EDK2282

USER MANUAL

FOR H8S/2282
ON-CHIP FLASH MICROCONTROLLER

Preface

Cautions

1. This document may be, wholly or partially, subject to change without notice.
2. All rights reserved. No one is permitted to reproduce or duplicate, in any form, a part or this entire document without Hitachi Micro Systems Europe Limited's written permission.

Trademarks

General
All brand or product names used in this manual are trademarks or registered trademarks of their respective companies or organisations.

Specific
Microsoft, MS and MS-DOS are registered trademarks and Windows and Windows NT are trademarks of Microsoft Corporation.

Document Information

Product Code: D004163_11

Version: 1.0

Date: 26/09/2002

Global: http://www.hitachisemiconductor.com

Europe: http://www.hmse.com

1. TABLE OF CONTENTS

2. START-UP INSTRUCTIONS

2.1. INSTALLING THE EVALUATION DEVELOPMENT KIT (EDK)

Please refer to the quick start guide provided for initial installation of the EDK.

A copy of the quick start guide and other information relating to this EDK at:

http://www.hmse.com/products/support.htm

Installing the EDK requires power and COM1serial connection to a host computer.

2.2. SERIAL CONNECTION

The serial communications cable for connecting the EDK to a host computer is supplied. The serial cable has 1:1 connectivity.

Figure 2-1 shows how to connect the EDK to a PC or notebook computer equipped with a nine pin D connector.

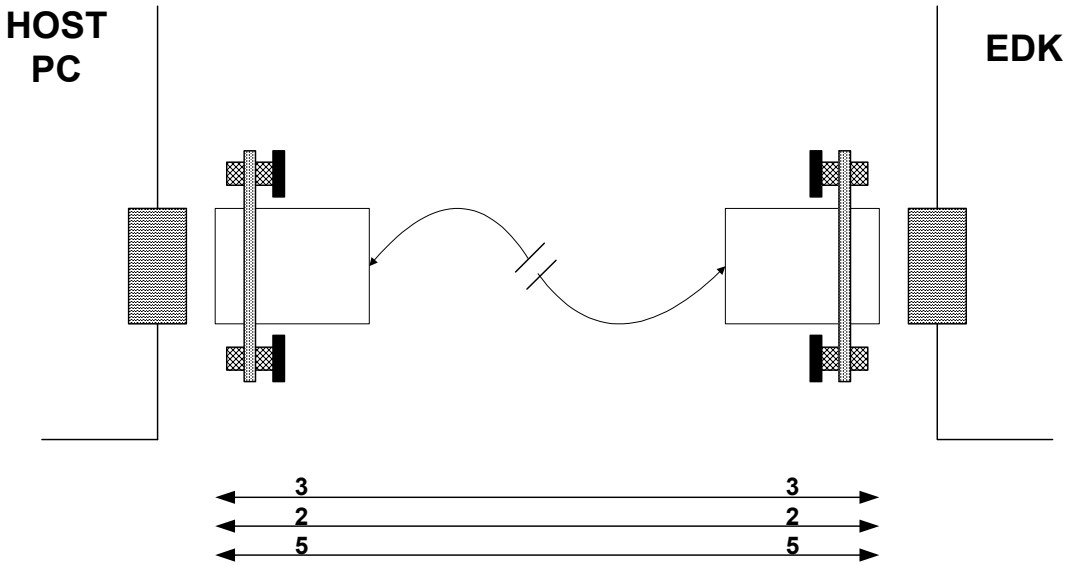


FIGURE 2-1: SERIAL CONNECTION TO PC/NOTEBOOK WITH DB-9 CONNECTOR (SUPPLIED)

2.3. POWER SUPPLY

The EDK hardware requires a power supply of +5V. Since total power consumption can vary widely due to external connections, port states, and memory configuration, use a power supply capable of providing at least 500mA at +5V DC ± 5%.

The design is specified for evaluation of the microcontroller and so does not include circuitry for supply filtering/noise reduction, under voltage protection, over current protection or reversed polarity protection. Caution should be used when selecting and using a power supply.

The power connector on the EDK is a 2.5mm Barrel connector. The center pin is the positive connection.

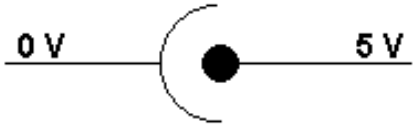


FIGURE 2-2: POWER SUPPLY CONNECTION

Caution: Existing customers using E6000 products note that the polarity of this board is opposite to that for the E6000. Use of the E6000 power supply with this board will damage both board and power supply.

3. EDK BOARD LAYOUT

The diagram shows a general layout of the EDK board.

