THE UNIVERSITY OF NEW BRUNSWICK

SCHOOL OF COMPUTER SCIENCE

CS/SC3 PROJECT REPORT

THE CONSTRUCTION OF A PRIMATIVE COMPUTER

APRIL 7, 1978

STEPHEN R. KYLE

GARY G. MORRISON

TABLE OF CONTENTS

CHAPTER ONE

The physical characteristics of the PRIMATIVE COMPUTER are outlined in table 1.1.

TABLE 1.1

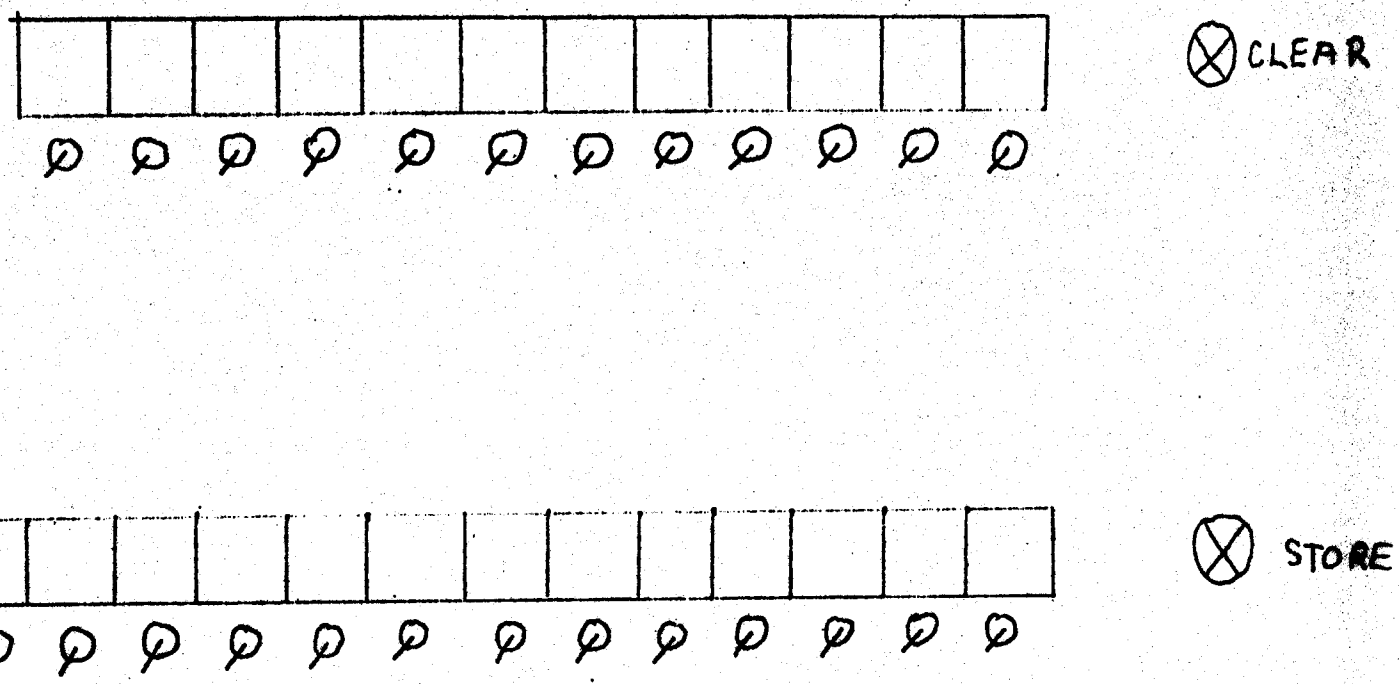
Size of byte	16 bits
Size of word	1 byte (16 bits)
Size of memory	4096 bytes
Maximum number of instructions	16
Instruction format	Fixed length, single address
Data path width	16 bits
Number system	Binary (2's complement arithmetic)
Number of data registers	one accumulator

The layout of the front panel is shown in figure 1.1.

The row of lights along the top of the panel display the memory address at all times. In the load mode, the row of switches directly beneath select the address to be accessed, while in the run mode these switches are inactive.

The push button to the right of these switches clears the

FIGURE 1.1



program counter to zero, thus causing execution to begin at location 000 in memory.

At all times the next row of lights display the contents of memory at the location pointed to by the address on the first row of lights. In the load mode the switches below these lights select the data to be written at the address specified. These switches also remain inactive in the run mode.

The push button on the right of these switches causes the data on the switches to be stored at the location shown on the top row of lights.

The switch on the bottom left hand corner is used for power and the light above it indicates the power is on. Since the power supply has not been built yet it merely turns the display on or off.

The start button is located next to the power switch. When pressed the computer will begin executing the program at location 000. The light above the start switch was intended for displaying the computer status (run state or halt state). This light is currently inactive as external circuitry would be required on the clock board which has yet to be constructed.

The leftmost black handled switch was intended to be used in the controlling of the clock. It would place the clock in step mode or in run mode. When the clock has been built this switch could be connected to it by the wires ties into the front panel wiring.

There are two modes in this computer, load and run. These modes are controlled by the right most black handled switch. When it is in the down position the computer is in run mode ready to execute a program. If the switch is in either the center or top position the computer is in the load mode. When it is in the load mode programs may be loaded or modified.

The light on the bottom right hand corner of the panel displays the status of the accumulator on the fourth clock phase. When in one of the other three phases it indicates whether the memory buffer equals zero. When lit there is something other than zero in the accumulator or memory buffer. The light directly to the left of this indicates an accumulator overflow when lit.

EXTERNAL CONNECTIONS

In order to use the primative computer the following external connections must be made:

1. Connect the four terminals marked P1, P2, P3, and P4 to the output of the the four phase clock. (See Appendix A.)

2. Connect the leads marked ACD and ACC' to the external flip flop as shown in Appendix B.

3. Connect the lead marked P4D to the delay portion of phase four on the clock.

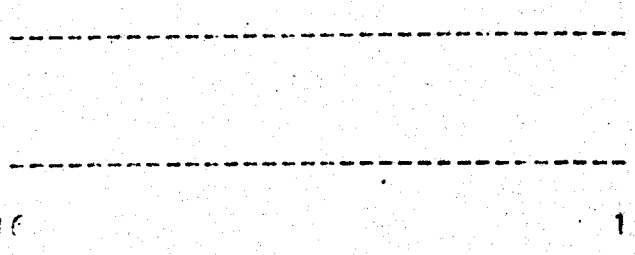
4. Connect the forty pin connectors to the memory boards. The arrow on the connectors indicates pin forty, while the memory boards have pins one and two marked.

5. Connect the battery and ground to B and G respectively on the terminal strip. The battery is five volts.

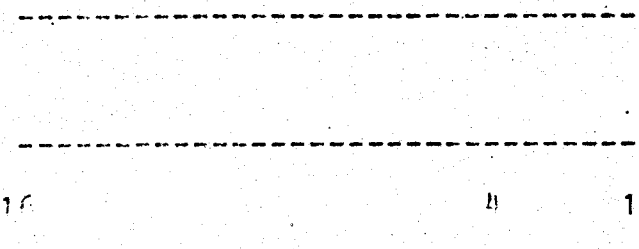
Turn the power on and the display should be lit. You should put the computer in the load mode and store a halt instruction at location zero. Then reset the program counter. When switched back to the run mode the computer should go to the halt state.

LOADING A PROGRAM

There are two storage formats used in this computer. The data format:



In this format the data is stored with the low order bit in location one and the high order bit in location sixteen. The instruction format is of the form:



In this format the opcode is stored in bit locations one to four and the address is stored in bit locations five to sixteen.

