NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

Microprocessor Generated
Vertical Gyrohorizon Instrument
for the Blue Bird Simulator

by

Marc A. Lucchesi

December 1980

Thesis Advisor: D. M. Layton

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Microprocessor Generated Vertical Gyrohorizon Instrument for the Blue Bird Simulator.		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; December 1980
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Marc A. Lucchesi, LT, USN		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE December 1980
		13. NUMBER OF PAGES 39
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Cathode Ray Tubes (CRT)
Microprocessor
Vertical Gyrohorizon Instrument (VGI)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number) An X-Y cathode ray tube display for use in a high-performance aircraft simulator facility as a Vertical Gyrohorizon Instrument was investigated. A microprocessor was used to generate the correct angle for the display corresponding to the analog equations of motion of the simulator. An unfavorable displayed result was obtained. Detailed conclusions and reccomendations for further study are presented.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601 |

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Approved for public release; distribution unlimited

Microprocessor Generated Vertical Gyrohorizon Instrument for the Blue Bird Simulator

by

Marc A. Lucchesi
Lieutenant, United States Navy
B.S., Miami University, 1974

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December, 1980

ABSTRACT

An X-Y cathode ray tube display for use in a high-performance aircraft simulator facility as a Vertical Gyrohorizon Instrument was investigated. A microprocessor was used to generate the correct angle for the display corresponding to the analog equations of motion of the simulator. An unfavorable displayed result was obtained. Detailed conclusions and recommendations for further study are presented.

TABLE OF CONTENTS

I. INTRODUCTION -- 7

A. BACKGROUND -- 7

B. THE PROBLEM -- 8

II. APPROACH -- 10

III. HARDWARE DEVELOPMENT -- 12

IV. SOFTWARE DEVELOPMENT -- 19

V. RESULTS AND CONCLUSIONS -- 24

VI. RECOMMENDATIONS -- 26

APPENDIX A: INTEL 8035 DETAILS -- 28

APPENDIX B: TEXAS INSTRUMENTS TI-59 PROGRAM -- 29

APPENDIX C: DISPLAY PROGRAM -- 32

LIST OF REFERENCES -- 38

INITIAL DISTRIBUTION LIST -- 39

LIST OF FIGURES

1. SET-UP TIMES -------------------------------------- 14

2. LOGIC FLOW -- 16

3. CIRCUIT DIAGRAM (TOP) ----------------------------- 17

4. CIRCUIT DIAGRAM (BOTTOM) -------------------------- 18

5. FLOW CHART -- 21

6. CRT SCALING --------------------------------------- 23

ACKNOWLEDGEMENTS

I would like to express my appreciation for the guidance and assistance provided by Professor Donald M. Layton that led to the completion of this thesis. Technical aid was provided throughout the construction of this project by Mr. Ted Dunton and Mr. Ray Garcia of the Aeronautics Department, Naval Postgraduate School.

The support and understanding of my wife and son, for which I am forever grateful, were invaluable to me in completing this task.

I. INTRODUCTION

In any academic environment there exists a vast separation between the course work (theory) and practical experience (laboratory). At the Naval Postgraduate School, in order to bridge this gap, some laboratory sessions of the Aeronautics Department use a large computer to simulate real world conditions. By inserting certain parameters into the computer one can observe the effects these parameters have on the modeled world. However, this type of simulation has several drawbacks, two of which are very detrimental to the learning process: one is the lack of instant response which causes a loss of interest in any problem and the other is the lack of realism which causes a loss of stimulation for the learning process itself.

A. BACKGROUND

For laboratory simulation of aircraft dynamics, it is desired to utilize a device that; (a) presents to the operator (pilot) a realistic cockpit environment and (b) provides external monitoring of inputs and outputs. Such a device may range from a relatively simple, fixed-based, two-degree-of-freedom simulator to a more complex, moving base, six-degree-of-freedom device. And, although a wide range of commercial simulators are available, not only are these devices costly, but they require extensive modifications to meet the demanding requirements of academic laboratory exercises.

Therefore, to circumvent this situation, it was decided to install a Cockpit Procedures Trainer (CPT) and to convert it to a six-degree-of-freedom, fixed-based simulator, the "Blue-Bird". In order to get this simulator to "fly", James H. Aldrich devised complex and extensive analog programs simulating the F-4 Phantom II aircraft equations of motion. After completion of this task (Ref. 1), the simulator could be used for supplemental instruction in courses in the Aeronautics Department (Static Stability and Control, AE 2036; Dynamic Stability, AE 4301; Flight Evaluation Techniques, AE 4323). Unfortunately these analog programs were so complex that there was little difference between using this system and putting numbers into a large digital computer. To simplify the programming, simple spring-mass-damper equations

$$m\ddot{x}+c\dot{x}+kx=f(t)$$

were used for the analog programs (Ref. 2) which allows for a quicker understanding of aircraft motion.

At this point one could sit in the cockpit, move the flight controls, and watch the results on strip chart recorders, but there was no visual display of longitudinal or lateral motion in the cockpit. This, of course, still did not provide all the realism desired, but it was a vast improvement over the large digital computer.

B. THE PROBLEM

The problem therefore, was to design and construct a two-dimensional visual display apparatus that would simulate

a Vertical Gyrohorizon Instrument (VGI) of an actual aircraft. This display would accept inputs from the new analog computer program output (pitch angle and bank angle), and display this information in a dynamic manner. The visual display was meant to simulate an actual VGI, but was not intended to have the exact visual characteristics of any actual instrument. It should have the generic characteristics acceptable by those pilots who might use the facility. This would provide one more step to the complete simulator.

II. APPROACH

There are three basic ways to address the problem of constructing a VGI: purely mechanical, purely video, or computer generated. Combination of these three are, or course, feasible, but will not be discussed in detail.

1. The purely mechanical approach would probably require a purchase of a VGI display specifically designed for the F-4 aircraft or a VGI instrument designed for flight simulator use. In either case, the installation would require a high frequency alternating current source and some sort of servo drive system that would respond to a varying voltage, direct-current output. In addition to being expensive, nothing would really be learned from this approach.
2. The purely video approach, again, would be very expensive, requiring the purchase of a video camera, and the building of a gimbled platform that would be linked to the equations of motion for roll and pitch. Although challenging from a design viewpoint, this is not very practical.
3. The computer generated approach, therefore, seemed the most fruitful. The low cost of computer chips, the available documentation to develop a circuit, the relatively small size of the computer board, and the fact that the simulator contains all the necessary power, made this approach the most practical one.

In any design procedure chosen, however, the input voltage to the VGI device (adjustable up to a positive/negative ten volts of direct current) needed to be massaged to produce a display with the following characteristics; (a) at least a sixty degree bank angle in either direction, (b) at least a twenty degree nose up/nose down pitch angle, and (c) at least a twenty-five degree per second roll rate. These parameters were considered the absolute minimum to insure realism of any type of VGI design.

