Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	HCS12
Core Size	16-Bit
Speed	32MHz
Connectivity	CANbus, EBI/EMI, I²C, IrDA, LINbus, SCI, SPI
Peripherals	LCD, Motor control PWM, POR, PWM, WDT
Number of I/O	80
Program Memory Size	48KB (48K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	100-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=s9s12hy48j0cll

[1] Read: Always reads 0x00
Write: Unimplemented

2.3.12 IRQ Control Register (IRQCR)

Address 0x001E — Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	IRQE	IRQEN	XIRQEN	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 2-10. IRQ Control Register (IRQCR)

[1] Read: See individual bit descriptions below.
Write: See individual bit descriptions below.

Table 2-11. IRQCR Register Field Descriptions

Field	Description
7 IRQE	**IRQ select edge sensitive only**— Special mode: Read or write anytime. Normal mode: Read anytime, write once. 1 $\overline{\text{IRQ}}$ pin configured to respond only to falling edges. Falling edges on the $\overline{\text{IRQ}}$ pin will be detected anytime IRQE=1 and will be cleared only upon a reset or the servicing of the $\overline{\text{IRQ}}$ interrupt. 0 $\overline{\text{IRQ}}$ pin configured for low level recognition
6 IRQEN	**IRQ enable**— Read or write anytime. 1 $\overline{\text{IRQ}}$ pin is connected to interrupt logic 0 $\overline{\text{IRQ}}$ pin is disconnected from interrupt logic
5 XIRQEN	**XIRQ enable**— Special mode: Read or write anytime. Normal mode: Read anytime, write once. 1 $\overline{\text{XIRQ}}$ pin is connected to interrupt logic 0 $\overline{\text{XIRQ}}$ pin is disconnected from interrupt logic

2.3.13 PIM Reserved Register

This register is reserved for factory testing of the PIM module and is not available in normal operation.

2.4.3.5 Port P

This port is associated with the PWM.

2.4.3.6 Port R

This port is associated with LCD/IIC.

2.4.3.7 Port S

This port is associated with SPI/SCI/IIC/PWM/CAN.

2.4.3.8 Port T

This port is associated with LCD and TIM.

2.4.3.9 Port U

This port is associated with the Motor Driver/TIM0.

2.4.3.10 Port V

This port is associated with the Motor Driver/TIM1/SPI/IIC/PWM.

2.4.4 Pin interrupts

Ports T, S, R, AD offer pin interrupt capability. The interrupt enable as well as the sensitivity to rising or falling edges can be individually configured on per-pin basis. All bits/pins in a port share the same interrupt vector. Interrupts can be used with the pins configured as inputs or outputs.

An interrupt is generated when a bit in the port interrupt flag register and its corresponding port interrupt enable bit are both set. The pin interrupt feature is also capable to wake up the CPU when it is in STOP or WAIT mode.

A digital filter on each pin prevents pulses (Figure 2-89) shorter than a specified time from generating an interrupt. The minimum time varies over process conditions, temperature and voltage (Figure 2-88 and Table 2-75).

5.1.3 Block Diagram

A block diagram of the BDM is shown in Figure 5-1.

