EDK2378

USER MANUAL

FOR H8S/2378
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 Renesas Technology 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: D004883_11

Version: 1

Date: 14/11/2003

Website: http://www.renesas.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.eu.renesas.com/tools

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

2.2. SERIAL CONNECTION

The serial communications cable for connecting the EDK to a host computer requires 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.

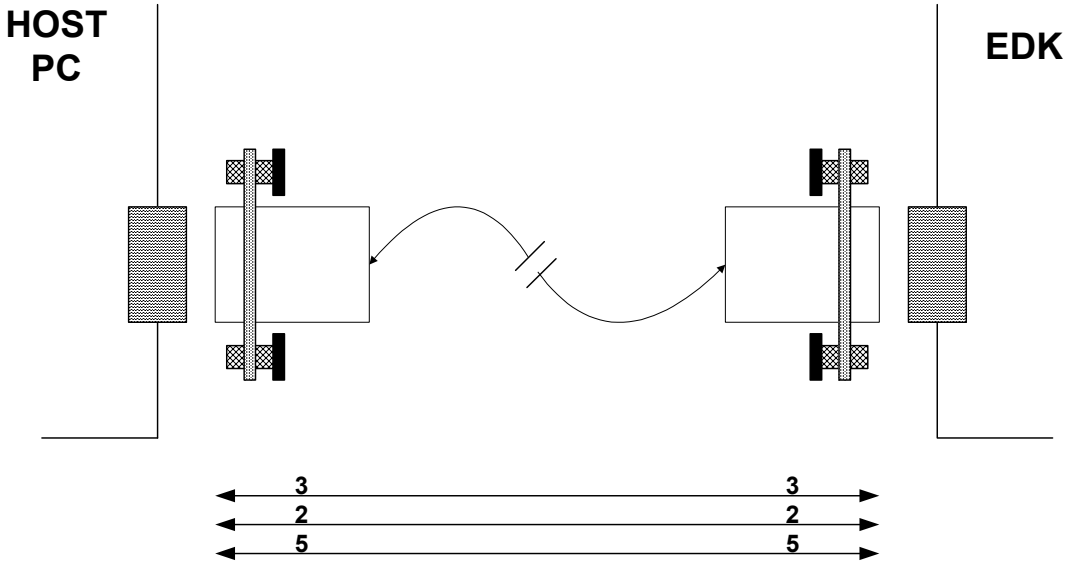


FIGURE 2-1: SERIAL CONNECTION TO PC/NOTEBOOK WITH DB-9 CONNECTOR (SUPPLIED)

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.

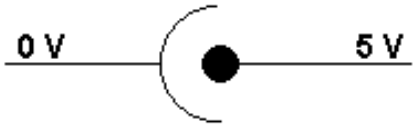


FIGURE 2-2: POWER SUPPLY CONNECTION

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.

