MOTOROLA
Semiconductor Products Inc.

AN942
Application Note

MC68704P2
8-BIT EPROM MICROCOMPUTER PROGRAMMING MODULE

INTRODUCTION

This application note provides programming information for the MC68704P2 8-bit EPROM Microcomputer Unit (MCU). This information enables the user to construct and operate the programming module which is used to program the EPROM MCU device. Figure 1 illustrates the programming module schematic diagram. All that is required to program the MC68704P2 EPROM MCU is the programming module, 2K EPROM, and a +5 volt dc power supply.

The MC68704P2 EPROM MCU programming module was designed to utilize either MC68705P3 or MC6805P2 MCUs. Four jumper pads (J1A, J1B, J2, and J3) are provided on the module to facilitate MCU interconnection as follows:

a. MC68705P3 MCU—Install insulated jumper wire between J1A and J3.
b. MC6805P2 MCU—Install insulated jumper wire between J1B and J2.

Programming module operation is identical when using either the MC68705P3 or MC6805P2 MCU.

PROGRAMMING OPERATING MODES

The programming module is designed to perform in four modes of operation. These modes of operation are as follows:

a. Zero check
b. Program
c. Verify
d. Test

ZERO CHECK

The zero check mode of operation allows the user to determine if the EPROM MCU is erased (blank). The erased value is $00 (hexadecimal). Upon completion of the zero check operation, the user is notified of the results via the module zero check LED (labeled "Z"). This mode of operation should be performed prior to any programming operation.

PROGRAM

The program mode of operation will store the data code located in the 2K EPROM into the EPROM MCU. Each byte programmed is also verified against the 2K EPROM contents. Upon completion of the program operation, the user is notified of the results via the module program LED (labeled "P").

VERIFY

The verify mode of operation compares the data code stored in the EPROM MCU against the 2K EPROM contents. Upon completion of the verify operation, the user is notified of the results via the module verify LED (labeled "V").

TEST

The test mode of operation tests the hardware operation of the serial to parallel conversion circuits (74LS164s), octal transparent latches (74LS374s), and the octal buffer (74LS241) which represents the majority of the interconnecting circuitry between the module MC68705P3 MCU and the MC68704P2 MCU programming socket.

PROGRAMMING MODE SELECTION

The programming module operating modes are selected by the placement of the mode select switches S1 and S2. The functions of these switches are as follows:

S1	S2	MODE
OFF	OFF	Test
OFF	ON	Zero check
ON	OFF	Program
ON	ON	Verify

After selecting the initial mode, reconfiguration of switches S1 and S2 (to any mode) followed by the placement of the RESET switch S3 to the ON and OFF positions will initiate a new mode of operation.

PRELIMINARY PROCEDURES

Prior to performing any programming operations, the following steps are performed:

1. Place mode select switches S1, S2, and POWER switch S4 to the OFF positions, and the RESET switch S3 to the ON position.
2. Connect +5 volt dc power supply to the programming module terminals labeled +5 and GND.
3. Install preprogrammed 2K EPROM device into the Zero Insertion Force (ZIF) 24-pin socket U10. Code stored in preprogrammed device is as follows:

Address	Contents
$012 — $017	Option bytes
$018 — $05F	User data space
$400 — $7F8	User program
$7FC — $7FD	IRQ vector
$7FE — $7FF	Restart vector

4. The EPROM MCU device should be erased by the exposure of a high-intensity ultraviolet (UV) light with a wavelength of 2537 Angstrom (Å). The recommended dose (UV intensity × exposure time) is 15 Ws/cm^2. UV lamps should be used without shortwave filters, and the EPROM MCU device positioned about one inch from the UV lamps.

OPERATING PROCEDURES

1. Insert erased MC68704P2 EPROM MCU into the Zero Insertion Force (ZIF) 28-pin programming socket U14.
2. Place POWER switch S4 to the ON position. Place RESET switch S3 to the ON position, and then to the OFF position. When S3 is placed to the OFF position, the hardware test of the programming module is initiated.
   a. If all LEDs remain illuminated, the module is operating correctly and the user proceeds to step 3.
   b. If all three LEDs flash for approximately 4 seconds, a problem exists with the programming module. Module POWER switch is placed to the OFF position, and the EPROM MCU device is removed from the programming socket. Upon fixing the module malfunction, the user proceeds to step 1.
3. Place mode select switch S2 to the ON position. Place RESET switch S3 to the ON position, and then to the OFF position. This step initiates the zero check of the EPROM MCU.
   a. If EPROM MCU is completely erased, the zero check LED (labeled "Z") will illuminate continuously, and the user proceeds to step 4.
   b. If EPROM MCU is not completely erased, the zero check LED will flash. Module POWER switch is placed to the OFF position, and the EPROM MCU device is removed from the programming socket and re-erased. Upon completion of the EPROM MCU erasing, the user places switch S2 to the OFF position and proceeds to step 1.
4. Place mode select switches S1 and S2 to the ON and OFF positions, respectively. Place RESET switch S3 to the ON position, and then to the OFF position. This step initiates the programming of the EPROM MCU. The EPROM MCU is programmed from the pre-programmed 2K EPROM residing in socket U10. EPROM MCU programming takes approximately two minutes to be completed.
   a. If no errors are encountered during the programming sequence, the program LED (labeled "P") will illuminate continuously, and the user proceeds to step 5.
   b. If errors are encountered, the program LED will flash. Module POWER switch is placed to the OFF position, and the EPROM MCU device is removed from the programming socket and re-erased. Upon completion of the EPROM MCU erasing, the user places switch S1 to the OFF position and proceeds to step 1.
5. Place mode select switch S2 to the ON position. Place RESET switch S3 to the ON position, and then to the OFF position. This step verifies the EPROM MCU programming operation just performed. The EPROM MCU contents is compared against the code stored in the 2K EPROM device. Verification process takes approximately 4 seconds.
   a. If a valid comparison is made, the verify LED (labeled "V") will illuminate continuously. The programming operation is now complete. Module POWER switch is placed to the OFF position, and the EPROM MCU device is removed from the programming socket.
   b. If a mismatch is detected, the verify LED will flash. Module POWER switch is placed to the OFF position, and the EPROM MCU device is removed from the programming socket and re-erased. Upon completion of the EPROM MCU erasing, the user places both switches S1 and S2 to the OFF position and proceeds to step 1.

PROGRAMMING MODULE CONSTRUCTION

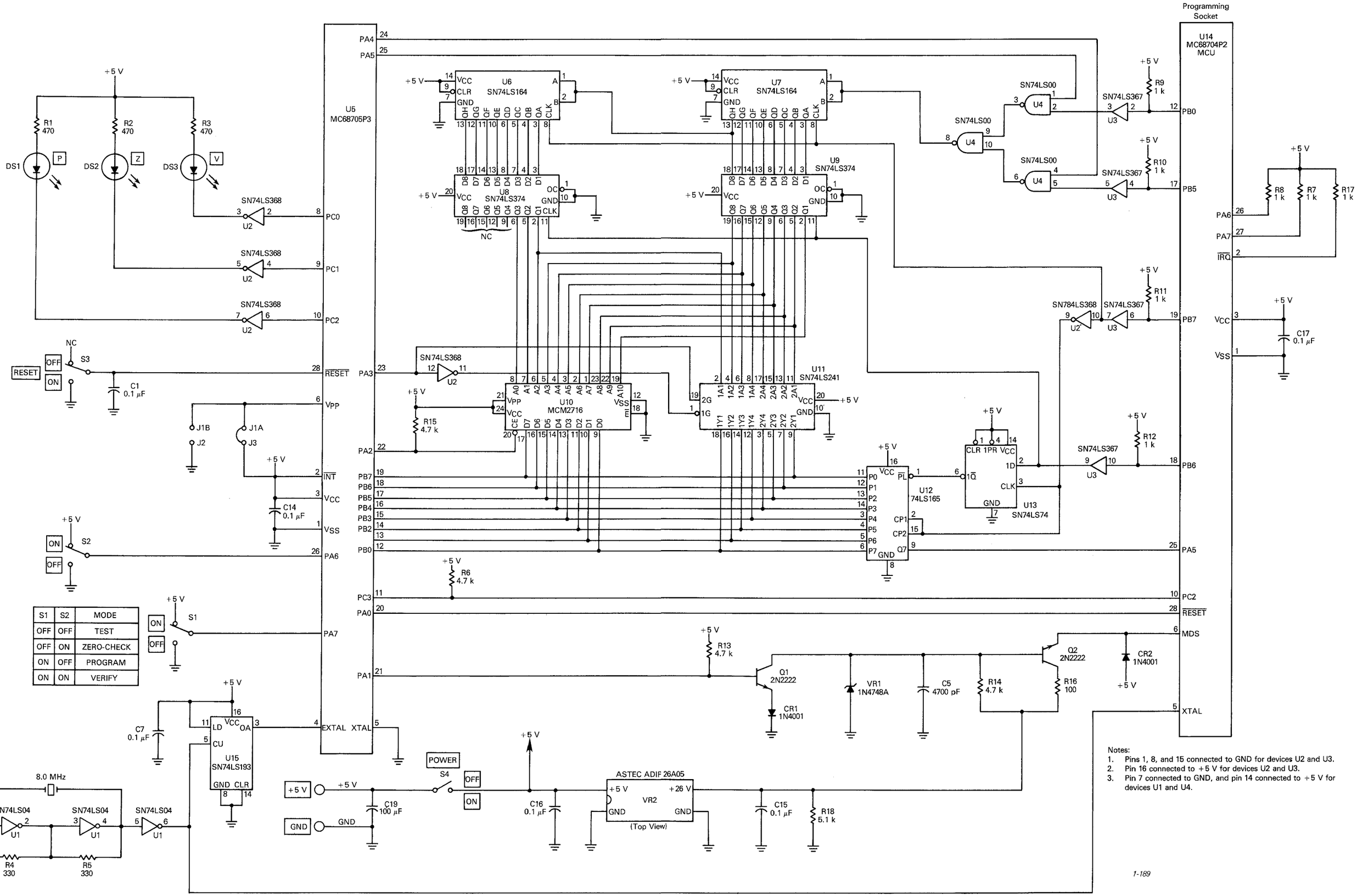
The programming module is a double-sided Printed Wiring Board (PWB) with plated through holes. Table 1 lists the parts list, and Figure 2 illustrates the part locations for the programming module. Figures 3 and 4 provide the top and bottom PWB printed wiring artwork layout diagrams, respectively.