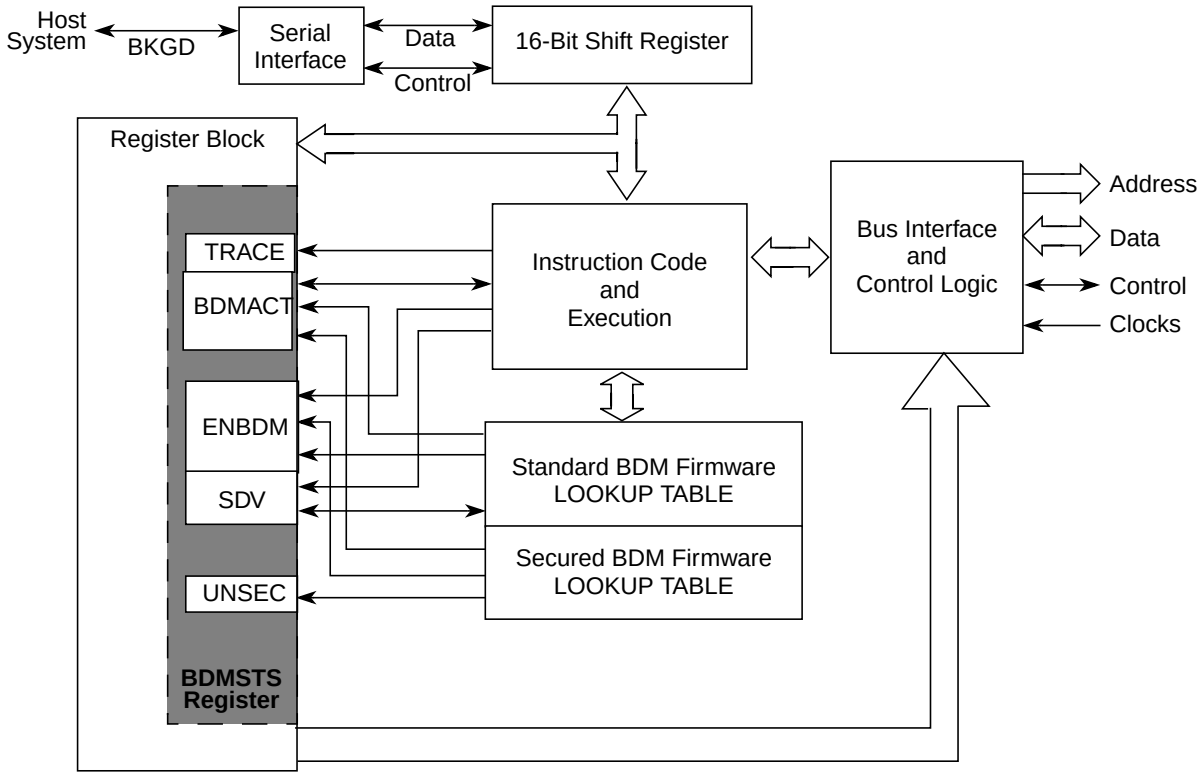

Figure 5-1. BDM Block Diagram

5.2 External Signal Description

A single-wire interface pin called the background debug interface (BKGD) pin is used to communicate with the BDM system. During reset, this pin is a mode select input which selects between normal and special modes of operation. After reset, this pin becomes the dedicated serial interface pin for the background debug mode.

5.3 Memory Map and Register Definition

5.3.1 Module Memory Map

Table 5-2 shows the BDM memory map when BDM is active.

Table 5-2. BDM Memory Map

Global Address	Module	Size (Bytes)
0x3_FF00–0x3_FF0B	BDM registers	12

Since the host knows the target serial clock frequency, the SYNC command (used to abort a command) does not need to consider the lower possible target frequency. In this case, the host could issue a SYNC very close to the 128 serial clock cycles length. Providing a small overhead on the pulse length in order to assure the SYNC pulse will not be misinterpreted by the target. See Section 5.4.9, “SYNC — Request Timed Reference Pulse”.

Figure 5-12 shows a SYNC command being issued after a READ_BYTE, which aborts the READ_BYTE command. Note that, after the command is aborted a new command could be issued by the host computer.

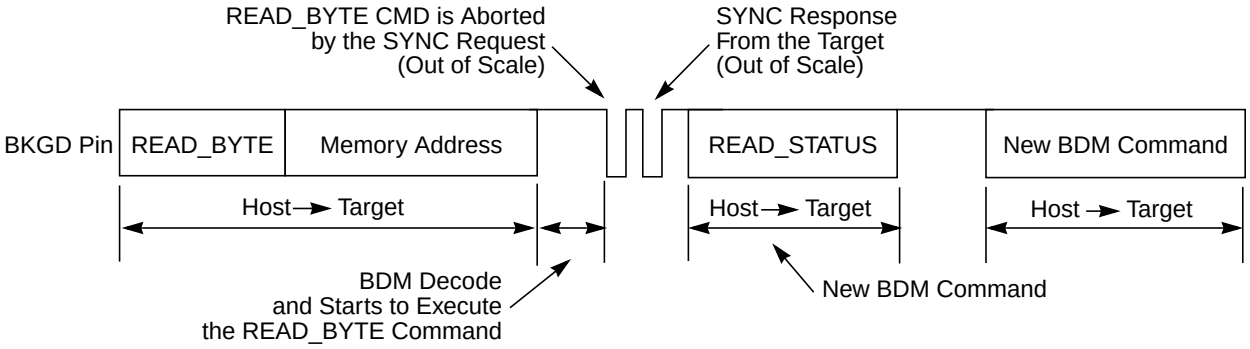

Figure 5-12. ACK Abort Procedure at the Command Level

NOTE

Figure 5-12 does not represent the signals in a true timing scale

Figure 5-13 shows a conflict between the ACK pulse and the SYNC request pulse. This conflict could occur if a POD device is connected to the target BKGD pin and the target is already in debug active mode. Consider that the target CPU is executing a pending BDM command at the exact moment the POD is being connected to the BKGD pin. In this case, an ACK pulse is issued along with the SYNC command. In this case, there is an electrical conflict between the ACK speedup pulse and the SYNC pulse. Since this is not a probable situation, the protocol does not prevent this conflict from happening.