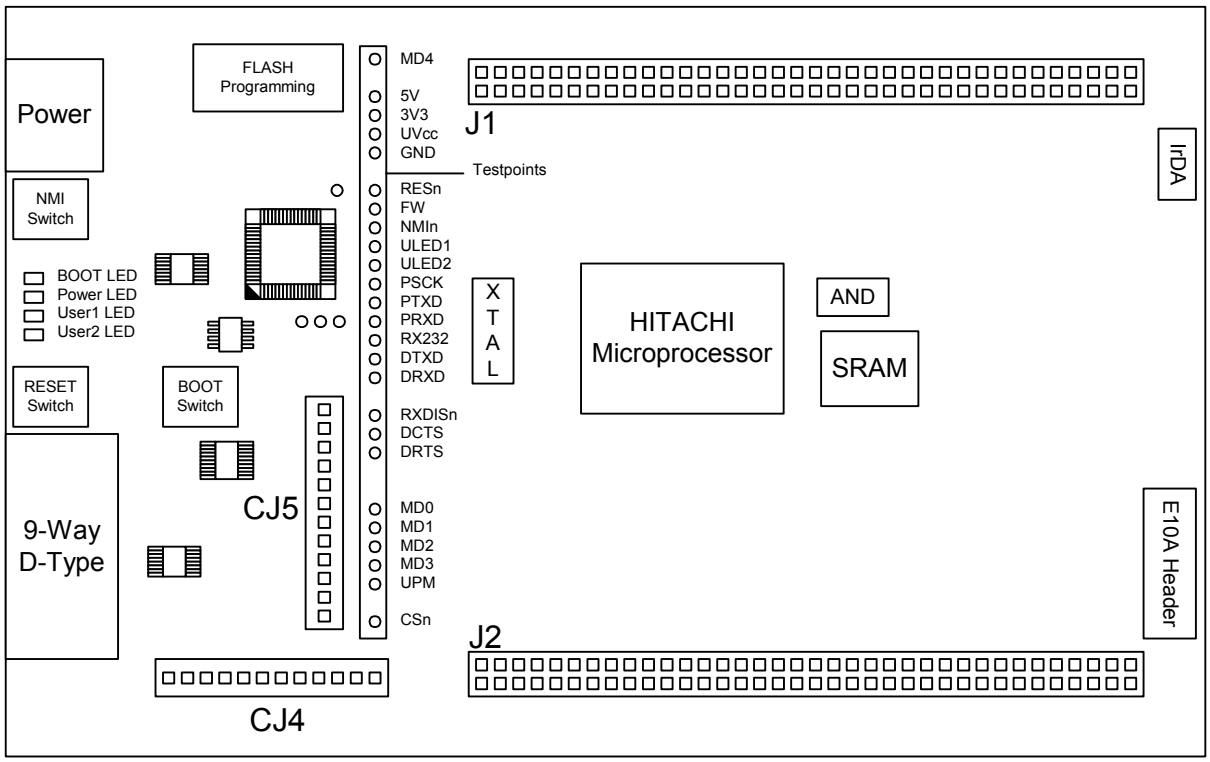


FIGURE 3-1: EDK BOARD LAYOUT

3.1. EDK BLOCK DIAGRAM

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

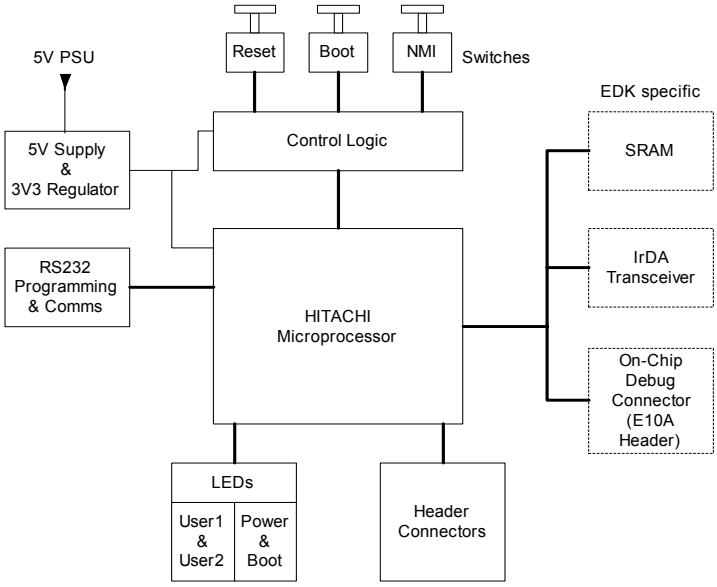


FIGURE 3-2: EDK BLOCK DIAGRAM

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 5.4 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.

