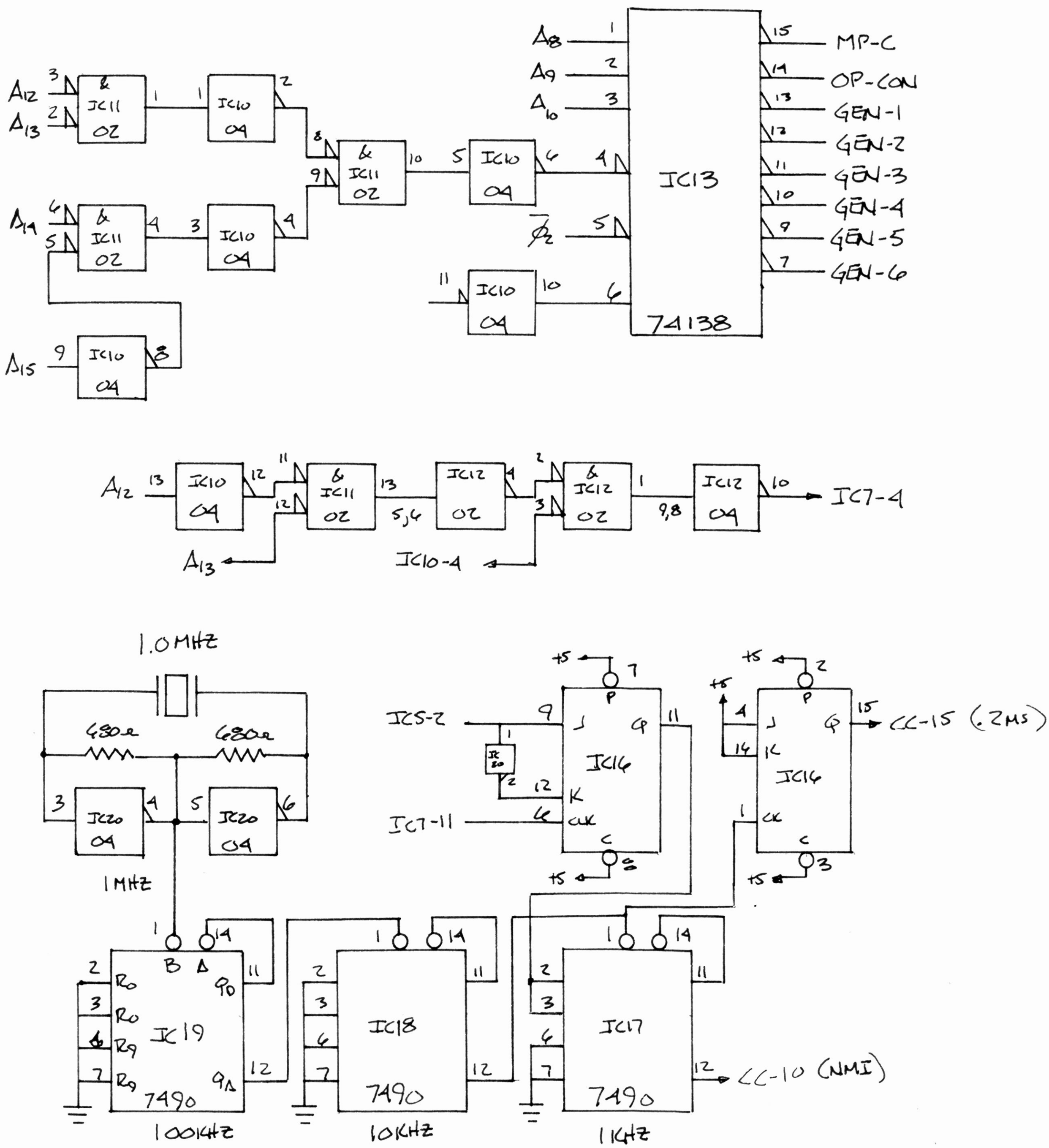


Figure 11

Address Decode Schematic

communication board, operator console board, and the pulse generator boards. A 74138 3 line to 8 line decoder is used along with supportive circuitry. It should be noted that the schematic representations being used are the new standard. The remaining section of this board contains some clock circuitry. Since a minimum pulse width of 100 μsec was requested, this was achieved by dividing down a 1 megahertz clock by 100. This 100 μsec period or 10 khz signal is obtained from IC18, pin 12. In addition, this signal is again divided by 10 and is output to the computer. This 1 khz signal is used as a periodic interrupt to the computer for timing purposes. The software associated with this will be discussed later.

Operator Console Board

This board pictured in Figure 12 contains all of the electronics necessary to interface the computer to the operator console, Figures 13 and 14. Notice the cabling that goes from the operator console board to the operator console. These cables carry all of the necessary signals from the interface to the operator console. The schematics for the operator console board are in Figures 15 and 16. Figure 15 illustrates all of the circuitry required to control the display on the operator console, along with all of the LED's (light emitting diodes). The basic prin-