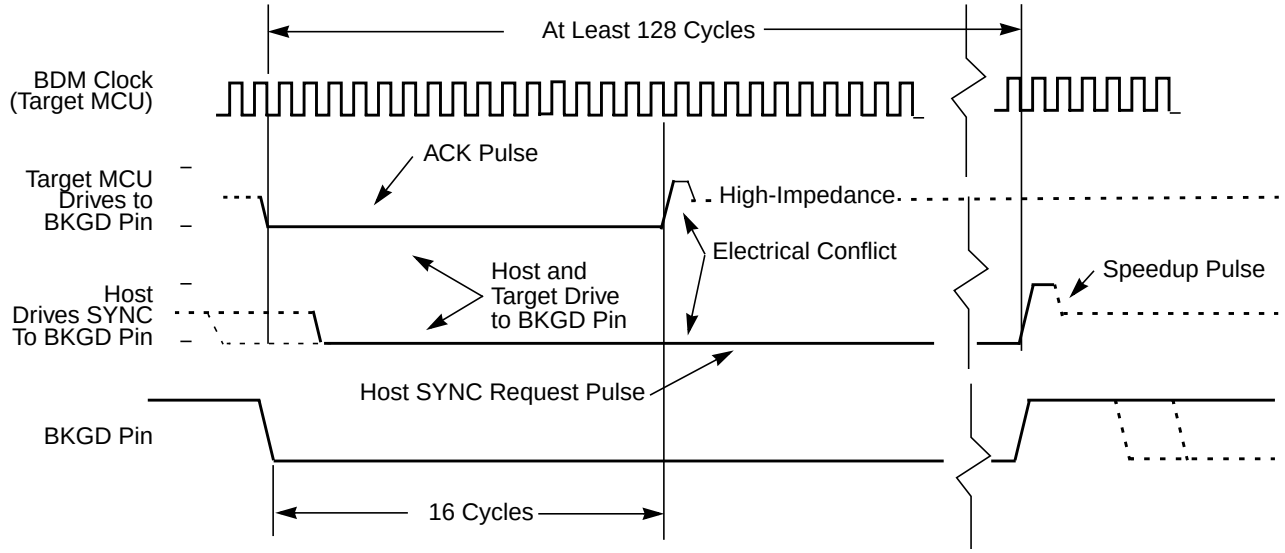

Figure 5-13. ACK Pulse and SYNC Request Conflict

7.3.2.15 Autonomous Periodical Interrupt Control Register (CPMUAPICTL)

The CPMUAPICTL register allows the configuration of the autonomous periodical interrupt features.

0x02F2

	7	6	5	4	3	2	1	0
R	APICLK	0	0	APIES	APIEA	APIFE	APIE	APIF
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 7-18. Autonomous Periodical Interrupt Control Register (CPMUAPICTL)

Read: Anytime

Write: Anytime

Table 7-15. CPMUAPICTL Field Descriptions

Field	Description
7 APICLK	**Autonomous Periodical Interrupt Clock Select Bit** — Selects the clock source for the API. Writable only if APIFE = 0. APICLK cannot be changed if APIFE is set by the same write operation. 0 Autonomous periodical interrupt clock used as source. 1 Bus Clock used as source.
4 APIES	**Autonomous Periodical Interrupt External Select Bit** — Selects the waveform at the external pin API_EXTCLK as shown in Figure 7-19. See device level specification for connectivity of API_EXTCLK pin. 0 If APIEA and APIFE are set, at the external pin API_EXTCLK periodic high pulses are visible at the end of every selected period with the size of half of the minimum period (APIR=0x0000 in Table 7-19). 1 If APIEA and APIFE are set, at the external pin API_EXTCLK a clock is visible with 2 times the selected API Period.
3 APIEA	**Autonomous Periodical Interrupt External Access Enable Bit** — If set, the waveform selected by bit APIES can be accessed externally. See device level specification for connectivity. 0 Waveform selected by APIES can not be accessed externally. 1 Waveform selected by APIES can be accessed externally, if APIFE is set.
2 APIFE	**Autonomous Periodical Interrupt Feature Enable Bit** — Enables the API feature and starts the API timer when set. 0 Autonomous periodical interrupt is disabled. 1 Autonomous periodical interrupt is enabled and timer starts running.
1 APIE	**Autonomous Periodical Interrupt Enable Bit** 0 API interrupt request is disabled. 1 API interrupt will be requested whenever APIF is set.
0 APIF	**Autonomous Periodical Interrupt Flag** — APIF is set to 1 when the in the API configured time has elapsed. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (APIE = 1), APIF causes an interrupt request. 0 API time-out has not yet occurred. 1 API time-out has occurred.

7.4.6.3 PLL Bypassed External Mode (PBE)

In this mode, the Bus Clock is based on the external oscillator clock. The reference clock for the PLL is based on the external oscillator. The adaptive spike filter and detection logic can be enabled which uses the VCOCLK to filter and qualify the external oscillator clock.

The clock sources for COP and RTI can be based on the internal reference clock generator or on the external oscillator clock.

This mode can be entered from default mode PEI by performing the following steps:

1. Make sure the PLL configuration is valid
2. Optionally the adaptive spike filter and detection logic can be enabled by calculating the integer value for the OSCFIL[4:0] bits and setting the bandwidth (OSCBW) accordingly.
3. Enable the external oscillator (OSCE bit)
4. Wait for the PLL being locked (LOCK = 1) and the oscillator to start-up and additionally being qualified if the adaptive spike filter is enabled (UPOSC =1).
5. Clear all flags in the CPMUFLG register to be able to detect any status bit change.
6. Optionally status interrupts can be enabled (CPMUINT register).
7. Select the Oscillator Clock (OSCCLK) as Bus Clock (PLLSEL=0)

Since the Adaptive Oscillator Filter (adaptive spike filter and detection logic) uses VCOCLK (from PLL) to continuously filter and qualify the external oscillator clock, loosing PLL lock status (LOCK=0) means loosing the oscillator status information as well (UPOSC=0).

The impact of loosing the oscillator status in PBE mode is as follows:

- PLLSEL is set automatically and the Bus Clock is switched back to the PLLCLK.
- The PLLCLK is derived from the VCO clock (with its actual frequency) divided by four until the PLL locks again.