III. HARDWARE DEVELOPMENT

Once a computer generated design was decided upon, the type of computer needed to be addressed. The design application called for a computer that would receive input from the analog equations of motion, massage the data, and put it out to some sort of display device. Inasmuch as this was to be a "real time" simulation, a computer was needed that was fast enough for real time. It was decided that, since the job required little actual memory, a microprocessor based system would be utilized. Of all the microprocessors available on the open market that would be useful, the Intel 8035 was chosen. Although the 8035 is not the fastest computer (cycle time of twenty-five micro-seconds) available, its all-in-one chip design, its quickness and variety of its instructions set (no instruction took longer than two cycles), and its availability made this chip the perfect choice. (Details on the 8035 are presented in Appendix A.)

Next, the matter of a cockpit display device needed to be addressed. The nature of a VGI lends itself to the concept of an X-Y plotter. In other words, if two sets of coordinates are put on a plotter, a straight line can be drawn between the two points. In order to accomplish this task, an oscilloscope with a horizontal input with calibration was needed. Since the two inputs (horizontal and vertical) were supplied to the oscilloscope, a X-Y cathode ray tube (CRT) was produced.

Next the circuit for connecting the computer to the input and output devices needed to be constructed. To change the analog data from the equations of motion, two eight-bit analog to digital converters were used. One converter was used for the pitch equation of motion, and the other for the roll equation of motion. The AD570S analog to digital converter was chosen because of its extremely fast conversion time (twenty-five micro-seconds) and its availability. These were connected to an Intel 8255A programmable peripheral interface chip which provided the necessary communication between the converters and the computer chip. These three chips comprised the input section.

The output section, on the other hand, was comprised of essentially only one chip: the Burr-Brown MP-10 microprocessor interfaced, eight-bit, analog output system. This chip contained one 8255 and two digital-to-analog converters on board. Therefore, only one chip provided the two outputs needed for the X-Y concept. Although these converters are slow for digital-to-analog converters (twenty-five micro-seconds), the one chip design far outweighed any increase in speed. The only problem with this device was synchronizing its timing with that of the 8035 computer. This was overcome by the use of two 74121 one shot chips. These chips were needed to delay the write pulse from the computer 600 nano-seconds to allow for a longer address set-up time on the MP-10. (figure 1)

The other chips required were from one to three Intel 8708 electrically programmable read only memory (EPROM) and on Intel

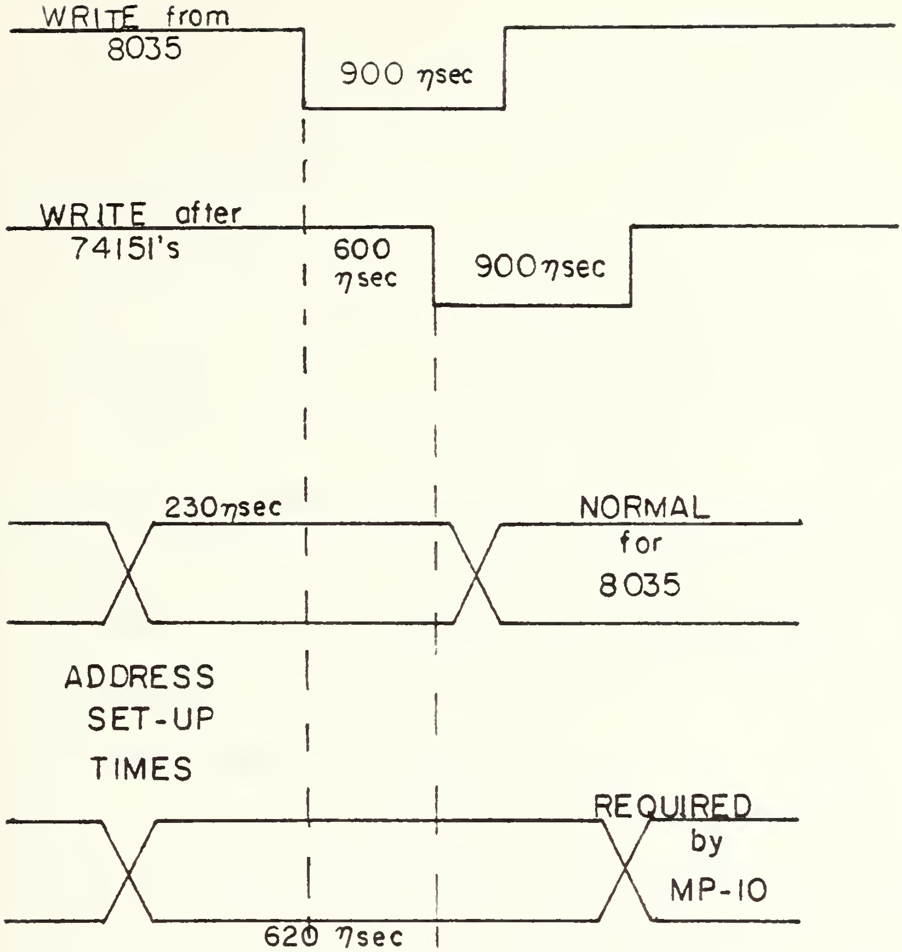

FIGURE 1 SET-UP TIMES

8212 eight-bit input/output port used as an address latch to hold data for addressing the 8255A, MP-10, and the 8700 chips.

The other two chips of the board were used as follows: the 7402 dual input NOR gate was used for external system reset and for external test for altitude and airspeed inputs, and the 74155 demultiplexer was used to select between the three 8708 memory chips.

Following the preliminary design it was necessary to construct the computer board. It was decided that since there were so many connections to be made (figure 2), "bread boarding" would not be the most practical approach. By using a photo-etching technique all the interconnections would automatically be made and, as a result, tracing probable errors would become relatively simple. Therefore, a two sided board was designed and etched (figures 3 & 4).

Finally, several power supplies were required. All the chips on the board required a positive five volts. Additionally, the MP-10 and the two AD570Ss required a positive and negative fifteen volts, and the 8708s required a positive twelve and a negative five volts.