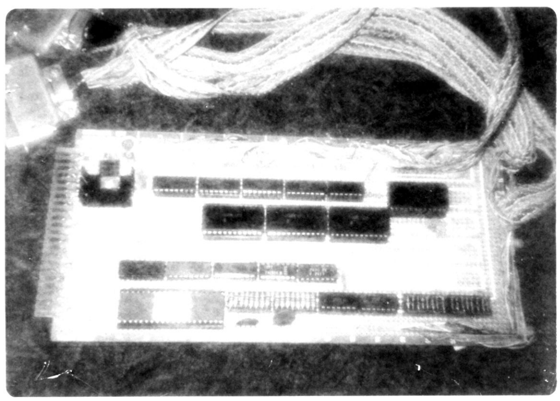


Figure 12

Operator Console Board

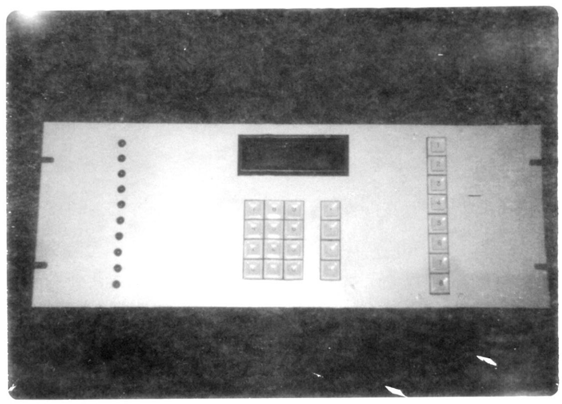


Figure 13

Operator Console - Front

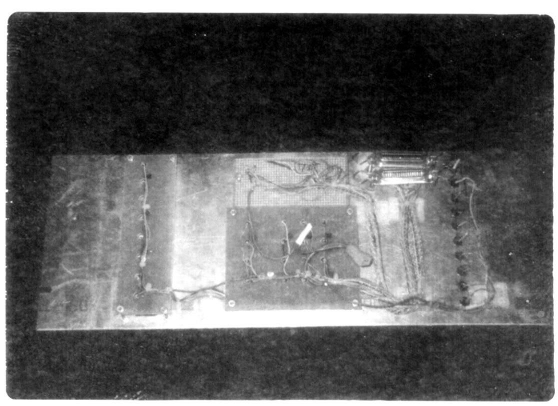


Figure 14

Operator Console - Rear

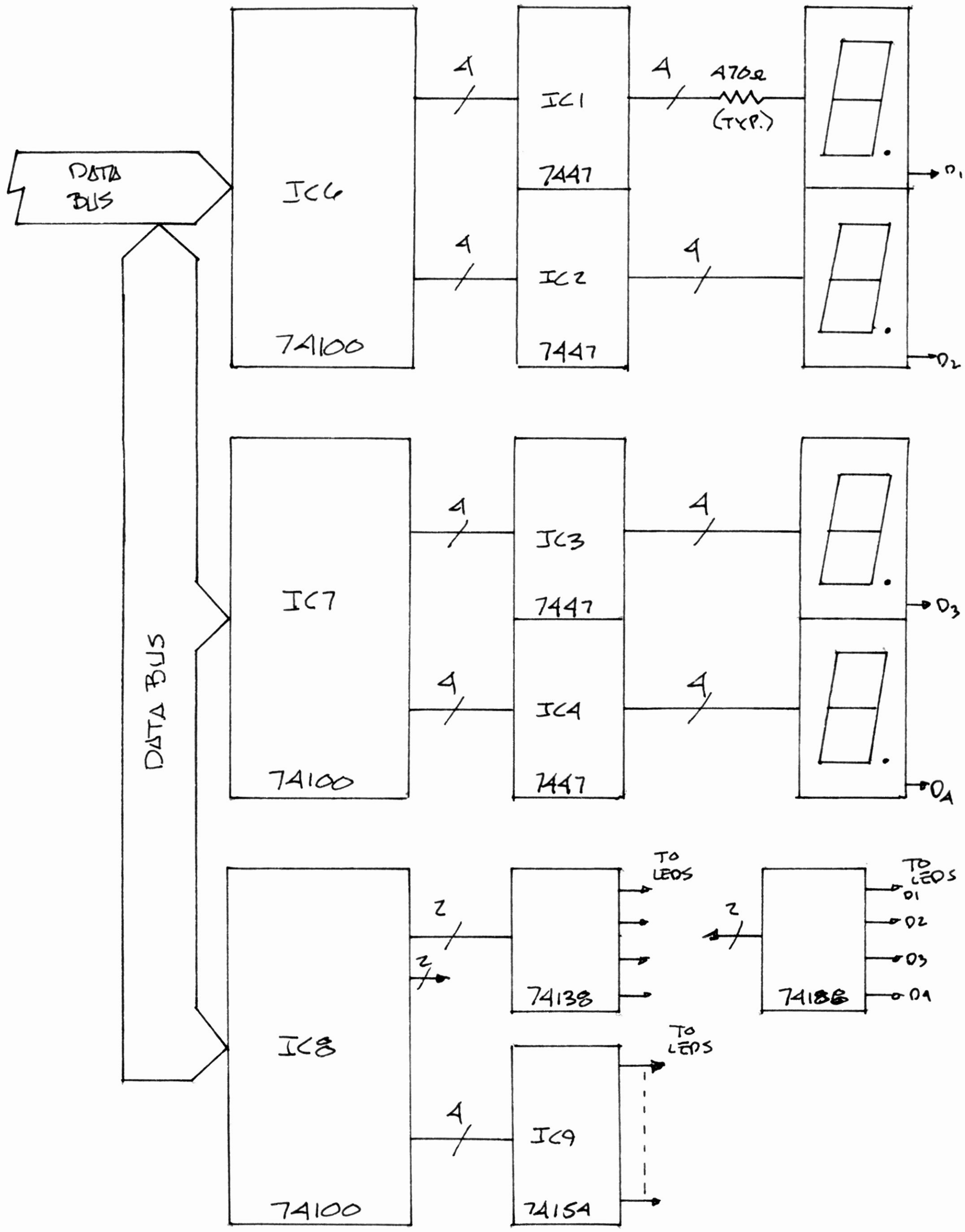


Figure 15

Operator Console Display Schematic

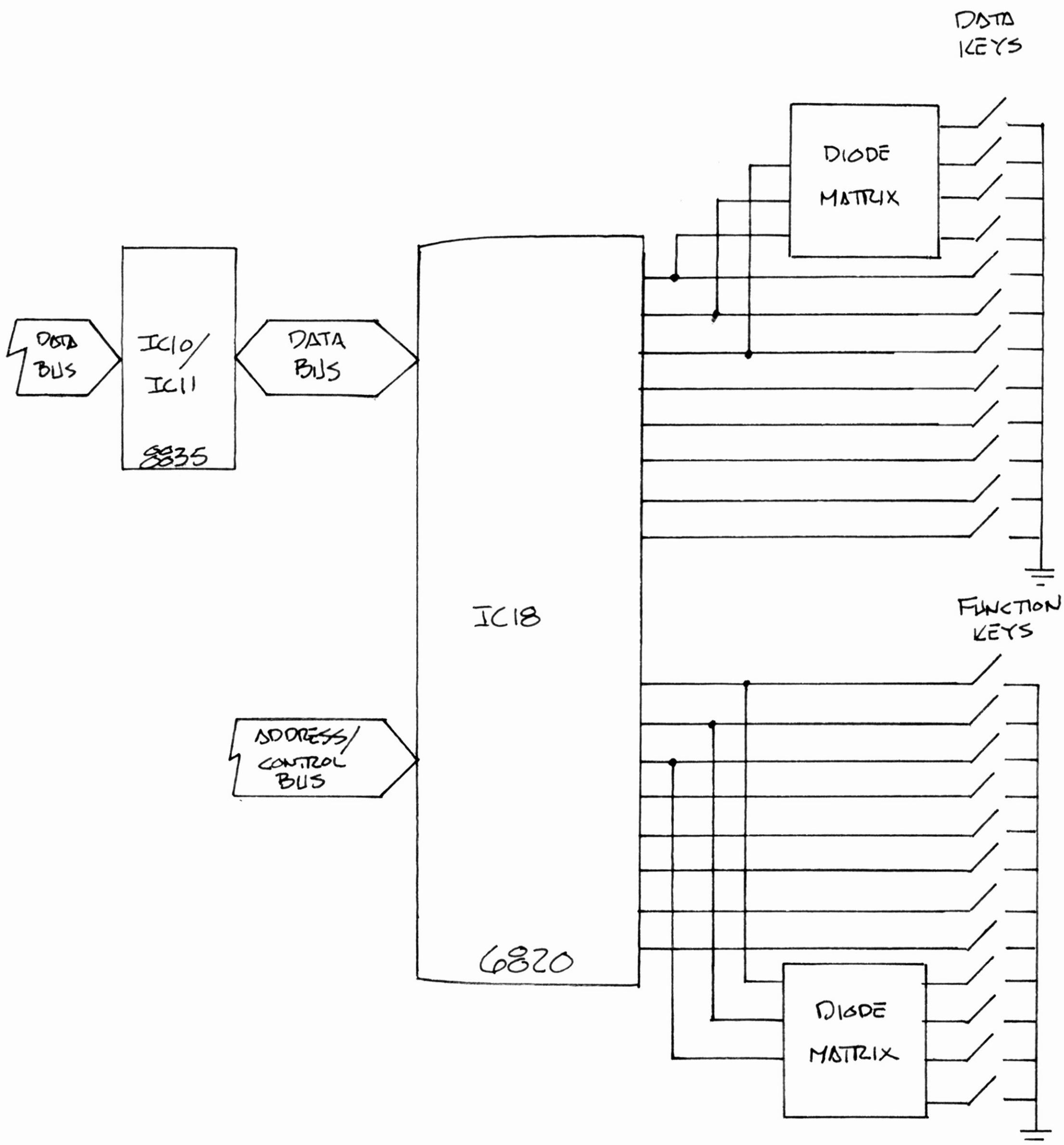


Figure 16

Operator Console Keypad Schematic

ciple is as follows. The computer selects which of the 74100's (data latches) it desires to communicate with via the 74138. The computer then outputs the appropriate information to that location. The information stored in the 74100 is processed by the 7447's which control the numerical display. In the same way, the computer controls the LED's by communicating with the appropriate 74100 whose outputs are then processed by 74138's and a 74154 to control the operator console LED's.

The computer receives information from the keypads by communicating with IC 18, which is a Motorola 6820 (see Figure 16). The 6820 is referred to as a Peripheral Interface Adapter (PIA) which is manufactured specifically for the purpose of interface design. The keypads are represented by switches on the right side of Figure 16. The keys are encoded by the diode matrix seen configured for each set of keys. When a key is depressed the computer can read the information stored in the PIA and can determine which key was depressed.

Programmable Pulse Generator Board

This board, pictured in Figures 17 and 18, generates the pulse signal. Note in Figure 18 that wire-wrap techniques were utilized to construct all of the prototype circuits. Figures 19, 20, 21 and 22 contain the schematic