Application software needs to be prepared to deal with the impact of loosing the oscillator status at any time.

In the PBE mode, not every noise disturbance can be indicated by bits LOCK and UPOSC (both bits are based on the Bus Clock domain). There are clock disturbances possible, after which UPOSC and LOCK both stay asserted while occasional pauses on the filtered OSCCLK and resulting Bus Clock occur. The adaptive spike filter is still functional and protects the Bus Clock from frequency overshoot due to spikes on the external oscillator clock. The filtered OSCCLK and resulting Bus Clock will pause until the PLL has stabilized again.

Table 9-4. CANCTL1 Register Field Descriptions (continued)

Field	Description
1 SLPAK	**Sleep Mode Acknowledge** — This flag indicates whether the MSCAN module has entered sleep mode (see Section 9.4.5.5, "MSCAN Sleep Mode"). It is used as a handshake flag for the SLPRQ sleep mode request. Sleep mode is active when SLPRQ = 1 and SLPAK = 1. Depending on the setting of WUPE, the MSCAN will clear the flag if it detects activity on the CAN bus while in sleep mode. 0 Running — The MSCAN operates normally 1 Sleep mode active — The MSCAN has entered sleep mode
0 INITAK	**Initialization Mode Acknowledge** — This flag indicates whether the MSCAN module is in initialization mode (see Section 9.4.4.5, "MSCAN Initialization Mode"). It is used as a handshake flag for the INITRQ initialization mode request. Initialization mode is active when INITRQ = 1 and INITAK = 1. The registers CANCTL1, CANBTR0, CANBTR1, CANIDAC, CANIDAR0–CANIDAR7, and CANIDMR0–CANIDMR7 can be written only by the CPU when the MSCAN is in initialization mode. 0 Running — The MSCAN operates normally 1 Initialization mode active — The MSCAN has entered initialization mode

9.3.2.3 MSCAN Bus Timing Register 0 (CANBTR0)

The CANBTR0 register configures various CAN bus timing parameters of the MSCAN module.

Module Base + 0x0002 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R W	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
Reset:	0	0	0	0	0	0	0	0

Figure 9-6. MSCAN Bus Timing Register 0 (CANBTR0)

1. Read: Anytime
   Write: Anytime in initialization mode (INITRQ = 1 and INITAK = 1)

Table 9-5. CANBTR0 Register Field Descriptions

Field	Description
7-6 SJW[1:0]	**Synchronization Jump Width** — The synchronization jump width defines the maximum number of time quanta (Tq) clock cycles a bit can be shortened or lengthened to achieve resynchronization to data transitions on the CAN bus (see Table 9-6).
5-0 BRP[5:0]	**Baud Rate Prescaler** — These bits determine the time quanta (Tq) clock which is used to build up the bit timing (see Table 9-7).

Table 9-6. Synchronization Jump Width

SJW1	SJW0	Synchronization Jump Width
0	0	1 Tq clock cycle
0	1	2 Tq clock cycles
1	0	3 Tq clock cycles
1	1	4 Tq clock cycles

Table 12-3. IRSCI Transmit Pulse Width

TNP[1:0]	Narrow Pulse Width
11	1/4
10	1/32
01	1/16
00	3/16

12.3.2.2 SCI Control Register 1 (SCICR1)

Module Base + 0x0002

	7	6	5	4	3	2	1	0
R W	LOOPS	SCISWAI	RSRC	M	WAKE	ILT	PE	PT
Reset	0	0	0	0	0	0	0	0

Figure 12-5. SCI Control Register 1 (SCICR1)

Read: Anytime, if AMAP = 0.

Write: Anytime, if AMAP = 0.

NOTE

This register is only visible in the memory map if AMAP = 0 (reset condition).

Table 12-4. SCICR1 Field Descriptions

Field	Description
7 LOOPS	**Loop Select Bit** — LOOPS enables loop operation. In loop operation, the RXD pin is disconnected from the SCI and the transmitter output is internally connected to the receiver input. Both the transmitter and the receiver must be enabled to use the loop function. 0 Normal operation enabled 1 Loop operation enabled The receiver input is determined by the RSRC bit.
6 SCISWAI	**SCI Stop in Wait Mode Bit** — SCISWAI disables the SCI in wait mode. 0 SCI enabled in wait mode 1 SCI disabled in wait mode
5 RSRC	**Receiver Source Bit** — When LOOPS = 1, the RSRC bit determines the source for the receiver shift register input. See Table 12-5. 0 Receiver input internally connected to transmitter output 1 Receiver input connected externally to transmitter
4 M	**Data Format Mode Bit** — MODE determines whether data characters are eight or nine bits long. 0 One start bit, eight data bits, one stop bit 1 One start bit, nine data bits, one stop bit
3 WAKE	**Wakeup Condition Bit** — WAKE determines which condition wakes up the SCI: a logic 1 (address mark) in the most significant bit position of a received data character or an idle condition on the RXD pin. 0 Idle line wakeup 1 Address mark wakeup

12.4.5.5 LIN Transmit Collision Detection

This module allows to check for collisions on the LIN bus.