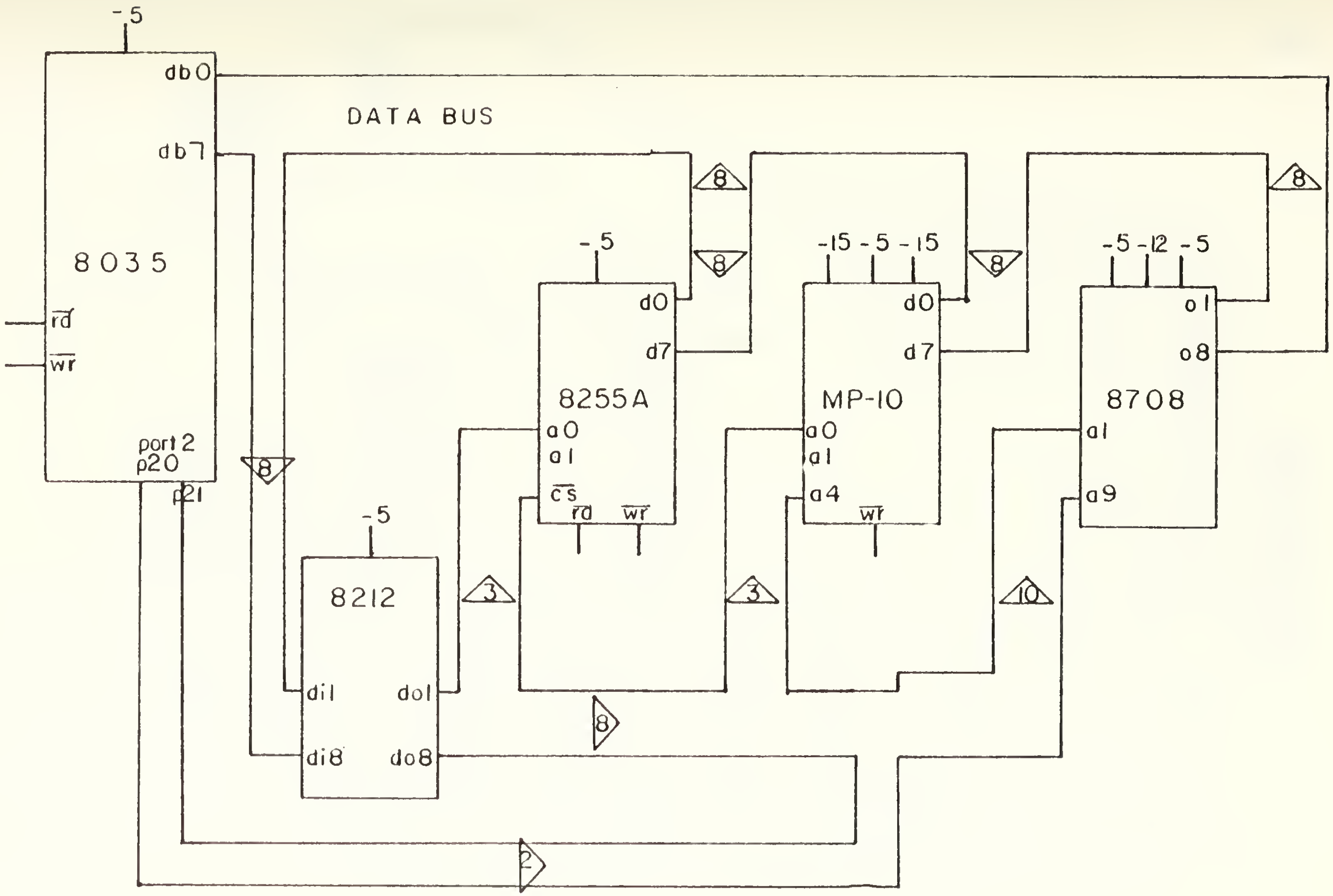

FIGURE 2 LOGIC FLOW

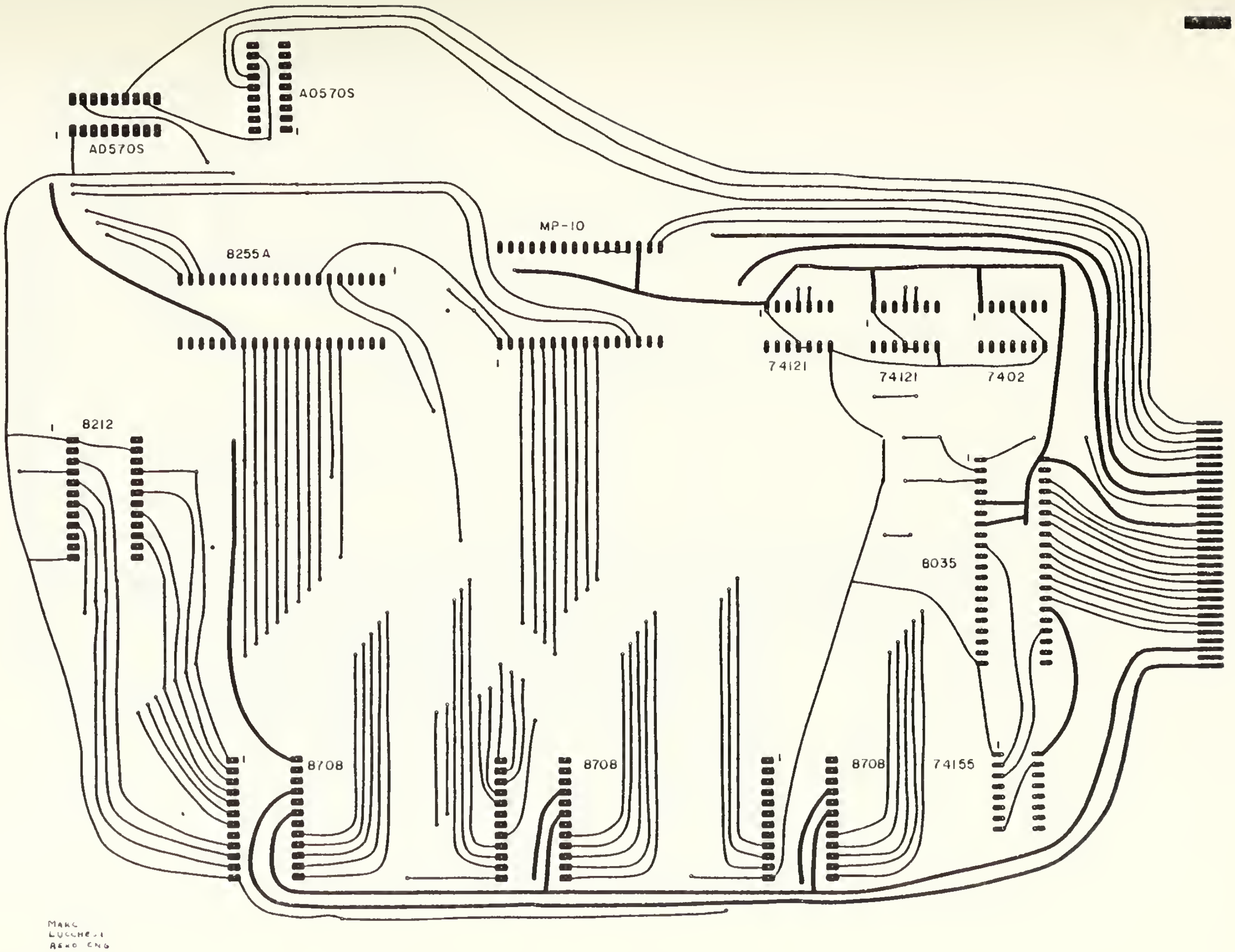

FIGURE 3 CIRCUIT DIAGRAM (TOP)