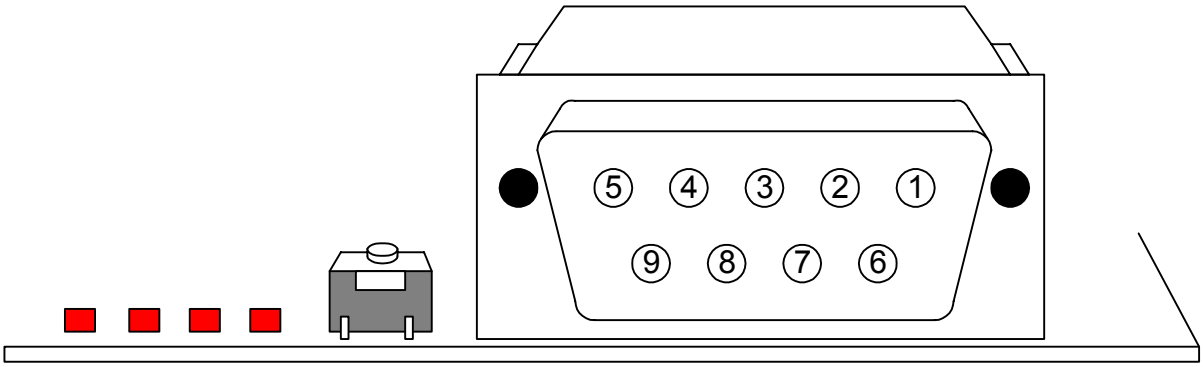


FIGURE 4-1: EDK SERIAL PORT PIN NUMBERING

4.2.2. CRYSTAL CHOICE

The operating crystal frequency has been chosen to support the fastest operation with the fastest serial operating speeds. The value of the crystal is 14.7456MHz. The HMON tutorial code configures the PLL control register (PLLCR) to multiply this internally by 2 to provide a 29.4912MHz internal clock.

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	130	110	-0.07
300	invalid	invalid	invalid	invalid	invalid	invalid	191	300	0.00	47	300	0.00
1200	invalid	invalid	invalid	191	1200	0.00	47	1200	0.00	11	1200	0.00
2400	invalid	invalid	invalid	95	2400	0.00	23	2400	0.00	5	2400	0.00
4800	191	4800	0.00	47	4800	0.00	11	4800	0.00	2	4800	0.00
9600	95	9600	0.00	23	9600	0.00	5	9600	0.00	1	7200	-25.00
19200	47	19200	0.00	11	19200	0.00	2	19200	0.00	invalid	invalid	invalid
38400	23	38400	0.00	5	38400	0.00	1	28800	-25.00	invalid	invalid	invalid
57600	15	57600	0.00	3	57600	0.00	0	57600	0.00	invalid	invalid	invalid
115200	7	115200	0.00	1	115200	0.00	invalid	invalid	invalid	invalid	invalid	invalid
230400*	3	230400	0.00	0	230400	0.00	invalid	invalid	invalid	invalid	invalid	invalid
460800*	1	460800	0.00	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 (X2)*	R6	1MΩ	0805 1%	Welwyn WCR Series
Load capacitors (X2)	C1,C2	22pF	0603 10% 25V	AVX 0603 3 A 220 KAT

TABLE 4-3: REMOVABLE COMPONENT INFORMATION

* Normally Not Fitted

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

The SRAM device fitted to the board is a 4Mbit device allowing 128kx16 or 256kx8 operation.

The onboard H8S2378 microcontroller has chip select management built in. There is no external chip selection hardware associated with this device. The SRAM is connected to Chip Select 0 (CS0), which can address the range H80000 – HFF4000. The usable address range is H'800000 – H'BFFFF.

4.4. MEMORY MAP

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

Section End Section Start	Section Allocation
H'000000 H'07FFFF	ROM
H'080000 H'0BFFFF	External SRAM
H'0A0000 H'1FFFFF	Reserved
H'200000 H'FF3FFF	External Address Space
H'FF4000 H'FFBFFF	On-chip RAM / External Address Space
H'FFC000 H'FFCFFF	Reserved
H'FFD000 H'FFFBFF	External Address Space
H'FFFC00 H'FFFEFF	Internal I/O registers
H'FFFF00 H'FFFF1F	External Address Space
H'FFFF20 H'FFFFFF	Internal I/O registers

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

4.5. SRAM ACCESS TIMING

External access timing is defined by several registers, allowing different types of devices to be addressed. The registers for the selection of wait states and signal extensions are given below with recommended values for the EDK.

Register	Address	Recommended Setting for EDK	Function
ABWCR	FEC0	0x00	Set all 8 memory areas to be 16 bit access
ASTCR	FEC1	0xFF	Selects bus width for memory accesses
WTCRBL	FEC5	0x01	Selects one wait state for memory accesses
RDNCR	FEC6	0x01	Selects RDN0 as Read Strobe signal for memory area 0
Port C DDR	FE2B	0xFF	Configures Port C for output of address line signals
Port B DDR	FE2A	0xFF	Configures Port B for output of address line signals
Port A DDR	FE29	0x07	Configures Port A for output of address line signals
Port G DDR	FE2F	0x01	Configures pin PG0 as CS0 (i.e. chip select for memory area 0)

TABLE 4-5: SRAM ACCESS CONTROL REGISTERS

Please refer to the hardware manual for the microcontroller for more information on these register settings.