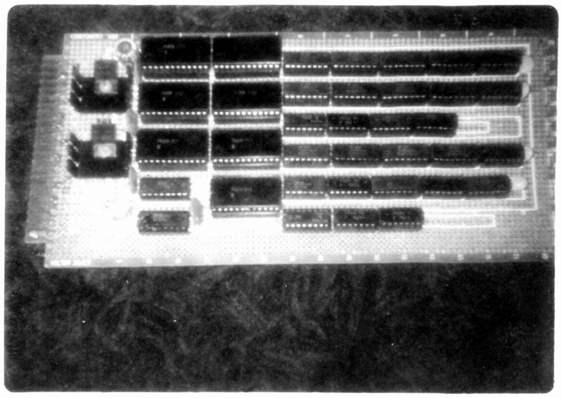


Figure 17

Programmable Pulse Generator Board - Front

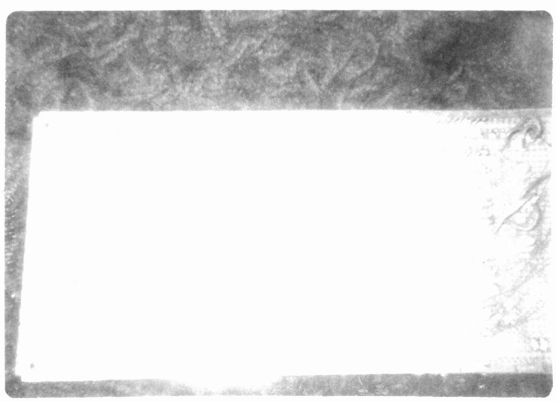


Figure 18

Programmable Pulse Generator Board - Rear

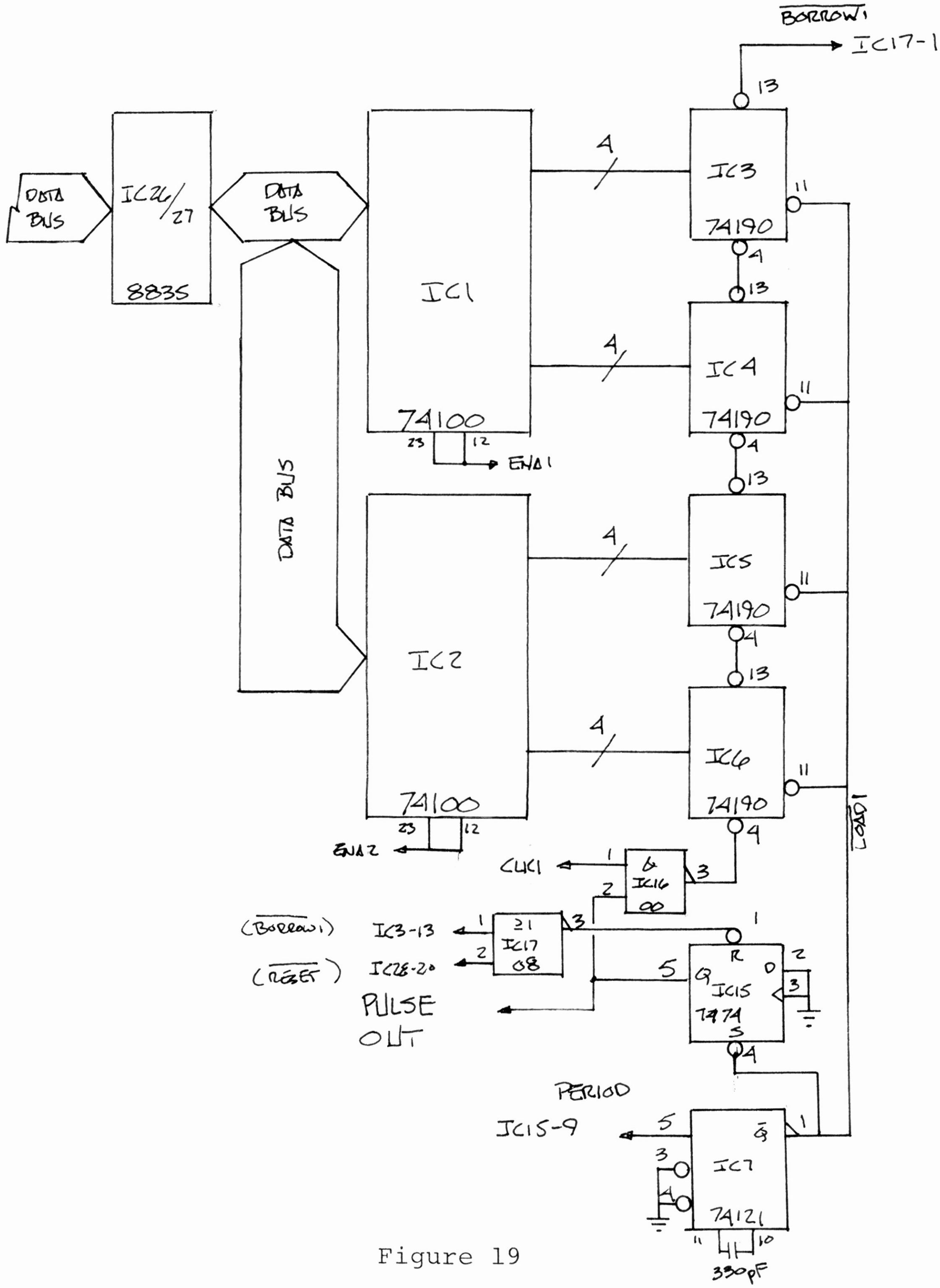


Figure 19

Pulse Width Schematic

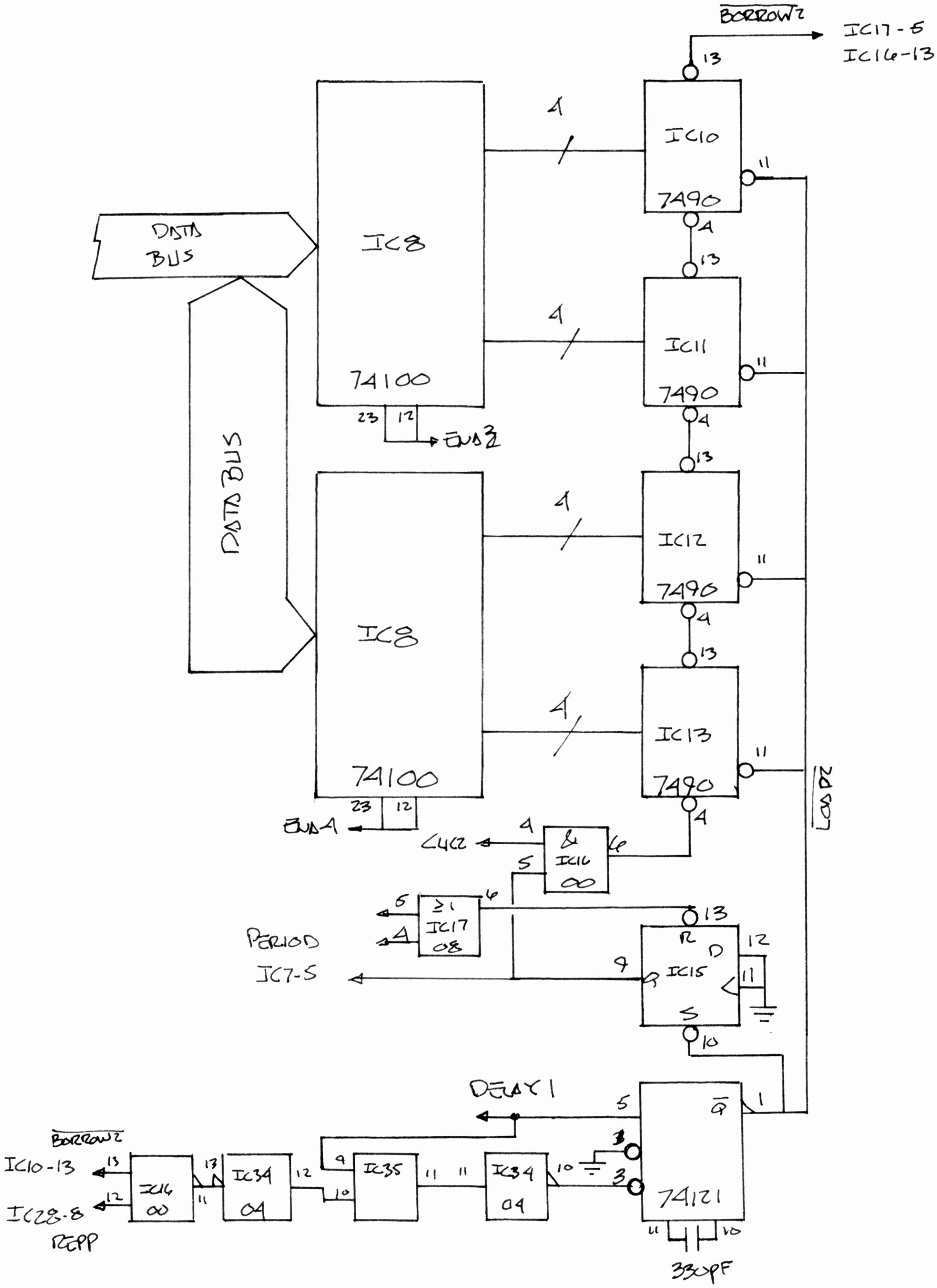


Figure 20

Pulse Period Schematic

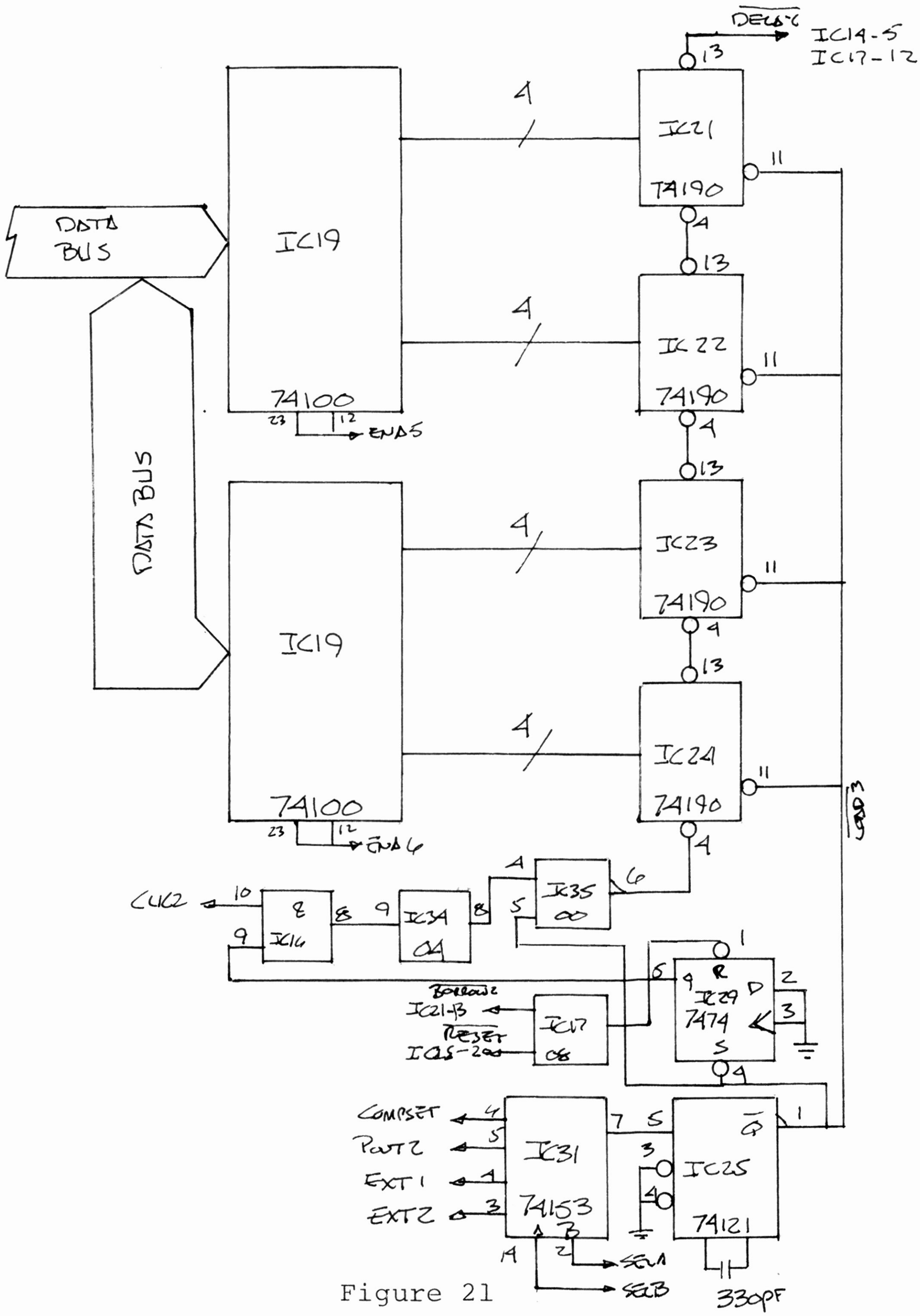


Figure 21

Pulse Delay Schematic

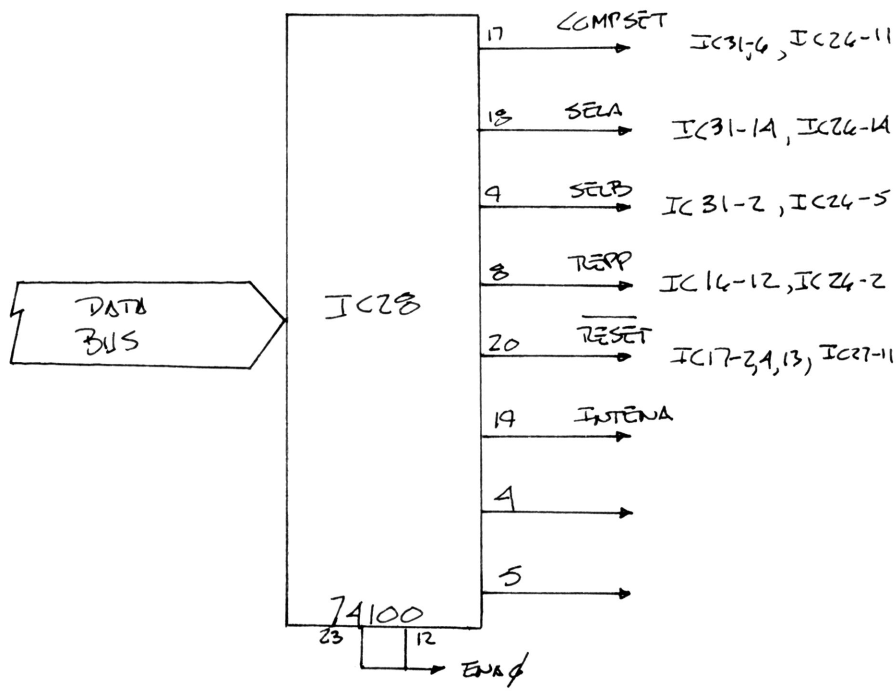


Figure 22

Pulse Generator Control Latch

diagrams for the pulse generator board. Figure 19 illustrates the circuitry used to generate the pulse width. Four binary coded decimal counters (IC 3,4,5 and 6) are utilized in the circuit. The basic principle of this portion of the circuit and the circuit in general is centered around these counters.

Basic circuit operation is more easily visualized by observing Figure 23. The counters are supplied with a clock signal of .1 ms period or 10 khz in frequency. In this example, the desired pulse width is .3 ms. In order to accomplish this, the computer first loads ØØØ3 into the 74100 data latches (IC's 1 and 2). When the computer starts the counters, they count from 3 to 1 and stop as in Figure 23. The associated control circuitry simultaneously generates a signal similar to the resulting output in Figure 23. The actual output is pulse out in the schematic diagram. It is by this method that the computer can determine the width of the pulse by merely loading the data latches with the appropriate count. Since the pulse width generator has four counters, the pulse width can vary from .1 ms to 999.9 ms = .9999 sec.

The pulse period and pulse delay portions of the circuit work in much the same manner, see Figures 20 and 21. The computer loads the desired count into the data latches and the pulse period counters then utilize this count as