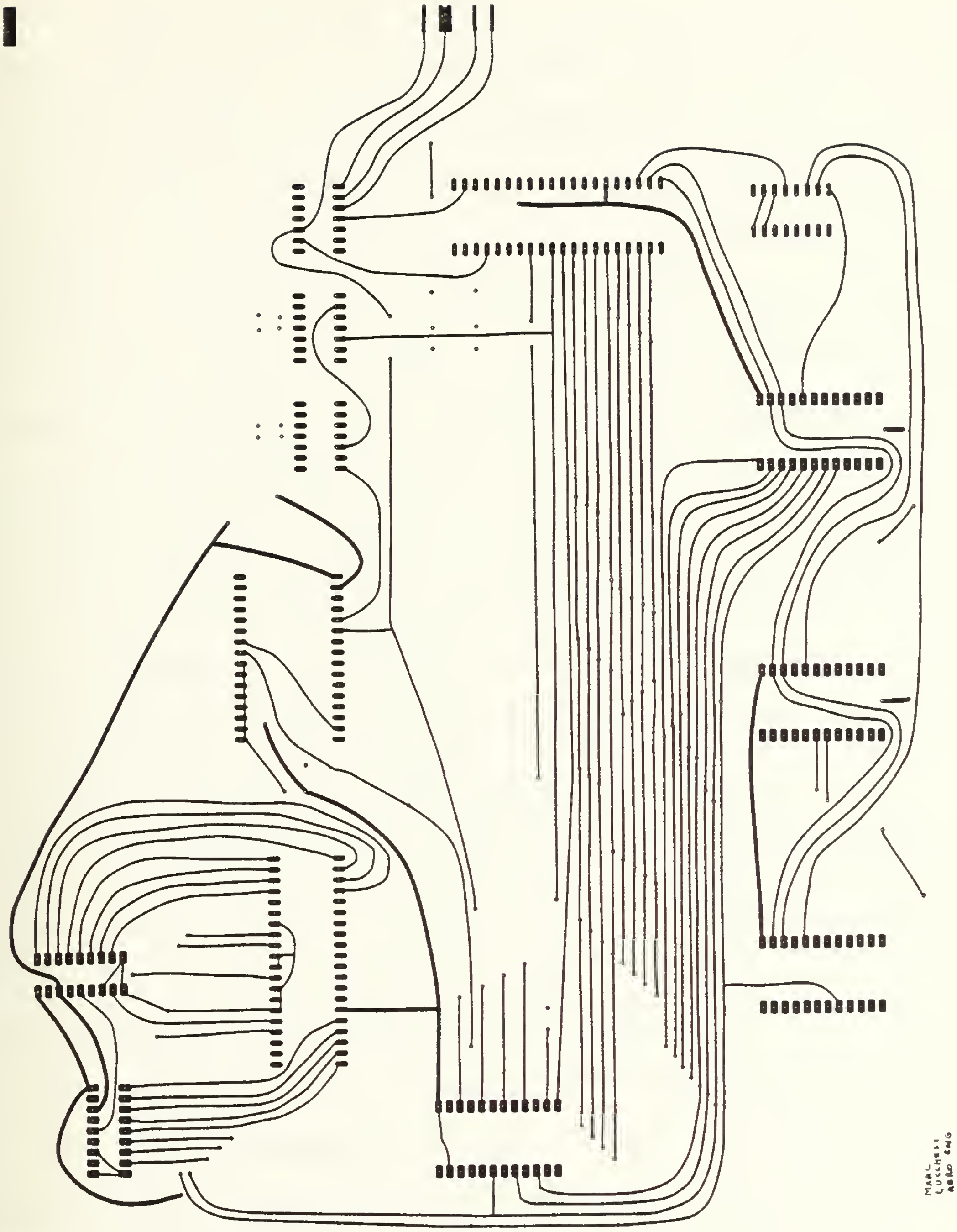

FIGURE 4 CIRCUIT DIAGRAM (BOTTOM)

IV. SOFTWARE DEVELOPMENT

The program logic flow was a relatively straight-forward process (figure 5). The basic format was to bring in the roll and pitch motions separately, couple them, and put them out to the X-Y CRT. The input needed to be brought in only once to produce the desired output. The design called for the first end point to be at the far left of the CRT and the second end-point to be at the far right. (Since the input and output devices are bipolar, i.e., accept both positive and negative voltages, the scaling of the output is as shown in figure 6.) The theory dictates that if a vertical voltage is applited to the CRT (roll motion) then the position on the Y axis should change in opposite directions at the two endpoints. And if the switching between these endpoints were done quickly enough a straight line at any angle should be formed.

With this theory as a reference point, the program began to take shape. To start the sequence of operation, the 8255A and the MP-10 chips needed to be initialized. The MP-10 is a straightforward, two step process (Ref. 3), while the 8255A is quite another story. There are many modes to the 8255A that can be programmed as either input or output. The design called for two input ports and one split input/output port for communication to the analog-to-digital converters. Therefore, ports A and B are pure input while port C was the split one (Ref. 4). Once the chips have been initialized, the computer

then requests the data from the analog-to-digital converters. The program then uses the data from the converters to select the proper output data that was fed to the MP-10. The output data is stored on a sine look-up table in order to arrive at the correct angle. (A Texas Instruments TI-59 programmable calculator was used to generate the sine look-up table appearing in the main program. The TI-59 program is in appendix B.) Once the data is received from the look-up table it was put out as follows: Y position on output 1 (sine of the angle), and X position on output 2 (cosine of the angle) of the MP-10. The pitch was added to the Y position to move the center of the line either up or down.

