M68HC711D3 BootROM Program Listing

By Joseph Haas, ke0ff
11/26/2013
joeh-*at*-rollanet-*dot*-org

I recently had need for the bootrom listing of the HC711D3 processor. After scouring the internet and finding only a few tantalizing references to same, I decided to simply download the ROM code from one of the D3 parts I have and disassemble it. By hand.

This is normally a daunting task, but it was aided greatly by several happy coincidences. First, there are a number of HC11 variants with bootrom listings that are available. Furthermore, the fact that the bootrom differences between variants are generally minor, and that the boot ROM is comprised of less than 255 bytes of code, all combined to make the process quite manageable.

Of course, being an operation that entails a great deal of human involvement means that it must be taken with a grain of salt. Still, I believe that this effort is as error-free as I can reasonably make it and I offer this listing to the internet on the slim chance that it might help someone, somewhere, who might need to have a quick gander at what is going on inside the D3 when it enters boot mode.

Bon Appétit!

```
                       **********************************************************
                       * BOOTLOADER FIRMWARE FOR 68HC711D3                      *
                       **********************************************************
                       * Dis-assembled by Joe Haas, ke0ff, 11-25-2013 from ROM    *
                       * code downloaded from:                                  *
                       *            P/N: XC68HC711D3                            *
                       *        mask ID: IEQC9239                               *
                       *                                                        *
                       * Comments reconstructed from listings presented in      *
                       * Freescale document# EB422.                             *
                       *                                                        *
                       * This code was carefully hand-dis-assembled and checked   *
                       * for accuracy, but may yet contain isolated transcription *
                       * errors.  As such, this code listing should be considered *
                       * as reference data only.                                *
                       * Note: All timing specs are based on Eclk = 2.00 MHz      *
                       **********************************************************
                       * Features of this bootloader are. . .                   *
                       *                                                        *
                       * Auto baud select between 7812.5 and 1200 (8 MHz)       *
                       * 0 - 192 byte variable length download                  *
                       * Jump to EPROM at $F000 if 1st input byte = $00         *
                       * PROGRAM - Utility subroutine to program EPROM          *
                       * UPLOAD - Utility subroutine to dump memory to host     *
                       *  Rev I.D. at $BFD1 = $42 ("B")                         *
                       * Mask I.D. at $BFD4 = $71D3                             *
                       **********************************************************
                       * Revision B                                             *
                       * Changes from previous revisions unknown                *
                       **********************************************************
                       *
                       * HC711D3 REG EQUATES
                       *
0008                   PORTD    EQU     $08
000E                   TCNT     EQU     $0E
0028                   SPCR     EQU     $28
                       * BIT EQUATES FOR SPCR
0020                   DWOM     EQU     $20

000E                   TCNT     EQU     $0E
0016                   TOC1     EQU     $16
0023                   TFLG1    EQU     $23
                       * BIT EQUATES FOR TFLG1
0080                   OCF1     EQU     $80

002B                   BAUD     EQU     $2B
002C                   SCCR1    EQU     $2C
002D                   SCCR2    EQU     $2D
                       * BIT EQUATES FOR SCCR2
0008                   TE       EQU     $08
0004                   RE       EQU     $04

002E                   SCSR     EQU     $2E
                       * BIT EQUATES FOR SCSR
0080                   TDRE     EQU     $80
0020                   RDRF     EQU     $20

002F                   SCDR     EQU     $2F
003B                   PPROG    EQU     $3B
                       * BIT EQUATES FOR PPROG
0020                   ELAT     EQU     $20
0001                   EPGM     EQU     $01
                       *
                       * MEMORY CONFIGURATION EQUATES
                       *
F000                   EPRMSTR  EQU     $F000                     Start of EPROM
FFFF                   EPRMEND  EQU     $FFFF                     End of EPROM

0040                   RAMSTR   EQU     $0040                     Start of RAM
00FF                   RAMEND   EQU     $00FF                     End of RAM
                       *
                       * DELAY CONSTANTS
                       *
0DB0                   DELAYS   EQU     3504                      Delay at slow baud
021B                   DELAYF   EQU     539                       Delay at fast baud
                       *
1068                   PROGDEL  EQU     4200                      2.1 ms prog delay
```

```
                       ************************************************************

BF00                            ORG      $BF00                      Start of ROM code

                       ************************************************************
                       *                                                          *
                       * The next two instruction locations provide a predictable *