To run the program place the computer in the run mode and press the start button. (NOTE: the start button must be held in until the first execution of phase two has been completed.)

It was found helpful to store zero at location 000 and start the program at location 001. This would help to identify the halt state.

At present there have been only seven instructions implemented.

They are found in table 1.2

TABLE 1.2

MNEMONIC	CODE	ACTION
LDA	0001	Load Accumulator from the address specified in the address field.
STA	0010	Store Accumulator in the address specified in the address field.
ADD	0100	Add the contents of the memory location specified to the accumulator.
SUB*	0101	Subtract the contents of the memory location specified from the accumulator.
JMP	0110	Jump to the instruction at the location given in the address field.
BNE	0111	Branch to the instruction at the location given in the address field if the contents of the accumulator is not equal to zero.
HLT	1000	Halt the execution of instructions and reset the program counter to zero.

The other instructions are mapped to NOP.

*NOTE: the SUB instruction actualy takes the 2's compliment of the contents of the accumulator. We believe that this is due to a change in the specifications after our documentation was released.

CHAPTER TWO

This chapter is broken down into four sections. The first three describe the individual boards that make up the computer while the fourth gives a general overview of the whole computer.

2.1 PROGRAM COUNTER AND MEMORY ADDRESS BOARD.

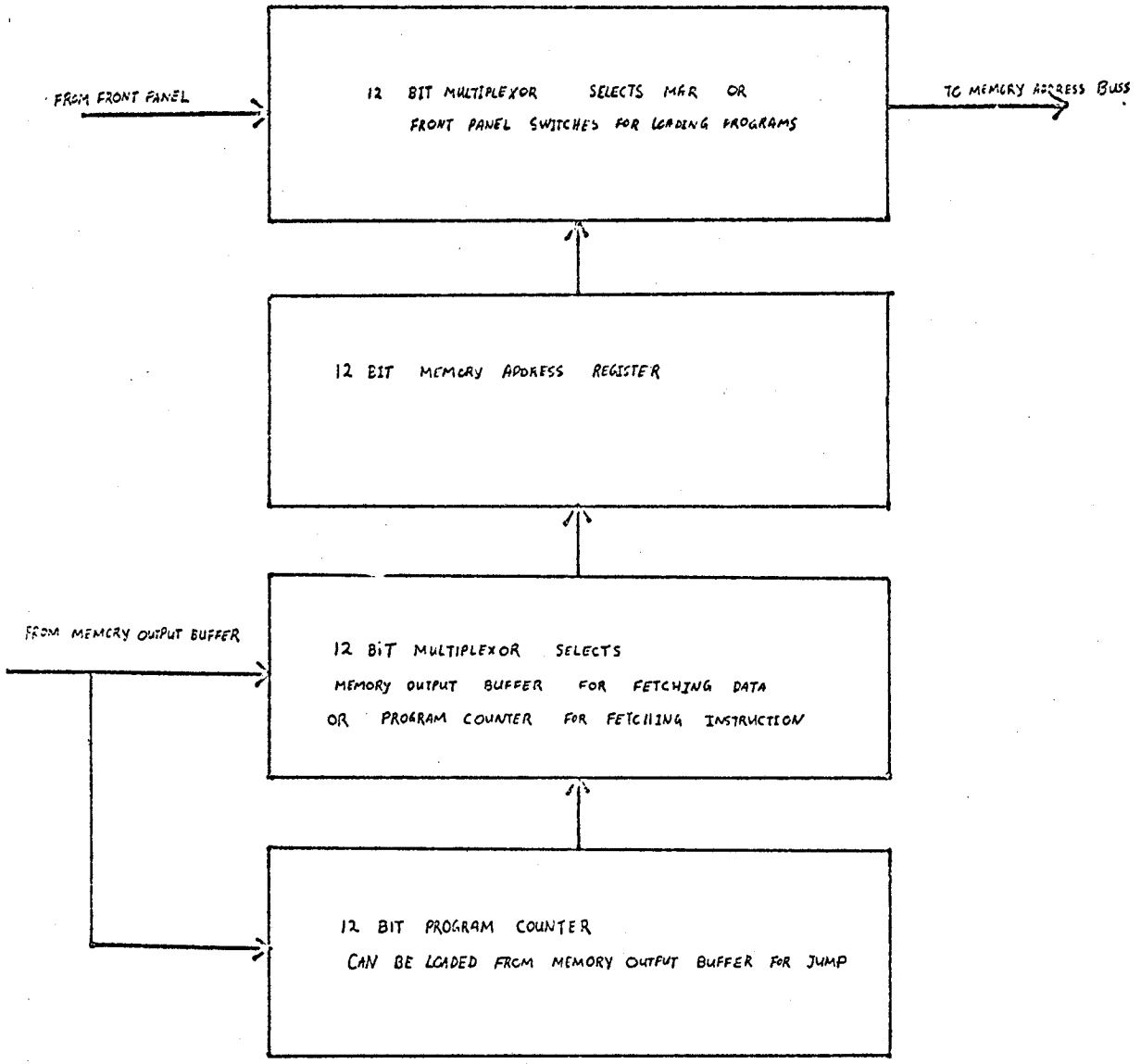
Refer to figure 2.1 for the relationship between the components that are to be described here.

The first block is composed of three four bit multiplexers which select the memory address from the memory address register during the the run mode and the front panel switches during the load mode.

Next are three four bit latches which are use to store the address of the next instruction or the address of the data to be used in this instruction. These latches comprise the memory address register.

Following the memory address register is three four bit multiplexers which are use to select the address from the program counter while in the fetch sequence and from the memory buffer during the execution sequence.

FIGURE 2.1



PROGRAM COUNTER AND MEMORY ADDRESS BOARD

The program counter is made up of three four bit binary counters which may be cleared or loaded with a value. Normal execution causes these counters count on phase one. When a branch instruction is executed they are loaded from the memory buffer.

The chip layout and the wiring diagram for this board are found in Appendix C.