Once the program was written it was keyed into the Intel Prompt 80/85. This device does not belong to the MCS-48 computer systems (it is part of the MCS-80 system or better known as the 8080A based system), but its ability to program the 8708 EPROMS made this system indispensible. (Appendix C)

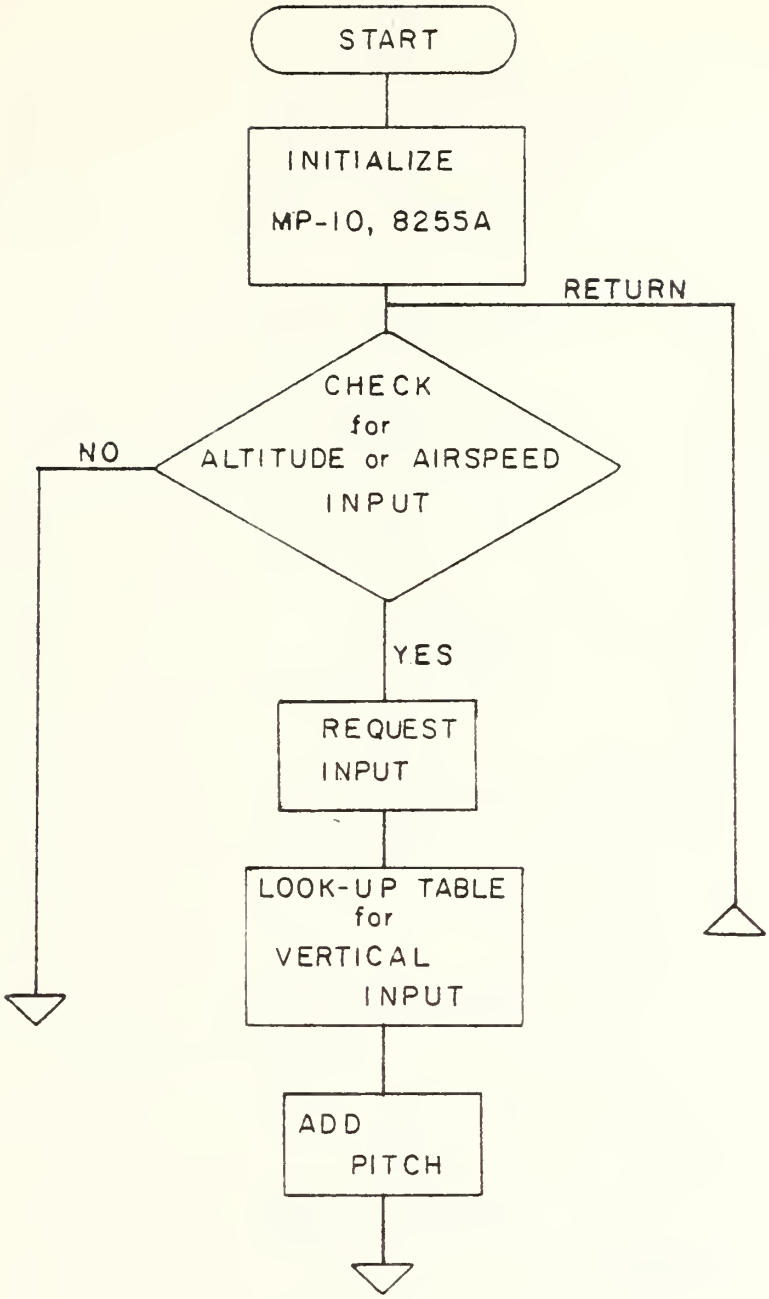

FIGURE 5 FLOW CHART

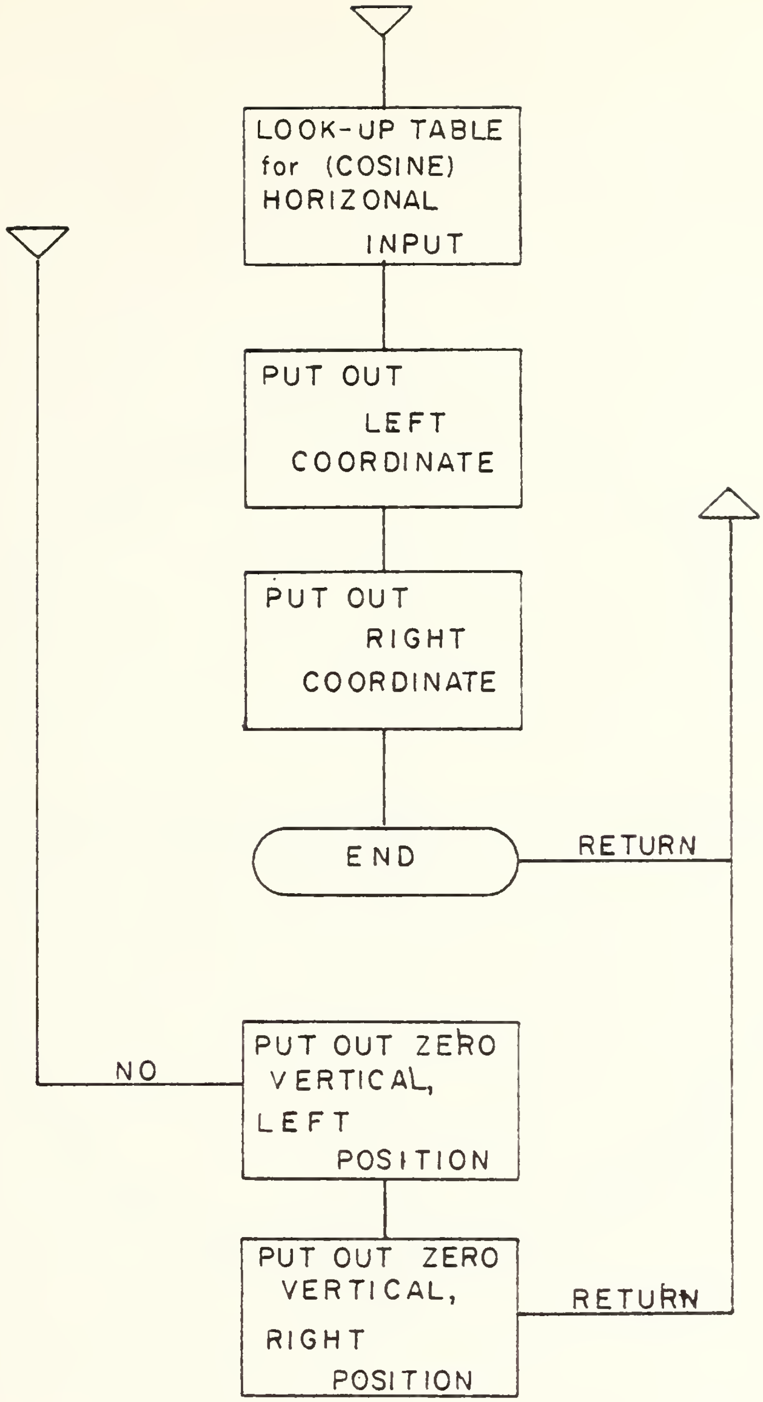

FIGURE 5 (CONTINUED)

