Software Emulation of DDC1 hardware using HC05BD3

Introduction

Data Display Channel (DDC) is a new standard defined by VESA for monitor devices in 1994 (Ref.1). There are two levels of DDC, namely, DDC level 1 (DDC1) and DDC level 2 (DDC2). DDC2 is a bidirectional communications protocol between host computer and monitor. It is based on the IIC protocol.

DDC1 is an unidirectional protocol. The monitor will continue to send out Extended Display Identification Data (EDID) to the host computer. It is a serial communication interface, with a clock and a data line. The vertical sync from the host is used as the clock input. Data will be clocked out to host from the monitor. Every nine clock pulses will complete one byte data transmission. Figure 1 shows block diagram of the DDC1 operation.

If monitor manufacturers need to include the DDC1 features into their designs, they have two alternatives. One is to built-in the DDC1 hardware into MCU. The other is to use a special EEPROM with DDC1 function. The first method will increase the MCU cost and not flexible for external EEPROM MCU. The second method requires a special EEPROM, which will also increase the cost. It is because the price for a special EEPROM may be as high as US$2.0. In this application, we will use the low cost HC05BD3 MCU to implement the DDC1 features. Since the HC05BD3 also has built-in IIC, it fully supports the DDC standard for both DDC1 and DDC2.

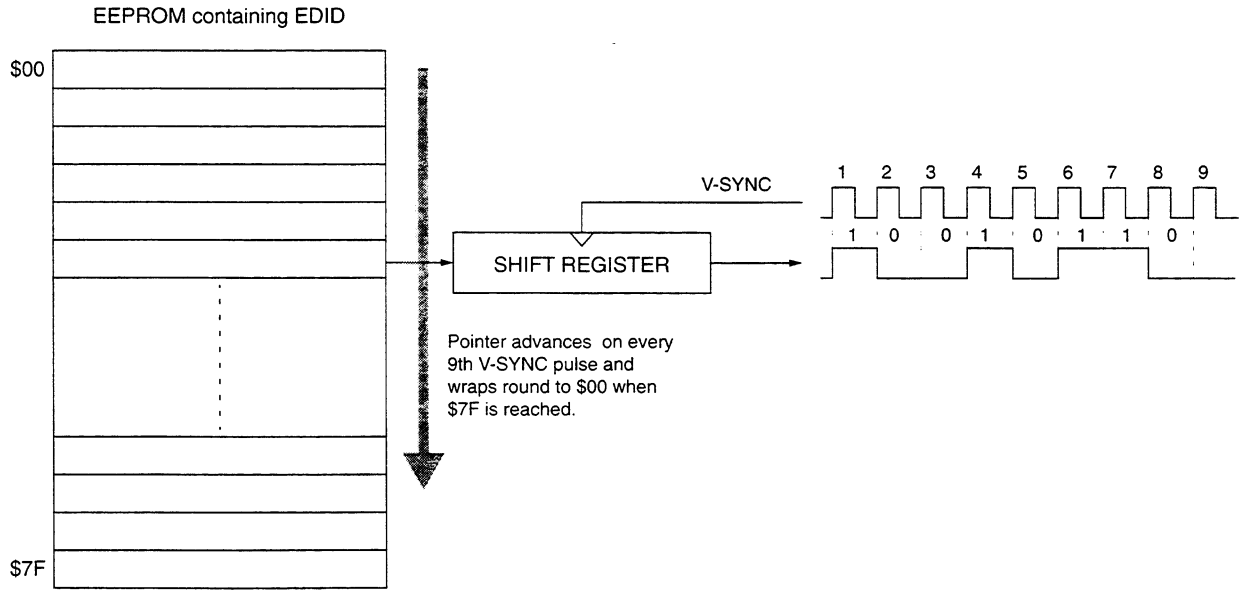

Figure 1 Block Diagram of DDC1 Operation

DDC1 Specification Summary (Ref.1)

Electrical

For DDC1, the VCLK signal should start at the normal frame frequency. Once data is sensed on the data line, then VCLK can be increased to a maximum rate of 25KHz. If no data is received at the normal video frame frequency, the display may be of type OLD. OLD type monitors may be damaged if a higher than normal vertical frequency is applied.