4.6. 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 5.7 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	P26	Pin 57	P26/PO6/TIOCA5/IRQ14n
USR2	P25	Pin 56	P25/PO5/TIOCB4/IRQ13n

TABLE 4-6: 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

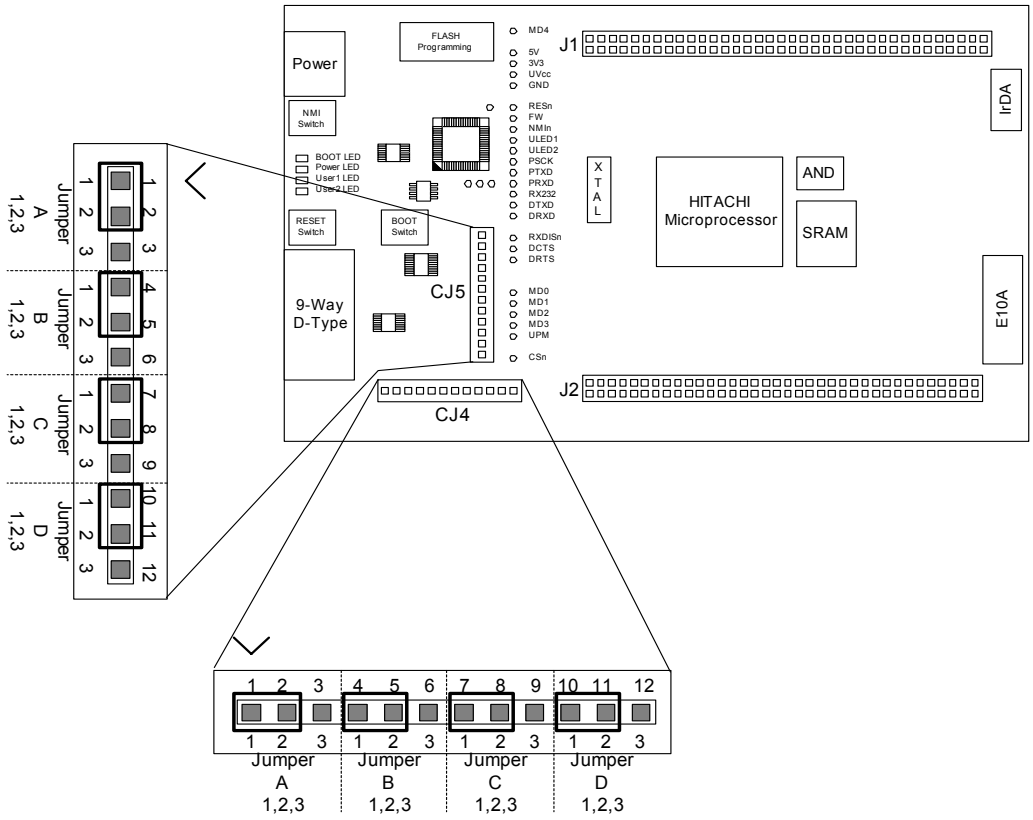


FIGURE 5-1: JUMPER CONFIGURATION

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 2-3	User Mode Setting Bit 0	MD0 pulled High	**MD0 pulled Low**
CJ 5-B Default 2-3	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 4.

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	UPM	Enables E10A Interface	**Disables E10A Interface**
CJ 4-C Default 1-2	CSn	**SRAM device enabled: Device selected by microcontroller.** **(DEFAULT)**	SRAM device Disabled: Device permanently de-selected.
CJ 4-D Default 2-3	-	Not Used	**Not Used**

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

*See section 5.5

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. E10A Enable / Disable Signal.
6	No Connection	No Connection
7	PG0	Microcontroller CSn(0) signal
8	CSn	SRAM CSn signal
9	No Connection	No Connection
10	No Connection	No Connection
11	No Connection	No Connection
12	No Connection	No Connection

5.4. 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	P31
CR23	Fitted	Receive data to EDK	P33
CR19	Not Fitted	Alternate Transmit data from EDK	P23
CR22	Not Fitted	Alternate Receive data to EDK	P24

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	P31
CR23	Not Fitted	Receive data to EDK	P33
CR19	Fitted	Alternate Transmit data from EDK	P23
CR22	Fitted	Alternate Receive data to EDK	P24

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	P22
CR13	Not Fitted	Alternate CTS232 – Clear to send – to EDK	P21

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.