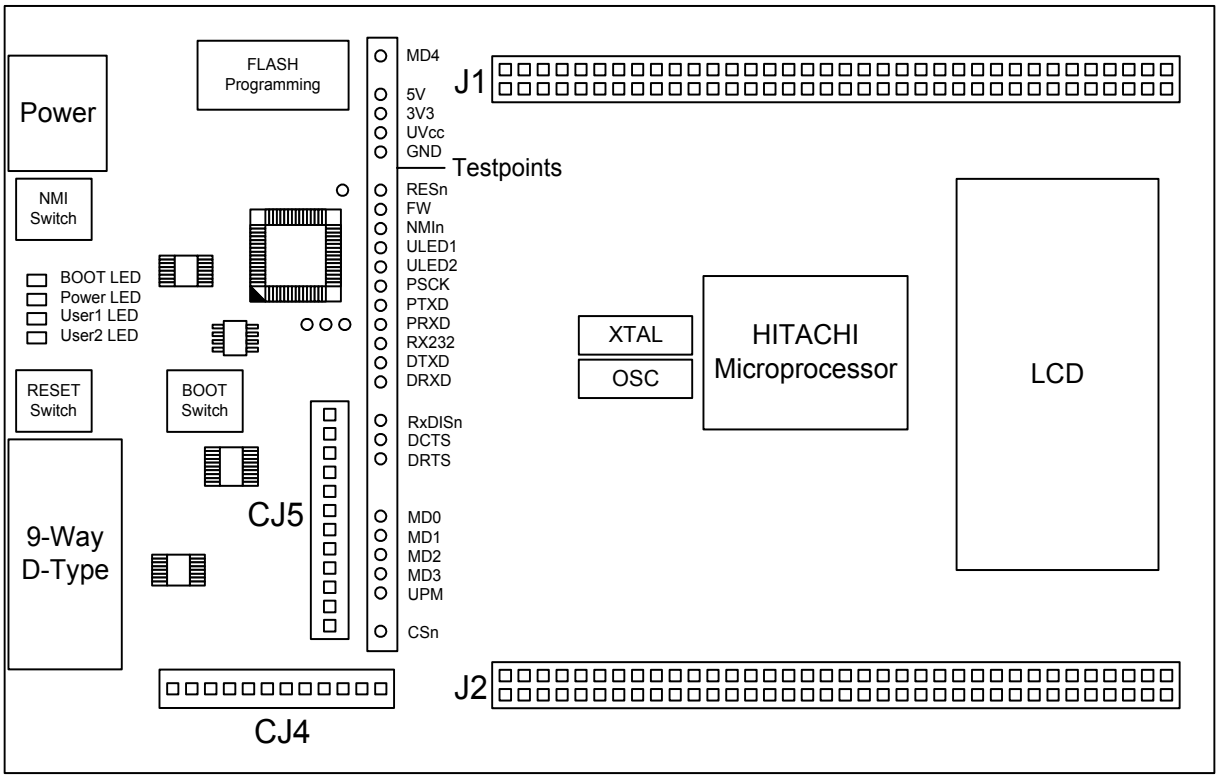


FIGURE 3-1: EDK BOARD LAYOUT

3.1. EDK BLOCK DIAGRAM

The diagram shows the connectivity of the components on the EDK board.

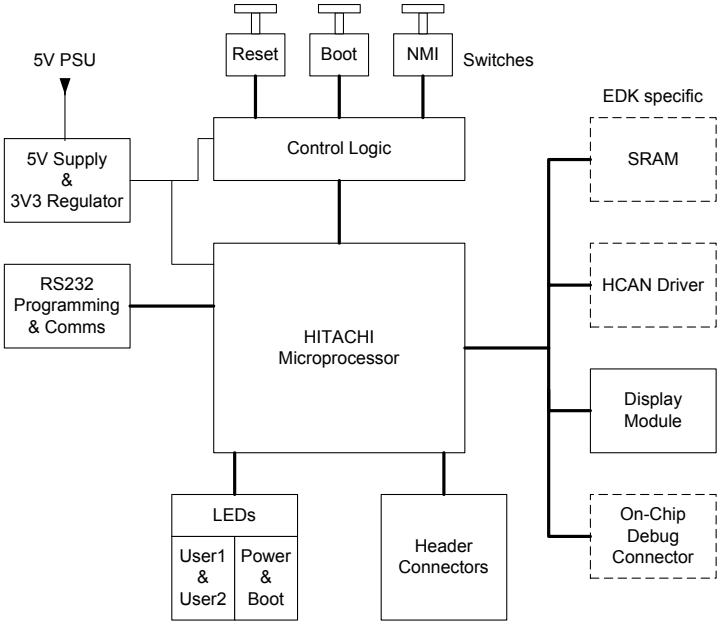


FIGURE 3-2: EDK BLOCK DIAGRAM

4. EDK OPERATION

4.1. USER INTERFACE

The EDK provides three buttons for influencing the operation of the board. The purpose of each button is clearly marked next to it. Refer to the board layout for positions (Section 3)

1. Reset Switch

This button provides the microcontroller with a timed reset pulse of at least 250mS.

2. Boot Switch

This button toggles the operating mode of the microcontroller. A complete description of this function is given in section 5.7.

3. NMI Switch

This button provides a de-bounced signal to the microcontroller for each operation of the button. There is no minimum or maximum activation time for this button.

4.2. SERIAL INTERFACE

The serial interface on the EDK board has several functions. The serial port on the microcontroller directly supports three wire serial interfaces. Options are provided on the board for the user to write handshaking routines using standard port pins. Other board option links allow users to control the entry and exit from boot mode using the same handshaking signals. Refer to section 5 for details on setting serial interface options.

4.2.1. CONNECTOR PIN DEFINITIONS

The EDK RS232 interface conforms to Data Communication Equipment (DCE) format allowing the use of 1-1 cables when connected to Data Terminal Equipment (DTE) such as an IBM PC. The cable used to connect to the EDK will affect the available board options. A fully wired cable can allow handshaking between the microcontroller and the host PC, subject to setting the board options and the availability of suitable host software. Handshaking is not supported as standard on the microcontroller so for normal use a minimal three-wire cable can be used. The minimum connections are unshaded in the following table.

EDK DB9 Connector Pin	Signal	Host DB9 Connector Pin
1	No Connection	1
2	EDK Tx Host Rx	2
3	EDK Rx Host Tx	3
4	No Connection	4
5	Ground	5
6	No Connection	6
7	* EDK CTS Host RTS	7
8	* EDK RTS Host CTS	8
9	No Connection	9

TABLE 4-1: RS232 INTERFACE CONNECTIONS

* These are not connected on the EDK by default. See section 5.4 for more details.