TABLE 1 — Programming Module Parts List

C1-C4, C6-C18	Capacitor, 0.1 μF @ 50 V, rf-bypass
C5	Capacitor, 4700 pF @ 100 V, ceramic/mylar
C19	Capacitor, 100 μF @ 35 V, electrolytic
CR1, CR2	Diode, 1N4001
DS1-DS3	LEDs, red
Q1, Q2	Transistor, 2N2222
R1-R3	Resistor, 470 Ω, carbon, 5%, ¼ W
R4, R5	Resistor, 330 Ω, carbon, 5%, ¼ W
R6, R13-R15	Resistor, 4.7 kΩ, carbon, 5%, ¼ W
R7-R12, R17	Resistor, 1.0 kΩ, carbon, 5%, ¼ W
R16	Resistor, 100 Ω, carbon, 5%, ¼ W
R18	Resistor, 5.1 kΩ, carbon, 5%, ¼ W
S1-S4	Switch, SPDT, Amer # ST1-1, PCB mtg.
U1	I.C., SN74LS04
U2	I.C., SN74LS368
U3	I.C., SN74LS367
U4	I.C., SN74LS00
U5	I.C., MC68705P3 (programmed EPROM MCU, see note below)
U6, U7	I.C., SN74LS164
U8, U9	I.C., SN74LS374
U10	I.C., MCM2716 (programmed 2K EPROM)
	I.C. Socket, 24-pin, ZIF, Textool # 224-3344
U11	I.C., SN74LS241
U12	I.C., SN74LS165
U13	I.C., SN74LS74
U14	I.C. Socket, programming, 28-pin, ZIF, Textool # 228-3345
U15	I.C., SN74LS193
VR1	Diode, Zener, 1N4748A (22 V, 5%)
VR2	+5 V to +26 V DC Converter, ASTEC/ADIP#26A05
Y1	Crystal, 8.0 MHz

NOTE: Software listing for the programmed EPROM MCU is provided at the end of this application note.

1-188



S1	S2	MODE
OFF	OFF	TEST
OFF	ON	ZERO-CHECK
ON	OFF	PROGRAM
ON	ON	VERIFY

Notes:
1. Pins 1, 8, and 15 connected to GND for devices U2 and U3.
2. Pin 16 connected to +5 V for devices U2 and U3.
3. Pin 7 connected to GND, and pin 14 connected to +5 V for devices U1 and U4.

