EDK2628

USER MANUAL

FOR H8S/2628
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: D004673_11

Version: 1

Date: 30/04/2003

Copyright © Renesas Technology Europe Ltd. 2003. All rights reserved.

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 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.

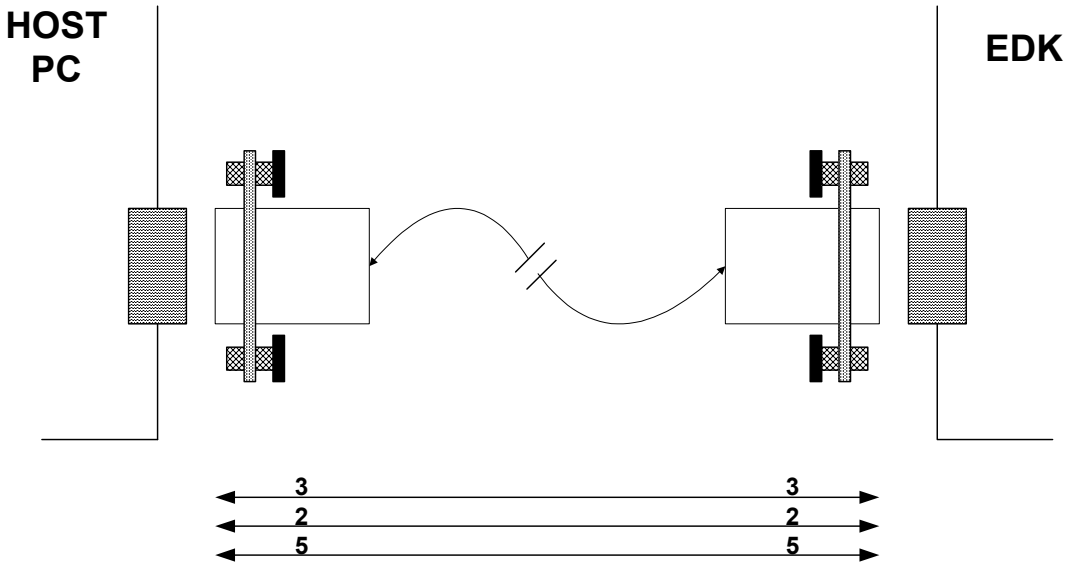


FIGURE 2-1: SERIAL CONNECTION TO PC/NOTEBOOK WITH DB-9 CONNECTOR (SUPPLIED)

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.

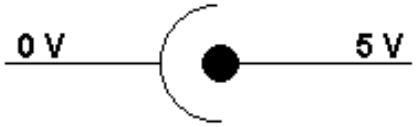


FIGURE 2-2: POWER SUPPLY CONNECTION

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.