                       * set of addresses to call PROGRAM and UPLOAD even if the  *
                       * routines change size in future versions.                 *

BF00: 7E BF10          PROGRAM  JMP      PRGROUT

                       ************************************************************
                       * UPLOAD - Utility subroutine to send data from            *
                       * inside the MCU to the host via the SCI interface.        *
                       * Prior to calling UPLOAD set baud rate, turn on SCI       *
                       * and set Y=first address to upload.                       *
                       * Bootloader leaves baud set, SCI enabled, and             *
                       * Y pointing at EPROM start ($D0000) so these default      *
                       * values do not have to be changed typically.              *
                       * Consecutive locations are sent via SCI in an             *
                       * infinite loop. Reset stops the upload process.           *

BF03: 18A6 00          UPLOAD   LDAA     0,Y                        get data to send
BF06: 13 2E 80 FC               BRCLR    SCSR,TDRE,*                wait for tx ready
BF0A: 97 2F                     STAA     SCDR                       send data
BF0C: 1808                      INY                                 next address
BF0E: 20 F3                     BRA      UPLOAD                     loop (forever)...

                       ************************************************************
                       * PROGRAM - Utility subroutine to program EPROM.           *
                       * Prior to calling PROGRAM set baud rate, turn on SCI      *
                       * set X=2ms prog delay constant, and set Y=first           *
                       * address to program. SP must point to RAM.               *
                       * Bootloader leaves baud set, SCI enabled, X=4200          *
                       * and Y pointing at EPROM start ($D000) so these           *
                       * default values don't have to be changed typically.       *
                       * Delay constant in X should be equivalent to 2 ms         *
                       * at 2.1 MHz X=4200; at 1 MHz X=2000.                      *
                       * At external voltage source is required for EPROM         *
                       * programming.                                             *
                       * This routine uses 2 bytes of stack space                 *
                       * Routine does not return. Reset to exit.                  *

BF10: 13 2E 80 FC      PRGROUT  BRCLR    SCSR,TDRE,*                wait for tx ready
BF14: 86 FF                     LDAA     #$FF                       send $ff to host to start
BF16: 97 2F                     STAA     SCDR
BF18: 13 2E 20 FC      WAIT1    BRCLR    SCSR,RDRF,*                wait for eprom data
BF1C: D6 2F                     LDAB     SCDR
BF1E: 18E1 00                   CMPB     0,Y                        already programmed?
BF21: 27 1D                     BEQ      DONEIT                     yes, skip pgm steps
BF23: 86 20                     LDAA     #ELAT                      put eprom in latch mode
BF25: 97 3B                     STAA     PPROG
BF27: 18E7 00                   STAB     0,Y                        store data
BF2A: 86 21                     LDAA     #ELAT+PGM                  put eprom in pgm mode
BF2C: 97 3B                     STAA     PPROG
BF2E: 3C                        PSHX
BF2F: 8F                        XGDX                                get 2.1 ms delay const in D
BF30: 38                        PULX                                keep delay in X
BF31: D3 0E                     ADDD     TCNT
BF33: DD 16                     STD      TOC1                       set 2.1 ms time delay
BF35: 86 80                     LDAA     #OC1F
BF37: 97 23                     STAA     TFLG1                      pre-clear flag
BF39: 13 23 80 FC               BRCLR    TFLG1,OC1F,*               wait for delay to expire
BF3D: 7F 003B                   CLR      PPROG                      turn off pgm voltage
BF40: 13 2E 80 FC      DONEIT   BRCLR    SCSR,TDRE,*                wait for tx ready
BF44: 18A6 00                   LDAA     0,Y                        get EPROM byte
BF47: 97 2F                     STAA     SCDR                       and send to host
BF49: 1808                      INY                                 go to next eprom location
BF4B: 20 CB                     BRA      WAIT1                      loop,
                                    ************************************************************
                                    * Main bootloader starts here out of reset.                *
                                    * Initializes SCI for 7812.5 baud, issues a break and      *
                                    *  waits for a response.  $FF at 7812.5 or 1200 baud will  *
                                    *  initiate dowload sequence to MCU RAM.  Download is at   *
                                    *  either 7812.5, or 1200 (if $FF was sent at any rate     *
                                    *  slower than about 7000 baud, the download will be at    *
                                    *  1200 baud). After 192 bytes, or a pause of greater than *
                                    *  5.12ms, the code loads Y with a timing constant, X with *
                                    *  $F000, and jumps to start of RAM.                       *
                                    * A response of $00 causes a JMP to $F000.                 *

BF4D: 8E 00FF           BEGIN    LDS      #RAMEND                    set stack
BF50: 14 28 20                   BSET     SPCR,DWOM
BF53: CC A20C                    LDD      #$A20C
BF56: 97 2B                      STAA     BAUD                       set BAUD = 7812.5
BF58: D7 2D                      STAB     SCCR2                      set SCCR2 = RE+TE