1-189

FIGURE 1 — Programming Module Schematic Diagram

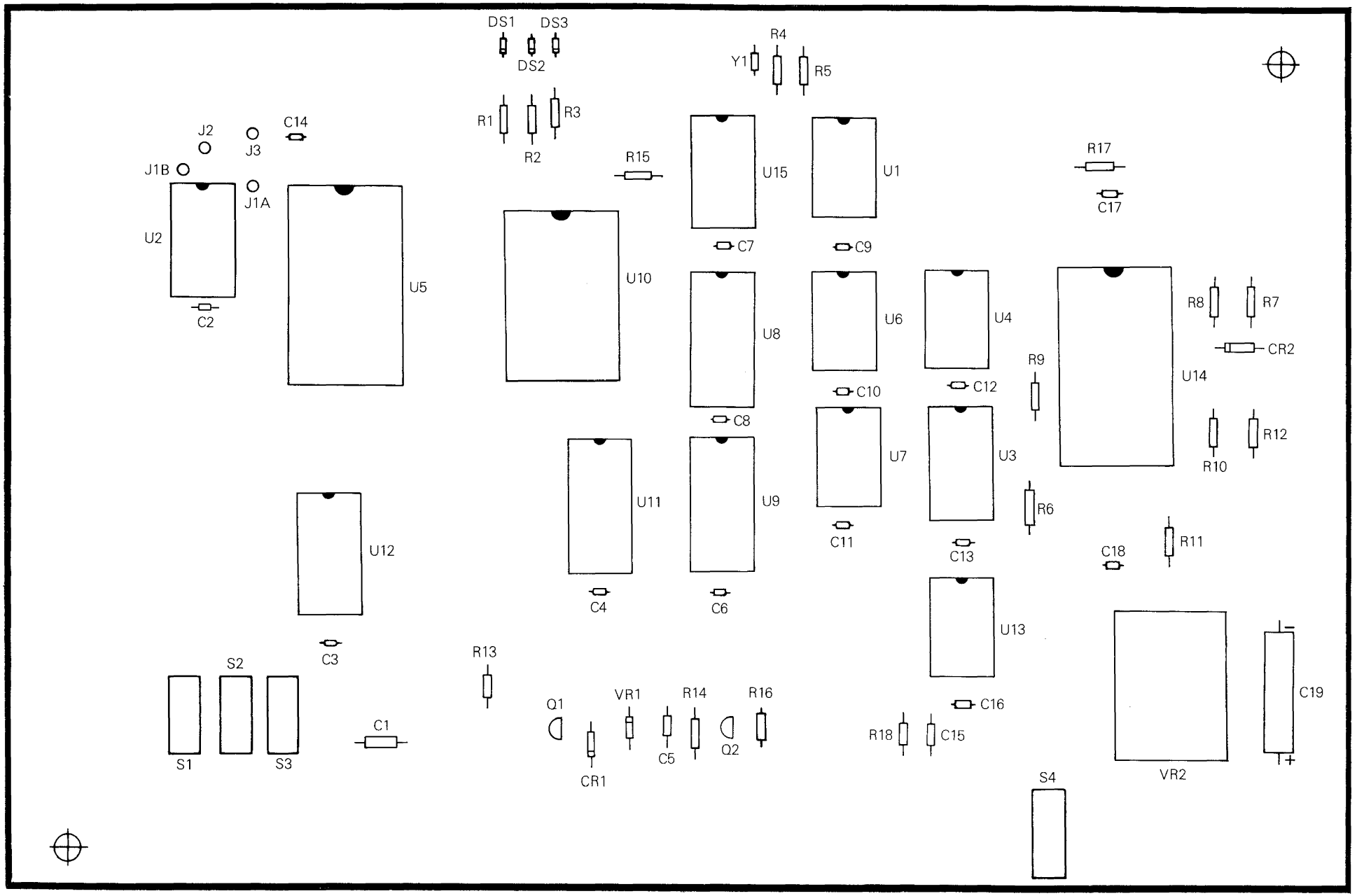


FIGURE 2 — Programming Module Parts Location Diagram

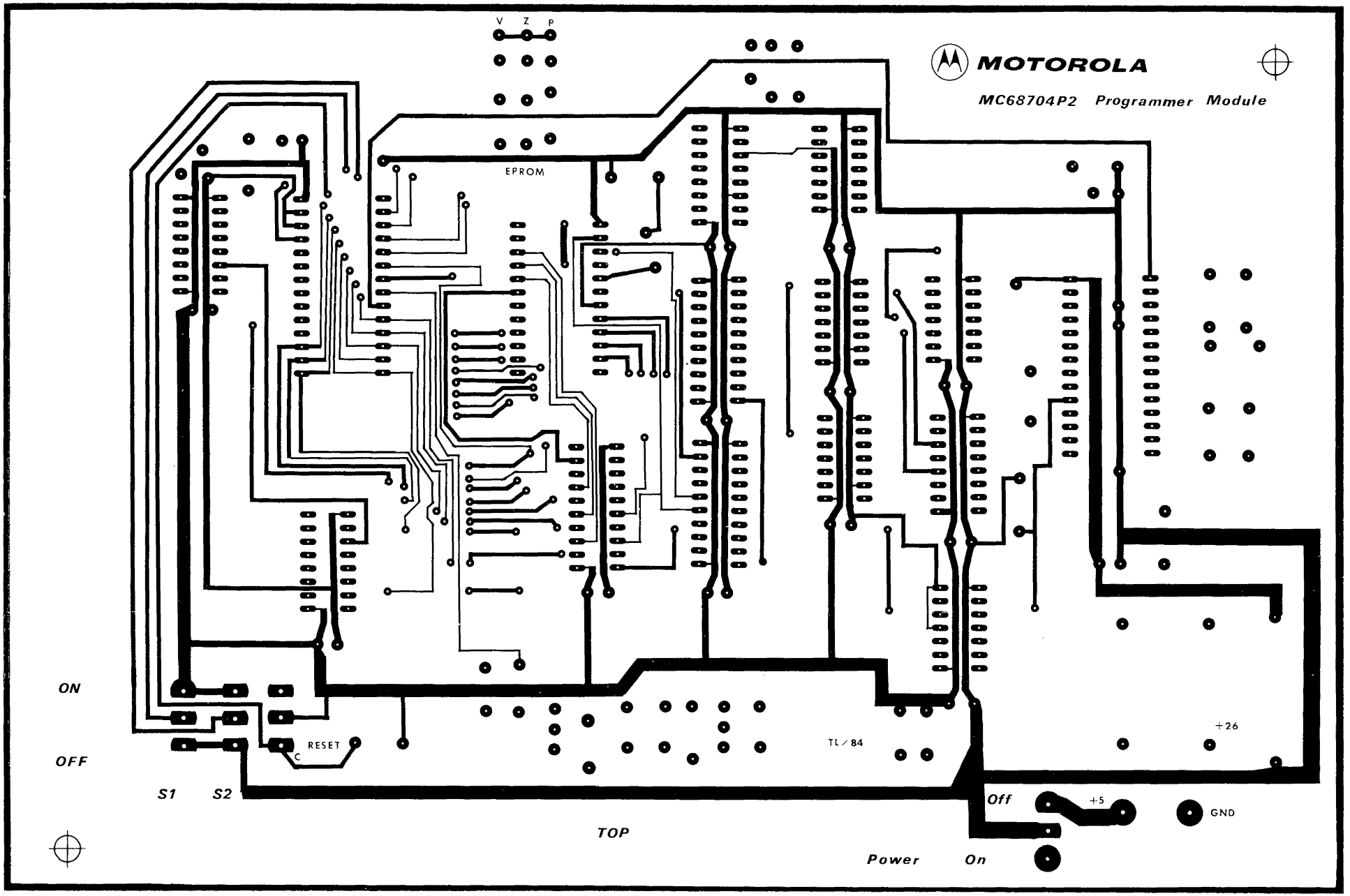


FIGURE 3 — PWB (Top) Printed Wiring Artwork Layout Diagram

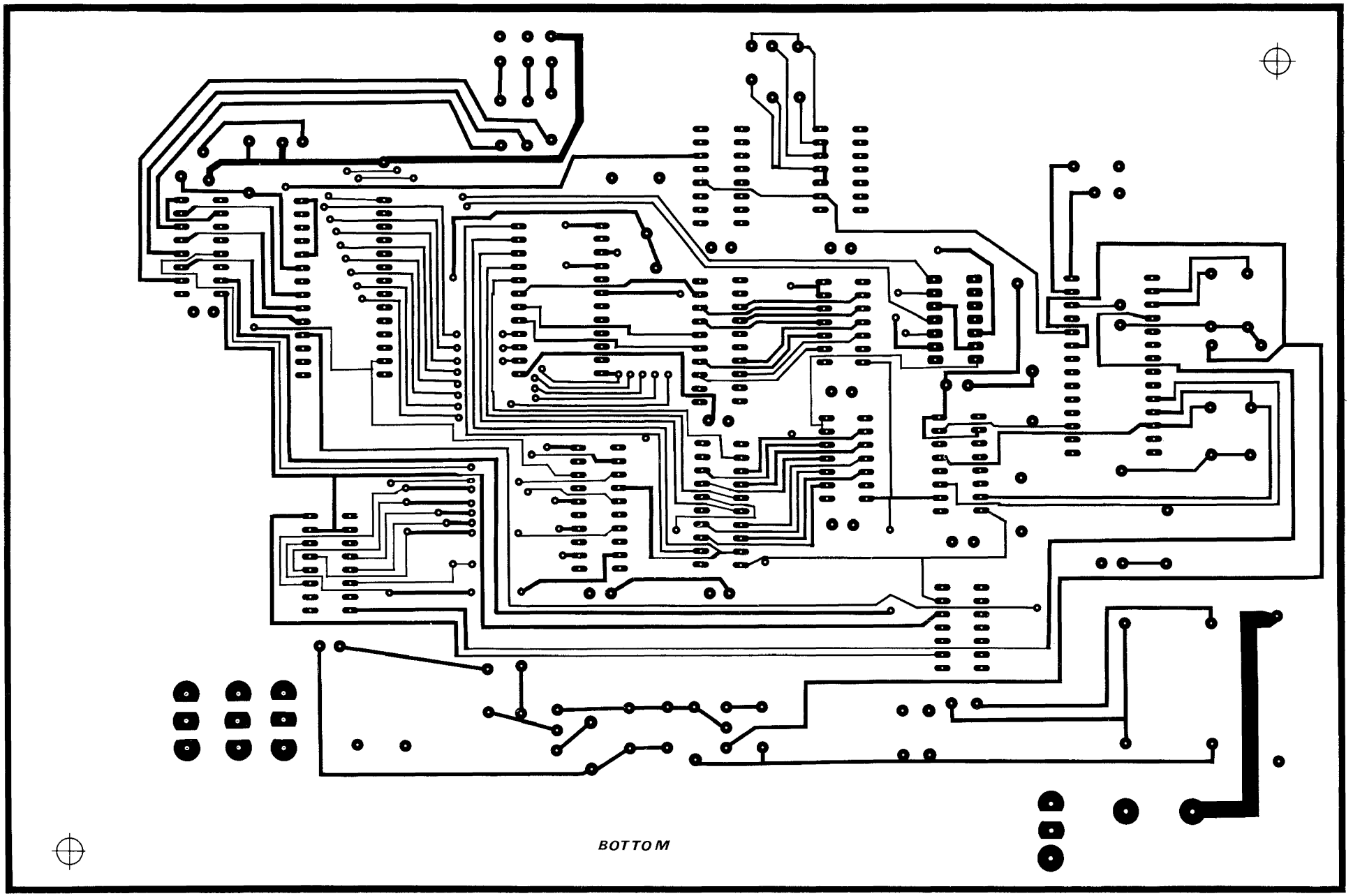


FIGURE 4 — PWB (Bottom) Printed Wiring Artwork Layout Diagram

```
00001                           NAM     PROGRAMMER
00002                           OPT     LLE=120
00003
00004                     ********************************************************************************
00005                     *                                                                              *
00006                     *    MC68704P2 PROGRAMMER MODULE SOFTWARE                                      *
00007                     *    THIS SOFTWARE PERMITS A USER TO ACTIVATE THE                              *
00008                     *    FOLLOWING FUNCTIONS:                                                      *
00009                     *                                                                              *
00010                     *     i)  HARDWARE TEST                                                        *
00011                     *    ii)  ZERO_CHECK      ( CHECKS FOR AN ERASED 704P2 )                       *
00012                     *   iii)  PROGRAM         ( STORES CODE 2716---> 704P2 )                       *
00013                     *    iv)  VERIFY          ( COMPARES 2716 CONTENTS AGAINST 7042P2'S CONTENTS )*
00014                     *                                                                              *
00015                     ********************************************************************************
00016
00017
00018           0000      A PORTA   EQU     0         PORTA   DATA           REGISTER
00019           0004      A DDRA    EQU     4         PORTA   DATA DIRECTION REGISTER
00020           0001      A PORTB   EQU     1         PORTB   DATA           REGISTER
00021           0005      A DDRB    EQU     5         PORTB   DATA DIRECTION REGISTER
00022           0002      A PORTC   EQU     2         PORTC   DATA           REGISTER
00023           0006      A DDRC    EQU     6         PORTC   DATA DIRECTION REGISTER
00024           0008      A TDATA   EQU     8
00025
00026
00027                       * THE FOLLOWING EQUATES ARE FOR PORTA
00028           0006      A SWITCH  EQU     6
00029           0005      A PSTRM   EQU     5
00030           0004      A VSTRM   EQU     4
00031           0003      A RDATA   EQU     3
00032           0002      A EPROM   EQU     2         ENABLES THE EPROM
00033           0001      A BURN    EQU     1         CONTROLS THE ANALOG CIRCUITRY, WHICH ENABLES THE
00034                       *                       VPP PULSE TO BE APPLIED TO THE MDS PIN ON THE
00035                       *                       704P2 MDS LINE.
00036                       *                       PORTA.1 = "0"       "BURNER"  ON
00037                       *                       PORTA.1 = "1"       "BURNER"  OFF
00038           0000      A RESET   EQU     0         HANDLES THE 704P2 RESET LINE
00039
00040
00041                       * THE FOLLOWING EQUATES ARE FOR PORTC
00042
00043           0003      A HALT    EQU     3
00044           0000      A VERFD   EQU     0
00045
00046           0000      A MPT     EQU     0         DEFAULT ERASED STATE OF EPROM
00047           0008      A TIMER   EQU     8         TIMER DATA REGISTER
00048           0009      A TSCR    EQU     9         TIMER STATUS CONTROL REGISTER
00049           000B      A PCR     EQU     $B
00050           0784      A MOR     EQU     $784
00051           0007      A TIR     EQU     7         TIMER INTERRUPT BIT
00052           0001      A KEY     EQU     1
00053
00054A 0010                        ORG     $10
00055
00056A 0010     0001      A FLAG    RMB     1
00057A 0011     0001      A PULSES  RMB     1
00058A 0012     0002      A BYTES   RMB     2
00059A 0014     0001      A EXPECT  RMB     1
00060A 0015     0001      A TEMP1   RMB     1
00061A 0016     0001      A TEMP2   RMB     1
00062A 0017     0001      A TEMP3   RMB     1
00063A 0018     0001      A TEMP4   RMB     1
00064A 0019     0001      A SELECT  RMB     1
00065A 001A     0001      A DATA1   RMB     1
00066A 001B     0001      A PHIGH   RMB     1
00067A 001C     0001      A PLOW    RMB     1
00068A 001D     0001      A SHIGH   RMB     1
00069A 001E     0001      A NOVERF  RMB     1
00070A 001F     0001      A SLOW    RMB     1
```

```
00071
00072
00073                    ************************************************
00074                    *                                              *
00075                    * THE FOLLOWING CODE WILL PERMIT THE CODE STORED *
00076                    * IN THE MCM2716 TO BE STORED INTO THE 68704P2   *
00077                    *                                              *
00078                    * BRIEF PIN-OUT FOLLOWS:                       *
00079                    *                                              *
00080                    *  PORT A:                                     *
00081                    *           BIT #1: HANDLES THE RESET LINE     *
00082                    *           BIT #2: HANDLES THE MDS LINE ON THE *
00083                    *                   68704P2.                   *
00084                    *           BIT #3: ENABLES THE MCM2716 EPROM  *
00085                    *           BIT #4: ENABLES THE DATA BUFFER    *
00086                    *           BIT #5: ENABLE THE VERIFY DATA INPUT *
00087                    *                   FROM THE 704P2.            *
00088                    *           BIT #6: ENABLE THE PC DATA INPUT   *
00089                    *                   FROM THE 704P2             *
00090                    *           BIT #7: PROG/VERIFY SWITCH         *
00091                    *                   BIT = "1"..PROGRAM         *
00092                    *                   BIT = "0"..VERIFY          *
00093                    *           BIT #8: ZERO_CHECK SWITCH          *
00094                    *                   BIT = "1" ZERO_CHECK       *
00095                    *                                              *
00096                    *  THE FOLLOWING CODE IS USED TO REPRESENT     *
00097                    *  DIFFERENT STATUS OF THE PROGRAMMER;         *
00098                    *                                              *
00099                    *            PA7  PA6   STATUS                 *
00100                    *             0    0    WAIT                   *
00101                    *             0    1    ZERO_CHECK             *
00102                    *             1    0    PROGRAM                *
00103                    *             1    1    VERIFY                 *
00104                    *                                              *
00105                    *                                              *
00106                    *                                              *
00107                    *  PORT B:                                     *
00108                    *           USED ONLY FOR DATA INPUT           *
00109                    *                                              *