2.2 ACCUMULATOR AND MEMORY OUTPUT BUFFER BOARD.

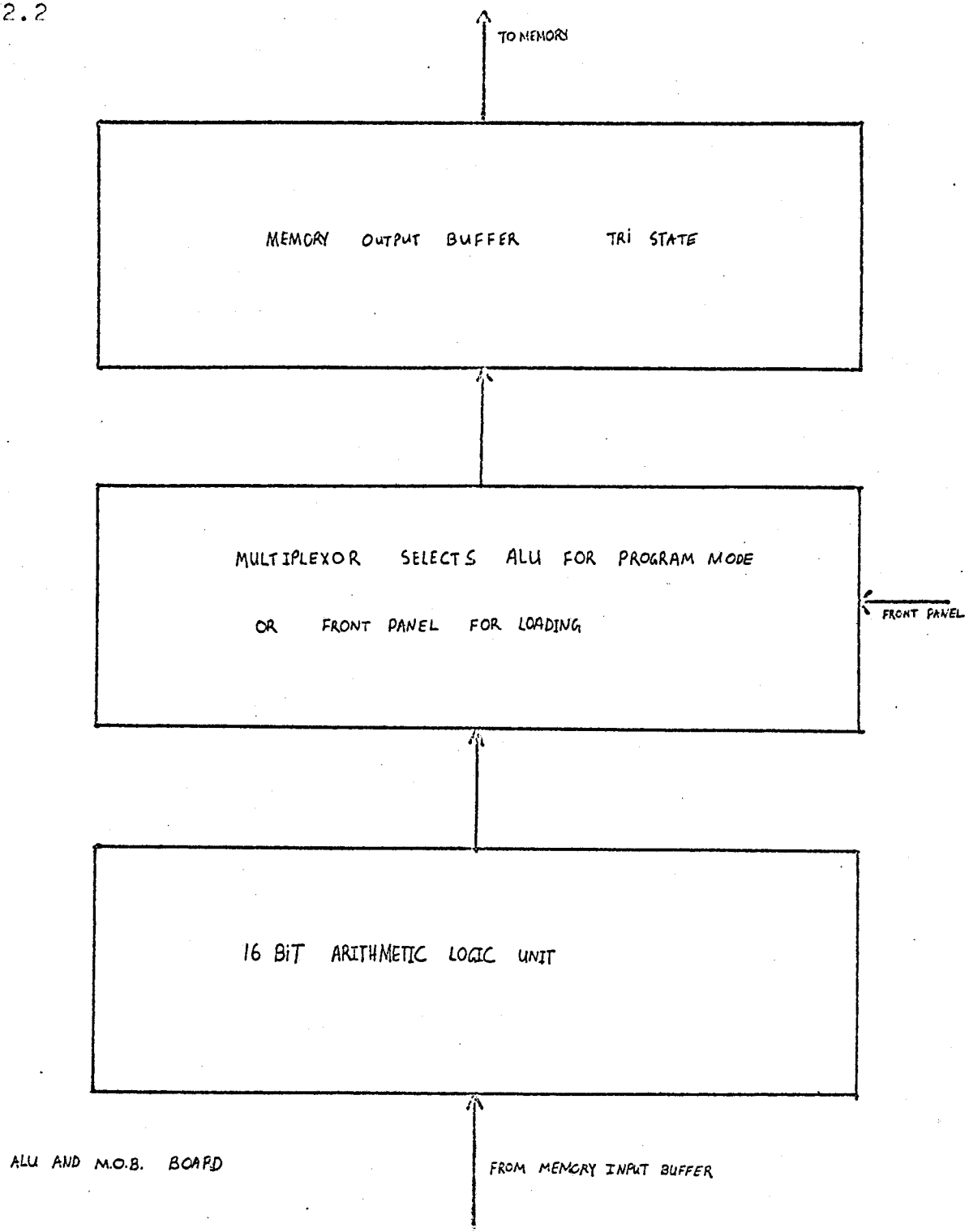
Refer to figure 2.2 for the relationship between the components that are to be described here.

Tri-state buffers form the sixteen bit memory output buffer. The input of this buffer is selected by the four bit multiplexers from the front panel switches (load mode) and from the accumulator (run mode).

Four cascaded four bit ALU function generators form a sixteen bit accumulator with the aid of a look ahead carry generator. This accumulator is used to perform the arithmetic computations requested by the program that may be running.

The chip layout and the wiring diagram for this board are found in Appendix C.

FIGURE 2.2


TO MEMORY
MEMORY OUTPUT BUFFER TRI STATE
MULTIPLEXOR SELECTS ALU FOR PROGRAM MODE
OR FRONT PANEL FOR LOADING
FRONT PANEL
16 BIT ARITHMETIC LOGIC UNIT
ALU AND M.O.B. BOARD
FROM MEMORY INPUT BUFFER

2.3 CONTROL NETWORK BOARD.

Refer to figure 2.3 for the relationship between the components that are to be described here.

The control network board contains the memory input buffer, opcode register, led drivers, and the decoder along with the necessary logic required to control the differnt sections of the computer. The memory input buffer is made up of four bit latches whose input comes from the memory. The opcode register is a four bit latch which gets its input from the low order four bits of the memory input register. The led drivers are hex inverters used to drive the leds.

The boolean expressions representing the control signals and their destinations are summarized in table 2.1 .

The chip layout and the wiring diagram for this board are found in Appendix C.

2.4 OVERVIEW OF THE COMPUTER AS A WHOLE.

Refer to the block diagram in figure 2.4 while reading this section.

FIGURE 2.3

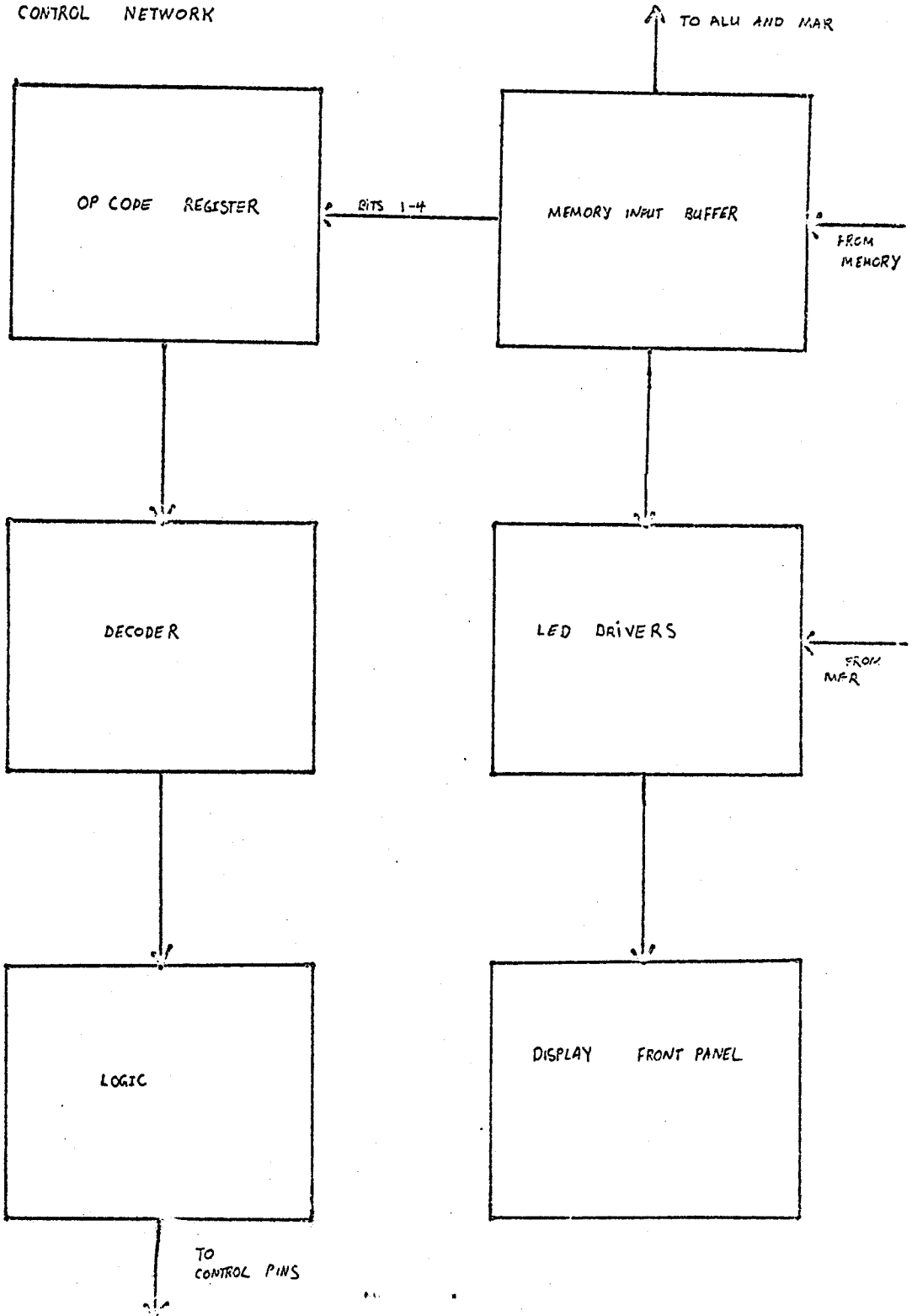

CONTROL NETWORK
TO ALU AND MAR
OP CODE REGISTER
BITS 1-4
MEMORY INPUT BUFFER
FROM MEMORY
DECODER
LED DRIVERS
FROM MFR
LOGIC
DISPLAY FRONT PANEL
TO CONTROL PINS