Timing

Data is clocked on the rising edge of the VCLK and shall be valid 30µs after the rising edge. It shall remain valid until the next rising edge. The minimum time between falling edge to the next rising edge is 20µs. Pulse width shall have a minimum of 20µs. The DDC1 data shall be clocked with nine clocks per byte. The data bit generated on the ninth clock pulse is an acknowledge bit that should be discarded by the host. Figure 2 shows the timing graphically.

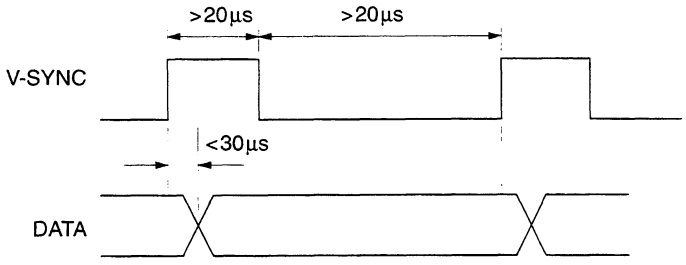

Figure 2 Timing Diagram for V-SYNC and DATA

Hardware Configurations

In this application, we will use the vertical sync interrupt provided by HC05BD3 and also the IIC bus to interface with an external 128bytes normal EEPROM. Figure 3 shows the configuration. PA0 will be used to be the DATA output from MCU. Actually, any PORT pin can be used to replace PA0. This horizontal sync (or H-sync) and vertical sync (or V-sync) is connected to the H-sync and V-sync input of HC05BD3.

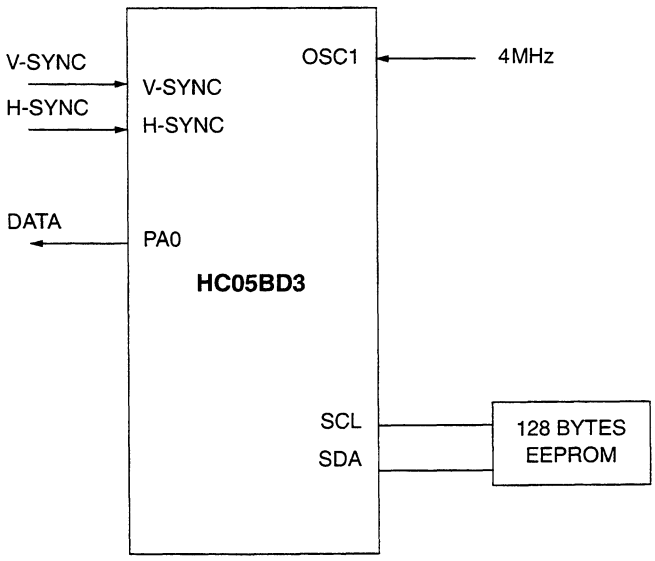

Figure 3 Hardware Connection

HC05BD3 is operated at 5V with 4MHz, and the IIC module is set to the highest speed of about 90KHz SCL.

Programming Procedures

initialize the MCU and IIC bus

bit_count = 9

get first data byte from EE at address $00 by

send start signal
send slave address (write mode)
send EE data address
send repeat start
send slave address (read mode)
get data from MCU data register and put into DATA

wait for v-sync interrupt
when v-sync interrupt occur, do
if bit_count not zero then do
decrease bit_count
rotate left DATA, if carry set then high PA0, otherwise low PA0
else read next data byte from EE and put into DATA and bit_count = 9
return interrupt

Since the EDID requires 128 bytes data, we used a 128 serial EEPROM. The EEPROM address will wrap round from 128 to 0 automatically. So that there is no need to reset the address pointer when the address reaches $7F. It is very important because we must read one byte from EEPROM within 40µs.

Results

Figure 4 shows the part of the transmission and the timing relationship between V-sync and DATA. The following results are observed:

1) The data will be valid within 20µs after the rising edge of V-sync.
2) It takes approx. 24µs to read a new data byte from external EEPROM.This time occurs during the 9th clock and NO data update on PA0 is required.
3) The total code size is below 180 bytes.