00110                    *  PORT C:                                     *
00111                    *           BIT #1: VERIFY  LED                *
00112                    *           BIT #2: ERASE   LED                *
00113                    *           BIT #3: PROGRAM LED                *
00114                    *           BIT #4: HANDLES THE HALT LINE ON THE *
00115                    *                   68704P2                    *
00116                    *                                              *
00117                    *                                              *
00118                    ************************************************
00119
00120A 0100                        ORG    $100
00121A 0100 CD 03D2      A START   JSR    LED
00122A 0103 CD 024D      A         JSR    DISCON   SET HARDWARE TO A STANDBY MODE
00123A 0106 CD 03C9      A         JSR    INIT     CONFIGURE PORTA, PLACE THE 704P2 RESET LINE LOW
00124A 0109 CD 0366      A         JSR    WAIT1    HOLD RESET LINE LOW
00125A 010C CD 0366      A         JSR    WAIT1
00126A 010F CD 0366      A         JSR    WAIT1
00127A 0112 10 00        A         BSET   RESET,PORTA SET THE 704P2 RESET LINE
00128                      *                         HIGH AGAIN
00129A 0114 B6 00        A START1  LDA    PORTA    READ CURRENT PORTA DATA
00130A 0116 A4 C0        A         AND    #%11000000 EXTRACT SWITCH DATA
00131A 0118 27 0E      0128        BEQ    HDWARE   HARDWARE TEST?
00132A 011A A1 40        A         CMPA   #%01000000 ERASE STATE?
00133A 011C 26 03      0121        BNE    START2   YES
00134A 011E CC 01A1      A         JMP    ZEROCK
00135A 0121 A1 80        A START2  CMPA   #%10000000 PROGRAM STATE?
00136A 0123 26 54      0179        BNE    VERIFY   IF NOT HARDWARE,ERASE OR PROGRAM...THEN VERIFY
00137A 0125 CC 01C7      A         JMP    PROGRM
```

```
00138
00139                    ************************************************************
00140                    *                                                          *
00141                    * HARDWARE TEST:                                           *
00142                    * THE FOLLOWING SECTION OF CODE WILL TEST THE OPERATION    *
00143                    * OF BOTH 74LS164'S, 74LS374'S AND THE TRI-STATE BUFFER    *
00144                    * 74LS241. IF THE HARWDARE PASSES THE THREE LED'S WILL     *
00145                    * BE TURNED ON. OTHERWISE, IF IT FAILS, THE LED'S WILL     *
00146                    * FLASH FOR   SECONDS.                                     *
00147                    *                                                          *
00148                    ************************************************************
00149
00150A 0128 A6 1F       A HDWARE LDA    #%00011111 VERIFY OFF, PC OUTPUT ON
00151                    *                          TRI-STATE BUFFER ON, EPROM
00152                    *                          OUPUTS DISABLED
00153A 012A B7 00       A        STA    PORTA
00154A 012C CD 0366     A        JSR    WAIT1     WAIT UNTIL DATA IS STABLE
00155A 012F CD 0366     A        JSR    WAIT1
00156A 0132 CD 0366     A        JSR    WAIT1
00157A 0135 A6 01       A        LDA    #$01
00158A 0137 B7 15       A        STA    TEMP1     INITIALIZE SOFTWARE P.C.
00159A 0139 CD 01FC     A        JSR    INCRM     INCREMENT  HARDWARE P.C.
00160A 013C B6 01       A HTLOOP LDA    PORTB     READ IN HARWDARE P.C.
00161A 013E B1 15       A        CMPA   TEMP1     CHECK FOR HARDWARE & SOFTWARE P.C. MATCH
00162A 0140 27 03     0145       BEQ    TLOOPX
00163A 0142 CC 0158     A        JMP    DANGER    HARDWARE FAILURE
00164A 0145 3C 15       A TLOOPX INC    TEMP1     INCREMENT SOFTWARE P.C.
00165A 0147 CD 01FC     A        JSR    INCRM     INCREMENT HARDWARE P.C.
00166A 014A B6 15       A        LDA    TEMP1     READ      SOFTWARE P.C.
00167A 014C A1 3C       A        CMPA   #$3C      HAS TEST FINISHED?
00168A 014E 27 02     0152       BEQ    COMPLT    IF SO, FLASH LED'S
00169A 0150 20 EA     013C       BRA    HTLOOP    OTHERWISE, KEEP TESTING
00170A 0152 A6 0F       A COMPLT LDA    #$0F
00171A 0154 B7 02       A        STA    PORTC     TURN ON LED'S
00172A 0156 20 1C     0174       BRA    HTEND     TEST FINISHED
00173A 0158 A6 08       A DANGER LDA    #$08      NUMBER OF LED FLASHES
00174A 015A B7 15       A        STA    TEMP1
00175A 015C B6 15       A TLOOP  LDA    TEMP1     FLASH LED'S A TOTAL OF 8 TIMES.
00176A 015E A1 00       A        CMPA   #$00
00177A 0160 27 12     0174       BEQ    HTEND
00178A 0162 A6 0F       A        LDA    #$0F      TURN LED'S ON
00179A 0164 B7 02       A        STA    PORTC
00180A 0166 CD 03AC     A        JSR    XBLINK
00181A 0169 A6 08       A        LDA    #$08      TURN LED'S OFF
00182A 016B B7 02       A        STA    PORTC
00183A 016D CD 03AC     A        JSR    XBLINK
00184A 0170 3A 15       A        DEC    TEMP1     DECREMENT FLASH_COUNT
00185A 0172 20 E8     015C       BRA    TLOOP
00186A 0174 CD 024D     A HTEND  JSR    DISCON    DISCONNECT HARDWARE
00187A 0177 20 FE     0177 HTS   BRA    HTS       END OF TEST, LOOP ENDLESSLY
00188
00189                    ****************************************************
00190                    *                                                  *
00191                    * VERIFY:                                          *
00192                    * THE VERIFY CODE FOLLOWS.                         *
00193                    * IT RECEIVES THE BYTE OF DATA JUST PROGRAMMED     *
00194                    * INTO THE 68704P2 AND COMPARES IT WITH A BYTE     *
00195                    * THAT IS LOCATED IN THE MCM2716.                  *
00196                    *                                                  *
00197                    * SHIGH....HIGH ORDER BYTE OF P.C.                 *
00198                    * SLOW.....LOW ORDER BYTE OF P.C.                  *
00199                    *                                                  *
00200                    ****************************************************
00201
00202
00203A 0179 CD 0229      A VERIFY JSR    SKIP0     SKIP $00---$12
00204A 017C 3F 1D       A        CLR    SHIGH     CLEAR HIGH ORDER BYTE OF P.C.
00205A 017E A6 17       A        LDA    #$17
00206A 0180 B7 1F       A        STA    SLOW      SET LOW ORDER BYTE OF P.C. TO $17
00207A 0182 CD 0372     A        JSR    SKIP      INCREMENT THE 704P2 PROGRAM COUNTER, SO THAT IT
00208                    *                       POINTS AT $17
00209A 0185 A6 01       A        LDA    #$01
00210A 0187 B7 19       A        STA    SELECT    SELECT VERIFY MODE FOR THE HANDLE ROUTINE
00211A 0189 CD 0252     A        JSR    HNDLA     VERIFY 704P2 CONTENTS AGAINST
00212                    *                       '2716 EPROM CONTENTS.
00213                    *                       [ DATA SPACE EPROM ]
00214                    *                       MEMORY LOCATION: $20--$5F
00215A 018C CD 0237     A        JSR    SKIPB     SKIP BYTES $60--$0BFF
00216A 018F CD 025D     A        JSR    HNDLB     VERIFY 704P2 CONTENTS AGAINST
00217                    *                       '2716 EPROM CONTENTS.
00218                    *                       [ PROGRAM SPACE EPROM ]
00219A 0192 CD 0241     A        JSR    SKIPC     SKIP BYTES $FF8---$FFB
00220A 0195 CD 0269     A        JSR    HNDLC     VERIFY 704P2 CONTENTS AGAINST '2716 EPROM CONTENTS