5.5. FLASH PROGRAMMING HEADER

The Flash Programming header is used with the Renesas Flash Debug Module (FDM). The FDM is a USB based programming tool for control and programming of Renesas microcontrollers, available separately from Renesas. This header provides direct access for the FDM 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 section 5.3.
2. Disable the UPM jumper (which enables the E10A) by setting CJ4-B to 2-3. Refer to section 5.3.

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

5.6. EXTERNAL DEBUG HEADER

The External debug header may be used with the Renesas E10A Debugger or a third party debugger.

The E10A is an on-chip debug emulator available separately from Renesas.

This header provides direct access for the debugger to control the EDK microcontroller.

To enable the external debug header, set the CJ4-B jumper to 1-2 to enable emulation. (See section 5.3).

5.7. BOOT CONTROL

The method for placing the microcontroller device into 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 3. 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

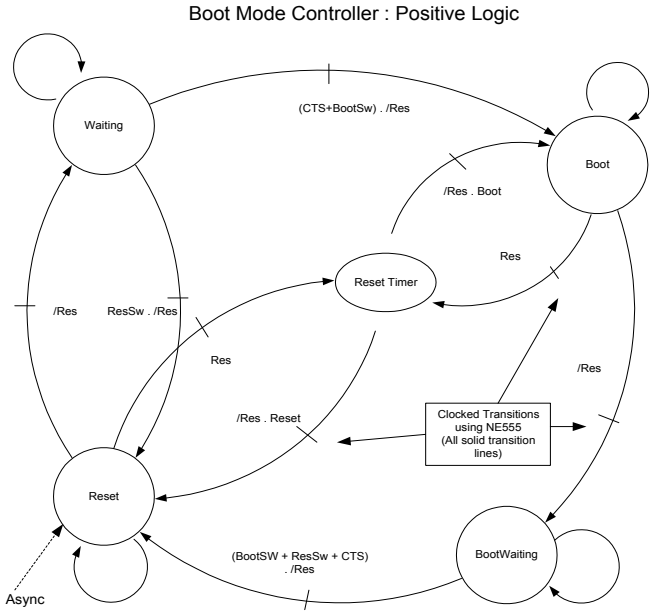


FIGURE 5-2: CPLD STATE DIAGRAM

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	Vss	GND	95	2	PF7/Φ	PF7	94
3	PLLVss	NC_J1_03	93	4	RESn	RESETn	N\A
5	PLLVcc	NC_J1_05	91	6	PF6/ASn	PF6	90
7	PF5/RDn	PF5	89	8	PF4/HWRn	PF4	88
9	PF3/LWRn	PF3	87	10	PF2/IRQ15n/LCASn/DQML	PF2	86
11	PF1/IRQ14n/UCASn/DQMU	PF1	85	12	PF0/WAITn	PF0	84
13	P62/TEND0n/TMCIO/IRQ10n	P62	83	14	P61/DREQ1n/TMRI1/IRQ9n	P61	82
15	P60/DREQ0n/TMRIO/IRQ8n	P60	81	16	PD7/D15	PD7	80
17	PD6/D14	PD6	79	18	PD5/D13	PD5	78
19	PD4/D12	PD4	77	20	PD3/D11	PD3	76
21	PD2/D10	PD2	75	22	PD1/D9	PD1	74
23	PD0/D8	PD0	73	24	Vcc	UVcc	72
25	PE7/D7	PE7	71	26	Vss	GND	70
27	PE6/D6	PE6	69	28	PE5/D5	PE5	68
29	PE4/D4	PE4	67	30	PE3/D3	PE3	66
31	PE2/D2	PE2	65	32	PE1/D1	PE1	64
33	PE0/D0	PE0	63	34	DCTL	DCTL	62
35	P85/IRQ5n/SCK3/EDACK3	P85	61	36	P84/IRQ4n/EDACK2	P84	60
37	P83/IRQ3n/RxD3/ETEND3	P83	59	38	P27/PO7/TIOCB5/IRQ15n	P27	58
39	P26/P06/TIOCA5/IRQ14n	ULED1	57	40	P25/PO5/TIOCB4/IRQ13n	ULED2	56
41	P24/PO4/TIOCA4/RxD4/IRQ12n	DRXD	55	42	P23/PO3/TIOCD3/TxD4/IRQ11n	DTXD	54
43	P22/PO2/TIOCC3/IRQ10n	DRTS	53	44	P21/PO1/TIOCB3/IRQ9n	DCTS	52
45	P20/PO0/TIOCA3/IRQ8n	P20	51	46	Vss	GND	50