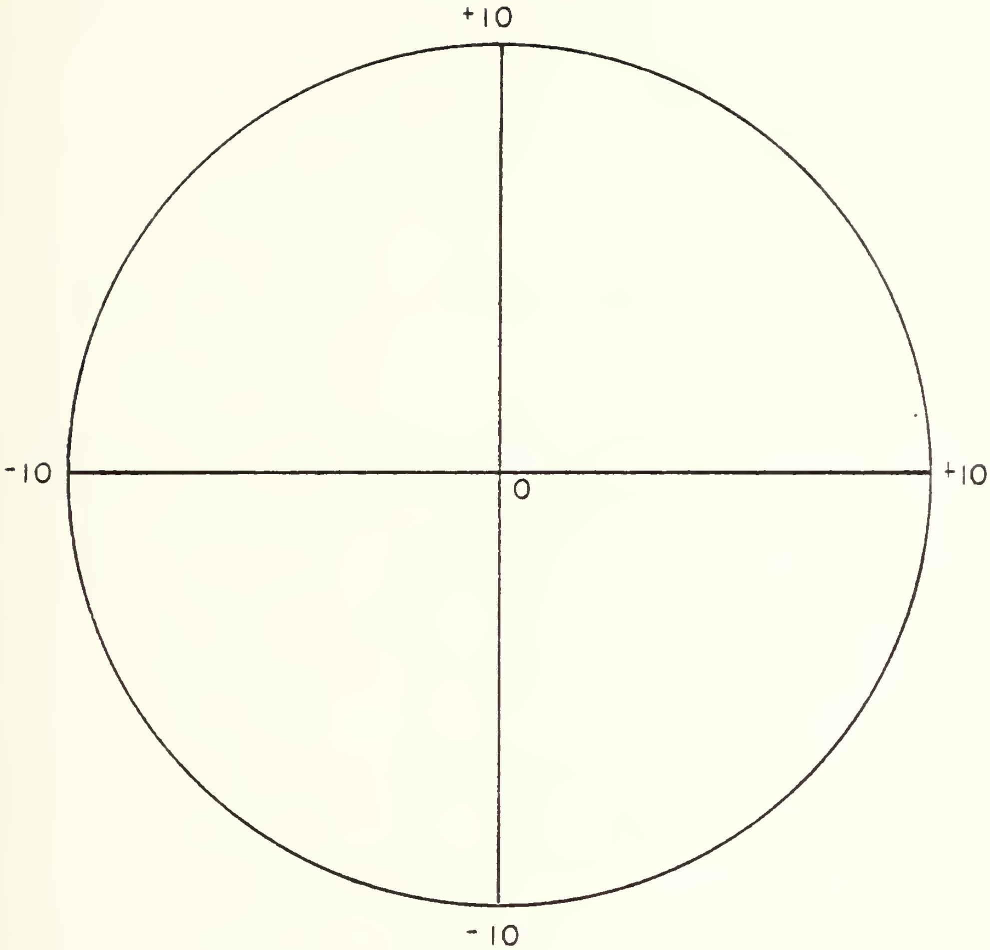

FIGURE 6 CRT SCALING

V. RESULTS AND CONCLUSIONS

Once the 8708 EPROM had been programmed it was put onto the board with the other chips, thereby making a complete computer.

Testing of the computer was accomplished with the use of a "bread board" box to supply the necessary power. A joy-stick was also connected through the "bread board" and used as the lateral and longitudinal inputs to simulate motion. The oscilloscope used for the X-Y CRT was the same as that for the cockpit display: the Textronix Type 504 single trace, tube type oscilloscope.

When power was applied to the computer with the joy-stick in the neutral position, i.e., center, the face of the CRT lit up with a straight line from left to right across the center of the scope. This was exactly as predicted, but as the joy-stick was moved, the straight line expanded into a rectangle instead of a skewed straight line. This rectangle formed a square at the forty-five degree position of the joy-stick, then another rectangle, until at the ninety degree position of the joy-stick a vertical straight line was formed.

What was not recognized from the outset of this project was the fact that a CRT does not behave the same as a normal X-Y plotter. In other words, on a plotter the X and Y coordinates are put to a device before a point is printed, whereas on the CRT each coordinate is displayed independently.

Therefore, unless the X and Y coordinates are outputed to the display device simultaneously a box will be formed. Since it is impossible to have simultaneous data when using only one computer, the conclusion must be made that using only the endpoints of a line will not produce the desired skewed straight line.

VI. RECOMMENDATIONS

There exist a few areas of study that could produce the desired results of a skewed straight line. Unfortunately, time constraints have prevented the author from pursuing any of these.

1. Using the basic program to the point just after the data is retrieved from the look-up-table, one can divide this data into sixteen parts thereby producing a seventeen segmented output. This produces at any angle a straight line of sorts. In other words, the actual coordinates produce a small stair-step line the width of the scope. The seventeen segmented display should be small enough, however, so that the output is not distracting. The only problem with this approach is that the output may not be "real time"; i.e., there may be too much delay between the stick motion in the cockpit and what is perceived on the scope.
2. A hardware add-on which might be addressed is that of a resistor network in conjunction with a 555 timing chip to produce a rastor scan on the oscilloscope. This essentially produces a stair-step, but the size of the step can be varied. Problems with this might lie in the fact that the line may not be able to be reversed; i.e., the scope would show only one direction of bank.

3. One other change is to use a different type of display. The Aeronautics Department has a television type monitor that accepts video like any other monitor, but is also gear driven to produce a skewed picture. By removing the video tube and supplying a motor to drive the screen, one can produce the desired effect. This makes for a much simpler software problem, but the current needed to drive the available motors are much beyond the output current of the computer.

4. If any of the above approaches accomplishes the desired effect, there should be further study to produce a "flying" simulator. The areas that need to be addressed are coupling for the airspeed and the altitude read outs. The present system allows only the operator to supply the necessary voltages to move the dials in the cockpit. Probably the best means of displaying the information would be to remove the present gauges and use digital displays. When this segment has been completed the simulator will be "flyable" in the true sense.

APPENDIX A

The Intel 8035 is part of the overall Intel MCS-48 family of computer systems. Designed as a special purpose system it can be adapted to most situations requiring small space and memory. (Reference 4)

The only difference between the 8035 and the other chips of the family was that the 8035 had no on board memory. This proved to be extremely useful because the number of 8708 memory chips available made program changes quicker than would have been trying to change only on 8748 computer chip.