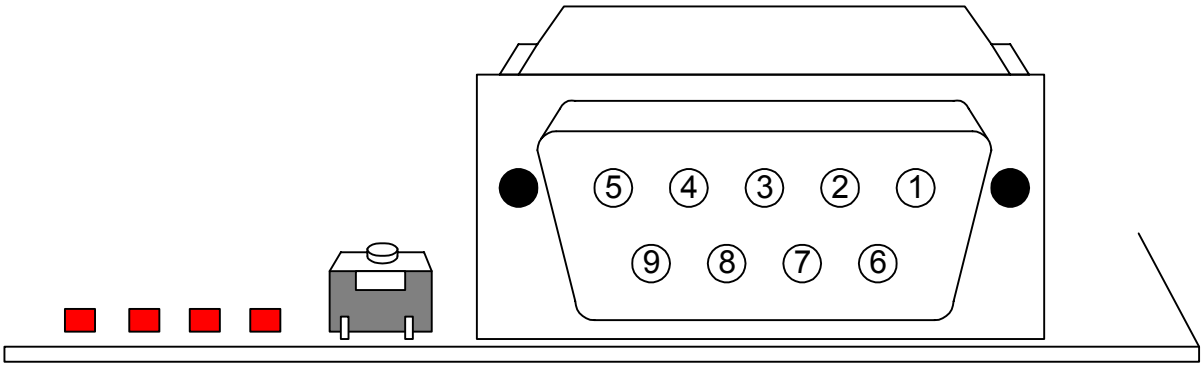


FIGURE 4-1: EDK SERIAL PORT PIN NUMBERING

4.2.2. CRYSTAL CHOICE

The operating crystal frequency has been chosen to support the fastest operation with a 500k baud CAN data rate. The value of the crystal is 20MHz. This affects the maximum possible serial baud rate.

The following table shows the baud rates and Baud Rate Register (BRR) setting required for each communication rate using the above default operating speed. It also confirms the resultant baud rate and the bit error rate that can be expected.

Baud Rate Register Settings for Serial Communication Rates												
SMR Setting:	**0**			**1**			**2**			**3**		
Comm. Baud	**BRR setting**	**Actual Rate**	**ERR (%)**	**BRR setting**	**Actual Rate**	**ERR (%)**	**BRR setting**	**Actual Rate**	**ERR (%)**	**BRR setting**	**Actual Rate**	**ERR (%)**
110	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	88	110	-0.25
300	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	129	300	0.16	32	296	-1.36
1200	Invalid	Invalid	Invalid	129	1202	0.16	32	1184	-1.36	7	1221	1.73
2400	Invalid	Invalid	Invalid	64	2404	0.16	15	2441	1.73	3	2441	1.73
4800	129	4808	0.16	32	4735	-1.36	7	4883	1.73	1	4883	1.73
9600	64	9615	0.16	15	9766	1.73	3	9766	1.73	0	9766	1.73
19200	32	18939	-1.36	7	19531	1.73	1	19531	1.73	Invalid	Invalid	Invalid
38400	15	39063	1.73	3	39063	1.73	0	39063	1.73	Invalid	Invalid	Invalid
57600	10	56818	-1.36	2	52083	-9.58	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
115200	4	125000	8.51	0	156250	35.63	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
230400*	2	208333	-9.58	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid
460800*	0	625000	35.63	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid	Invalid

TABLE 4-2 CRYSTAL FREQUENCIES FOR RS232 COMMUNICATION

** Note: The device used to convert the RS232 serial information to logic signals for the microcontroller is limited to 120kBaud. The rates above this level can only be utilised if the user provides direct logic level communications.*

The user may replace the HC49/U surface mounted AT cut crystal with another of similar type within the operating frequency of the microcontroller device. Please refer to the hardware manual for the microcontroller for the valid operating range.

Alternatively the user may fit an oscillator module – or provide an external clock source. When providing an oscillator module or external source it is highly recommended that the load capacitors for the AT crystal are removed from the PCB. These are physically placed within the PCB outline of the oscillator module for easy location and to ensure they are removed when using this option.

When changing the crystal frequency the pre-loaded debugging monitor will not function. In this situation the user is responsible for providing code to evaluate the device away from the default operating speed.

4.2.3. REMOVABLE COMPONENT INFORMATION.

This information is provided to allow the replacement of components removed from the board as described in section 4.2.2.

Component	Cct. Ref	Value	Rating	Manufacturer
Load Resistor (X1)	R9	1MΩ	0805 1%	Welwyn WCR Series
Load capacitors (X1)	C1,C2	22pF	0603 10% 25V	AVX 0603 3 A 220 KAT

TABLE 4-3: REMOVABLE COMPONENT INFORMATION

Care must be taken not to damage the tracking around these components. Only use soldering equipment designed for surface mount assembly and rework.

4.3. SRAM

This EDK does not support external address and data bus connections so there is no SRAM on this EDK

4.4. MEMORY MAP

Table 4-4 illustrates the EDK memory map for mode 7.

Section Start Section End	Section Allocation
H'000000	On Chip ROM (FLASH)
H'01FFFF	
H'020000	Reserved
H'FFDFFF	
H'FFE000	On-Chip RAM
H'FFEFBF	
H'FFEFC0	Reserved
H'FFF7FF	
H'FFF800	Internal I/O Registers
H'FFFF3F	
H'FFFF40	Reserved
H'FFFF5F	
H'FFFF60	Internal I/O Registers
H'FFFFBF	
H'FFFFC0	On-Chip RAM
H'FFFFFF	

TABLE 4-4: MEMORY MAP (DEFAULT MODE 7)

4.5. LEDS

The EDK has four red LEDs. The function of each LED is clearly marked on the silk screen of the PCB. Please refer to the board layout diagram for position information (Section 3).

When the board is connected to a power source the Power (PWR) led will illuminate. The Boot mode indication LED will illuminate when the microcontroller has been placed into Boot mode. Please see section 5.6 for more details of this function.