Considerations

1) During normal operation, the V-sync should be less than 1KHz or 1ms. So every vertical period, the CPU will have (1000-15) or 985µs to do other jobs. The efficiency will be 98.5% at 1KHz V-sync input.
2) Since IIC is used, we can connect two external EEPROMs at the same time. One for EDID data and one for video mode information. However, no EDID information will be output to the HOST PC if the CPU accesses the video mode EEPROM for more than 8 V-sync periods. 8 V-sync periods because we can pending the video mode info access until the 9th clock is completed. The monitor will access the video EEPROM in two conditions: 1. is during factory testing, 2. is during video mode change. During condition 1, no EDID info will be output, so no conflict. During condition 2, the time is long enough for the CPU to wait for the 9th clock.

Summary

It is shown that by using the V-sync interrupt and IIC features provided by HC05BD3, it is possible to emulate the DDC1 hardware. The CPU efficiency is greater than 98.5% during normal operation. The code size is below 180 bytes. The user can also use the built-in IIC to implement the DDC2 standard.

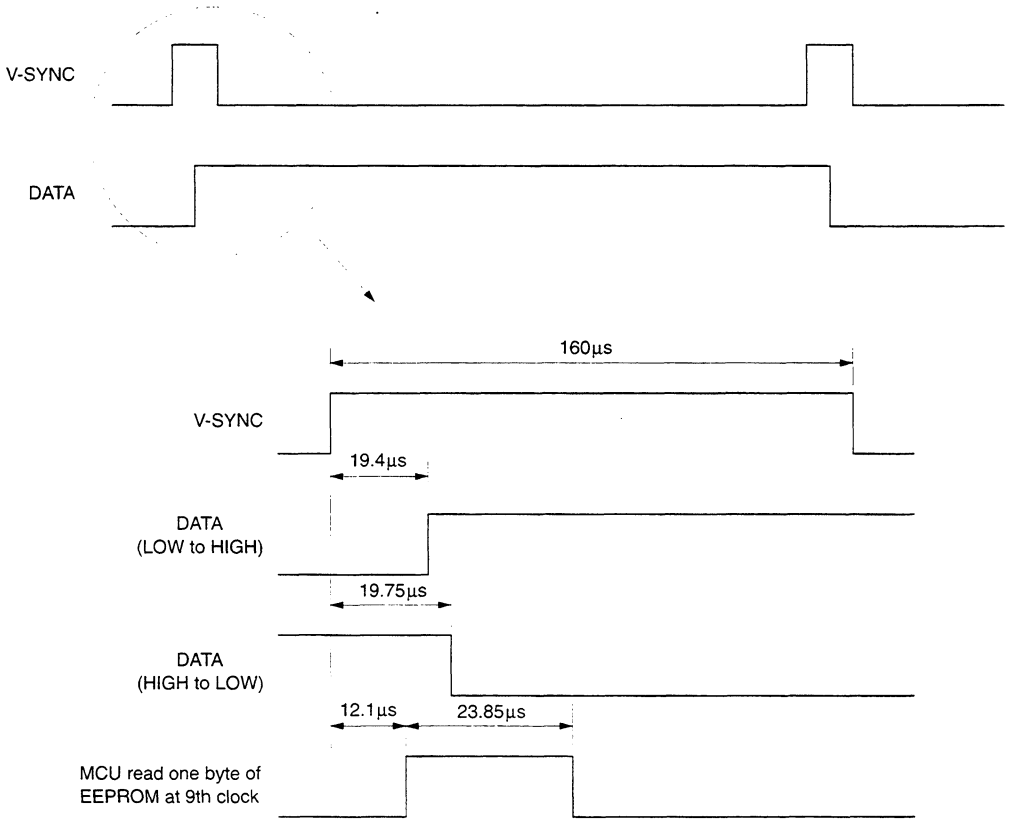

Figure 4 Timing Relationship between V-SYNC and DATA achieved by this application