00221                    *                       [ IRQ/ & RESET/ ADDRESSES ]
00222A 0198 CD 024D     A        JSR    DISCON    DE-ACTIVATE EXTERNAL HARDWARE
00223A 019B A6 09       A        LDA    #%00001001
00224A 019D B7 02       A        STA    PORTC     TURN ON VERIFIED LED
00225A 019F 20 FE     019F VEREND BRA    VEREND
```

```
00226
00227
00228                    ************************************************
00229                    *                                              *
00230                    *  ZERO_CHECK:                                 *
00231                    *  THE ZERO_CHECK CODE FOLLOWS.                *
00232                    *  THE PURPOSE OF THIS CODE IS TO DETERMINE    *
00233                    *  WHETHER OR NOT THE 704P2 EPROM IS ERASED.   *
00234                    *  ERASED CONDITION FOR ANY GIVEN LOCATION;    *
00235                    *     "00" HEX.                                *
00236                    *                                              *
00237                    ************************************************
00238
00239
00240A 01A1 CD 0229     A ZEROCK JSR    SKIP0    SKIP BYTES $00--$12
00241A 01A4 3F 19       A        CLR    SELECT
00242A 01A6 3F 1D       A        CLR    SHIGH
00243A 01A8 A6 1F       A        LDA    #$1F
00244A 01AA B7 1F       A        STA    SLOW
00245A 01AC CD 027E     A        JSR    HANDLE   ZERO_CHECK OPTION BYTES
00246A 01AF CD 0252     A        JSR    HNDLA    ZERO_CHECK DATA   BYTES
00247A 01B2 CD 0237     A        JSR    SKIPB    SKIP BYTES $60--$0BFF
00248A 01B5 CD 025D     A        JSR    HNDLB    ZERO_CHECK PROG.  BYTES
00249A 01B8 CD 0241     A        JSR    SKIPC    SKIP BYTES $0FF8--$0FFB
00250A 01BB CD 0269     A        JSR    HNDLC    CHECK ERASED IRQ/,
00251                    *                        RESET/ VECTORS
00252A 01BE CD 024D     A        JSR    DISCON
00253A 01C1 A6 0A       A        LDA    #%00001010 TURN ON ERASED LED
00254A 01C3 B7 02       A        STA    PORTC
00255A 01C5 20 FE    01C5 ZEROK  BRA    ZEROK
00256
00257                    ************************************************
00258                    *                                              *
00259                    * PROGRAM MODE:                                *
00260                    * THE PROGRAM CODE FOLLOWS                     *
00261                    * THE PURPOSE OF THIS CODE IS TO "BURN" DATA   *
00262                    * INTO THE 68704P2. THE DATA IS EXTRACTED      *
00263                    * FROM A MCM2716.                              *
00264                    *                                              *
00265                    ************************************************
00266
00267A 01C7 CD 0229     A PROGRM JSR    SKIP0    SKIP BYTES $00--$12
00268A 01CA 3F 1D       A        CLR    SHIGH
00269A 01CC A6 16       A        LDA    #$16
00270A 01CE B7 1F       A        STA    SLOW
00271A 01D0 A6 FF       A        LDA    #$0FF
00272A 01D2 B7 1E       A        STA    NOVERF   INHIBIT VERIFY AFTER BURN
00273A 01D4 A6 02       A        LDA    #$02
00274A 01D6 B7 19       A        STA    SELECT   PROGRAM THE DATA SPACE
00275A 01D8 CD 027E     A        JSR    HANDLE   BURN LOCATIONS $12---$1E
00276A 01DB A6 1F       A        LDA    #$1F
00277A 01DD B7 1F       A        STA    SLOW
00278A 01DF 3F 1E       A        CLR    NOVERF   ACTIVATE VERIFY AFTER BURN
00279A 01E1 CD 027E     A        JSR    HANDLE   BURN & VERIFY LOCATION $1F
00280A 01E4 CD 0252     A        JSR    HNDLA    EPROM CONTENTS
00281A 01E7 CD 0237     A        JSR    SKIPB    SKIP BYTES $60--$0BFF
00282A 01EA CD 025D     A        JSR    HNDLB    EPROM
00283A 01ED CD 0241     A        JSR    SKIPC    SKIP BYTES $0FF8--$0FFB
00284                    *                    VECTORS
00285A 01F0 CD 0269     A        JSR    HNDLC
00286A 01F3 CD 024D     A        JSR    DISCON   DE-ACTIVATE EXTERNAL
00287                    *                    HARDWARE
00288A 01F6 A6 0C       A        LDA    #$0C
00289A 01F8 B7 02       A        STA    PORTC    TURN ON PROGRAM LED
00290A 01FA 20 FE    01FA PROGED BRA    PROGED
```

```
00291
00292
00293                    ********************************************
00294                    *                                          *
00295                    * ALL SUB-ROUTINES USED IN THE MAIN PROGRAM *
00296                    * WILL FOLLOW.                             *
00297                    *                                          *
00298                    ********************************************
00299
00300                    ********************************************
00301                    *                                          *
00302                    *  INCRM SUB-ROUTINE:                      *
00303                    *  THIS SUB-ROUTINE INCREMENTS THE 704P2   *
00304                    *  PROGRAM COUNTER BY FOUR COUNTS. IT IS   *
00305                    *  USED IN THE HARDWARE TEST MODE.         *
00306                    *                                          *
00307                    ********************************************
00308
00309A 01FC 17 02       A INCRM  BCLR   HALT,PORTC
00310A 01FE 16 02       A        BSET   HALT,PORTC
00311A 0200 9D                    NOP
00312A 0201 9D                    NOP
00313A 0202 9D                    NOP
00314A 0203 9D                    NOP
00315A 0204 9D                    NOP
00316A 0205 9D                    NOP
00317A 0206 9D                    NOP
00318A 0207 17 02       A        BCLR   HALT,PORTC
00319A 0209 16 02       A        BSET   HALT,PORTC
00320A 020B 9D                    NOP
00321A 020C 9D                    NOP
00322A 020D 9D                    NOP
00323A 020E 9D                    NOP
00324A 020F 9D                    NOP
00325A 0210 9D                    NOP
00326A 0211 9D                    NOP
00327A 0212 17 02       A        BCLR   HALT,PORTC
00328A 0214 16 02       A        BSET   HALT,PORTC
00329A 0216 9D                    NOP
00330A 0217 9D                    NOP
00331A 0218 9D                    NOP
00332A 0219 9D                    NOP
00333A 021A 9D                    NOP
00334A 021B 9D                    NOP
00335A 021C 9D                    NOP
00336A 021D 17 02       A        BCLR   HALT,PORTC
00337A 021F 16 02       A        BSET   HALT,PORTC
00338A 0221 9D                    NOP
00339A 0222 9D                    NOP
00340A 0223 9D                    NOP
00341A 0224 9D                    NOP
00342A 0225 9D                    NOP
00343A 0226 9D                    NOP
00344A 0227 9D                    NOP
00345A 0228 81                    RTS
```

```
00346
00347
00348                    ********************************************
00349                    *                                          *
00350                    *    SKIP SUB-ROUTINES                     *
00351                    *                                          *
00352                    *  THESE SUB-ROUTINES WORK IN THE FOLLOWING *
00353                    *  MANNER:                                 *
00354                    *                                          *
00355                    *  THE SOFTWARE 704P2 P.C. VALUE STORED    *
00356                    *  IN THE 705P3 IS SEPARATED INTO TWO BYTES *
00357                    *     *