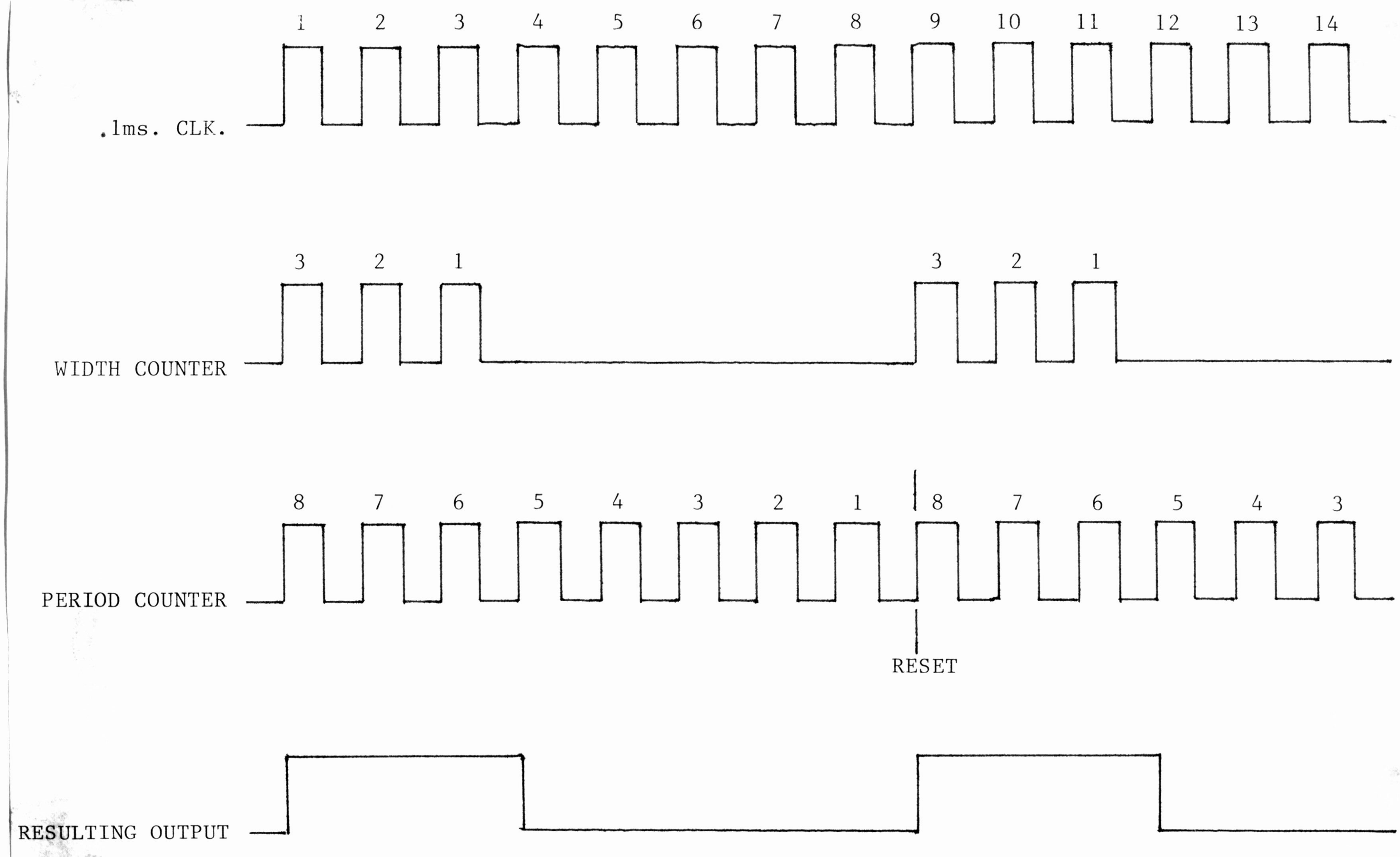


Figure 23

Basic Programmable Pulse Generator Operation a starting count. A repeating pulse signal can be obtained by cycling the pulse period counters (see Figure 23). In this manner the pulse width counter is, for this example, started every .8 ms. The pulse delay counter (see Figure 21) delays the output signal by some preset count of time by the computer. This is used in experiments where it is desired to delay activation of a stimulation from the receipt of some triggering signal.

Figure 22 illustrates the pulse generator control latch which enables the computer to start and stop the pulse generator, reset the generator, configure it for repetitive operation and control the input triggering source.

SYSTEM SOFTWARE

The stimulator's operating system or programs, reside in the read-only-memory chips (ROM's) on the ROM/address decode/clock board. When the stimulator is first turned on or "powered-up" the computer jumps to the ROM to determine what it is supposed to do. The software for the system consists of two basic packages, the operator interface software package and the control package. The basic configuration is illustrated in Figure 24. The operator interface software consists of the higher level decision routines which build the pulse train parameters and manage the operator console communications. Information processed by this package is then stored in data memory. The other package consists of the control routine which take the information processed by the operator interface software and transfers it to the programmable pulse generator boards and the operator console. The control routine also coordinates input information from the operator console keypads.

Operator Interface Software

The structure of this software package is illustrated in Figure 25. The main routine is a command sequence routine which calls several subroutines during execution.

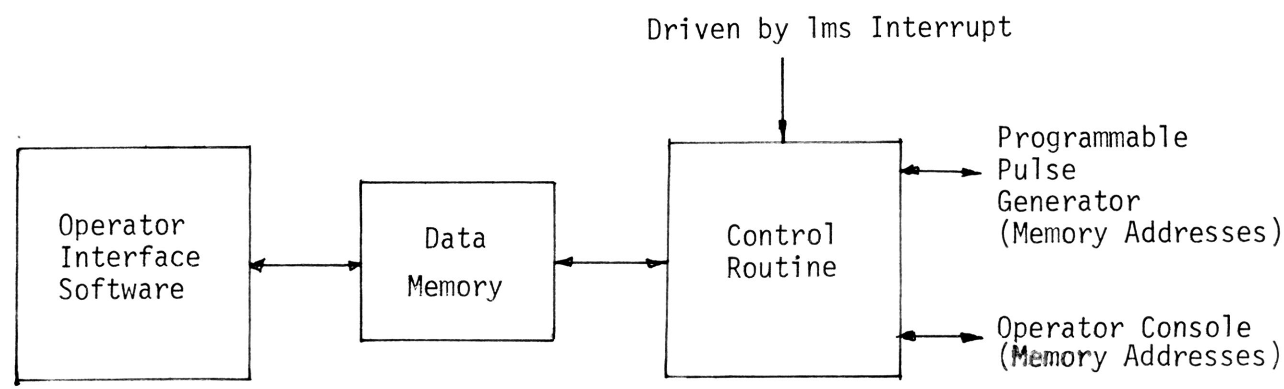


Figure 24

Basic Stimulator Software Configuration

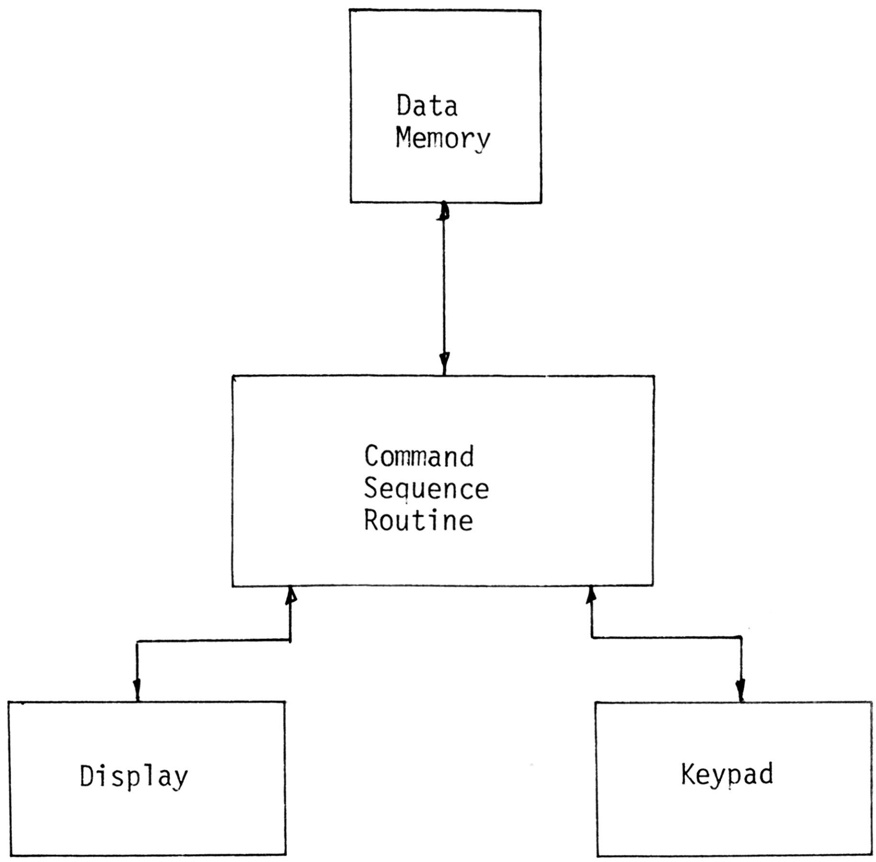


Figure 25

Operator Interface Software Diagram

The command sequence routine manages the building of the pulse train parameters by deciding which LED (light) to activate on the front panel and what data to display. Having decided what to display, the command sequence routine (CSR) calls the routine DISPLAY which then formats and stores the desired information from the operator. The CSR uses the routine KEYPAD when input from the keypad is necessary. The CSR then takes this information and determines if it is valid information. If the entry is invalid, then the CSR transmits an error condition to the operator via DISPLAY and waits for proper re-entry. All of the logic for the function keys are contained in the CSR which can build specified pulse trains for each function key. The CSR also contains the logic to communicate with the external computer via the communication board.