Reference Documents

1) "Display Data Channel Proposal", version 1.0p 1994 by VESA
2) "MC68HC06T16 Product Preview", version 1.1 1994 by Motorola
3) "MC68HC05BD3 Product Preview", version 1.1 1994 by Motorola
4) Philips IIC bus specification (Data Handbook IIC Peripherals for Microcontroller 1/92)

Appendix A - Program Listing (T16 version)

```
0001                    ********************************************************************
0002                    *                                                                  *
0003                    * MOTOROLA SEMICONDUCTORS HK LTD                                   *
0004                    * A/P MCU TECHNICAL MARKETING                                      *
0005                    *                                                                  *
0006                    * DATA DISPLAY CHANNEL 1 DEMO                                     *
0007                    *                                                                  *
0008                    * TARGET MCU  : MC68HC05T16                                       *
0009                    *               ALSO APPLICABLE                                    *
0010                    *               OTHER MCU WITH V-SYNC INT AND IIC                  *
0011                    * CLOCK SPEED : 4MHZ                                              *
0012                    *                                                                  *
0013                    * AUTHOR      : C.P. CHEN                                         *
0014                    * DATE        : 10, AUG 1994                                      *
0015                    *                                                                  *
0016                    ********************************************************************
0017
0018                    ***********************************
0019                    *         SYSTEM EQUATES          *
```

```
0020                    **********************************
0021
0022 0050               RAMST1   EQU       $50       * START OF 1ST PART OF RAM
0023 0100               RAMST2   EQU       $100      * START OF 2ND PART OF RAM
0024
0025 A000               ROMST    EQU       $A000     * START OF USER ROM
0026
0027 0000               PORTA    EQU       $00       * PORT A
0028 0001               PORTB    EQU       $01       * PORT B
0029 0006               DDRA     EQU       $06       * PORT A DDR
0030 0007               DDRB     EQU       $07       * PORT B DDR
0031 000D               CFG_F    EQU       $0D       * PORT F CONFIG REG
0032
0033 001D               OSD_CTL1 EQU       $1D       * OSD FRAME CONTROL 1
0034 0007               PLLEN    EQU       $7        * PLL ENABLE
0035 0004               ONOFF    EQU       $4        * OSD ON/OFF
0036 002B               OSD_CTL3 EQU       $2B       * OSD FRAME CONTROL 3
0037 0007               VFINTE   EQU       $7        * V-SYNC INT ENABLE
0038 0004               VFLB     EQU       $4        * V-SYNC INT FLAG
0039
0040 0050               MASADR   EQU       $50       *MASTER ADDRESS
0041 00A0               SLV_W    EQU       $A0       *SLAVE WRITE ADDRESS
0042 00A1               SLV_R    EQU       $A1       *SLAVE READ ADDRESS
0043 0000               PRESCAL  EQU       $00       *PRESCALAR
0044 0001               READ     EQU       $1        *READ MASK
0045 00FE               WRITE    EQU       $FE       *WRITE MASK
0046
0047 0037               MADR     EQU       $37       *M-BUS ADDRESS REGISTER
0048 0038               FDR      EQU       $38       *FREQ DIVIDER REG
0049 0039               MCR      EQU       $39       *M-BUS CONTROL REG
0050 003A               MSR      EQU       $3A       *M-BUS STATUS REG
0051 003B               MDR      EQU       $3B       *M-BUS DATA REG