TABLE 2.1

DESTINATION	LOGIC EXPRESSION
CLEAR PC	(HLT' ∧ CLEAR)'
LOAD OPCODE	P2∧(HLT∧STORE)'
LOAD MIB	P1∨P4
LOAD MAR	P2∨P4
LOAD PC	(((Q'D ∨BNE')∧JMP')'∧P2)'
*RW	(STORE'∧LOAD) ∨(STA'∧P3)
*M	*RW'
ALU	SEE TABLE IN APPENDIX C

FIGURE 2.4

P1: INCREMENT PC
FETCH INSTRUCTION

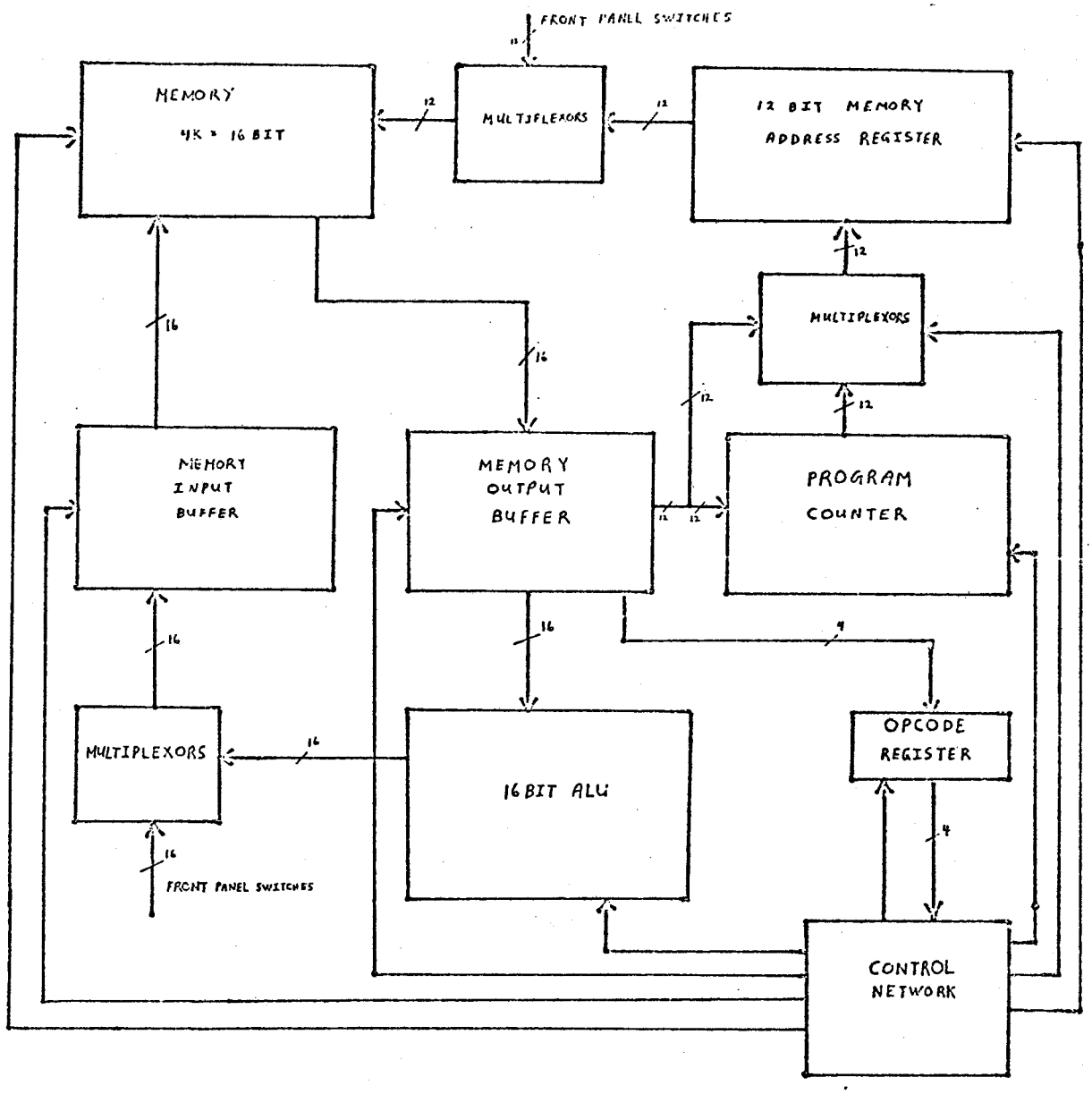
P2: LOAD MAR (DATA)
LOAD OCR
LOAD PC (BRANCH)

P3: FETCH DATA
STORE DATA

P4: EXECUTE INSTRUCTION (ALU)
LOAD MAR (INSTRUCTION)

INSTRUCTION SET

LDA	0001
STA	0010
ADD	0100
SUB	0101
JMP	0110
BNE	0111
HLT	1000
NOP	1111



A four phase clock is used to control the operation of this computer.

PHASE ONE:

The program counter is incremented and the memory buffer is loaded from the contents of memory pointed to by the memory address register. This loads the instruction while setting the program counter to point to the next instruction.

PHASE TWO:

The memory address register is loaded from bits five to sixteen of the memory buffer with the address of the data to be used in this instruction. The opcode register is loaded from bits one to four of of the memory buffer which contain the opcode of the instruction to be executed. This opcode is then decoded by the four to sixteen decoder and the appropiate control signals are set up to execute the instruction. If the instruction is a JMP or a BNE with the contents of the accumulator not equal to zero the program counter is loaded with the address contained in bits five to sixteen of the memory buffer.

PHASE THREE:

The data in the accumulator is stored in the location referenced by the address register or the data is retrieved from memory depending on the instruction that is being executed.

PHASE FOUR:

The accumulator operations are performed and the state of the accumulator is stored in the external flip flop. The memory address register is loaded from the program counter with the address of the next instruction.

CHAPTER THREE

This chapter is divided into two sections, changes made in the design and problems encountered in all stages of this project.

CHANGES:

A two phase clock with separate fetch and execute cycles was originally proposed. However we found it was much easier to impliment a four phase clock with hard wired fetch and execution sequences. Thus the first two phases of the clock cause the fetch sequence while the last two phases perform the execution cycle. This required less circuitry as there would no longer be a need to force a fetch instruction. The hardware required for control was much simpler with a four phase clock as the phases of the clock could be used to control some of the sections directly.

Orginally sixteen instructions were to be implemented however due to lack of time, seven were actualy completed. A list of these can be found in chapter one. The remaining nine instructions were mapped to NOPS so as to simplify the logic to control the accumulator.

Wire wrap was chosen over pre-printed boards as the boards available could hold more circuitery than the pre-printed ones. Also, wire wrap lends itself to easy modification. We found this was an asset as the original wiring had to be modified due to the unavailabilty

of parts.