00358                    *  PHIGH: HIGH ORDER BYTE OF THE "SOFT" PC  *
00359                    *         VALUE.                           *
00360                    *  PLOW : LOW ORDER BYTE OF THE "SOFT" PC   *
00361                    *         VALUE.                           *
00362                    *                                          *
00363                    * WHEN THE CONCATENATED VALUE OF PHIGH &    *
00364                    * PLOW IS EQUAL TO SHIGH & SLOW, THE PROPER *
00365                    * NUMBER OF BYTES  WILL HAVE BEEN SKIPPED.  *
00366                    *                                          *
00367                    ********************************************
00368
00369
00370
00371                    ********************************************
00372                    *                                          *
00373                    * SKIPØ SUB-ROUTINE:                       *
00374                    * THIS SUB-ROUTINE WILL INCREMENT THE 704P2  *
00375                    * PROGRAM COUNTER SO THAT IT POINTS AT $12   *
00376                    *                                          *
00377                    ********************************************
00378A 0229 3F 1B       A SKIPØ  CLR    PHIGH
00379A 022B 3F 1C       A        CLR    PLOW
00380A 022D 3F 1D       A        CLR    SHIGH
00381A 022F A6 12       A        LDA    #$12
00382A 0231 B7 1F       A        STA    SLOW
00383A 0233 CD 0372     A        JSR    SKIP
00384A 0236 81                   RTS
00385
00386                    ********************************************
00387                    *                                          *
00388                    * SKIPB SUB-ROUTINE:                       *
00389                    * THIS SUB-ROUTINE WILL INCREMENT THE 704P2  *
00390                    * PROGRAM COUNTER SO THAT IT POINTS AT $CØØ  *
00391                    *                                          *
00392                    ********************************************
00393A 0237 A6 ØC       A SKIPB  LDA    #$ØC
00394A 0239 B7 1D       A        STA    SHIGH
00395A 023B 3F 1F       A        CLR    SLOW
00396A 023D CD 0372     A        JSR    SKIP
00397A 0240 81                   RTS
00398
00399                    ********************************************
00400                    *                                          *
00401                    * SKIPC SUB-ROUTINE:                       *
00402                    * THIS SUB-ROUTINE WILL INCREMENT THE 704P2  *
00403                    * PROGRAM COUNTER SO THAT IT POINTS AT $FFC  *
00404                    *                                          *
00405                    ********************************************
00406A 0241 A6 ØF       A SKIPC  LDA    #$ØF
00407A 0243 B7 1D       A        STA    SHIGH
00408A 0245 A6 FC       A        LDA    #$ØFC
00409A 0247 B7 1F       A        STA    SLOW
00410A 0249 CD 0372     A        JSR    SKIP
00411A 024C 81                   RTS
```

```
00412
00413                     ********************************************
00414                     *                                          *
00415                     *  DISCON SUB-ROUTINE:                     *
00416                     *  THIS SUB-ROUTINE DISABLES THE HARDWARE. *
00417                     *  IT DE-ACTIVATES THE SERIAL TO PARALLEL  *
00418                     *  CONVERSION, DESELECTS THE EPROM, AND ALSO *
00419                     *  DISABLES THE TRI-STATE BUFFER.          *
00420                     *                                          *
00421                     ********************************************
00422
00423
00424A 024D A6 07       A DISCON LDA    #%00000111 TURN EPROM OFF,
00425                     *                  BUFFER OFF,SERIAL
00426                     *                  STREAMS OFF,BURNER OFF
00427A 024F B7 00       A        STA    PORTA
00428A 0251 81                   RTS
00429
00430                     ********************************************
00431                     *                                          *
00432                     *  HNDLA SUB-ROUTINE:                      *
00433                     *  THIS SUB-ROUTINE WILL PERMIT THE PROPER *
00434                     *  FUNCTION( ZERO_CHECK,VERIFY OR PROGRAM ) *
00435                     *  TO OPERATE ON ADDRESSES $18---$5F.      *
00436                     *                                          *
00437                     ********************************************
00438A 0252 4F           HNDLA  CLRA
00439A 0253 B7 1D       A        STA    SHIGH
00440A 0255 A6 5F       A        LDA    #$5F
00441A 0257 B7 1F       A        STA    SLOW
00442A 0259 CD 027E     A        JSR    HANDLE
00443A 025C 81                   RTS
00444                     ********************************************
00445                     *                                          *
00446                     *  HNDLB SUB-ROUTINE:                      *
00447                     *  THIS SUB-ROUTINE WILL PERMIT THE PROPER *
00448                     *  FUNCTION( ZERO_CHECK,VERIFY OR PROGRAM ) *
00449                     *  TO OPERATE ON ADDRESSES $C00---$FF7     *
00450                     *                                          *
00451                     ********************************************
00452A 025D A6 0F       A HNDLB  LDA    #$0F
00453A 025F B7 1D       A        STA    SHIGH
00454A 0261 A6 F7       A        LDA    #$0F7
00455A 0263 B7 1F       A        STA    SLOW
00456A 0265 CD 027E     A        JSR    HANDLE
00457A 0268 81                   RTS
00458
00459                     ********************************************
00460                     *                                          *
00461                     *  HNDLC SUB-ROUTINE:                      *
00462                     *  THIS SUB-ROUTINE WILL PERMIT THE PROPER *
00463                     *  FUNCTION( ZERO_CHECK,VERIFY OR PROGRAM ) *
00464                     *  TO OPERATE ON ADDRESSES $FFC---$FFF.    *
00465                     *                                           *
00466                     ********************************************
00467A 0269 A6 0F       A HNDLC  LDA    #$0F
00468A 026B B7 1D       A        STA    SHIGH
00469A 026D A6 FF       A        LDA    #$0FF
00470A 026F B7 1F       A        STA    SLOW
00471A 0271 CD 027E     A        JSR    HANDLE
00472A 0274 81                   RTS
00473
00474A 0275 A6 02       A WAIT   LDA    #2
00475A 0277 B7 11       A        STA    PULSES
00476A 0279 3A 11       A WAITX  DEC    PULSES
00477A 027B 26 FC     0279       BNE    WAITX
00478A 027D 81                   RTS
```

```
00479
00480
00481                    ************************************************
00482                    *                                              *
00483                    *  HANDLE SUB-ROUTINE:                         *
00484                    *  DEPENDING ON THE VALUE OF THE VARIABLE SELECT, *
00485                    *  EITHER THE ZERO_CHECK,VERIFY OR PROGRAM MODE  *
00486                    *  OF OPERATION WILL BE SELECTED.              *
00487                    *                                              *
00488                    ************************************************
00489A 027E B6 19       A HANDLE LDA    SELECT