0052
0053 0007               MEN      EQU       7         *MCR BIT7 - M-BUS ENABLE
0054 0006               MIEN     EQU       6         *MCR BIT6 - M-BUS INTERRUPT ENABLE
0055 0005               MSTA     EQU       5         *MCR BIT5 - MASTER/SLAVE MODE SELECT
0056 0004               MTX      EQU       4         *MCR BIT4 - TX/RX MODE SELECT
0057 0003               TXAK     EQU       3         *MCR BIT3 - TRANSMIT ACK ENABLE BIT
0058
0059 0007               MCF      EQU       7         *MSR BIT7 - M-BUS DATA TRANSMITTING
0060 0006               MAAS     EQU       6         *MSR BIT6 - M-BUS ADDRESSED AS SLAVE
0061 0005               MBB      EQU       5         *MSR BIT5 - M-BUS BUSY
0062 0001               MIF      EQU       1         *MSR BIT1 - M-BUS INTERRUPT PENDING
0063 0000               RXAK     EQU       0         *MSR BIT0 - M-BUS RECEIVED ACKNOWLEDGE
0064
0065                    **********************************
0066                    *      VARIABLE DEFINITIONS      *
0067                    **********************************
0068
0069                    *
0070                    * RAM USEAGE :
0071                    *
0072 0050                        ORG     RAMST1
0073 0050               SAVE_X   RMB     1              * TEMP X
0074 0051               SAVE_A   RMB     1              * TEMP A
0075 0052               EEP_AD   RMB     1              * EEPROM BYTE ADDRESS
0076 0053               BYTE_CNT RMB     1              * EDID COUNTER 0-128
0077 0054               DATA     RMB     1              * RECEIVE DATA SHIFT REGISTER
0078 0055               BIT_CNT  RMB     1              * COUNT NUMBER OF BIT RECEIVED
0079
0080                    **********************************
0081                    *          MAIN PROGRAM          *
0082                    **********************************
0083 A000                        ORG     ROMST
0084
0085                    *
0086                    * PORT AND DATA INIT
0087                    *
0088 A000 9B            MAIN     SEI
0089 A001 4F                     CLRA                   * INIT
0090 A002 B7 52                  STA     EEP_AD         * EEP_AD = 0
0091 A004 B7 53                  STA     BYTE_CNT       * BYTE_CNT = 0
0092 A006 B7 00                  STA     PORTA          * PORT A DATA = 0
0093 A008 B7 01                  STA     PORTB          * PORT B DATA = 0
0094 A00A A6 07                  LDA     #$7            * PORTA 0,1 & 2 SET TO OUTPUT
```

```
0095 A00C B7 06                     STA     DDRA            * PORT A DDR = $FF (ALL OUTPUT)
0096 A00E A6 60                     LDA     #$60            * CONFIG PF5 AND PF6 TO BE SDA AND SCL RESP.
0097 A010 B7 0D                     STA     CFG_F
0098 A012 A6 09                     LDA     #9
0099 A014 B7 55                     STA     BIT_CNT
0100 A016 A6 90                     LDA     #$90            * ENABLE PLL AND OSD
0101 A018 B7 1D                     STA     OSD_CTL1
0102 A01A A6 80                     LDA     #$80            * ENABLE V-SYNC INT
0103 A01C B7 2B                     STA     OSD_CTL3
0104
0105 A01E CD A0 6C                  JSR     INIT_MB         * INIT M-BUS
0106 A021 5F                        CLRX                    * START FROM EEPROM ADDRESS $00
0107 A022 CD A0 7D                  JSR     CALL            * SEND SLAVE ADDRESS & GET DATA
0108 A025 B7 54                     STA     DATA
0109 A027 3C 53                     INC     BYTE_CNT        * COUNT FOR NUMBER OF BYTE READ
0110
0111 A029 9A                        CLI
0112 A02A 20 FE             LOOP    BRA     LOOP
0113
0114 A02C                   OSD_ISR EQU     *
0115 A02C 09 2B 29                  BRCLR   VFLB,OSD_CTL3,EX_OSD  *CHECK IF V-SYNC INT