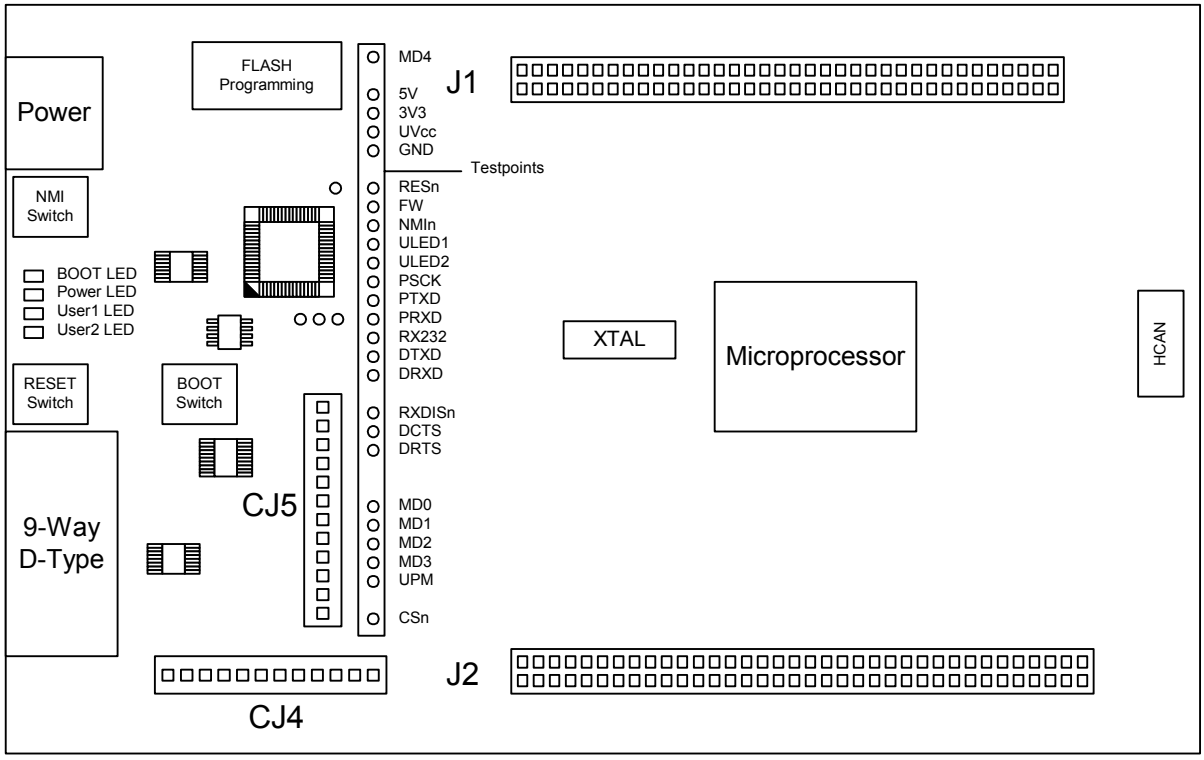


FIGURE 3-1: EDK BOARD LAYOUT

3.1. EDK BLOCK DIAGRAM

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

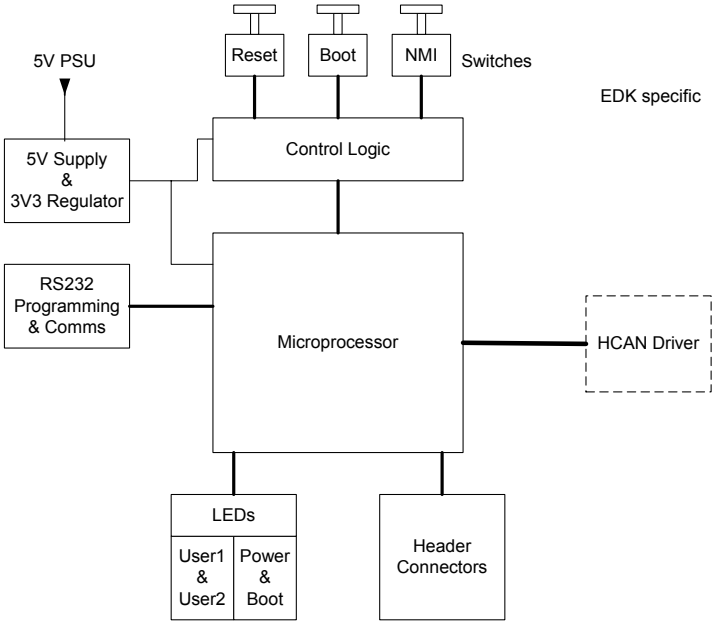


FIGURE 3-2: EDK BLOCK DIAGRAM

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 5.6.

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.