00490A 0280 A1 02       A        CMPA   #$02
00491A 0282 27 12    0296        BEQ    HANDL1
00492A 0284 A1 01       A        CMPA   #$01
00493A 0286 27 1C    02A4        BEQ    HANDL2
00494A 0288 A6 27       A        LDA    #%00100111 TURN OFF EPROM
00495A 028A B7 00       A        STA    PORTA
00496A 028C A6 2F       A        LDA    #%00101111 TURN ON BUFFER
00497A 028E B7 00       A        STA    PORTA
00498A 0290 CD 0275     A        JSR    WAIT
00499A 0293 CC 02CA     A        JMP    HANDL4
00500
00501A 0296 A6 13       A HANDL1 LDA    #%00010011 PC STREAM ON,
00502                    *                  VERIFY STREAM OFF,
00503                    *                  BURNER OFF, EPROM ON,
00504                    *                  BUFFER OFF
00505A 0298 B7 00       A        STA    PORTA
00506A 029A CD 0275     A        JSR    WAIT
00507A 029D B6 01       A        LDA    PORTB    READ EPROM CONTENTS
00508A 029F B7 1A       A        STA    DATA1    BACKUP THE VALUE
00509A 02A1 CC 02D3     A        JMP    HANDL5
00510
00511A 02A4 A6 27       A HANDL2 LDA    #%00100111 TURN OFF EPROM
00512A 02A6 B7 00       A        STA    PORTA
00513A 02A8 A6 2F       A        LDA    #%00101111 TURN ON BUFFER
00514A 02AA B7 00       A        STA    PORTA
00515A 02AC CD 0275     A        JSR    WAIT     WAIT UNTIL DATA IS STABLE
00516
00517A 02AF B6 01       A HANDL3 LDA    PORTB
00518A 02B1 B7 1A       A        STA    DATA1    BACKUP VERIFY DATA
00519A 02B3 A6 13       A        LDA    #%00010011 PC STREAM ON, VERIFY OFF,
00520                    *                  EPROM ON, DATA OFF
00521A 02B5 B7 00       A        STA    PORTA
00522A 02B7 CD 0275     A        JSR    WAIT
00523A 02BA B6 1A       A        LDA    DATA1
00524A 02BC B1 01       A        CMPA   PORTB    COMPARE VERIFY DATA WITH EPROM
00525                    *                  CONTENTS
00526A 02BE 27 03    02C3        BEQ    HANDLX   IF THEY MATCH, CONTINUE THE ANALYSIS.
00527A 02C0 CC 032A     A        JMP    VFAIL    OTHERWISE, GO TO THE VERIFY FAIL SUB-ROUTINE.
00528A 02C3 A6 2F       A HANDLX LDA    #%00101111
00529A 02C5 B7 00       A        STA    PORTA    ENABLE SERIAL OUTPUT OF 704P2 EPROM CONTENTS
00530A 02C7 CC 02ED     A        JMP    HANDL6
00531
00532A 02CA B6 01       A HANDL4 LDA    PORTB
00533A 02CC A1 00       A        CMPA   #$00
00534A 02CE 26 4A    031A        BNE    EFAIL
00535A 02D0 CC 02ED     A        JMP    HANDL6
00536
00537A 02D3 CD 034A     A HANDL5 JSR    BURNIT
00538A 02D6 B6 1E       A        LDA    NOVERF
00539A 02D8 A1 FF       A        CMPA   #$0FF    SEE IF THE VERIFY AFTER
00540                    *                  BURN HAS BEEN ACTIVATED.
00541A 02DA 27 11    02ED        BEQ    HANDL6
00542A 02DC A6 27       A        LDA    #%00100111 TURN OFF EPROM
00543A 02DE B7 00       A        STA    PORTA
00544A 02E0 A6 2F       A        LDA    #%00101111 TURN ON DATA BUFFER
00545A 02E2 B7 00       A        STA    PORTA
00546A 02E4 CD 0275     A        JSR    WAIT
00547A 02E7 B6 01       A        LDA    PORTB
00548A 02E9 B1 1A       A        CMPA   DATA1
00549A 02EB 26 4D    033A        BNE    PFAIL
00550
00551A 02ED B6 1B       A HANDL6 LDA    PHIGH
00552A 02EF B1 1D       A        CMPA   SHIGH
00553A 02F1 25 06    02F9        BLO    HANDL7
00554A 02F3 B6 1C       A        LDA    PLOW
00555A 02F5 B1 1F       A        CMPA   SLOW
00556A 02F7 27 14    030D        BEQ    HANDL9
```

```
00557
00558A 02F9 CD 03A1      A HANDL7 JSR     TOGGLE
00559A 02FC 98                    CLC
00560A 02FD B6 1C        A        LDA     PLOW
00561A 02FF AB 01        A        ADD     #$01
00562A 0301 25 04     0307        BCS     HANDL8
00563A 0303 B7 1C        A        STA     PLOW
00564A 0305 20 07     030E        BRA     HNDL10
00565A 0307 3C 1B        A HANDL8 INC     PHIGH
00566A 0309 3F 1C        A        CLR     PLOW
00567A 030B 20 01     030E        BRA     HNDL10
00568A 030D 81                HANDL9 RTS
00569
00570A 030E B6 19        A HNDL10 LDA     SELECT
00571A 0310 A1 02        A        CMPA    #$02
00572A 0312 27 82     0296        BEQ     HANDL1   IF SELECT = $02
00573                             *              THEN GO BACK AND BURN
00574                             *              ANOTHER BYTE OF DATA
00575A 0314 A1 01        A        CMPA    #$01
00576A 0316 27 97     02AF        BEQ     HANDL3   IF SELECT = $03, THEN
00577                             *              GO BACK AND VERIFY THE
00578                             *              EPROM CONTENTS
00579A 0318 20 B0     02CA        BRA     HANDL4   SELECT = $00, THEN GO
00580                             *              BACK AND CHECK ON ANOTHER
00581                             *              ERASED BYTE
00582
00583
00584
00585
00586                          ***********************************************
00587                          *                                             *
00588                          *       EFAIL SUB-ROUTINE                     *
00589                          *                                             *
00590                          ***********************************************
00591
00592A 031A A6 0A        A EFAIL  LDA     #%00001010 BLINK THE ERASED  LED
00593                             *                  TO INDICATE ERASE FAILURE
00594A 031C B7 02        A        STA     PORTC
00595A 031E CD 03AC      A        JSR     XBLINK
00596A 0321 A6 08        A        LDA     #%00001000
00597A 0323 B7 02        A        STA     PORTC
00598A 0325 CD 03AC      A        JSR     XBLINK
00599A 0328 20 F0     031A        BRA     EFAIL
00600
00601                          ***********************************************
00602                          *                                             *
00603                          *       VFAIL SUB-ROUTINE                     *
00604                          *       THIS SUB-ROUTINE WILL BE ACTIVATED    *
00605                          *       TO INDICATE A VERIFY FAILURE. THE     *
00606                          *       ADDRESS WILL BE AVAILABLE ON THE OUT- *
00607                          *       PUT PINS OF THE LS374'S               *
00608                          *                                             *
00609                          ***********************************************
00610A 032A A6 09        A VFAIL  LDA     #%00001001 VERIFIED LED ON
00611A 032C B7 02        A        STA     PORTC
00612A 032E CD 03AC      A        JSR     XBLINK   SMALL DELAY LOOP
00613A 0331 A6 08        A        LDA     #%00001000 VERIFIED LED OFF
00614A 0333 B7 02        A        STA     PORTC
00615A 0335 CD 03AC      A        JSR     XBLINK   SMALL DELAY LOOP
00616A 0338 20 F0     032A        BRA     VFAIL
00617
00618                          ***********************************************
00619                          *                                             *
00620                          *       PFAIL SUB-ROUTINE                     *
00621                          *       THIS SUB-ROUTINE IS CALLED WHENEVER A *
00622                          *       PROBLEM OCCURS DURING THE PROGRAMMING *
00623                          *       MODE.                                 *
00624                          *                                             *
00625                          ***********************************************