There are two LEDs dedicated for user control these are marked USR1 and USR2. Each LED will illuminate when the port pin is in a logical high state.

The user LEDs are connected to the following ports:

LED Identifier	Port Pin	Microcontroller Pin	Pin Functions on Port Pin
USR1	P10	57	TIOCA0
USR2	P11	58	TIOCB0

TABLE 4-5: LED PORT CONNECTIONS

5. BOARD OPTIONS

The EDK has a number of configuration settings set by jumpers CJ4 (A, B, C, D) CJ5 (A, B, C, D) and zero-ohm links. Common EDK functions can be set using the jumpers as described in sections 5.3 and 5.2. The additional zero-ohm links provide additional features that may be required to interface with other systems.

All the Jumper link settings are three pin options. There are four sets of options on each header.

The headers are numbered from 1 to 12 with pin 1 marked on the PCB by an arrow pointing to the pin. The diagram below shows the numbering of these jumper links and indicates jumpers fitted 1-2 for each three-pin jumper.

5.1. JUMPER LINKS

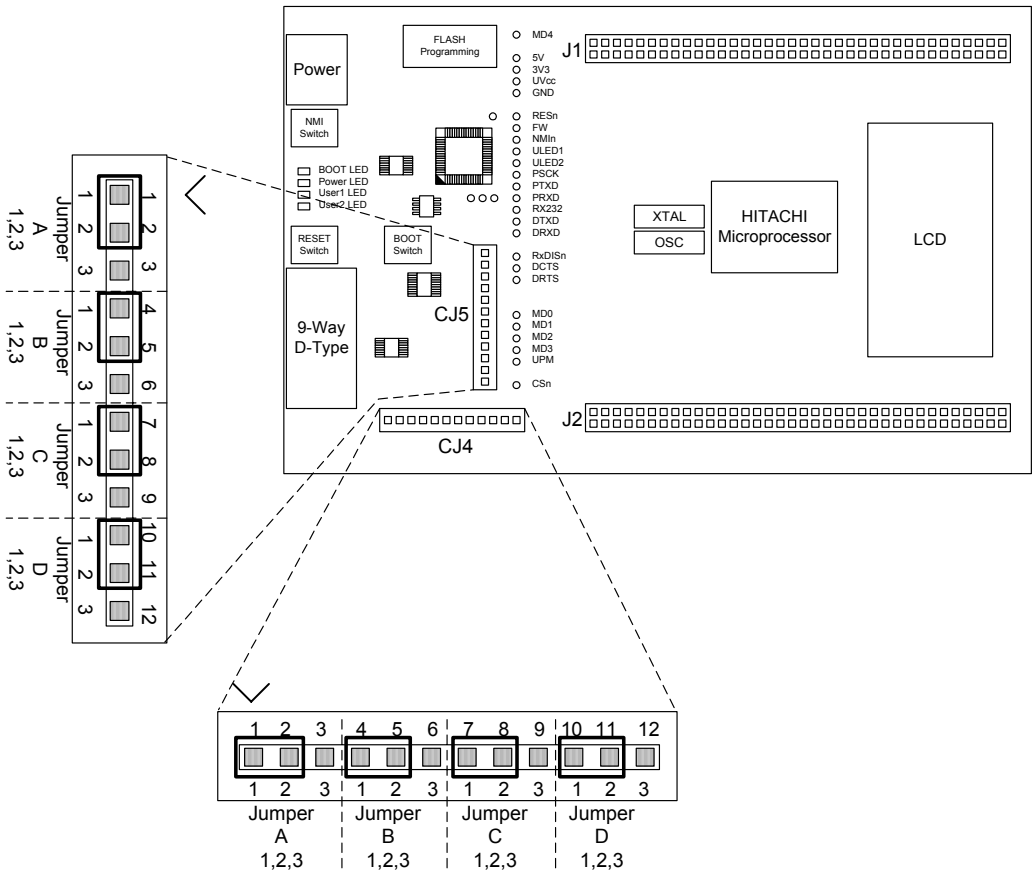


FIGURE 5-1: JUMPER CONFIGURATION

The following tables define each jumper and its settings.

5.2. USER MODE SETTINGS – CJ5

CJ5 is used to set the operating mode of the microcontroller.

These jumpers must be fitted at all times to ensure correct operation of the EDK.

Jumper	Function	Setting 1-2	Setting 2-3
CJ 5-A Default 1-2	User Mode Setting Bit 0	**MD0 pulled High**	MD0 pulled Low
CJ 5-B Default 1-2	User Mode Setting Bit 1	**MD1 pulled High**	MD1 pulled Low
CJ 5-C Default 1-2	User Mode Setting Bit 2	**MD2 pulled High**	MD2 pulled Low
CJ 5-D Default 1-2	User Mode Setting Bit 3	**MD3 pulled High**	MD3 pulled Low

TABLE 5-1: USER MODE: JUMPER SETTINGS (DEFAULT SETTINGS IN BOLD)

The default settings indicated in bold text place the microcontroller into Mode 7.

5.3. EDK OPTIONS – CJ4

The EDK options provide access to commonly used features of the EDK range.

These jumpers must be fitted at all times to ensure correct operation of the EDK.

Jumper	Function	Setting 1-2	Setting 2-3
CJ 4-A Default 2-3	Serial Receive Source	Disables the RS232 receive signal to enable the use of the Flash Programming Header	**Enables the RS232 receive signal. The Flash Programming Header* must not be used in this state.**
CJ 4-B Default 2-3	User Programming Mode	Disables the Flash write hardware protection. The flash can be overwritten in User Mode.	**Enables the Flash write hardware protection. The flash can not be overwritten in User Mode.**
CJ 4-C	Not Used		
CJ 4-D	Not Used		

TABLE 5-2: BOARD OPTION: JUMPER SETTINGS (DEFAULT SETTINGS IN BOLD)

*See section 5.6

The following table lists the connections to each jumper pin.