The front panel that was used is a modified version of that described in the original proposal. This was done to simplify the loading and inspection of instructions. The outline presented in LeRoy Johnson's book, DIGITAL COMPUTER ORGINIZATION, called for the data lights to display the contents of the accumulator rather than the data bus. It was felt that displaying the data bus would make loading programs and displaying memory simpler. The need for the LDA and SET buttons was eliminated and a simple STA button was found to be sufficient.

The wait indicator light was deemed unnecessary as it was the opposite of the compute light. The stop switch was not installed but may be at a later date if so desired.

The error light was to indicate that the program counter had overflowed. Since the counters automatically reset to zero and the speed of the clock was too great in free running mode to cause noticable indication, we deemed this light unnecessary. We decided to use it to display the status of the accumulator on phase four as a debugging aid.

We could not obtain a proper selector switch making it necessary to use two separate toggle switches to control the mode of the computer. The first switch controls the clock (see chapter one), while the second

controls the mode of the computer.

PROBLEMS:

The parts needed for construction of the computer were ordered in early November, however, some of them did not arrive until mid March. We knew we would be pressed for time if the project was to be completed. We had to order subsitutes for some of the parts as the ones we had ordered were not available. This resulted in the rewiring of most of the boards.

There were two main problems in debugging the computer after it had been constructed. We found that there was not enough of a delay between each clock phase to prevent the overlap of control signals. The delay between phase four and phase one of the clock was used to control the address multiplexers.

Insufficent documentation for the accumulator led to much wasted time in designing the external logic to control its operation. As the output of the accumulator was only available on the fourth phase of the clock an external D-type flip flop was needed to retain the status of the accumulator until needed.

SUGGESTIONS FOR FUTURE PROJECTS:

It is hoped that the following suggestions will be of value for future projects.

1) Construction of the memory boards.

2) Construction of a power supply.

3) Construction of a four phase clock.

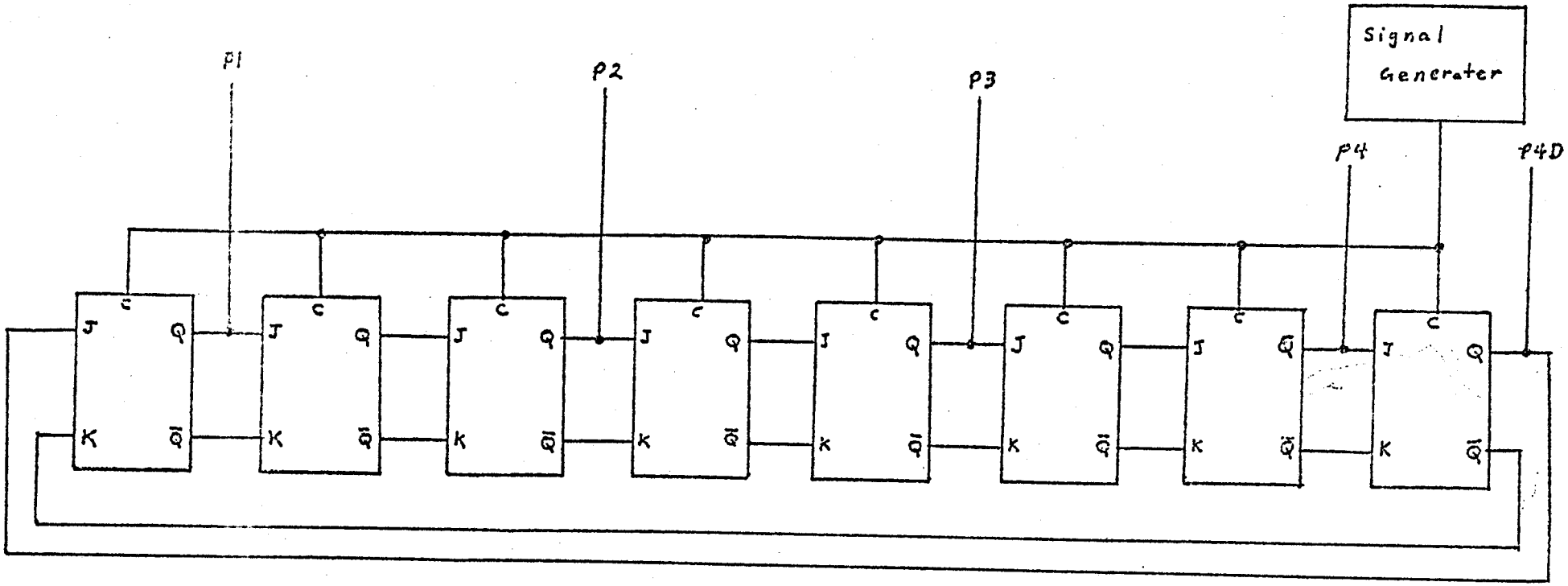
4) Implimentation of the remaining instructions.

5) An input output package.

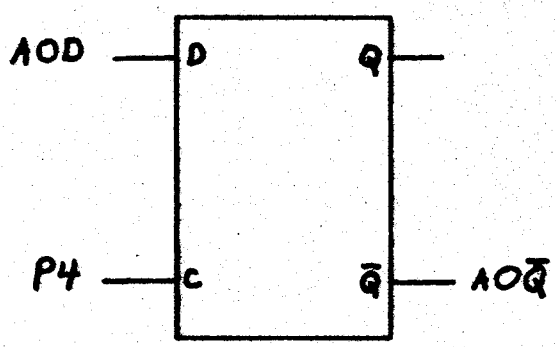
CONCLUSION:

We found this project very interesting and educational. However, it was much too large an undertaking for even two students.