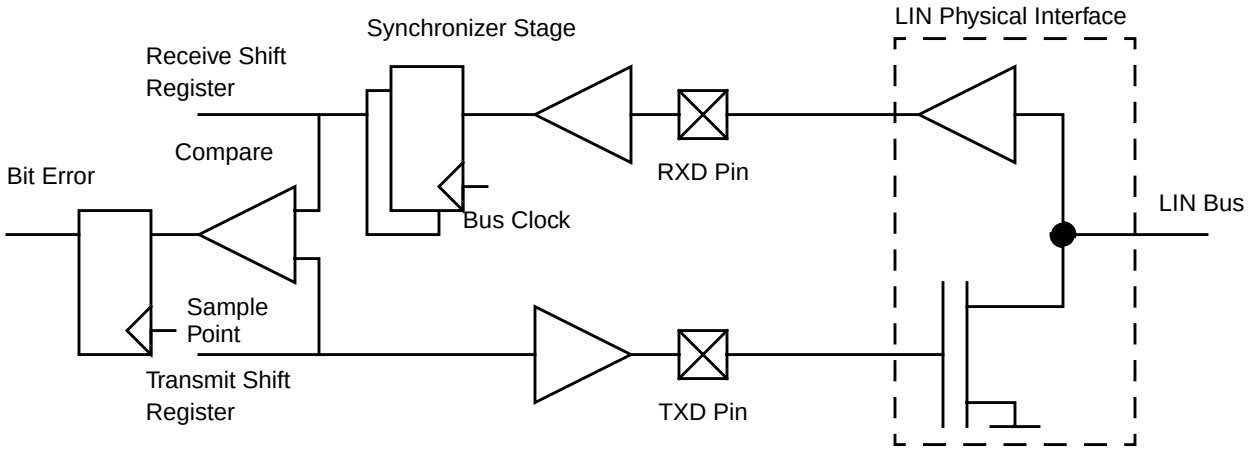

Figure 12-18. Collision Detect Principle

If the bit error circuit is enabled (BERRM[1:0] = 0:1 or = 1:0]), the error detect circuit will compare the transmitted and the received data stream at a point in time and flag any mismatch. The timing checks run when transmitter is active (not idle). As soon as a mismatch between the transmitted data and the received data is detected the following happens:

- The next bit transmitted will have a high level (TXPOL = 0) or low level (TXPOL = 1)
- The transmission is aborted and the byte in transmit buffer is discarded.
- the transmit data register empty and the transmission complete flag will be set
- The bit error interrupt flag, BERRIF, will be set.
- No further transmissions will take place until the BERRIF is cleared.

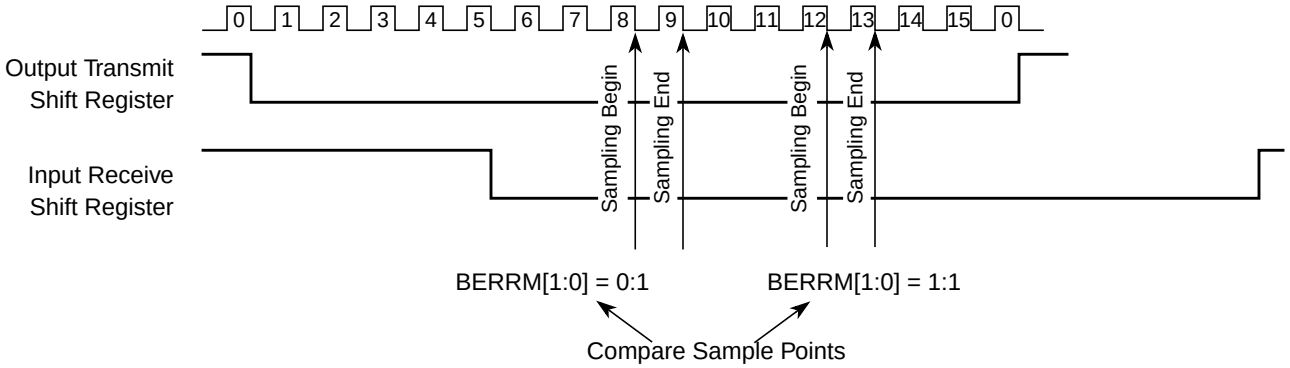

Figure 12-19. Timing Diagram Bit Error Detection

If the bit error detect feature is disabled, the bit error interrupt flag is cleared.

NOTE

The RXPOL and TXPOL bit should be set the same when transmission collision detect feature is enabled, otherwise the bit error interrupt flag may be set incorrectly.

