TECHNICAL MEMORANDUM NASA 51

A FOUR DIGIT MEMORY-MAPPED DISPLAY

An interface board has been fabricated for the Ohio University Microcomputer-Based Navigation Receiver to display data from the microprocessor.

by

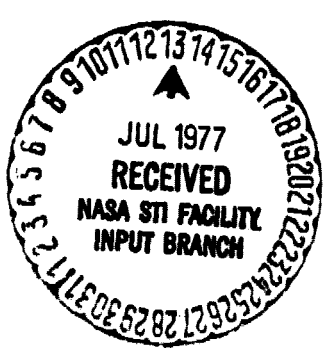
Ralph E. Smith

Avionics Engineering Center
Department of Electrical Engineering
Ohio University
Athens, Ohio 45701

May 1977

Supported by

National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia
Grant NGR 36-009-017



I. INTRODUCTION

A circuit for interfacing the JOLT(TM) microprocessor used in the Ohio University Navigation Receiver Base to four digital displays has been designed and fabricated. This circuit will take data from the microprocessor, which has been software-decoded, and display it on four seven-segment displays. Because the data is decoded in the microprocessor, not only numbers can be displayed, but also certain letters and other combinations of segments that might be desired.

II. CIRCUIT DESCRIPTION

Each digit can be addressed as if it were a memory location. This is similar to a memory-mapped output port. Since this is used as an output port to display data, data can only be stored and not read back. The addresses for the four digits are B0XX, B1XX, B2XX, and B3XX. As shown in the logic diagram, Figure 1, the most significant four address bits are decoded by an inverter and a NAND gate. This output, two inverters, and the R/W pulse are used as inputs to the NOR gates. The outputs strobe latches for data from the microcomputer bus. Since only one latch can be stored with a single store instruction, four store instructions are needed to fill the latches. The four DM8865 I.C.'s are used as drivers for the L.E.D. seven-segment displays.

There is an input from a switch on the front panel. When the switch is operated this line goes high causing all latches to be reset. With the latches reset all the segments in the displays will be on. This feature can be used to clear or test the displays. A slide switch on the front panel is used to turn off the displays. This is accomplished by switching off the 5V to the displays.

An auxiliary decoded output from the most significant four address bits is sent to the keyboard interface (described in NASA TM 50 by Lee Wright) where it is used to output data to the microcomputer when address BXXX is called. Figure 2 shows the seven-segment display wiring. Figure 3 shows the actual layout of the board as used in the Ohio University Microcomputer Navigation Receiver Base.

III. TEST PROGRAM

The program described here will take a four-bit hexadecimal and display it on one of the four displays. Three bits are used to designate which display digit is to be used and ti display a decimal point. This data is stored in location 0100 as shown in Table II. The four-bit hexadecimal number is shifted right, masked off, and loaded into the index register (X). The two bits specifying the digit number are masked off and effectively added to B016. This number is loaded into memory location 011B16 which is the high order byte of the display address. The bit specifying a decimal point is inverted, masked off, and inserted into the correct position in the display constant. This result is now in the accumulator and is stored to the display digit.

The segment coding constants are stored in zero page. The correct constant is obtained by using the AND instruction in the zero page, X mode. The X register was loaded with the hexadecimal number earlier in the program.

Figure 4 gives detailed digit and segment address data for reference.