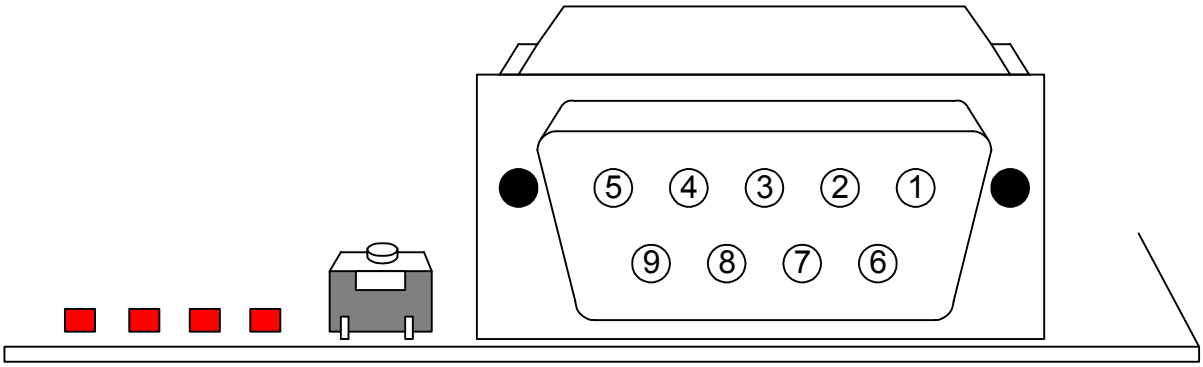


FIGURE 4-1: EDK SERIAL PORT PIN NUMBERING

4.2.2. CRYSTAL CHOICE

The operating crystal frequency has been chosen to support the fastest operation and provide the most reliable CAN communication. The value of the crystal is 20MHz.

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)	R4	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. MEMORY MAP

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

Section End Section Start	Section Allocation
H'00000000 H'0001FFFF	On-Chip ROM
H'00FFD800 H'00FFEFBF	On-Chip RAM
H'00FFF800 H'00FFFF3F	Internal I/O Registers
H'00FFFF60 H'00FFFFBF	Internal I/O Registers
H'00FFFFC0 H'00FFFFFF	On-Chip RAM

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

4.4. 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	PB6	PB6/TIOCA5	39
USR2	PB7	PB7/TIOCB5	40