Pin	Net Name	Description
1	UVCC	Microcontroller Supply Voltage
2	RXDISn	Disable Flash Header functions. Pulled low. (Enables RX232)
3	No Connection	No Connection
4	UVCC	Microcontroller Supply Voltage
5	UPM	CPLD Controlled option to set Flash Write (FW). Pulled low.
6	No Connection	No Connection
7	No Connection	No Connection
8	No Connection	No Connection
9	No Connection	No Connection
10	No Connection	No Connection
11	No Connection	No Connection
12	No Connection	No Connection

5.4. LCD

The H8/2282 features a 28 segment LCD driver.

Two different LCD footprints are to be provided on the EDK2282 PCB (only one in use at any one time):

- VI-422-DPRC (direct drive 4 digits) (NORMALLY FITTED)
- VI-503-DPRC-LV (multiplexed 4.5 digits)

The H8S/2282 internal LCD controller is connected as follows:

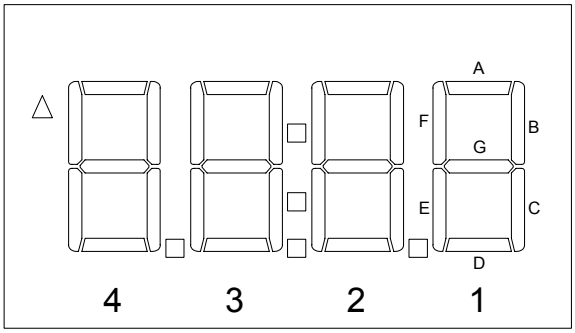
H8S/2282 Pin	LCD Controller Function	VI-422 Pin	VI-422 Function	VIM-503 Pin	VIM-503 Function	H8S/2282 Pin	LCD Controller Function
3	SEG1	21	1A	12	1B-1C-CONT	3	SEG1
4	SEG2	20	1B	11	1A-1G-1D	4	SEG2
5	SEG3	19	1C	10	1F-1E-2DP	5	SEG3
6	SEG4	18	1D	9	2B-2C-LOW	6	SEG4
7	SEG5	17	1E	8	2A-2G-2D	7	SEG5
8	SEG6	22	1F	7	2F-2E-3DP	8	SEG6
9	SEG7	23	1G	6	3B-3C-MIN	9	SEG7
10	SEG8	25	2A	5	3A-3G-3D	10	SEG8
11	SEG9	24	2B	4	3F-3E-4DP	11	SEG9
12	SEG10	15	2C	3	4B-4C-BC	12	SEG10
13	SEG11	14	2D	2	4A-4G-4D	13	SEG11
14	SEG12	13	2E	1	4F-4E-5DP	14	SEG12
15	SEG13	26	2F	-	-	15	SEG13
16	SEG14	27	2G	-	-	16	SEG14
17	SEG15	30	3A	-	-	17	SEG15
18	SEG16	29	3B	-	-	18	SEG16
21	SEG17	11	3C	-	-	21	SEG17
22	SEG18	10	3D	-	-	22	SEG18

Both common pins on the VI-422 LCD footprint are connected to COM1 of the LCD controller along with the 'ARROW' symbol to use only 4 ports for the LCD and leave port N free. This also means that for the clock tutorial, as requested by HEL, the colon is available.

COM4 from the H8S/2268 is tracked to pin (no. 16) on the VIM-503 LCD footprint – this allows multiplexed LCDs that use four common pins to be fitted instead. The VIM-503 only uses 15 pins, therefore pin 16 is a 'not connect' under normal use.

The Pin out of the VI-422 normally fitted is;

Pin No	Segment	Pin No	Segment	Pin No	Segment	Pin No	Segment
1	COM	11	3C	21	1A	31	3F
2	NC	12	3DP	22	1F	32	3G
3	NC	13	2E	23	1G	33	NC
4	NC	14	2D	24	2B	34	4B
5	4E	15	2C	25	2A	35	4A
6	4D	16	2DP	26	2F	36	4F
7	4C	17	1E	27	2G	37	4G
8	4DP	18	1D	28	L	38	UP-Arrow
9	3E	19	1C	29	3B	39	NC
10	3D	20	1B	30	3A	40	COM



5.5. SERIAL PORT SELECTION

The programming serial port is connected to the RS232 connector by default. This allows direct programming of the EDK using the supplied software tools. A secondary serial port is available on the microcontroller and can be connected to the RS232 connector by changing some board option links. The additional port option allows the user to write messages or connect to other devices via the serial port while programming support is provided by the Flash programming header.

The following surface mount, zero-ohm link settings are fitted by default and connect the RS232 header to the programming serial port of the microcontroller.

Zero-ohm Link ID	Default	Function	Microcontroller Port Pin
CR20	Fitted	Transmit data from EDK	P33
CR23	Fitted	Receive data to EDK	P34
CR19	Not Fitted	Alternate Transmit data from EDK	P30
CR22	Not Fitted	Alternate Receive data to EDK	P31

TABLE 5-3: OPTION LINKS – DEFAULT SETTINGS

To enable the use of this alternate port the user must change the settings to those in the following table.

Zero-ohm Link ID	Default	Function	Microcontroller Port Pin
CR20	Not Fitted	Transmit data from EDK	P33
CR23	Not Fitted	Receive data to EDK	P34
CR19	Fitted	Alternate Transmit data from EDK	P30
CR22	Fitted	Alternate Receive data to EDK	P31

TABLE 5-4: OPTION LINKS – ALTERNATE SERIAL PORT

The user may implement a handshaking protocol on the EDK. This is not supported with the software tools supplied. To support this option two spare port pins have been allocated on the microcontroller. Using these port pins the CTS and RTS lines of the host serial interface can be controlled.