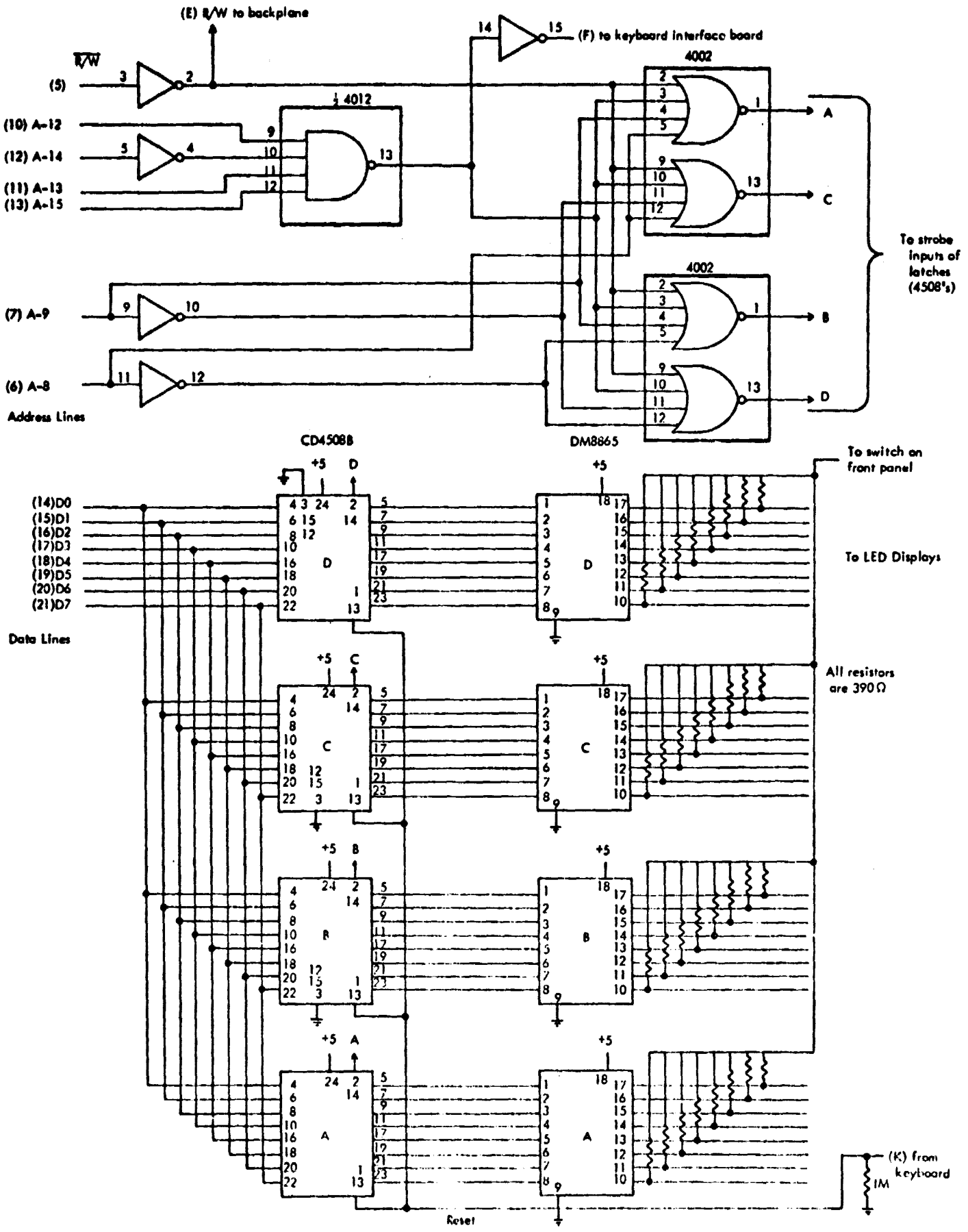


Figure 1. Display Interface Logic Diagram.