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

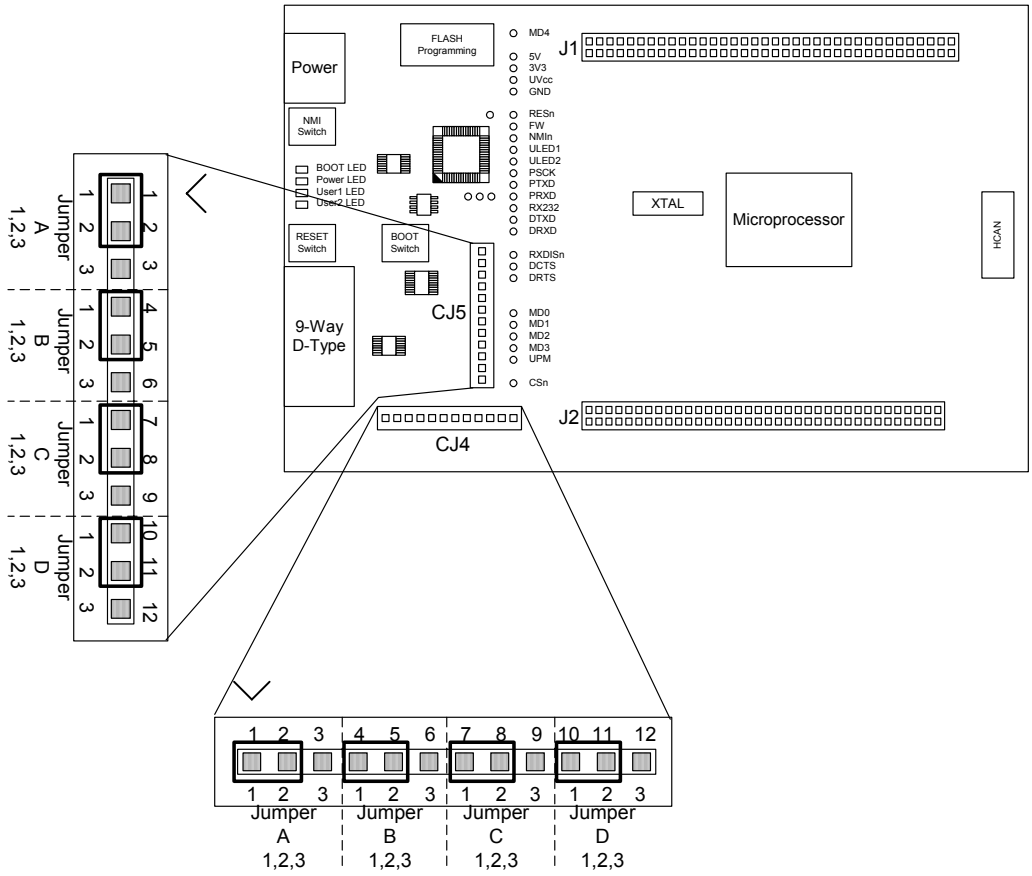


FIGURE 5-1: JUMPER CONFIGURATION

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 cannot be overwritten in User Mode.**
CJ 4-C		Not Used	Not Used
CJ 4-D		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.
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. 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	TxD2/PA1
CR23	Fitted	Receive data to EDK	RxD2/PA2
CR19	Not Fitted	Alternate Transmit data from EDK	TxD0/P30
CR22	Not Fitted	Alternate Receive data to EDK	RxD0/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	TxD2/PA1
CR23	Not Fitted	Receive data to EDK	RxD2/PA2
CR19	Fitted	Alternate Transmit data from EDK	TxD0/P30
CR22	Fitted	Alternate Receive data to EDK	RxD0/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:5.6. 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	P33
CR13	Not Fitted	Alternate CTS232 – Clear to send – to EDK	P35/IRQ5n

TABLE 5-5: OPTION LINKS – SERIAL PORT CONTROL

* See section 5.6

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 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.

5.6. 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 reset 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.

5.6.1. 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.