The user may also control the operation of the board via the same handshaking lines. This is not supported with the software tools supplied but may be written by the user. Using the CTS line the user may simulate pressing the boot button, see section:5.7. This will cause the EDK to swap into and out of Boot mode on each low-level activation of CTS. Feedback of the current mode is provided on the RTS line. A high level indicates boot mode and a low level indicates user mode.

The following settings are made by default, and ensure that there are no conflicts on unnecessary microcontroller pins.

Zero-ohm Link ID	Default	Function	Microcontroller Port Pin
CR12	Not Fitted	Mode State out from EDK	N/A (From CPLD*)
CR7	Not Fitted	Change Mode request to EDK	N/A (From CPLD*)
CR16	Not Fitted	Alternate RTS232 – Ready to send – from EDK	P15
CR13	Not Fitted	Alternate CTS232 – Clear to send – to EDK	P32

TABLE 5-5: OPTION LINKS – SERIAL PORT CONTROL

* See section 5.7

Note: These setting pairs are exclusive:
If CR12 and CR7 are fitted; CR16 and CR13 must not be fitted.
If CR16 and CR13 are fitted; CR12 and CR7 must not be fitted.

5.6. FLASH PROGRAMMING HEADER

The Flash Programming header is used with the Hitachi Flash Debug Board (FDB). The FDB is a USB based programming tool for control and programming of Hitachi microcontrollers, available separately from Hitachi. This header provides direct access for the FDB to control the EDK microcontroller.

To utilise this header the user must make the following changes to the board configuration.

1. Disable the RX232 signal from the RS232 transceiver.
   Jumper link CJ4-A is provided for this purpose. Please refer to section5.3.
2. Disable User Program Mode using jumper CJ4-B. Please refer to section5.3.

Caution: Do not operate the board with the user mode jumpers removed and the FDB disconnected as the microcontroller mode pins will float to an indeterminate state. This may damage the microcontroller device.

5.7. BOOT CONTROL

The method for placing the microcontroller device in to Boot mode for reprogramming has been incorporated into a complex programmable logic device (CPLD). This is not necessary for most user designs but allows a measure of increased flexibility for the EDK designs. Mode transitions including boot mode transitions only require the reset to be held active while the mode settings are presented. On releasing reset the microcontroller will be in the required mode.

The logic design detects a power up event and provides a timed reset pulse to guarantee the reset of the device. At the end of the rest pulse the processor will be placed in user mode and any code in the device will execute.

During user mode the NMI button can be pressed at any time. This will provide a single de-bounced NMI interrupt to the device.

Pressing the boot button will cause the boot mode controller to reset the device and, during the reset period, present the required mode settings to start the device in boot mode. At the end of the reset period the boot mode settings will have been latched into the device which will then be ready to accept a boot mode connection via the RS232 interface or the flash programming header. Pressing the boot button during a normal reset will not cause the EDK to enter boot mode.

The boot mode settings are fixed at mode 0. The required mode settings are made using a tri-state capable buffer.

Note: The boot control device is programmed to support all possible EDK products.
For this reason the reset pulse is over 500ms. Repetitive activation of either the Boot or Reset buttons will restart the reset timer and extend the reset period. Pressing the boot button within the 500mS period of a reset will not cause the board to enter boot mode.

5.7.1. CPLD CODE

The code is based upon a four state machine providing a guaranteed reset period which can be extended by holding the relevant control input in the active state. When released the timer will extend the reset for approximately 500mS.

The states are split into two functions, one for User mode and one for Boot mode. The first state of each is used to hold the reset line active. When the timer expires then the second state is used to hold the device in the selected mode and wait for an external control signal to either move back into the user reset state or into the boot reset state.

5.7.2. STATE DIAGRAM

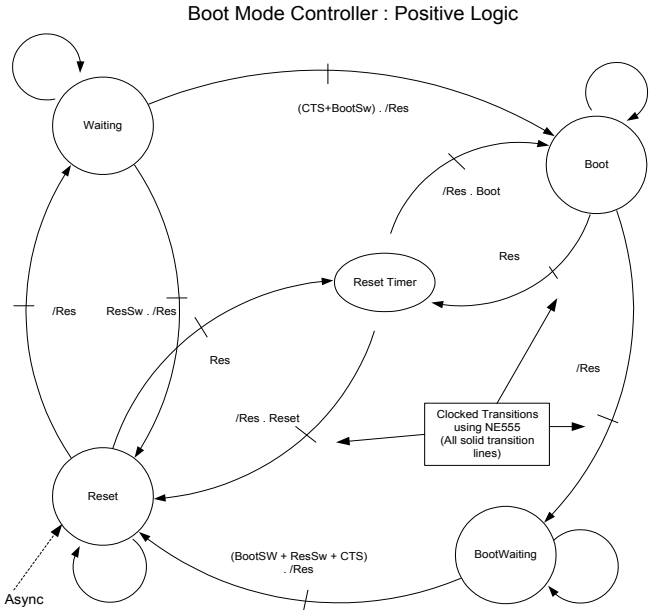


FIGURE 5-2: CPLD STATE DIAGRAM

6. MICROCONTROLLER HEADER CONNECTIONS

The following table lists the connections to each or the headers on the board.