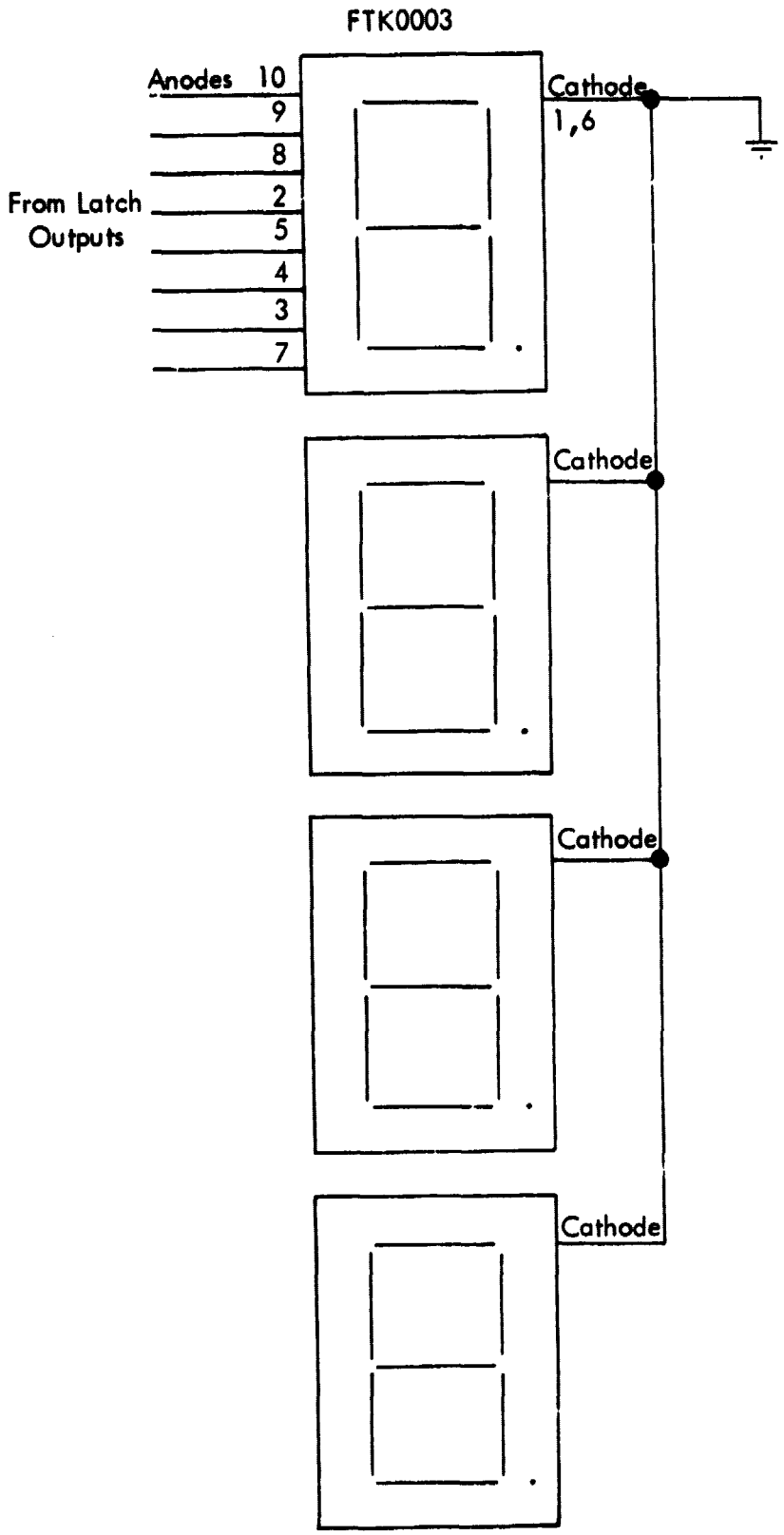


Figure 2. Display Details

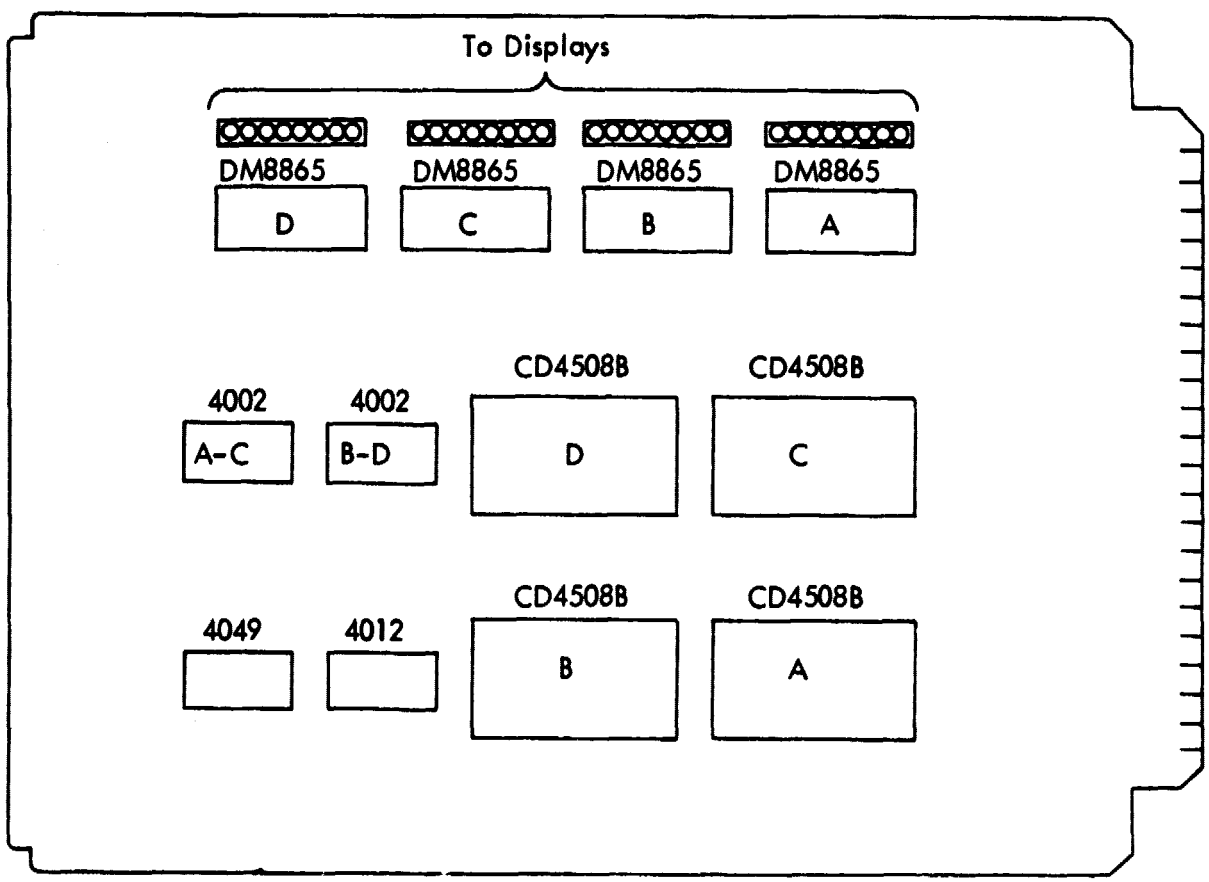
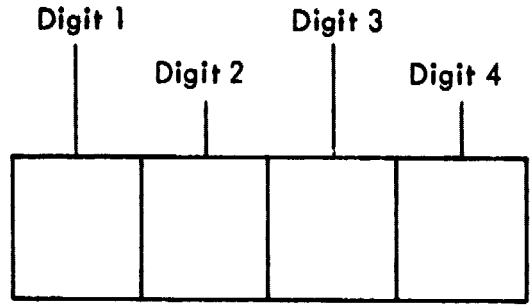
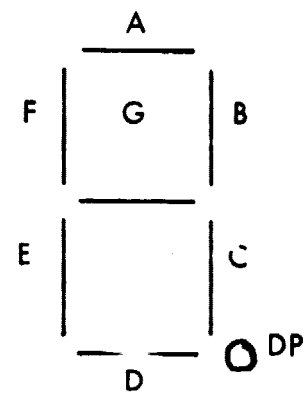