5.6.2. STATE DIAGRAM

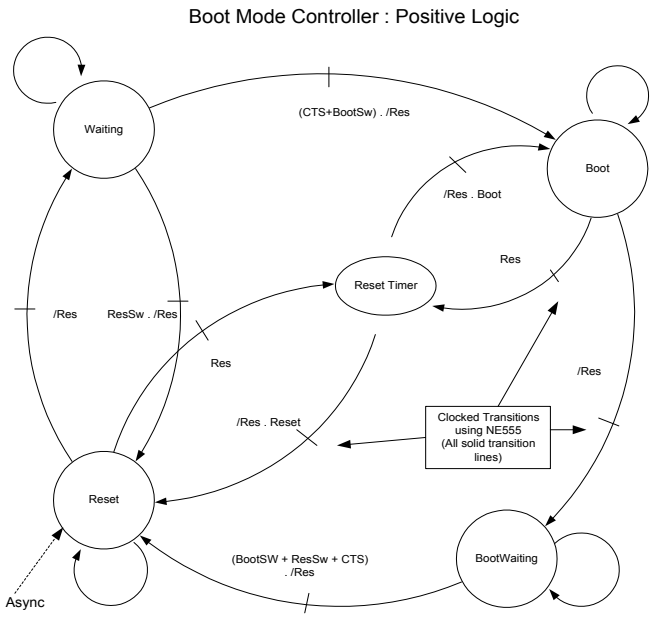


FIGURE 5-2: CPLD STATE DIAGRAM

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	64	2	EXTAL	CON_EXTAL	63
3	XTAL	CON_XTAL	62	4	VCC	UVCC	61
5	NMI	NMIn	60	6	STBYn	STBYn	59
7	VCL	NC_J2_07	58	8	RESn	RESn	57
9	VSS	GND	56	10	MD2	MD2	55
11	MD1	MD1	54	12	MD0	MD0	53
13	P30/TxD0	DTxD	52	14	P31/RxD0	DRxD	51
15	P32/SCK0/IRQ4n	P32	50	16	P33	DRTS	49
17	P34	P34	48	18	P35/IRQ5n	DCTS	47
19	P36	P36	46	20	P37	P37	45
21	PA3/SCK2	PSCK	44	22	PA2/RxD2	PRxD	43
23	PA1/TxD2	PTxD	42	24	PA0	PA0	41
25	PB7/TIOCB5	ULED2	40	26	PB6/TIOCA5	ULED1	39
27	PB5/YIOCB4	PB5	38	28	PB4/TIOCA4	PB4	37
29	PB3/TIOCD3	PB3	36	30	PB2/TIOCC3	PB2	35
31	VSS	GND	34	32	PB1/TIOCB3	PB1	33
33	VCC	UVCC	32	34	PB0/TIOCA3	PB0	31
35	PC7/SCS1n	PC7	30	36	PC6/SSCK1	PC6	29
37	PC5/SSI1	PC5	28	38	PC4/SSO1	PC4	27
39	PC3/SCSOn	PC3	26	40	PC2/SSCK0	PC2	25
41	PC1/SSI0	PC1	24	42	PC0/SSO0	PC0	23
43	PD7	PD7	22	44	PD6	PD6	21
45	PD5	PD5	20	46	PD4	PD4	19
47	PD3	PD3	18	48	PD2	PD2	17
49	PD1	PD1	16	50	PD0	PD0	15

6.2. HEADER J2

J2							
Pin No	**Function**	**EDK Symbol**	**Device pin**	**Pin No**	**Function**	**EDK Symbol**	**Device pin**
1	FWE	FW	66	2	PLLVSS	NC_J2_01	65
3	PF7	PF7	68	4	PLLCAP	NC_J2_03	67
5	PF5	PF5	70	6	PF6	PF6	69
7	PF3/ADTRGn/IRQ3n	PF3	72	8	PF4	PF4	71
9	PF1	PF1	74	10	PF2	PF2	73
11	P97/AN15	P97	76	12	PF0/IRQ2n	PF0	75
13	P95/AN13	P95	78	14	P96/AN15	P96	77
15	P93/AN11	P93	80	16	P94/AN12	P94	79
17	P91/AN9	P91	82	18	P92/AN10	P92	81
19	AVSS	CON_AVSS	84	20	P90/AN8	P90	83
21	AVCC	CON_AVCC	86	22	VREF	CON_VREF	85
23	P46/AN6	P46	88	24	P47/AN7	P47	87
25	P44/AN4	P44	90	26	P45/AN5	P45	89
27	P42/AN2	P42	92	28	P43/AN3	P43	91
29	P40/AN0	P40	94	30	P41/AN1	P41	93
31	P11/P09/TIOCB0	P11	96	32	P10/P08/TIOCA0	P10	95
33	P13/P011/TCLKB	P13	98	34	P12/PO10/TIOCC0/TCLKA	P12	97
35	P15/PO13/TIOCB1/TCLKC	P15	100	36	P14/PO12/TIOCA1/IRQ0n	P14	99
37	VCC	UVCC	2	38	P16/PO14/TIOCA2/IRQ1n	P16	1
39	VSS	GND	4	40	P17/PO15/TIOCB2/TCLKD	P17	3
41	HTxD	HTxD	6	42	HRxD	HRxD	5
43	P71/TMCI23	P71	8	44	P70/TMCIO1/TMRIO1	P70	7
45	P73/TMO1	P73	10	46	P72/TMO0	P72	9
47	P75/TMO3	P75	12	48	P74/TMO2	P74	11
49	P77	P77	14	50	P76	P76	13

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 7.

7.1.2. BREAKPOINT SUPPORT

The monitor utilises the PC Break Controller for code located in ROM, allowing a single breakpoint to be set in the code. 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 FLASH / ROM

Double clicking in the breakpoint column in the code sets the breakpoint. Adding a further breakpoint elsewhere in the code removes the previous one.