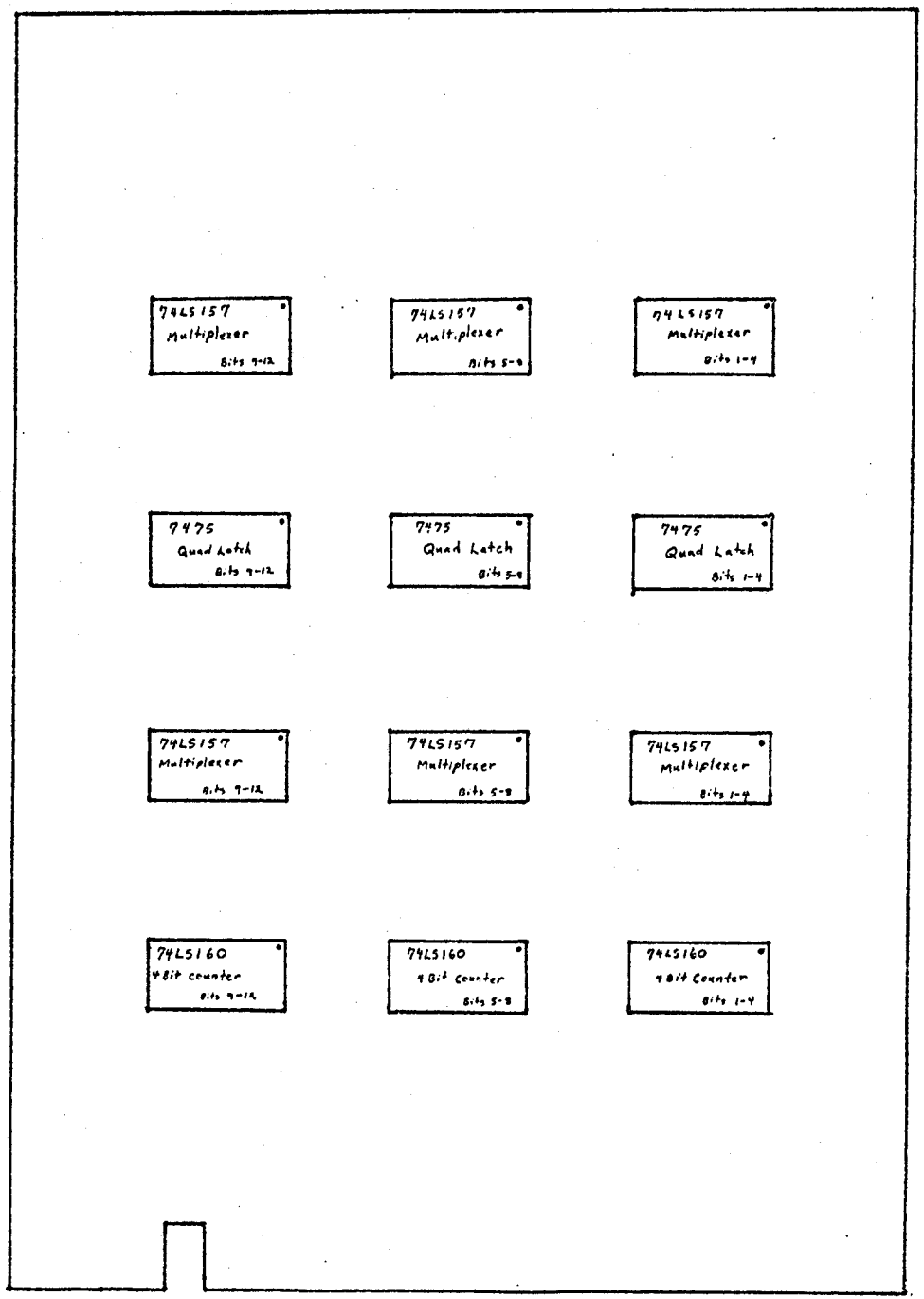
APPENDIX A


Signal Generater
P1
P2
P3
P4
P4D
J
K
C
Q
$\bar{Q}$

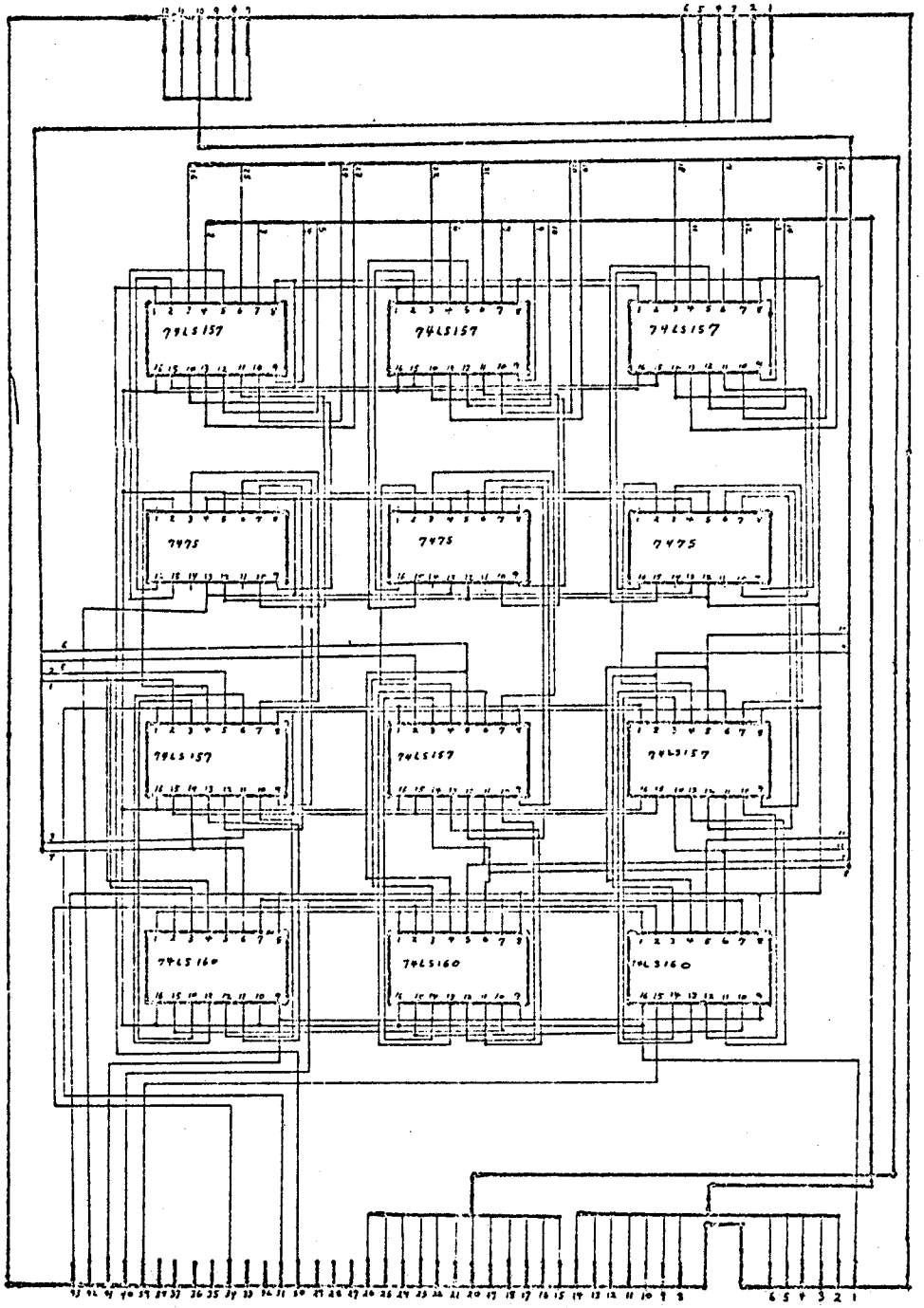
APPENDIX B


10D
D
Q
P4
C
$\bar{Q}$
10$\bar{Q}$

APPENDIX C



PROGRAM COUNTER AND MEMORY ADDRESS BOARD