Figure 3. Circuit Board Layout.



Digit	Address Bus (digit addressing) A_{15} through A_8	Hexadecimal Address
1	1011XX00	B0XX
2	1011XX01	B1XX
3	1011XX10	B2XX
4	1011XX11	B3XX

Data		
D_0	-	A
D_1	-	B
D_2	-	C
D_3	-	F
D_4	-	D
D_5	-	E
D_6	-	G
D_7	-	DP



Logic 1 = segment off
Logic 0 = segment on

Figure 4. Segment Coding and Digit Addressing.

MEMORY LOCATION

Start	0101	LDA AD 00 01 load A with data
	0104	TAY A8 transfer A to Y for later use
	0105	LSR 4A } shift right two-bits to position hexadecimal number
	0106	LSR 4A
	0107	AND 29 OF mask off number
	0109	TAX AA load index register with number
	010A	TYA transfer data from Y to A
	010B	AND 29 03 mask off digit number
	010D	ORA 09 B0 add B0 to number
	0112	TYA 98 transfer from Y to A
	0113	EOR 49 FF invert data
	0115	ORA 09 7F mask off decimal point bit
	0117	AND 35 00 and with constant
	0119	STA 80 XXXX store to display
	010F	STA 8D 1B 01 store number in display address

Digit	Contents of Location 0100
1	X X X X X X 0 0
2	X X X X X X 0 1
3	X X X X X X 1 0
4	X X X X X X 1 1

Decimal pt. (bit 7) -1 = ON, 0 = OFF

Figure 5a. Example Programming.

MEMORY LOCATION	CONTENTS
0000	C0
0001	F9
0002	8C
0003	A8
0004	B1
0005	A2
0006	82
0007	F8
0008	80
0009	B0
000A	90
000B	83
000C	C6
000D	89
000E	86
000F	96

Figure 5b. Code Table for BCD-to-7 Segment Decoding.

Character	Hexadecimal Data	
	Without Decimal Put.	With Decimal Put.
0	C0	40
1	F9	79
2	8C	0C
3	A8	28
4	B1	31
5	A2	22
6	82	02
7	F8	78
8	80	00
9	B0	30
A	90	10
B	83	03
C	C6	46
D	89	09
E	86	06
F	96	16
G	C2	40
H	91	11
Blank	FF	7F
--	BF	3F

Table I. Hexadecimal Equivalent of Character Display.