0116 A02F 3A 55                     DEC     BIT_CNT               *DECREASE BIT_CNT
0117 A031 27 0F                     BEQ     NXT_R                 *IF 0 THEN READ NEXT BYTE
0118 A033 B6 54                     LDA     DATA                  *OTHERWISE ROTATE DATA 1BIT LEFT
0119 A035 49                        ROLA
0120 A036 B7 54                     STA     DATA
0121 A038 24 04                     BCC     LOW                   *IF MSB LOW THEN OUTPUT LOW
0122 A03A 10 00                     BSET    0,PORTA               *OTHERWISE OUTPUT HIGH
0123 A03C 20 1A                     BRA     EX_OSD                *RETURN INT
0124 A03E 11 00             LOW     BCLR    0,PORTA
0125 A040 20 16                     BRA     EX_OSD
0126
0127
0128
0129 A042                   NXT_R   EQU     *
0130                        ********************************************************************
0131                        * THIS PART IS ONLY REQUIRED FOR EXTERNAL EEPROM IS GREATER
0132                        * THAN 128 BYTES
0133                        ********************************************************************
0134 A042 BE 53                     LDX     BYTE_CNT              *CHECK IF 128 BYTES READ
0135 A044 5C                        INCX
0136 A045 A3 80                     CPX     #$80
0137 A047 27 12                     BEQ     NXT_CYC               *IF SO RESET EEPROM ADDRESS TO $00
0138 A049 BF 53                     STX     BYTE_CNT
0139                        ****************************************OTHERWISE GET 1 BYTE FROM EEPROM
0140 A04B 03 3A 64                  BRCLR   MIF,MSR,W_RB1         *WAIT UNTIL DATA RECEIVED
0141 A04E 13 3A                     BCLR    MIF,MSR               *RESET INTERRUPT FLAG
0142 A050 B6 3B                     LDA     MDR                   *GET RECEIVED DATA
0143 A052 B7 54                     STA     DATA
0144 A054 A6 09             INIT_BIT LDA    #9                    *RESET BIT_CNT
0145 A056 B7 55                     STA     BIT_CNT
0146 A058 19 2B             EX_OSD  BCLR    VFLB,OSD_CTL3         *RESET INT FLAG
0147 A05A 80                        RTI
0148
0149                        ********************************************************************
0150                        * THIS PART IS ONLY REQUIRED FOR EXTERNAL EEPROM IS GREATER
0151                        * THAN 128 BYTES
0152                        ********************************************************************
0153 A05B                   NXT_CYC EQU     *
0154 A05B 1A 39                     BSET    MSTA,MCR        *GENERATE (STOP)
0155 A05D 1B 39                     BCLR    MSTA,MCR
0156 A05F CD A0 BA                  JSR     DELAY           *DELAY REQUIRED BETWEEN STOP/START
0157 A062 5F                        CLRX
0158 A063 BF 53                     STX     BYTE_CNT
0159 A065 CD A0 7D                  JSR     CALL
0160 A068 B7 54                     STA     DATA
0161 A06A 20 E8                     BRA     INIT_BIT
0162
0163                        *-------------------------------------------------------------------
0164                        *INIT_MB:   INIT M-BUS
0165                        *        - SET PRESCALAR
0166                        *        - SET MASTER ADDRESS
0167                        *        - RELEASE M-BUS IF IN USE
0168                        * SIDE EFF: A
0169                        *-------------------------------------------------------------------
```

```
0170 A06C A6 00          INIT_MB  LDA      #PRESCAL  ;SET CLOCK PRESCALAR
0171 A06E B7 38                   STA      FDR
0172 A070 A6 50                   LDA      #MASADR   ;SET MASTER ADDRESS
0173 A072 B7 37                   STA      MADR
0174 A074 1E 39                   BSET     MEN,MCR   ;ENABLE M-BUS
0175 A076 17 39                   BCLR     TXAK,MCR  ;ENABLE ACKNOWLEDGE