APPENDIX B

TEXAS INSTRUMENTS TI-59 PROGRAM

000	76	1BL	
001	11	A	; DECIMAL TO BINARY ; CONVERSION SUBROUTINE
002	29	CP	;CLEAR T REGISTER
003	42	STO	
004	10	10	;STORE NUMBER AT REGISTER 10
005	00	0	
006	42	STO	
007	11	11	;STORE ZERO IN REGISTER 11
008	42	STO	
009	02	02	;AND REGISTER 2
010	76	LBL	
011	87	IFF	;WORKING SUBROUTINE
012	53	(	
013	43	RCL	
014	10	10	;PUT NUMBER INTO WORKING
015	55	/	;REGISTER
016	02	2	
017	54	)	
018	42	STO	
019	01	01	;STORE NUMBER/2 IN 1
020	59	INT	;INTEGER VALUE OF NUMBER/2
021	42	STO	;AND STORE IT IN 10
022	10	10	
023	53	(	
024	43	RCL	;PUT NUMBER/2 INTO
025	01	01	;WORKING REGISTER
026	22	INV	
027	59	INT	;KEEP ONLY NUMBER RIGHT ;OF DECIMAL POINT
028	69	OP	
029	10	10	
030	65	*	
031	01	1	
032	00	0	
033	45	Y↑T	;RAISE THIS NUMBER TO THE
034	43	RCL	;POWER IN REGISTER 2
035	02	02	
036	54	)	
037	44	SUM	
038	11	11	;ADD THIS TO REGISTER 11
039	69	OP	

```
040   22   22     ;INCREMENT REGISTER 2
041   43   RCL
042   10   10
043   22   INV
004   67   EQ     ;IF THIS NUMBER IS NOT
045   87   IFF    ;EQUAL TO ZERO GO BACK TO
                  ;START OF THE SUBROUTINE
046   43   RCL
047   11   11     ;DISPLAY THIS NUMBER AND
048   92   RTN    ;RETURN TO CALLED PLACE
049   76   L      ;START OF THE PROGRAMBL
050   13   C      ;SEQUENCE
051   42   STO
052   20   20     ;STORE BEGINNING NUMBER
053   01   1
054   02   2
055   08   8
056   42   STO    ;STORE 128 IN ZERO FOR A
057   00   00     ;COUNTER
058   01   1
059   42   STO
060   05   05     ;STORE 1
061   00   0
062   42   STO
063   06   06     ;CLEAR REGISTER 6
064   43   RCL
065   05   05
066   69   OP     ;PRINT REGISTER 5
067   06   06
068   43   RCL
069   06   06
070   85   +
071   43   RCL
072   20   20     ;ADD REGISTERS 6&20
073   95   =
074   42   STO
075   06   06     ;STORE THIS SUM IN 6 AGAIN
076   38   SIN
077   65   *
078   05   5
079   55   /
080   93   .
081   00   0
082   03   3
083   09   9
084   03   3
085   07   7
086   95   =      ;5*SIN(X)/.03937
087   69   OP
088   06   06     ;PRINT THIS
```

```
089    59     INT            ;INTERGERIZE IT
090    11     A              ;AND CALL SUBROUTINE A
091    69     OP
092    06     06             ;PRINT THE OUTPUT OF SUB-
                             ;ROUTINE A
093    98     ADV            ;ADVANCE THE PAPER
094    69     OP
095    25     25             ;INCREMENT REGISTER 5
096    97     DSZ            ;DECREMENT REGISTER ZERO
097    00     00             ;AND SKIP TO THE END IF
                             ;IT IS ZERO
098    00     00             ;OTHERWISE GO TO
099    64     64             ;STEP 64
100    92     RTN            ;STOP
```

APPENDIX C

DISPLAY PROGRAM

```
0000  00    NOP
0001  27    CLR A          ;CLEAR ACCUMULATOR
0002  17    INC A          ;
0003  AF    MOV R7,A       ;PUT "1" INTO REG 7
0004  17    INC A
0005  AE    MOV R6,A       ;PUT "2" INTO REG 6
0006  17    INC A
0007  A8    MOV R0,A       ;PUT "3" INTO REG 0-ADDRESS
                           ;FOR 8255A
0008  23    MOV A,#        ;SELECT 8255 AND PUT MODE
                           ;WORD OUT
0009  93    10010011B      ;(MODE 0; A, B, C LOWER ARE
000A  90    MOVX @R0,A     ;INPUT, C UPPER IS OUTPUT)
000B  B9    MOV R1,#       ;PUT ADDRESS FOR MP-10 INI-
000C  83    10000011B      ;TIALIZATION INTO REG 1
000D  23    MOV A,#
000E  80    10000000B
000F  AD    MOV R5,A       ;STORE INITIALIZATION DATA
0010  91    MOVX @R1,A     ;SELECT MP-10 AND INITIALIZE
0011  FE    MOV A,R6       ;
0012  A9    MOV R1,A       ;PUT 2 INTO REG 1 (PORT C OF
                           ;8255A)
0013  47    SWAP A         ;NOW THE BLANK AND CONVERT
                           ;PIN IS SET
0014  AC    MOV R4,A       ;PUT 00100000 INTO REG 4
0015  91    MOVX @R1,A     ;INSURE BIT 6 IS HIGH TO
                           ;START DATA CONVERSION
0016  56    JT 1           ;JUMP IF THERE IS NO A/S OR
0017  53    01010011B      ;ALT INPUT
0018  27    CLR A
0019  91    MOVX @R1,A     ;INSURE BIT  6 IS LOW TO HOLD
                           ;DATA
001A  81    MOVX A.@R1     ;CHECK FOR DATA READY BITS
001B  53    ANL A,#
001C  0C    00001100B
001D  96    JNZ            ;TRY AGAIN IF BITS 2&3 ARE
001E  1A    00011010B      ;HIGH, BECAUSE THEY ARE NOT
                           ;READY
001F  85    CLR F0
0020  95    CPL F0         ;INSURE FLAG IS HIGH
0021  A8    MOV R0,A       ;INSURE R0 IS CLEARED
0022  80    MOVX A,@R0     ;BRING IN FORE AND AFT STICK
```

```
0023  AB     MOV R3,A         ;POSITION
0024  18     INC R0
0025  80     MOVX A,@R0       ;BRING IN LEFT AND RIGHT
0026  AA     MOV R2,A         ;STICK POSITION
0027  FC     MOV A,R4
0028  91     MOVX @R1,A       ;INSURE BIT IS HIGH FOR FREE
                              ;DATA CONVERSION
0029  FA     MOV A,R2         ;BRING BACK LEFT/RIGHT STICK
                              ;POSITION
002A  F2     JB 7             ;JUMP IF BIT 7 IS HIGH
002B  2D     00101101B
002C  85     CLR R0           ;IF L/R INPUT IS NEG THEN
                              ;CLEAR FLAG
002D  53     ANL A,#          ;STRIP OFF BIT 7 AND DISCARD
002E  7F     01111111B
002F  AA     MOV R2,A
0030  E3     MOVP3 A,@A       ;BRING VALUE FROM LOOK UP
                              ;TABLE (5*SIN(Y))
0031  B6     JF 0             ;IF FLAG 0 IS SET DO NOT
0032  34     00110100B        ;COMPLEMENT (5*SIN(Y))
0033  37     CPL A
0034  6B     ADD A,R3         ;ADD 5*SIN(Y)+X FOR VERTICAL
                              ;SCOPE INPUT
0035  AB     MOV R3,A
0036  FA     MOV A,R2         ;BRING BACK LEFT/RIGHT STICK
                              ;POSITION
0037  37     CPL A            ;COMPLEMENT THE INPUT FOR
                              ;COSINE LOOK UP TABLE
0038  53     ANL A,#          ;STRIP OFF BIT 7 AND DISCARD
0039  7F     01111111B
003A  E3     MOVP3 A,@A       ;BRING IN VALUE FROM LOOK
                              ;UP TABLE
003B  B6     JF 0             ;IF F0 IS SET DON'T
003C  3E     00111110B        ;COMPLEMENT
003D  37     CPL A
003E  37     CPL A
003F  AA     MOV R2,A
0040  FD     MOV A,R5
0041  A8     MOV R0,A         ;REG 0 CONTAINS 10000000
0042  FB     MOV A,R3
0043  37     CPL A
0044  90     MOVX @R0,A       ;OUTPUT VERTICAL(LEFT)
0045  FA     MOV A,R2
0046  37     CPL A
0047  18     INC R0
0048  90     MOVX @R0,A       ;OUTPUT HORZ. (LEFT)
0049  C8     DEC R0
004A  FB     MOV A,R3
004B  90     MOVX @R0,A       ;OUTPUT VERTICAL (RIGHT)
004C  18     INC R0
```

```
004D  FA     MOV A,R2
004E  90     MOVX @R0,A       ;OUTPUT HORZ. (RIGHT)
004F  FE     MOV A,R6         ;PUT 2 INTO ACC
0050  A9     MOV R1,A
0051  04     JMP
0052  16     00010110B
0053  FD     MOV A,R5
0054  A8     MOV R0,A         ;PUT 10000000 INTO REG 0
0055  90     MOVX @R0,A       ;OUTPUT ZERO VERTICAL (LEFT)
0056  18     INC R0
0057  23     MOV A,#
0058  FF     11111111B
0059  90     MOVX @R0,A       ;OUTPUT NEG HORZ (LEFT)
005A  C8     DEC R0
005B  FD     MOV A,R5
005C  90     MOVX @R0,A       ;OUTPUT ZERO VERTICAL (RIGHT)
005D  27     CLR A
005E  18     INC R0
005F  90     MOVX @R0,A       ;OUTPUT POS HORZ (RIGHT)
0060  04     JMP
0061  16     00010110B
```

```
0300  80    10000000B     ;LOOK UP TABLE
0301  81    10000001B
0302  83    10000011B
0303  84    10000100B
0304  86    10000110B
0305  87    10000111B
0306  89    10001001B
0307  8A    10001010B
0308  8C    10001100B
0309  8D    10001101B
030A  8F    10001111B
030B  91    10010001B
030C  92    10010010B
030D  94    10010100B
030E  95    10010101B
030F  97    10010111B
0310  98    10011000B
0311  9A    10011010B
0312  9B    10011011B
0313  9D    10011101B
0314  9E    10011110B
0315  A0    10100000B
0316  A1    10100001B
0317  A3    10100011B
0318  A4    10100100B
0319  A6    10100110B
031A  A7    10100111B
031B  A9    10101001B
031C  AA    10101010B
031D  AC    10101100B
031E  AD    10101101B
031F  AF    10101111B
0320  B0    10110000B
0321  B2    10110010B
0322  B3    10110011B
0323  B4    10110100B
0324  B6    10110110B
0325  B7    10110111B
0326  B9    10111001B
0327  BA    10111010B
0328  BB    10111011B
0329  BD    10111101B
032A  BE    10111110B
032B  BF    10111111B
032C  C1    11000001B
032D  C2    11000010B
032E  C3    11000011B
032F  C5    11000101B
0330  C6    11000110B
0331  C7    11000111B
```