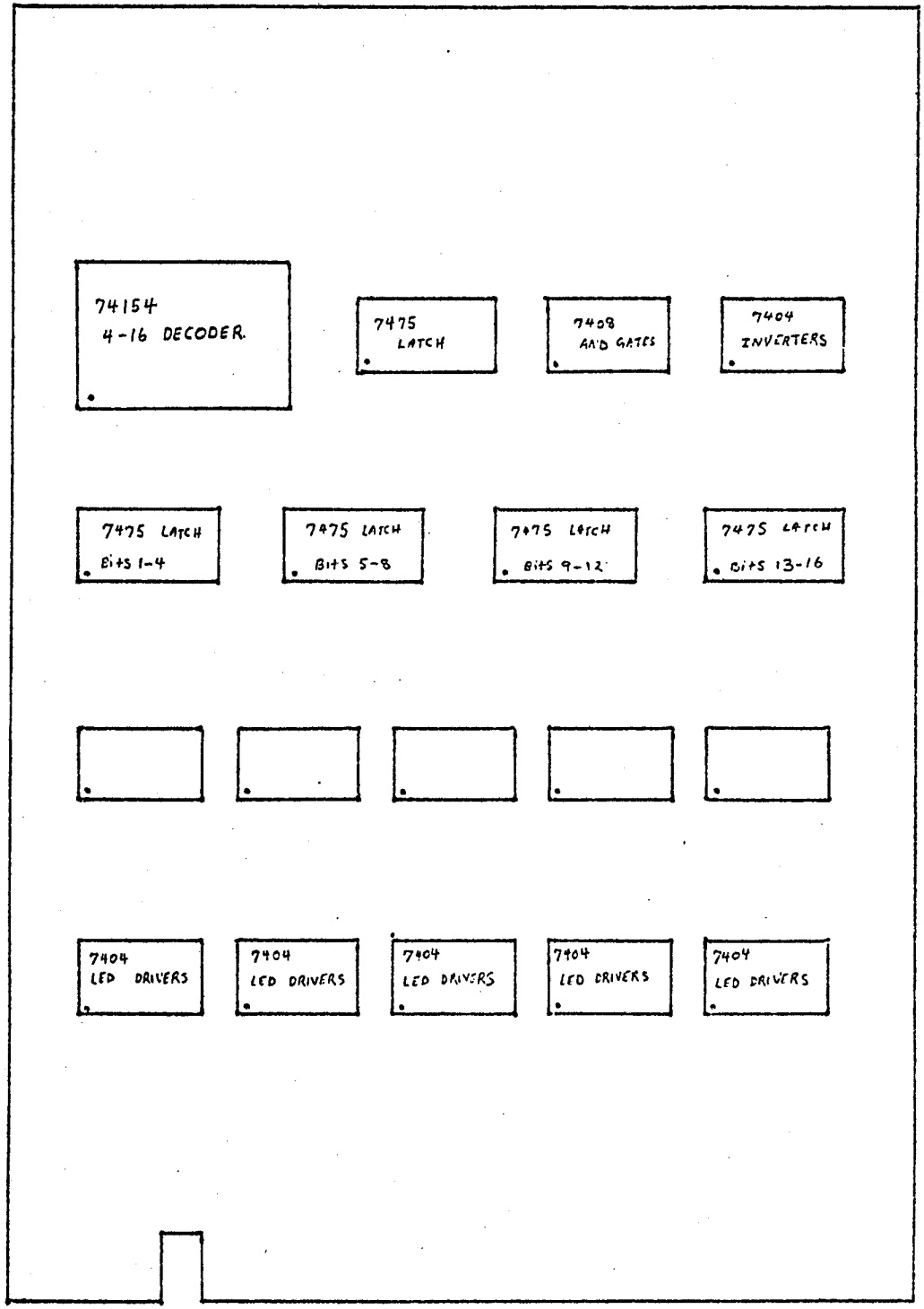


PROGRAM COUNTER AND MEMORY ADDRESS BOARD

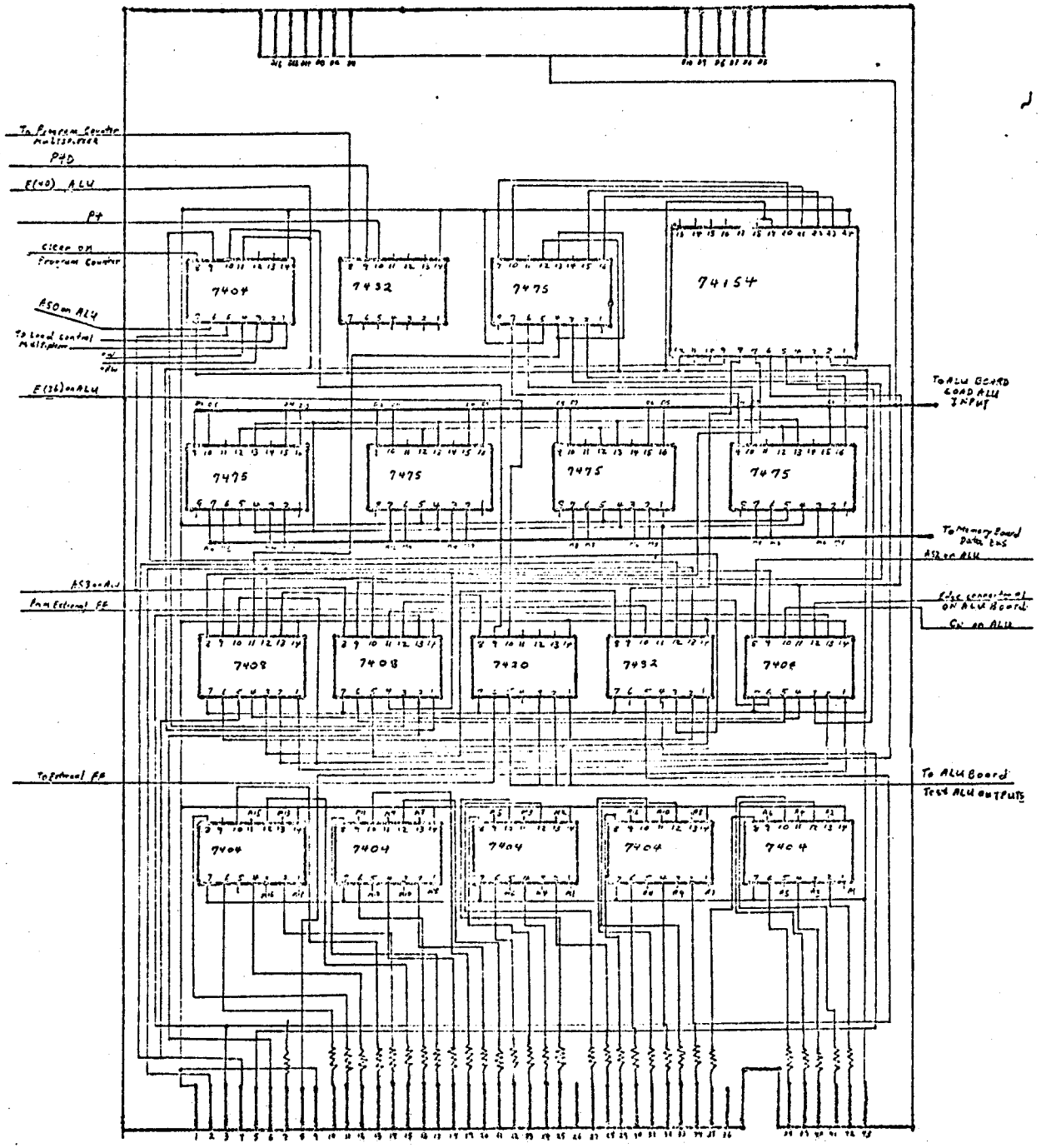
PIN CROSS REFFERENCE FOR COUNTERS.

The counters for the program counter had to be changed from 74LS160 to 74LS193. Below is a table showing the rewiring connections.