7.1.2.2. CODE LOCATED IN RAM

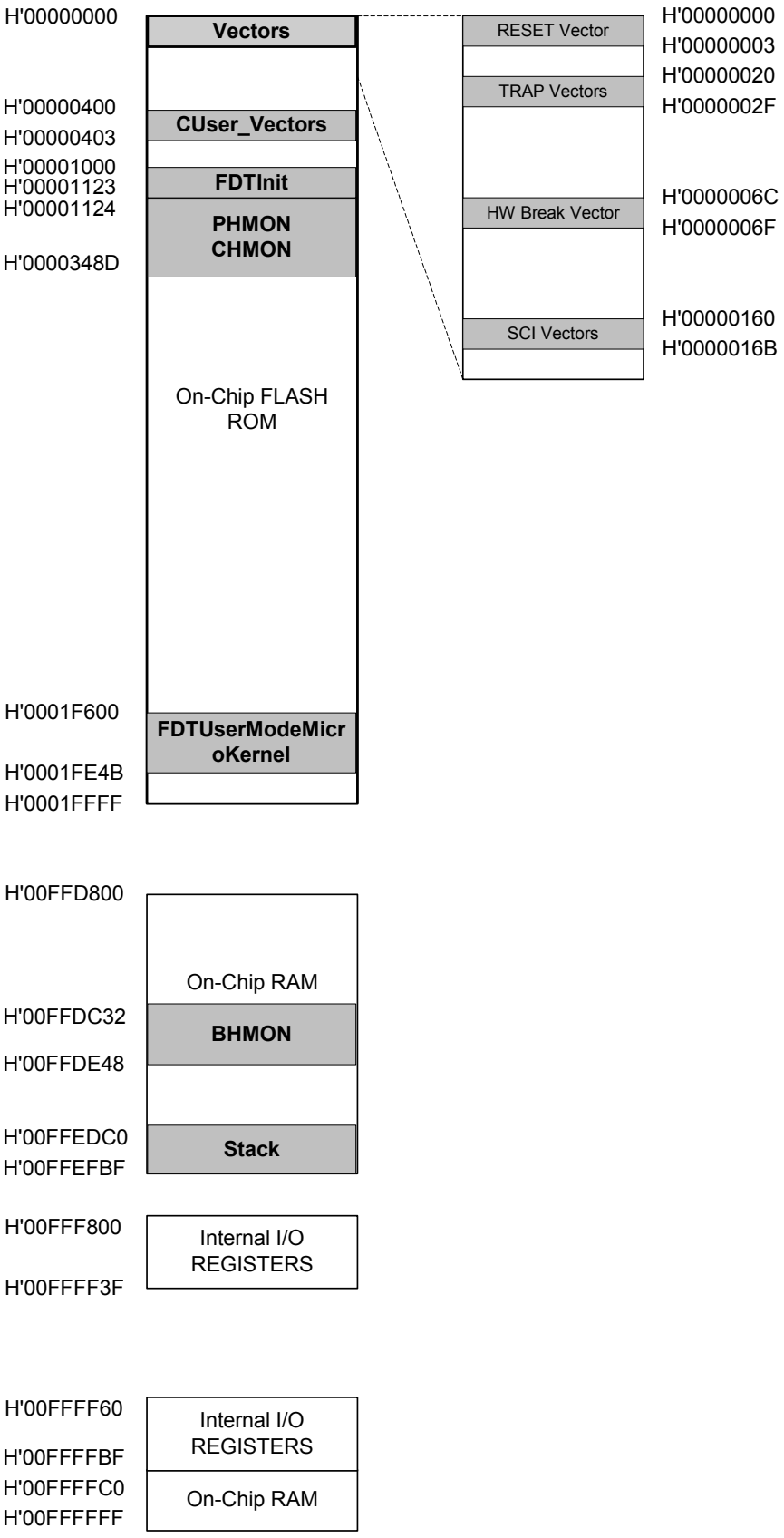
Double clicking in the breakpoint column in the code sets the breakpoint. Breakpoints will remain unless they are double clicked to remove them.

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 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' 00000160	C
PHMON	HMON Code	H' 00001124	222C
CHMON	HMON Constant Data	H' 00003350	13E
BHMON	HMON Uninitialised data	H' 00FFDC32	217
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	124
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.	H' 0000400	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 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/2628 series microcontrollers refer to the H8S/2628 *Series Hardware Manual*

For information about the H8S/2628 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/