00626A 033A A6 0C        A PFAIL  LDA     #%00001100 PROGRAMMED LED ON
00627A 033C B7 02        A        STA     PORTC
00628A 033E CD 03AC      A        JSR     XBLINK   SMALL DELAY LOOP
00629A 0341 A6 08        A        LDA     #%00001000 PROGRAMMED LED OFF
00630A 0343 B7 02        A        STA     PORTC
00631A 0345 CD 03AC      A        JSR     XBLINK   SMALL DELAY LOOP
00632A 0348 20 F0     033A        BRA     PFAIL
```

```
00633
00634
00635                    ***********************************************
00636                    *                                             *
00637                    *  BURNIT SUB-ROUTINE                         *
00638                    *  THIS SUB-ROUTINE IS CALLED WHENEVER A BYTE *
00639                    *  OF DATA IS TO BE PROGRAMMED INTO THE 704P2 *
00640                    *                                             *
00641                    ***********************************************
00642           034A     A BURNIT EQU    *
00643A 034A A6 08        A        LDA    #08
00644A 034C B7 11        A        STA    PULSES   NO MORE THAN A 50 MSEC. PULSE
00645A 034E 13 00        A BURN1  BCLR   BURN,PORTA TURN ON "BURNER"
00646A 0350 CD 0393      A        JSR    MSEC1
00647A 0353 CD 0393      A        JSR    MSEC1
00648A 0356 CD 0393      A        JSR    MSEC1
00649A 0359 CD 0393      A        JSR    MSEC1
00650A 035C CD 0393      A        JSR    MSEC1
00651A 035F 12 00        A        BSET   BURN,PORTA TURN OFF "BURNER"
00652A 0361 3A 11        A        DEC    PULSES
00653A 0363 26 E9     034E        BNE    BURN1
00654A 0365 81                    RTS             PROGRAMMING OF BYTE
00655                    *                          IS COMPLETE
00656
00657
00658           0366     A WAIT1  EQU    *
00659A 0366 A6 1E        A        LDA    #30
00660A 0368 B7 11        A        STA    PULSES
00661A 036A CD 0393      A WAIT2  JSR    MSEC1
00662A 036D 3A 11        A        DEC    PULSES
00663A 036F 26 F9     036A        BNE    WAIT2
00664A 0371 81                    RTS
00665
00666
00667
00668
00669
00670A 0372 B6 1B        A SKIP   LDA    PHIGH    LOAD HIGH ORDER "SOFT" PC
00671                    *                        VALUE & DETERMINE IF IT
00672                    *                        HAS REACHED SHIGH YET.
00673A 0374 B1 1D        A        CMPA   SHIGH
00674A 0376 25 06     037E        BLO    SKIP1    IF THEY AREN'T EQUAL,
00675                    *                        THEN KEEP ON INCREMENTING
00676                    *                        THE "SOFT" PLOW. IF THEY
00677                    *                        ARE EQUAL, THEN CHECK ON
00678                    *                        THE LOW ORDER BYTES OF PC
00679A 0378 B6 1F        A        LDA    SLOW
00680A 037A B1 1C        A        CMPA   PLOW
00681A 037C 27 14     0392        BEQ    SKIP3    PROPER # OF BYTES HAVE
00682                    *                         BEEN SKIPPED.
00683A 037E CD 03A1      A SKIP1  JSR    TOGGLE   INCREMENT THE 704P2 P.C.
00684A 0381 98                    CLC
00685A 0382 B6 1C        A        LDA    PLOW     INCREMENT THE "SOFT" PC VALUE
00686A 0384 AB 01        A        ADD    #$01
00687A 0386 25 04     038C        BCS    SKIP2    IF PLOW OVER-FLOWED, THEN
00688                    *                          INCREMENT PHIGH
00689A 0388 B7 1C        A        STA    PLOW
00690A 038A 20 E6     0372        BRA    SKIP
00691A 038C 3C 1B        A SKIP2  INC    PHIGH
00692A 038E 3F 1C        A        CLR    PLOW
00693A 0390 20 E0     0372        BRA    SKIP
00694A 0392 81              SKIP3  RTS
00695
00696                    ***********************************************
00697                    *                                             *
00698                    *  MSEC1 SUB-ROUTINE                          *
00699                    *  THIS SUB-ROUTINE IS CALLED WHENEVER AN     *
00700                    *  APPROXIMATE 1 MSEC. DELAY IS REQUIRED.     *
00701                    *                                             *
00702                    ***********************************************
00703           0393     A MSEC1  EQU    *
00704A 0393 A6 4A        A        LDA    #%0100101010 DIVIDE BY 4
00705A 0395 B7 09        A        STA    TSCR
00706A 0397 A6 B4        A        LDA    #180
00707A 0399 B7 08        A        STA    TDATA
00708A 039B 0F 09 FD 039B        BRCLR  TIR,TSCR,*
00709A 039E 1F 09        A        BCLR   TIR,TSCR
00710A 03A0 81                    RTS
```

```
00711
00712                    ***********************************************
00713                    *                                             *
00714                    *  TOGGLE SUB-ROUTINE                         *
00715                    *  THIS SUB-ROUTINE IS USED TO INCREMENT THE  *
00716                    *  704P2'S PROGRAM COUNTER.                   *
00717                    *                                             *
00718                    ***********************************************
00719           03A1     A TOGGLE EQU    *
00720A 03A1 17 02        A        BCLR   HALT,PORTC
00721A 03A3 16 02        A        BSET   HALT,PORTC
00722A 03A5 9D                    NOP
00723A 03A6 9D                    NOP
00724A 03A7 9D                    NOP
00725A 03A8 9D                    NOP
00726A 03A9 9D                    NOP
00727A 03AA 9D                    NOP
00728A 03AB 81                    RTS
00729
00730
00731           03AC     A XBLINK EQU    *
00732A 03AC A6 FF        A        LDA    #$FF
00733A 03AE B7 17        A        STA    TEMP3
00734A 03B0 A6 02        A        LDA    #$02
00735A 03B2 B7 18        A        STA    TEMP4
00736A 03B4 CD 0393      A LOOP2  JSR    MSEC1
00737A 03B7 B6 17        A        LDA    TEMP3
00738A 03B9 4A                    DECA
00739A 03BA B7 17        A        STA    TEMP3
00740A 03BC 26 F6     03B4        BNE    LOOP2
00741A 03BE CD 0393      A        JSR    MSEC1
00742A 03C1 B6 18        A        LDA    TEMP4
00743A 03C3 4A                    DECA
00744A 03C4 B7 18        A        STA    TEMP4
00745A 03C6 26 EC     03B4        BNE    LOOP2
00746A 03C8 81                    RTS
00747           03C9     A INIT   EQU    *
00748A 03C9 A6 3F        A        LDA    #%00111111
00749A 03CB B7 04        A        STA    DDRA
00750A 03CD A6 06        A        LDA    #%00000110
00751A 03CF B7 00        A        STA    PORTA
00752A 03D1 81                    RTS
00753
00754
00755A 03D2 A6 0F        A LED    LDA    #$0F      SET LOW NIBBLE OF PORTC
00756                     *                         TO OUTPUT MODE
00757A 03D4 B7 06        A        STA    DDRC
00758A 03D6 A6 08        A        LDA    #$08      TURN LED'S OFF
00759A 03D8 B7 02        A        STA    PORTC
00760A 03DA 81                    RTS
00761
00762
00763A 0784                       ORG    MOR
00764A 0784      03       A        FCB    %00000011 DIV BY 8, XTAL MODE
00765
00766A 07F8                       ORG    $7F8
00767A 07F8      0100     A        FDB    START
00768A 07FA      0100     A        FDB    START
00769A 07FC      0100     A        FDB    START
00770A 07FE      0100     A        FDB    START
00771                             END
TOTAL ERRORS 00000--00000
```

MOTOROLA *Semiconductor Products Inc.*

3501 ED BLUESTEIN BLVD., AUSTIN, TEXAS 78721 • A SUBSIDIARY OF MOTOROLA INC.