0332	C9	11001001B
0333	CA	11001010B
0334	CB	11001011B
0335	CC	11001100B
0036	CE	11001110B
0337	CF	11001111B
0338	DO	11010000B
0339	D1	11010001B
033A	D2	11010010B
033B	D4	11000100B
033C	D5	11000101B
033D	D6	11000110B
033E	D7	11000111B
033F	D8	11001000B
0340	D9	11011001B
0341	DA	11011010B
0342	DB	11011011B
0343	DD	11011101B
0344	DE	11011110B
0345	DF	11011111B
0346	E0	11100000B
0347	E1	11100001B
0348	E2	11100010B
0349	E3	11100011B
034A	E4	11100100B
034B	E5	11100101B
034C	E6	11100110B
034D	E6	11100110B
034E	E7	11100111B
034F	E8	11101000B
0350	D9	11101001B
0351	EA	11101010B
0352	EB	11101011B
0353	EC	11101100B
0354	EC	11101100B
0355	ED	11101101B
0356	EE	11101110B
0357	EF	11101111B
0358	F0	11110000B
0359	F0	11110000B
035A	F1	11110001B
035B	F2	11110010B
035C	F2	11110010B
035D	F3	11110011B
035E	F4	11110100B
035F	F4	11110100B
0360	F5	11110101B
0361	F5	11110101B
0362	F6	11110110B
0363	F7	11110111B

0364	F7	11110111B
0365	F8	11111000B
0366	F8	11111000B
0367	F9	11111001B
0368	F9	11111001B
0369	F9	11111001B
036A	FA	11111010B
036B	FA	11111010B
036C	FB	11111011B
036D	FB	11111011B
036E	FB	11111011B
036F	FC	11111100B
0370	FC	11111100B
0371	FC	11111100B
0372	FD	11111101B
0373	FD	11111101B
0374	FD	11111101B
0375	FD	11111101B
0376	FE	11111110B
0377	FE	11111110B
0378	FE	11111110B
0379	FE	11111110B
037A	FE	11111110B
037B	FF	11111111B
037C	FF	11111111B
037D	FF	11111111B
037E	FF	11111111B
037F	FF	11111111B

LIST OF REFERENCES

1. Aldrich, J.H., A Simulator Evaluation of Pilot Response to an Aircraft Cockpit Spin Indicator System, MS Thesis, Naval Postgraduate School, California, 1979.

2. Lucchesi, M.A., Longitudinal Equations of Motion verses Spring Mass Damper System, paper presented for course AE3815, spring, 1980.

3. Burr-Brown General Catalog, p. 5-95-5-102, 1979.

4. Intel, MCS-48 Microcomputer User's Manual, 1976.

5. Intel, Application Techniques for the MCS-48 Family, p. 3-8, 1977.

6. National, Digital Integrated Circuits, p. 1-5, 1-75, 1-87, 1974.

7. Intel, Prompt 80 Microcomputer User's Manual, 1976.

INITIAL DISTRIBUTION LIST

		No. copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93940	2
3.	Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, CA 93940	1
4.	Professor D. M. Layton Code 67Ln Department of Aeronautics Naval Postgraduate School Monterey, CA 93940	5
5.	Lt Marc A. Lucchesi, USN VA-128 Naval Air Station, Whidbey Island Oak Harbor, WA 98277	2

191473

Thesis
L8913
c.1

Lucchesi
Microprocessor generated vertical gyrohorizon instrument for the blue bird simulator.