6.1. HEADER J1

J1							
Pin No	**Function**	**EDK Symbol**	**Device pin**	**Pin No**	**Function**	**EDK Symbol**	**Device pin**
1	RESn	RESn	73	2	VSS	GROUND	74
3	VCL	NC71	71	4	FWE	FW	72
5	PLLCAP	NC69	69	6	PLLVSS	GROUND	70
7	NMI	NMIn	67	8	STBYn	STBYn	68
9	MD2	MD2	65	10	MD0	MD0	66
11	P16, TIOCA2, IRQ1n	P16	63	12	P17, TIOCB2, TCLKD	P17	64
13	P14, TIOCA1, IRQ0n	P14	61	14	P15, TIOCB1, TCLKC	DRTS	62
15	P12, TIOCD0, TCLKA	P12	59	16	P13, TIOCD0, TCLKB	P13	60
17	P10, TIOCA0	ULED1	57	18	P11, TIOCB0	ULED2	58
19	PJ6, PWM2G	PJ6	55	20	PJ7, PWM2H	PJ7	56
21	PJ4, PWM2E	PJ4	53	22	PJ5, PWM2F	PJ5	54
23	PWMVSS	GROUND	51	24	PWMVCC	UVCC	52
25	PJ2, PWM2C	PJ2	49	26	PJ3, PWM2D	PJ3	50
27	PJ0, PWM2A	PJ0	47	28	PJ1, PWM2B	PJ1	48
29	PH6, PWM1G	PH6	45	30	PH7, PWM1H	PH7	46
31	PH4, PWM1E	PH4	43	32	PH5, PWM1F	PH5	44
33	PWMVSS	GROUND	41	34	PWMVCC	UVCC	42
35	PH2, PWM1C	PH2	39	36	PH3, PWM1D	PH3	40
37	PH0, PWM1A	PH0	37	38	PH1, PWM1B	PH1	38
39	PA2, COM3	PA2	35	40	PA3, COM4	PA3	36
41	PA0, COM1	PA0	33	42	PA1, COM2	PA1	34
43	PA6, SEG27	PA6	31	44	PA7, SEG28	PA7	32
45	PA4, SEG25	PA4	29	46	PA5, SEG26	PA5	30
47	PF5, SEG23	PF5	27	48	PF6, SEG24	PF6	28
49	PF2, SEG21	PF2	25	50	PF4, SEG22	PF4	26

6.2. HEADER J2

J2							
Pin No	**Function**	**EDK Symbol**	**Device pin**	**Pin No**	**Function**	**EDK Symbol**	**Device pin**
1	EXTAL	CON_Extal	75	2	XTAL	CON_Xtal	76
3	VCC	UVcc	77	4	PF7, PHI	PF7	78
5	PF3, ADTRGn, IRQ3n	PF3	79	6	P30, TXD0	DTXD	80
7	P31, RXD0	DRXD	81	8	P32, SCK0, IRQ4n	DCTS	82
9	P33, TXD1	PTXD	83	10	P34, RxD1	PRXD	84
11	P35, SCK1, IRQ4n	PSCK	85	12	HRXD, IRQ2	HRXD	86
13	HTXD	HTXD	87	14	P40, AN0	P40	88
15	P41, AN1	P41	89	16	P42, AN2	P42	90
17	P43, AN3	P43	91	18	P44, AN4	P44	92
19	P45, AN5	P45	93	20	P46, AN6	P46	94
21	P47, AN7	P47	95	22	AVCC	CON_Avcc	96
23	AVSS	CON_AVss	97	24	V1	V1	98
25	V2	V2	99	26	V3	V3	100
27	VSS	GROUND	1	28	VCC	UVCC	2
29	PD4, SEG1	PD4	3	30	PD5, SEG2	PD5	4
31	PD6, SEG3	PD6	5	32	PD7, SEG4	PD7	6
33	PC0, SEG5	PC0	7	34	PC1, SEG6	PC1	8
35	PC2, SEG7	PC2	9	36	PC3, SEG8	PC3	10
37	PC4, SEG9	PC4	11	38	PC5, SEG10	PC5	12
39	PC6, SEG11	PC6	13	40	PC7, SEG12	PC7	14
41	PB0, SEG13	PB0	15	42	PB1, SEG14	PB1	16
43	PB2, SEG15	PB2	17	44	PB3, SEG16	PB3	18
45	LPVCC	UVCC	19	46	VSS	GROUND	20
47	PB4, SEG17	PB4	21	48	PB5, SEG18	PB5	22
49	PB6, SEG19	PB6	23	50	PB7, SEG20	PB7	24

7. CODE DEVELOPMENT

7.1. HMON

7.1.1. MODE SUPPORT

The HMON library is built to support Advanced Mode only. The Device supports only Mode 7.

7.1.2. BREAKPOINT SUPPORT

The monitor utilises the device RAM for user code requiring breakpoints. Code located in RAM may have multiple breakpoints limited only by the size of the On-Chip RAM.

7.1.2.1. CODE LOCATED IN ROM/FLASH

This device does not have a PC break controller. Breakpoints are limited to compiled in Trap instructions. HEW will not allow breakpoints to be set in the ROM/FLASH area.