Control Routine

This is the time critical routine which executes every millesecond. The basic flow diagram for this routine is in Figure 26. This routine coordinates communication directly with the programmable pulse generator boards and the operator console. The first portion of the code controls the pulse generators. This routine contains the logic that counts the number of pulses in a pulse train and determines the pulse train frequency.

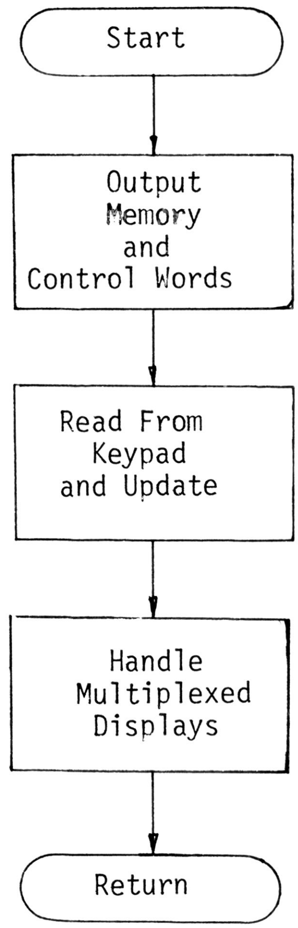


Figure 26

Control Routine Flowchart

This routine also reads the status of the operator console keypad and transfers the information to the data memory. In addition, this routine also outputs information to update the operator console display. Since this routine is executed every millesecond, its execution time is critical and has therefore been coded as efficiently as possible.

SYSTEM EVALUATION

The completed system can be seen in Figure 27. A chassis was obtained and the operator console manufactured. Card connectors were installed in the chassis and wired for operation. The computer interconnection cabling was installed. A power supply supplying +8 Vdc unregulated and -15 Vdc was constructed. Appropriate switches and fuses were installed on the back panel. All of the custom designed wire wrap boards were constructed and installed. The SWTPC 6800 computer boards were assembled and made operational. The computer functioned well in the stimulator. All of the functions of the operator console were verified as operational. The programmable pulse generator board was constructed and tested in the system. Figure 28 illustrates a basic pulse output of the stimulator. Figure 29 illustrates another pulse with different parameters. Figure 30 illustrates a basic pulse train output off the simulator in which the software is controlling the number of pulses in the pulse train and the train frequency. All of the basic modules have been verified as operational. The stimulator system software has been developed and is in the final development stages.

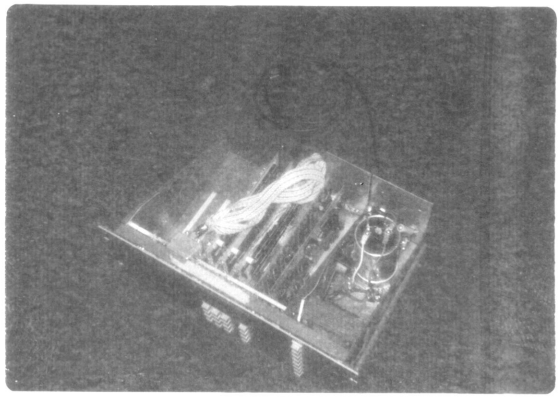


Figure 27

Completed Stimulator System in Chassis

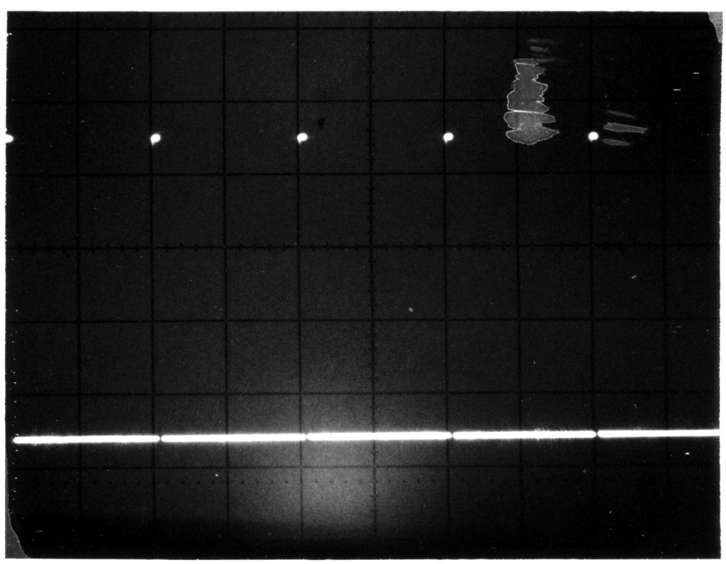


Figure 28

Pulse Signal #1

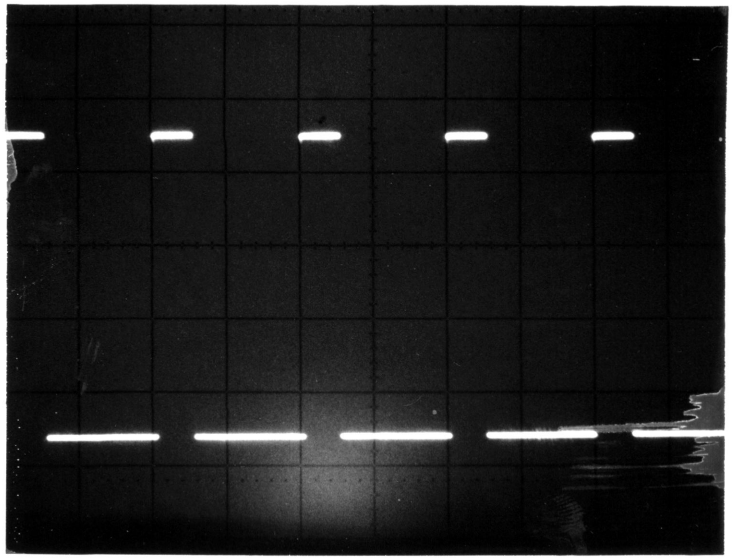


Figure 29

Pulse Signal #2

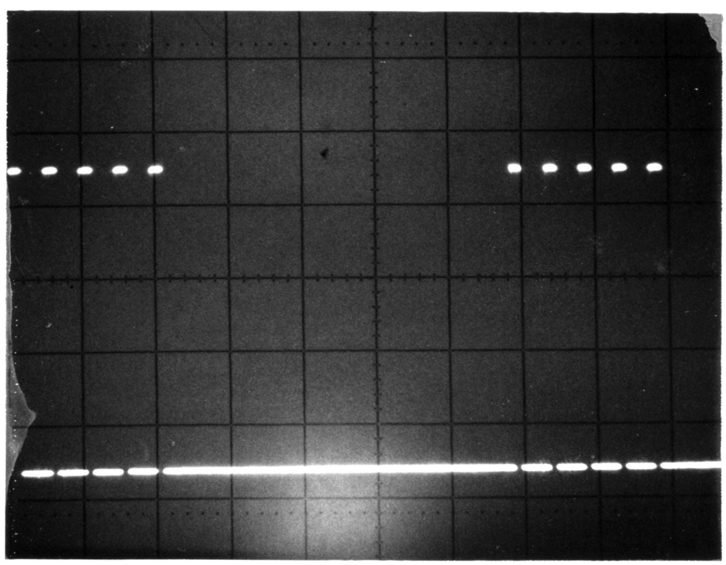


Figure 30

Pulse Train Signal

PROJECT EVALUATION

The design of the microcomputer based neurophysiological stimulator required not only technical expertise, but some management was required also. Since the author was responsible for all aspects of the project, from initial research and problem definition to completion and evaluation of the final product, a record of progress and procedure was easily maintained. Weekly progress meetings were maintained with the author's advisor, for which a weekly progress report was generated. The appendix contains a sample progress report and also contains a man-hours versus project weeks chart. Total time spent on the project as of 4/15/79 was 520 hours. This averages out to approximately 20.8 hrs/week counting only school weeks. $500 was allocated for the project of which approximately $450 was expended. It is interesting to note that the Hewlett-Packard Company has recently announced the production of a general purpose programmable pulse generator with much the same characteristics as the stimulator for approximately $16,000.