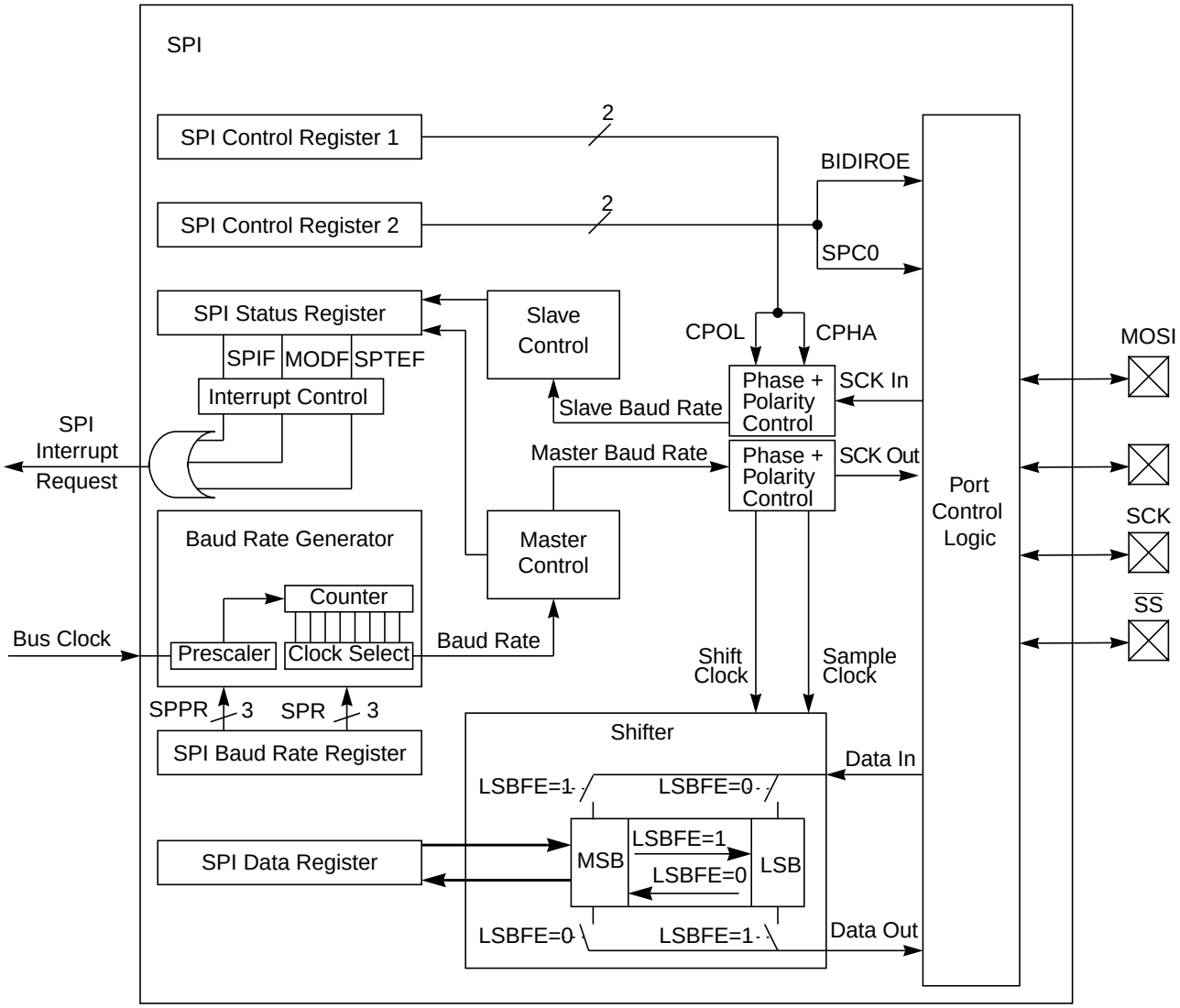

Figure 13-1. SPI Block Diagram

13.2 External Signal Description

This section lists the name and description of all ports including inputs and outputs that do, or may, connect off chip. The SPI module has a total of four external pins.

13.2.1 MOSI — Master Out/Slave In Pin

This pin is used to transmit data out of the SPI module when it is configured as a master and receive data when it is configured as slave.

13.2.2 MISO — Master In/Slave Out Pin

This pin is used to transmit data out of the SPI module when it is configured as a slave and receive data when it is configured as master.

13.3.2.3 SPI Baud Rate Register (SPIBR)

Module Base +0x0002

	7	6	5	4	3	2	1	0
R	0	SPPR2	SPPR1	SPPR0	0	SPR2	SPR1	SPR0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 13-5. SPI Baud Rate Register (SPIBR)

Read: Anytime

Write: Anytime; writes to the reserved bits have no effect

Table 13-6. SPIBR Field Descriptions

Field	Description
6–4 SPPR[2:0]	**SPI Baud Rate Preselection Bits** — These bits specify the SPI baud rates as shown in Table 13-7. In master mode, a change of these bits will abort a transmission in progress and force the SPI system into idle state.
2–0 SPR[2:0]	**SPI Baud Rate Selection Bits** — These bits specify the SPI baud rates as shown in Table 13-7. In master mode, a change of these bits will abort a transmission in progress and force the SPI system into idle state.

The baud rate divisor equation is as follows:

$$\text{BaudRateDivisor} = (\text{SPPR} + 1) \bullet 2^{(\text{SPR} + 1)} \qquad \textit{Eqn. 13-1}$$

The baud rate can be calculated with the following equation:

$$\text{Baud Rate} = \text{BusClock} / \text{BaudRateDivisor} \qquad \textit{Eqn. 13-2}$$

NOTE

For maximum allowed baud rates, please refer to the SPI Electrical Specification in the Electricals chapter of this data sheet.