BF5A: CC 021B                    LDD      #DELAYF                    set fast baud delay
BF5D: DD 16                      STD      TOC1                       (5.12ms @ E=2MHz)
BF5F: 14 2D 01                   BSET     SCCR2,SBK                  send break
BF62: 12 08 01 FC                BRSET    PORTD,RXD,*                watch for RXD low
BF66: 15 2D 01                   BCLR     SCCR2,SBK
BF69: 13 2E 20 FC                BRCLR    SCSR,RDRF,*                wait for rcv data
BF6D: 96 2F                      LDAA     SCDR                       get rcv data
BF6F: 26 03                      BNE      NOTZERO                    no break rcvd,
BF71: 7E F000                    JMP      EPRMSTR                    break rcvd, jmp to EPROM

BF74: 81 FF             NOTZERO  CMPA     #$FF                       was $ff rcvd?
BF76: 27 08                      BEQ      BAUDOK                     yes,
BF78: 14 2B 33                   BSET     BAUD,SCP1+SCP0+SCR1+SCR0   set 1200 baud
BF7B: CC 0DB0                    LDD      #DELAYS                    set slow baud delay (33.29ms)
B7BE: DD 16                      STD      TOC1
BF80: 18CE 0040         BAUDOK   LDY      #RAMSTR                    set start of RAM
BF84: DE 16             WAIL     LDX      TOC1                       dly count (19~~ = 9.5us/loop)
BF86: 12 2E 20 09       WTLOOP   BRSET    SCSR,RDRF,NEWONE           ~6 got data,
BF8A: 09                         DEX                                 ~3 decrement delay loop count
BF8B: 01                         NOP                                 ~2  { run these idle instr
BF8C: 01                         NOP                                 ~2  { to establish delay
BF8D: 21 00                      BRN      NOWHERE                    ~3  { increment
BF8F: 26 F5             NOWHERE  BNE      WTLOOP                     ~3 loop until X = 0
BF91: 20 0F                      BRA      STAR                       timeout, quit loop

BF93: 96 2F             NEWONE   LDAA     SCDR                       get rcv data
BF95: 18A7 00                    STAA     0,Y                        put in RAM
BF98: 97 2F                      STAA     SCDR                       echo to host
BF9A: 18 08                      INY                                 next RAM location
BF9C: 188C 0100                  CPY      #RAMEND+1                  is last?
BFA0: 26 E2                      BNE      WAIL                       no, keep looping

BFA2: CE 1068           STAR     LDX      #PROGDEL                   set EPROM delay const
BFA5: 18CE F000                  LDY      #EPRMSTR                   set start of EPROM
BFA9: 7E 0040                    JMP      RAMSTR                     jump to RAM

                                    *****************************************************
                                    * block fill unused bytes with 0's

BFAC:   00 00 00 00 00 00        BSZ      $BFD1-*
        00 00 00 00 00 00
        00 00 00 00 00 00
        00 00 00 00 00 00
        00 00 00 00 00 00
        00 00 00 00 00 00
        00
```

```
                      ****************************************************
                      * 711D3 Bootrom revision
BFD1: 42                       FCC      "B"

                      ****************************************************
                      * Mask set ID ($0000 for EPROM parts)
BFD2: 0000                     FDB      $0000

                      ****************************************************
                      * 711D3 ID - used to determine MCU type
BFD4: 71D3                     FDB      $71D3

                      ****************************************************
                      * 711D3 Vectors point to RAM jump table (except RESET)
BFD6: 00C4                     FDB      $00C4                            SCI
BFD8: 00C7                     FDB      $00C7                            SPI
BFDA: 00CA                     FDB      $00CA                            PAI edge
BFDC: 00CD                     FDB      $00CD                            PAIOF
BFDE: 00D0                     FDB      $00D0                            TOF
BFE0: 00D3                     FDB      $00D3                            TOC5/IC4
BFE2: 00D6                     FDB      $00D6                            TOC4
BFE4: 00D9                     FDB      $00D9                            TOC3
BFE6: 00DC                     FDB      $00DC                            TOC2
BFE8: 00DF                     FDB      $00DF                            TOC1
BFEA: 00E2                     FDB      $00E2                            TIC3
BFEC: 00E5                     FDB      $00E5                            TIC2
BFEE: 00E8                     FDB      $00E8                            TIC1
BFF0: 00EB                     FDB      $00EB                            RTI
BFF2: 00EE                     FDB      $00EE                            IRQ
BFF4: 00F1                     FDB      $00F1                            XIRQ
BFF6: 00F4                     FDB      $00F4                            SWI
BFF8: 00F7                     FDB      $00F7                            ILLOP
BFFA: 00FA                     FDB      $00FA                            COPF
BFFC: 00FD                     FDB      $00FD                            CMON
BFFE: BF4D                     FDB      $BF4D                            RESET

                               END
```

711D3 bootrom S-record listing:

```
S123BF007EBF1018A600132E80FC972F180820F3132E80FC86FF972F132E20FCD62F18E1F9
S123BF2000271D8620973B18E7008621973B3C8F38D30EDD1686809723132380FC7F003BC6
S123BF40132E80FC18A600972F180820CB8E00FF142820CCA20C972BD72DCC021BDD161478
S123BF602D01120801FC152D01132E20FC962F26037EF00081FF2708142B33CC0DB0DD16DF
S123BF8018CE0040DE16122E2009090101210026F5200F962F18A700972F1808188C010095
S123BFA026E2CE106818CEF0007E0040000000000000000000000000000000000000009B
S123BFC000000000000000000000000000000000000042000071D300C400C700CA00CD00D0E5
S123BFE000D300D600D900DC00DF00E200E500E800EB00EE00F100F400F700FA00FDBF4D99
S9030000FC
```