CONCLUSION

As was intended at the outset of the project, a final stimulator system has been designed, constructed, and verified as operational. The stimulator possesses all of the capabilities specified by the personnel as TIRR plus the capacity for easy expandability. In addition, the microcomputer based stimulator will enable the researchers at the Texas Institute of Research and Rehabilitation to perform interactive experiments that were previously impossible to perform with their present equipment. It is hoped that the inherent capability of the stimulator to be modified and adapted to future demand, will be fully exploited by the personnel at TIRR. Once again interdisciplinary cooperation has resulted in the creation of a new tool for research.

REFERENCES

1. T. R. Ebeling, W. M. Caldwell and T. W. McIntyre, "A Pulse Train Generator For Biological Stimulation," Medical and Biological Engineering, pp. 463-466, May 1975.

2. H. G. Gooverts, G. Koning and H. Schneider, "A Programmable Stimulator For Physiological Applications," Medical and Biological Engineering, pp. 112-118, January 1975.

APPENDIX

Progress for week of 2/6/78:

1. Made CPU Operational.

   a. Required Modification of some address decoding for the MP-C board.

   b. Tested all basic functions using MIKBUG. Memory also functions well.

   c. Modified data buffer enables.

2. Worked on operator console board.

   a. Ran basic test of board.

   b. PIA on board functions, however there are some addressing and enable problems with the 74100's that are causing trouble.

   c. Modified keypad entry design. Changed configuration for CMOS debouncing circuitry. Required some rewiring and addition of 16 100 k resistors.

   d. Added two 74367's for buffering of data lines to 74100's. The 6820 on board was experiencing some loading difficulties.

Projected Progress for Week of 2/13/78:

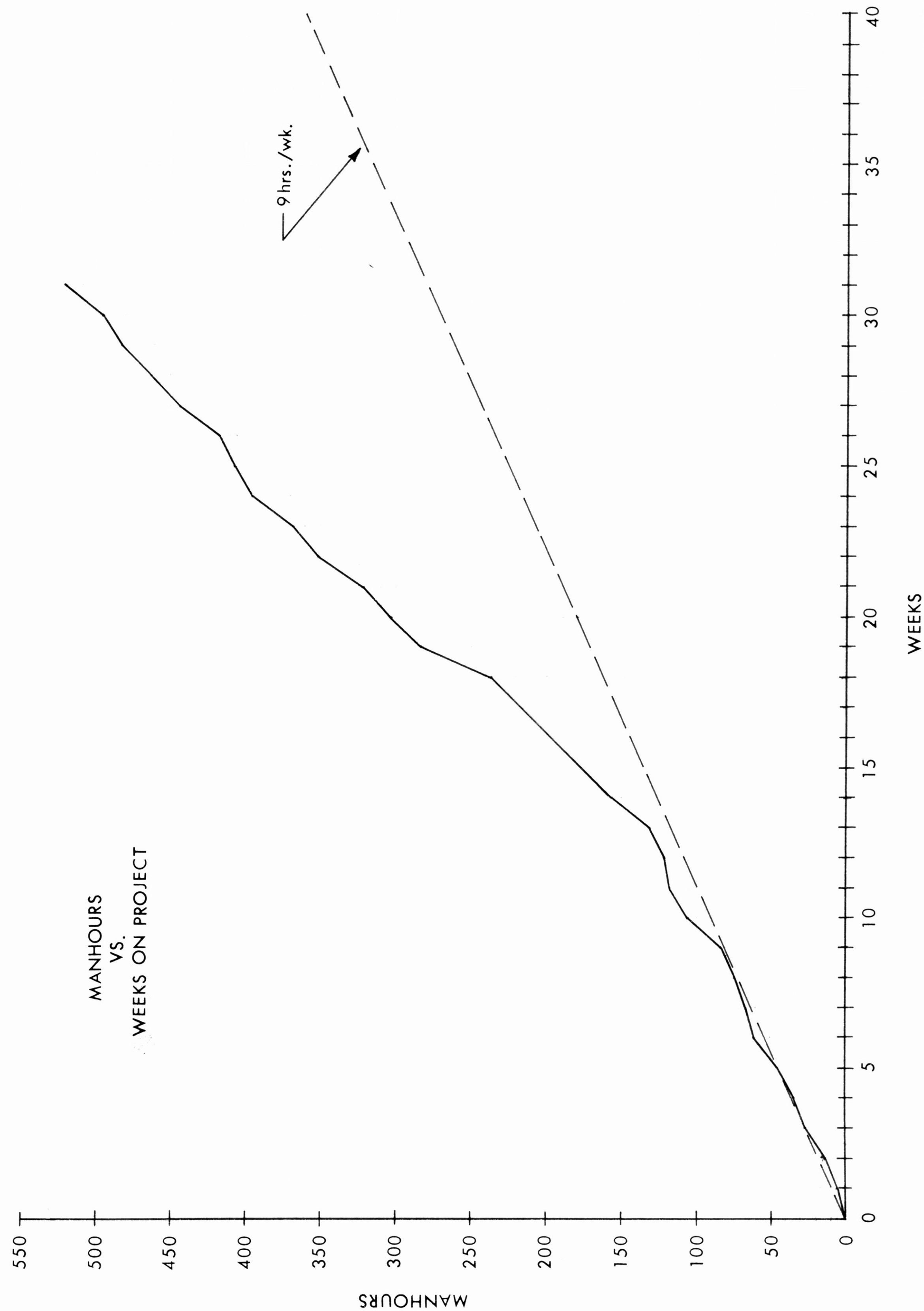
1. Make front panel electronics operational, i.e., operator console board, etc.

2. Finalize and order remaining components for pulse generator board.

Project Timesheet

Week of: 2/6

Task	Sun	Mon	Tues	Wed	Thur	Fri	Sat	Week Tot.	Task Tot.
Testing/Modification	6	10	4	5	4			29	37
Reporting					1			1	16.5
Advisor Meeting						1		1	19.5
Totals								31	352


MANHOURS
VS.
WEEKS ON PROJECT
9 hrs./wk.
MANHOURS
0
50
100
150
200
250
300
350
400
450
500
550
WEEKS
0
5
10
15
20
25
30
35
40