47	P17/PO15/TIOCB2/TCLKD/EDRAK3n	P17	49	48	P16/PO14/TIOCA2/EDRAK2n	P16	48
49	P15/PO13/TIOCB1/TCLKC	P15	47	50	P14/PO12/TIOCA1	P14	46
51	P13/PO11/TIOCD0/TCLKB	P13	45	52	P12/PO10/TIOCC0/TCLKA	P12	44
53	P11/PO9/TIOCB0	P11	43	54	P10/PO8/TIOCA0	P10	42
55	Vcl*	UVcc	41	56	NMI	NMIn	40
57	WDTOVFn	WDTOVFn	39	58	PH3/CS7n/IRQ7n/Oen/CKE	PH3	38
59	PH2/CS6n/IRQ6n	PH2	37	60	PH1/CS5n/RAS5n/SDRAMΦ	PH1	36
61	PH0/CS4n/RAS4n/WEn	PH0	35	62	P82/IRQ2n/ETEND2n	P82	34
63	P81/IRQ1n/TxD3/EDREQ3n	P81	33	64	EMLE	FW	32
65	PA7/A23/IRQ7n	PA7	31	66	PA6/A22/IRQ6n	PA6	30
67	PA5/A21/IRQ5n	PA5	29	68	PA4/A20/IRQ4n	PA4	28
69	PA3/A19	PA3	27	70	PA2/A18	PA2	26
71	Vss	GND	25	72	PA1/A17	PA1	24

* Pin 41 is Vcl for H8S/2378 and VCC for H8S/2377

6.2. HEADER J2

J2							
Pin No	**Function**	**EDK Symbol**	**Device pin**	**Pin No**	**Function**	**EDK Symbol**	**Device pin**
1	XTAL	CON_XTAL	96	2	EXTAL	CON_EXTAL	97
3	Vcc	UVcc	98	4	Vcc	UVcc	99
5	NC	NC_J2_05	100	6	NC	NC_J2_06	101
7	Vss	GND	102	8	STBYn	STBYn	103
9	P63/TEND1n/TMCI1/IRQ11n	P63	104	10	P64/DACK0n/TMO0/IRQ12n	P64	105
11	P65/DACK1n/TMO1/IRQ13n	P65	106	12	PG0/CS0n	PG0	107
13	PG1/CS1n	PG1	108	14	PG2/CS2n/RAS2n/RASn	PG2	109
15	PG3/CS3n/RAS3n/CASn	PG3	110	16	Avcc	CON_AVCC	111
17	Vref	CON_VREF	112	18	P40/AN0	P40	113
19	P41/AN1	P41	114	20	P42/AN2	P42	115
21	P43/AN3	P43	116	22	P44/AN4	P44	117
23	P45/AN5	P45	118	24	P46/AN6/DA0	P46	119
25	P47/AN7/DA1	P47	120	26	P90/AN8	P90	121
27	P91/AN9	P91	122	28	P92/AN10	P92	123
29	P93/AN11	P93	124	30	P94/AN12/DA2	P94	125
31	P95/AN13/DA3	P95	126	32	P96/AN14/DA4	P96	127
33	P97/AN15/DA5	P97	128	34	Avss	CON_AVSS	129
35	PG4/BREQ0n	PG4	130	36	PG5/BACKn	PG5	131
37	PG6/BREQn	PG6	132	38	P50/TxD2/IRQ0n	P50	133
39	P51/RxD2/IRQ1n	P51	134	40	P52/SCK2/IRQ2n	P52	135
41	P53/ADTRGn/IRQ3n	P53	136	42	P35/SCK1/SCL0/0En/CKE	PSCK	137
43	P34/SCK0/SCK4/SDA0	P34	138	44	P33/RxD1/SCL1	PRXD	139
45	P32/RxD0/IrRxD/SDA1	P32	140	46	P31/TxD1	PTXD	141
47	P30/TxD0/IrTxD	P30	142	48	MD0	MD0	143
49	MD1	MD1	144	50	MD2	MD2	1
51	Vss	GND	2	52	P80/IRQ0n/EDREQ2n	P80	3
53	Vcc	Uvcc	4	54	PC0/A0	PC0	5
55	PC1/A1	PC1	6	56	PC2/A2	PC2	7
57	PC3/A3	PC3	8	58	PC4/A4	PC4	9
59	Vss	GND	10	60	PC5/A5	PC5	11
61	PC6/A6	PC6	12	62	PC7/A7	PC7	13
63	PB0/A8	PB0	14	64	PB1/A9	PB1	15
65	PB2/A10	PB2	16	66	PB3/A11	PB3	17
67	Vss	GND	18	68	PB4/A12	PB4	19
69	PB5/A13	PB5	20	70	PB6/A14	PB6	21
71	PB7/A15	PB7	22	72	PA0/A16	PA0	23