0176 A078 1A 39                   BSET     MSTA,MCR  ;GENERATE (STOP)
0177 A07A 1B 39                   BCLR     MSTA,MCR
0178 A07C 81                      RTS
0179
0180 A07D 0A 3A FD       CALL     BRSET    MBB,MSR,CALL     ;WAIT UNTIL THE M-BUS IS NOT BUSY
0181 A080 18 39                   BSET     MTX,MCR          ;SET TRANSMIT MODE (START)
0182 A082 1A 39                   BSET     MSTA,MCR         ;SET MASTER MODE
0183 A084 13 3A                   BCLR     MIF,MSR          ;CLEAR M-BUS INTERRUPT FLAG
0184 A086 A6 A0                   LDA      #SLV_W           ;GET SLAVE ADDRESS
0185 A088 B7 3B                   STA      MDR              ;SEND OUT SLAVE ADDRESS
0186 A08A 03 3A FD       W_CALL   BRCLR    MIF,MSR,W_CALL   ;WAIT UNTIL SLAVE ADDRESS IS SENT
0187 A08D 13 3A                   BCLR     MIF,MSR          ;RESET INTERRUPT FLAG
0188 A08F 00 3A 31                BRSET    RXAK,MSR,ERROR   ;CHECK THE ACNOWLEDGE FLAG
0189 A092 BF 3B                   STX      MDR              ;SEND BYTE ADDRESS
0190 A094 03 3A FD       W_CALL1  BRCLR    MIF,MSR,W_CALL1  ;WAIT UNTIL SLAVE ADDRESS IS SENT
0191 A097 13 3A                   BCLR     MIF,MSR          ;RESET INTERRUPT FLAG
0192 A099 00 3A 27                BRSET    RXAK,MSR,ERROR   ;CHECK THE ACNOWLEDGE FLAG
0193                     *         RTS
0194
0195 A09C 18 39          READ_B   BSET     MTX,MCR          ;SET TRANSMITE MODE (START)
0196 A09E 1B 39                   BCLR     MSTA,MCR         ;GENERATE REPEAT START/START
0197 A0A0 1A 39                   BSET     MSTA,MCR         ;SET MASTER MODE
0198 A0A2 A6 A1                   LDA      #SLV_R           ;GET SLAVE ADDRESS
0199 A0A4 B7 3B                   STA      MDR              ;SEND OUT SLAVE ADDRESS
0200 A0A6 03 3A FD       W_RB     BRCLR    MIF,MSR,W_RB     ;WAIT UNTIL SLAVE ADDRESS IS SENT
0201 A0A9 13 3A                   BCLR     MIF,MSR          ;RESET INTERRUPT FLAG
0202 A0AB 00 3A 15                BRSET    RXAK,MSR,ERROR   ;CHECK THE ACNOWLEDGE FLAG
0203 A0AE 19 39                   BCLR     MTX,MCR          ;SET RECEIVE MODE
0204 A0B0 B6 3B                   LDA      MDR              ;DUMMY READ FOR RECEIVE TIMING
0205 A0B2 03 3A FD       W_RB1    BRCLR    MIF,MSR,W_RB1    ;WAIT UNTIL DATA RECEIVED
0206 A0B5 13 3A                   BCLR     MIF,MSR          ;RESET INTERRUPT FLAG
0207 A0B7 B6 3B                   LDA      MDR              ;GET RECEIVED DATA
0208 A0B9 81                      RTS
0209
0210 A0BA B7 51          DELAY    STA      SAVE_A
0211 A0BC 4F                      CLRA
0212 A0BD 4C             DELAY1   INCA
0213 A0BE 26 FD                   BNE      DELAY1
0214 A0C0 B6 51                   LDA      SAVE_A
0215 A0C2 81                      RTS
0216
0217 A0C3 20 FE          ERROR    BRA      ERROR     ;ERROR ENCOUNTERED
0218
0219
0220 A0C5 80             MFT_ISR  RTI
0221 A0C6 80             PAC_ISR  RTI
0222 A0C7 80             IIC_ISR  RTI
0223 A0C8 80             IC_ISR   RTI
0224 A0C9 80             IRQ_ISR  RTI
0225 A0CA 80             NUL_ISR  RTI
0226                     **********************************
0227                     *  RESET AND INTERRUPT VECTORS   *
0228                     **********************************
0229 FFF0                         ORG    $FFF0
0230 FFF0 A0 C5          MFT_VEC  FDB    MFT_ISR
0231 FFF2 A0 C6          PAC_VEC  FDB    PAC_ISR
0232 FFF4 A0 C7          IIC_VEC  FDB    IIC_ISR
0233 FFF6 A0 C8          TIM_VEC  FDB    IC_ISR
0234 FFF8 A0 C9          IRQ_VEC  FDB    IRQ_ISR
0235 FFFA A0 2C          OSD_VEC  FDB    OSD_ISR
0236 FFFC A0 CA          SWI_VEC  FDB    NUL_ISR
0237 FFFE A0 00          RESET_VE FDB    MAIN
0238
0239                              END
```

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com