Table 13-7. Example SPI Baud Rate Selection (25 MHz Bus Clock) (Sheet 1 of 3)

SPPR2	SPPR1	SPPR0	SPR2	SPR1	SPR0	Baud Rate Divisor	Baud Rate
0	0	0	0	0	0	2	12.5 Mbit/s
0	0	0	0	0	1	4	6.25 Mbit/s
0	0	0	0	1	0	8	3.125 Mbit/s
0	0	0	0	1	1	16	1.5625 Mbit/s
0	0	0	1	0	0	32	781.25 kbit/s
0	0	0	1	0	1	64	390.63 kbit/s
0	0	0	1	1	0	128	195.31 kbit/s
0	0	0	1	1	1	256	97.66 kbit/s
0	0	1	0	0	0	4	6.25 Mbit/s
0	0	1	0	0	1	8	3.125 Mbit/s
0	0	1	0	1	0	16	1.5625 Mbit/s
0	0	1	0	1	1	32	781.25 kbit/s

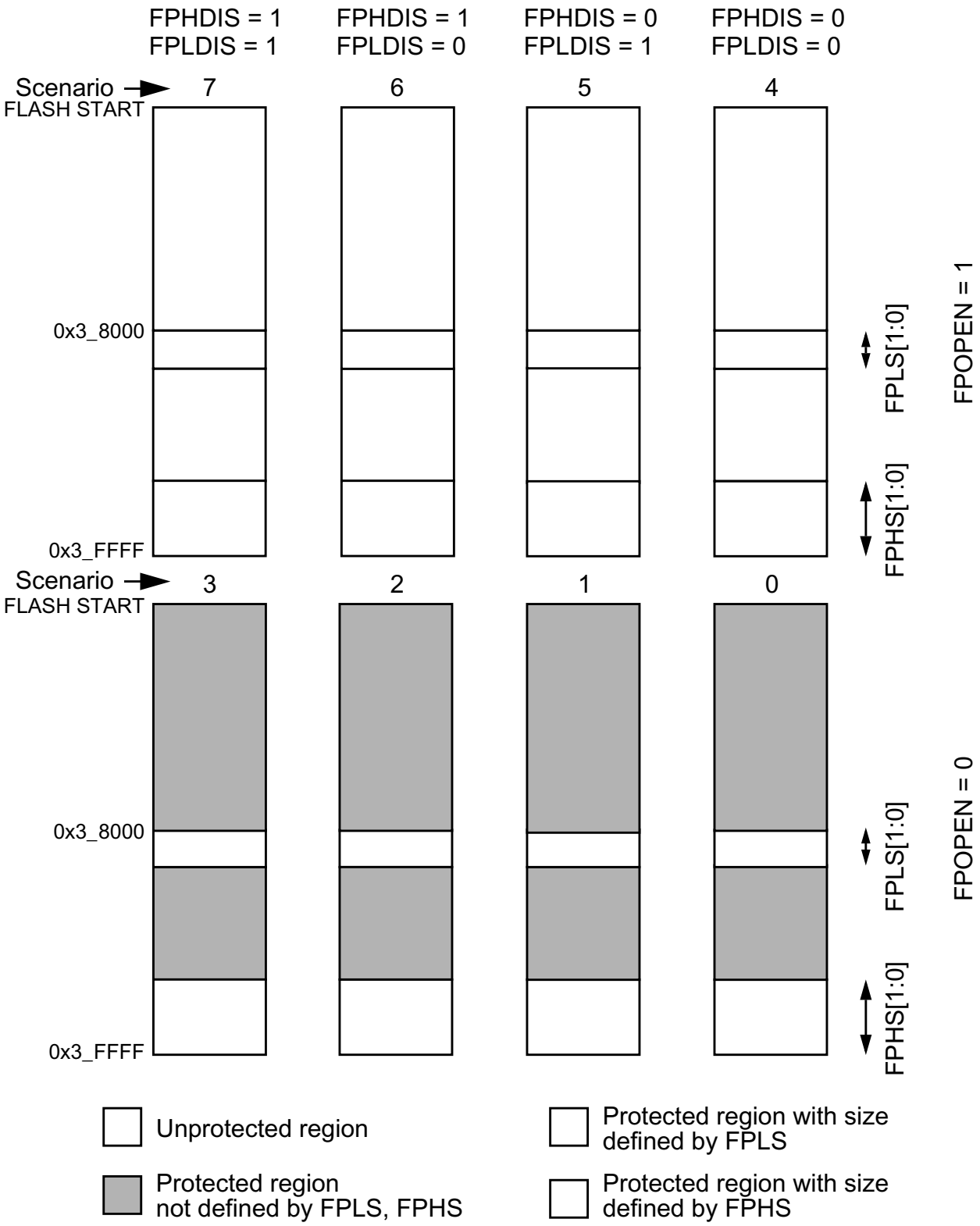

Figure 15-14. P-Flash Protection Scenarios

1 As defined by the memory map for FTMRC64K1.
2 As found in the memory map for FTMRC64K1.

16.4.5.4 Read Once Command

The Read Once command provides read access to a reserved 64 byte field (8 phrases) located in the nonvolatile information register of P-Flash. The Read Once field is programmed using the Program Once command described in Section 16.4.5.6. The Read Once command must not be executed from the Flash block containing the Program Once reserved field to avoid code runaway.