7. CODE DEVELOPMENT

7.1. HMON

7.1.1. MODE SUPPORT

The HMON library is built to support Advanced Expanded Mode only. The Device supports only Mode 4.

7.1.2. BREAKPOINT SUPPORT

The monitor has no PC Break Controller, therefore no breakpoints can be located in ROM. 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 RAM

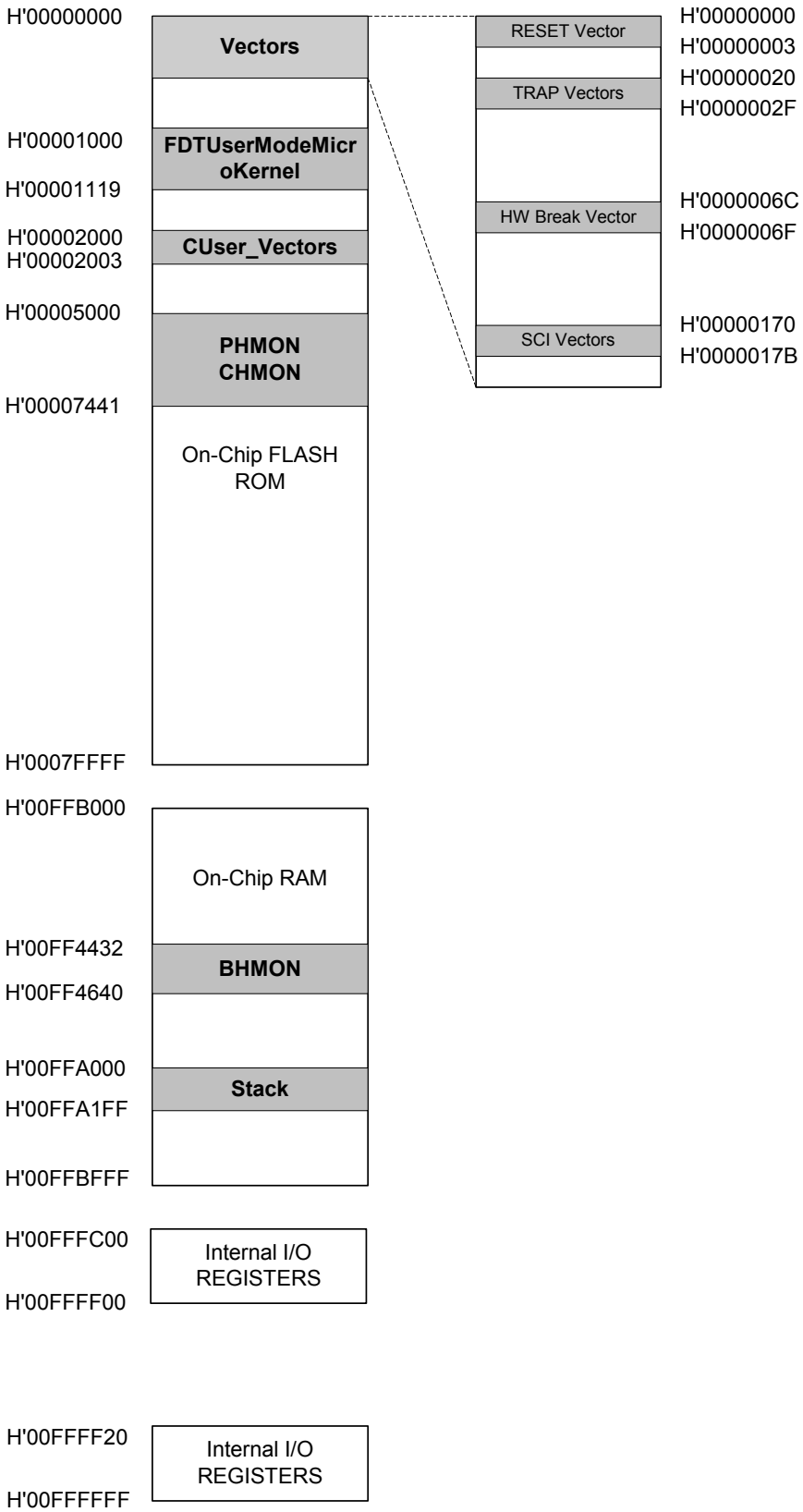
Double clicking in the breakpoint column in the code sets the breakpoint. Breakpoints will remain unless they are double clicked to remove them.