7.1.2.2. CODE LOCATED IN RAM

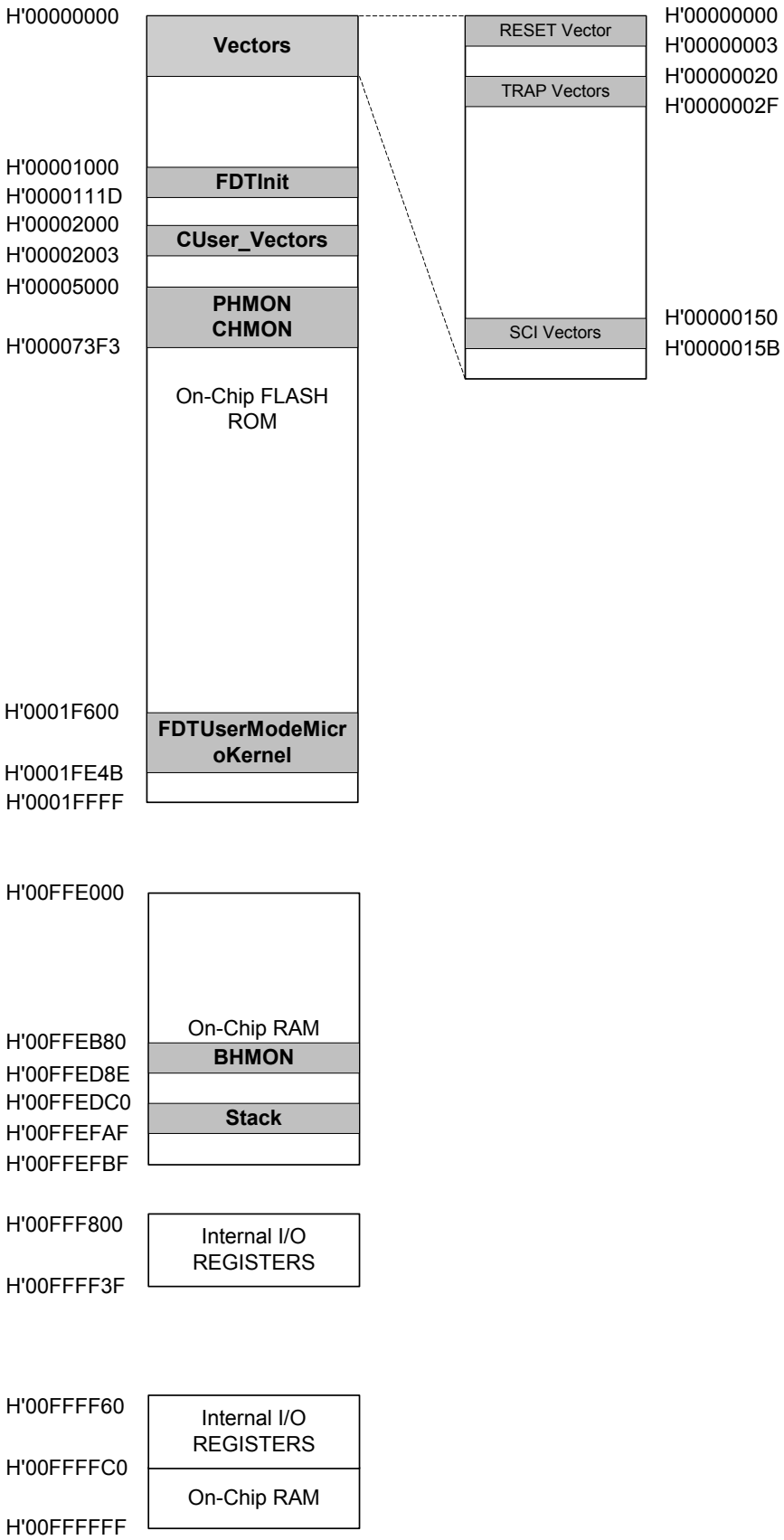
Double clicking in the breakpoint column in the code sets the breakpoint. Breakpoints will remain unless they are double clicked to remove them.

7.1.3. HMON CODE SIZE

HMON is built along with the debug code. Certain elements of the HMON code must remain at a fixed locationin memory. The following table details the HMON components and their size and location in memory. For more information, refer to the map file when building code.

Section	Description	Start Location	Size (H'bytes)
RESET_VECTOR	HMON Reset Vector (Vector 0) Required for Startup of HMON	H' 000000000	4
TRAP_VECTORS	Trap Vectors (Vector 8, 9, 10, 11) Required by HMON to create Trap Breakpoints in RAM	H' 00000020	10
SCI_VECTORS	HMON Serial Port Vectors (Vector 84, 85, 86, 87) Used by HMON when EDK is configured to connect to the default serial port.	H' 00000150	C
PHMON	HMON Code	H' 00005148	22AC
CHMON	HMON Constant Data	H' 00005000	148
BHMON	HMON Uninitialised data	H' 00FFEB80	20F
FDTInit	FDT User Mode Kernel. This is at a fixed location and must not be moved. Should the kernel need to be moved it must be re-compiled.	H' 00001000	11E
FDTUserModeMicroKernel	FDT User Mode Kernel. This is at a fixed location and must not be moved. Should the kernel need to be moved it must be re-compiled.	H' 0001F600	84C
CUser_Vectors	Pointer used by HMON to point to the start of user code. This is at a fixed location and must not be moved for the Reset CPU, and Go Reset commands to function.	H' 00002000	4

7.1.4. MEMORY MAP



7.1.5. BAUD RATE SETTING

HMON has initially set to connect at 57600 Baud. Should the user wish to change this, the value for the BRR in HMONserialconfiguser.c will need to be changed and the project re-built. Please refer to the HMON User Manual for further information.

7.1.6. INTERRUPT MASK SECTIONS

HMON has an interrupt priority of 6. The serial port has an interrupt priority of 7. Modules using interrupts should be set to lower than this value (6 or below), so that serial communications and debugging capability is maintained.

7.2. ADDITIONAL INFORMATION

For details on how to use Hitachi Embedded Workshop (HEW), with HMON, `refer to the HEW manual available on the CD or from the web site.

For information about the H8S/2282 series microcontrollers refer to the H8S/2282 *Series Hardware Manual*

For information about the H8S/2282 assembly language, refer to the H8S/2282 *Series Programming Manual*

Further information available for this product can be found on the HMSE web site at:

http://www.hmse.com/products/support.htm

General information on Hitachi Microcontrollers can be found at the following URLs.

Global: http://www.hitachisemiconductor.com

Europe: http://www.hmse.com