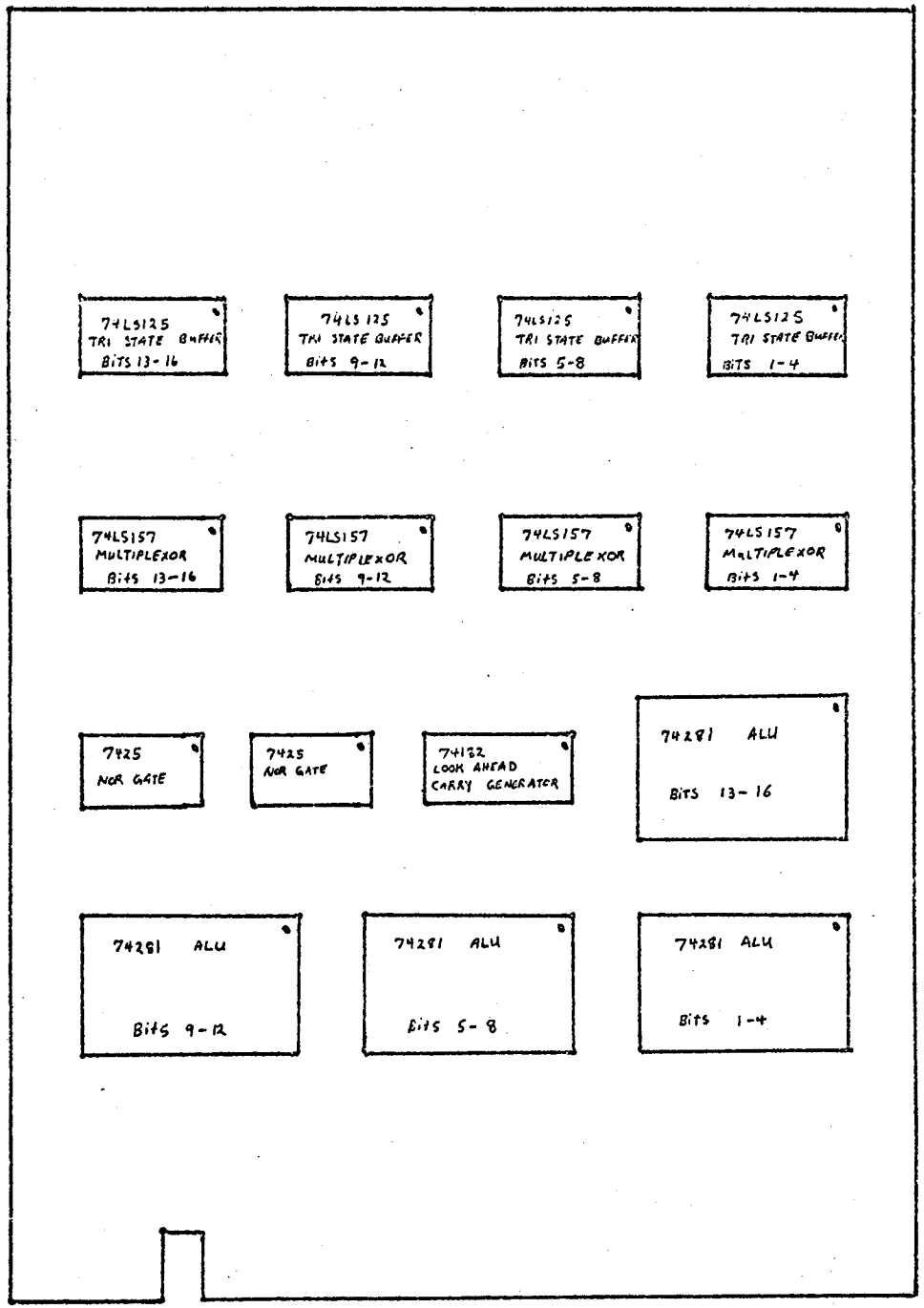
74LS160	74LS193	NOTES
1	14	CLEAR
2	5	CLOCK
3	15	A-input
4	1	B-input
5	10	C-input
6	9	D-input
7	4	Tie HIGH
8	8	GROUND
9	11	LOAD
10	13	No connection
11	7	D-output
12	6	C-output
13	2	B-output
14	3	A-output
15	12	CARRY OUT
16	16	Vcc



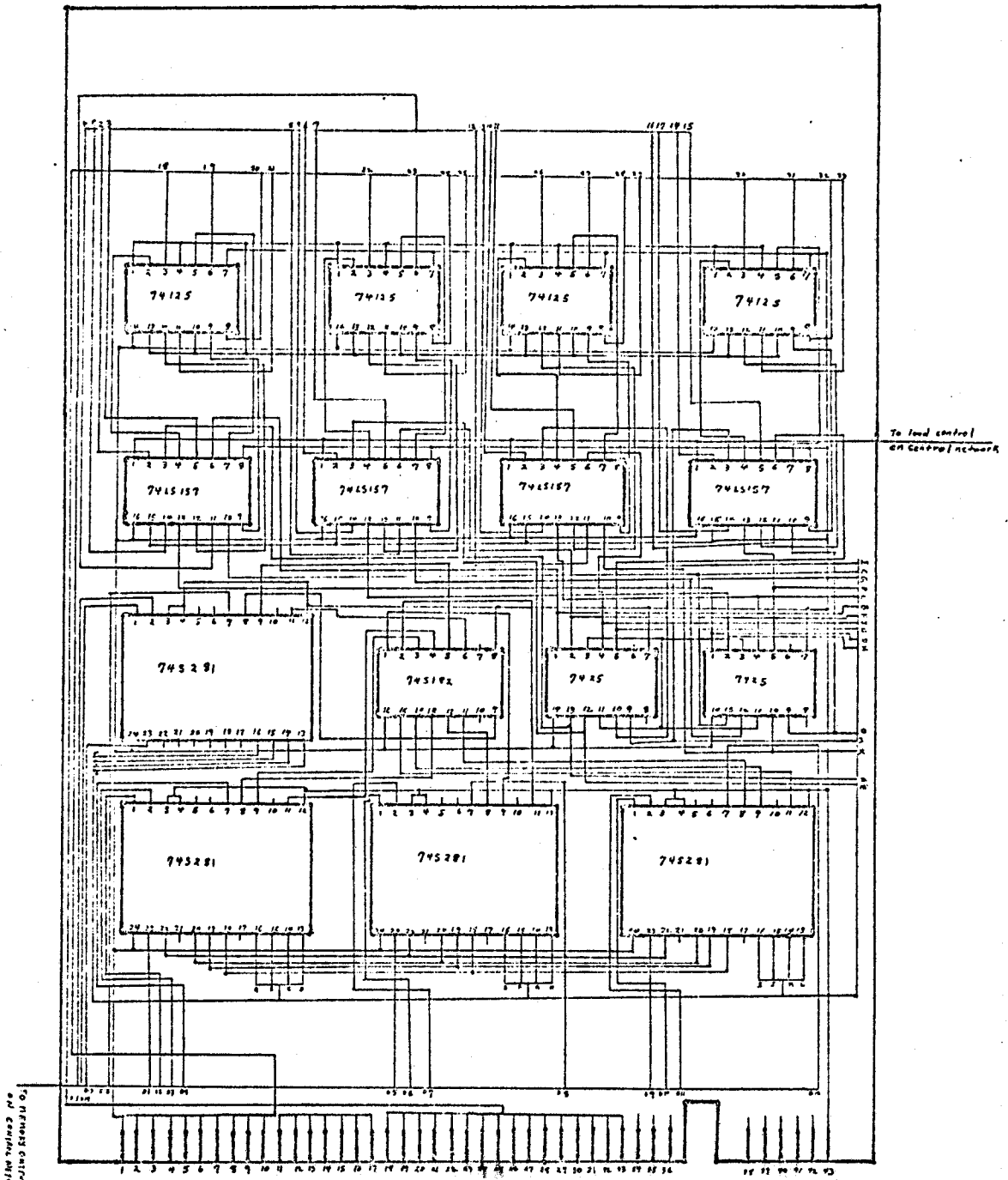
CONTROL NETWORK AND LED DRIVERS



CONTROL NETWORK AND LED DRIVERS



ALU AND MEMORY BUFFER BOARD



ALU AND MEMORY BUFFER BOARD

ACCUMULATOR CONTROL TABLE

LDA	0001	0	1	1	0
STA	0010	0	1	0	0
ADD	0100	0	0	1	1
SUB	0101	1	0	0	1
JMP	0110	0	1	0	0
BNE	0111	0	1	0	0
HLT	1000	0	1	0	0
NOP	----	0	1	0	0