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 location in 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
HW_BREAK_VECTORS	HMON Break Controller (Vector 27) Required by HMON to create Breakpoints in ROM	H' 0000006C	4
SCI_VECTORS	HMON Serial Port Vectors (Vector 80, 81, 82, 83) Used by HMON when EDK is configured to connect to the default serial port.	H' 00000170	C
PHMON	HMON Code	H' 00005000	22FA
CHMON	HMON Constant Data	H' 000072FA	148
BHMON	HMON Uninitialised data	H' 00FF4432	20F
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' 0001000	D03
CUser_Vectors	Pointer used by HMON to point to the start of user code.	H' 00002000	4

7.1.4. MEMORY MAP



7.1.5. BAUD RATE SETTING

HMON has initially set to connect at 115200Baud. 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. HEW3 WORKSPACE FOR KERNEL GENERATION AND MODIFICATION

This CD provides a Hew3 workspace for re-creating the user kernels. This potentially allows the user to rebuild the user mode kernel files uGenu.cde and 2378F.mot if required. This would be the case if, for example, the crystal was replaced with that of a differing value.

The HEW3 kernel workspace is located under the FDT3.1 installation in the 1_0_00.edk folder and is called “2378F”. The workspace contains two build configurations ‘BinaryForHMON” and “FDT_USER_MOT”. The FDT_USER_MOT configuration builds the file “2378F.mot”. This can be included in an FDT project and used to program the device in both user and user boot modes. The BinaryForHMON build configuration generates the binary uGenu.cde file. This is used by HMON to program the device in user mode.

The workspace created by the EDK project generator stores this file on the device chip at an address given by in the section “FDTUserModeMicroKernel”. This must not be moved from the default memory location. The user can view the FDTUserModeMicroKernel section address location in Hew3 by going to ‘Options’ -> Toolchain, selecting ‘Link/Library’ Tab in the window that appears and then in the drop down menu next to Category and selecting ‘section’.

If the user wishes to rebuild the kernels it is strongly recommended that both configurations are built at the same time. This will reflect any source or workspace changes in both 2378F.mot and uGenu.cde files. Multiple builds can be performed automatically in Hew3 by going to Build -> Build Multiple… and in the window that opens ticking both build configurations. The box ‘Build All’ should then be pressed to initiate builds of both uGenu.cde and 2378F.mot files. This should be done each time a new build of the kernels is required. The generated uGenu.cde file will then need to be copied into the users target code workspace overwriting the existing file.

7.3. ADDITIONAL INFORMATION

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

For information about the H8S/2378 series microcontrollers refer to the H8S/2378 *Series Hardware Manual*

For information about the H8S/2378 assembly language, refer to the H8S *Series Programming Manual*

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

http://www.eu.renesas.com/tools

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

Global: http://www.renesas.com/