Table 16-37. Read Once Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x04	Not Required
001	Read Once phrase index (0x0000 - 0x0007)	
010	Read Once word 0 value	
011	Read Once word 1 value	
100	Read Once word 2 value	
101	Read Once word 3 value	

Upon clearing CCIF to launch the Read Once command, a Read Once phrase is fetched and stored in the FCCOB indexed register. The CCIF flag will set after the Read Once operation has completed. Valid phrase index values for the Read Once command range from 0x0000 to 0x0007. During execution of the Read Once command, any attempt to read addresses within P-Flash block will return invalid data.

Table 16-38. Read Once Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 001 at command launch
		Set if command not available in current mode (see Table 16-27)
		Set if an invalid phrase index is supplied
	FPVIOL	None
	MGSTAT1	Set if any errors have been encountered during the read
	MGSTAT0	Set if any non-correctable errors have been encountered during the read

16.4.5.5 Program P-Flash Command

The Program P-Flash operation will program a previously erased phrase in the P-Flash memory using an embedded algorithm.

A P-Flash phrase must be in the erased state before being programmed. Cumulative programming of bits within a Flash phrase is not allowed.

Table 17-29. D-Flash Commands

FCMD	Command	Function on D-Flash Memory
0x10	Erase Verify D-Flash Section	Verify that a given number of words starting at the address provided are erased.
0x11	Program D-Flash	Program up to four words in the D-Flash block.
0x12	Erase D-Flash Sector	Erase all bytes in a sector of the D-Flash block.

17.4.4 Allowed Simultaneous P-Flash and D-Flash Operations

Only the operations marked ‘OK’ in Table 17-30 are permitted to be run simultaneously on the Program Flash and Data Flash blocks. Some operations cannot be executed simultaneously because certain hardware resources are shared by the two memories. The priority has been placed on permitting Program Flash reads while program and erase operations execute on the Data Flash, providing read (P-Flash) while write (D-Flash) functionality.

Table 17-30. Allowed P-Flash and D-Flash Simultaneous Operations

	Data Flash				
Program Flash	**Read**	**Margin Read[1]**	**Program**	**Sector Erase**	**Mass Erase[3]**
Read		OK	OK	OK	
Margin Read[1]		OK[2]			
Program					
Sector Erase				OK	
Mass Erase[3]					OK

1 A ‘Margin Read’ is any read after executing the margin setting commands ‘Set User Margin Level’ or ‘Set Field Margin Level’ with anything but the ‘normal’ level specified.

2 See the Note on margin settings in Section 17.4.5.12 and Section 17.4.5.13.

3 The ‘Mass Erase’ operations are commands ‘Erase All Blocks’ and ‘Erase Flash Block’

17.4.5 Flash Command Description

This section provides details of all available Flash commands launched by a command write sequence. The ACCERR bit in the FSTAT register will be set during the command write sequence if any of the following illegal steps are performed, causing the command not to be processed by the Memory Controller:

- Starting any command write sequence that programs or erases Flash memory before initializing the FCLKDIV register
- Writing an invalid command as part of the command write sequence
- For additional possible errors, refer to the error handling table provided for each command

Table 17-50. Unsecure Flash Command Error Handling

Register	Error Bit	Error Condition
FSTAT	ACCERR	Set if CCOBIX[2:0] != 000 at command launch
		Set if command not available in current mode (see Table 17-27)
	FPVIOL	Set if any area of the P-Flash or D-Flash memory is protected
	MGSTAT1	Set if any errors have been encountered during the verify operation
	MGSTAT0	Set if any non-correctable errors have been encountered during the verify operation

17.4.5.11 Verify Backdoor Access Key Command

The Verify Backdoor Access Key command will only execute if it is enabled by the KEYEN bits in the FSEC register (see Table 17-9). The Verify Backdoor Access Key command releases security if user-supplied keys match those stored in the Flash security bytes of the Flash configuration field (see Table 17-3). The Verify Backdoor Access Key command must not be executed from the Flash block containing the backdoor comparison key to avoid code runaway.

Table 17-51. Verify Backdoor Access Key Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x0C	Not required
001	Key 0	
010	Key 1	
011	Key 2	
100	Key 3	

Upon clearing CCIF to launch the Verify Backdoor Access Key command, the Memory Controller will check the FSEC KEYEN bits to verify that this command is enabled. If not enabled, the Memory Controller sets the ACCERR bit in the FSTAT register and terminates. If the command is enabled, the Memory Controller compares the key provided in FCCOB to the backdoor comparison key in the Flash configuration field with Key 0 compared to 0x3_FF00, etc. If the backdoor keys match, security will be released. If the backdoor keys do not match, security is not released and all future attempts to execute the Verify Backdoor Access Key command are aborted (set ACCERR) until a reset occurs. The CCIF flag is set after the Verify Backdoor Access Key operation has completed.

register (see Table 17-10) will be changed to unsecure the MCU. Key values of 0x0000 and 0xFFFF are not permitted as backdoor keys. While the Verify Backdoor Access Key command is active, P-Flash memory and D-Flash memory will not be available for read access and will return invalid data.

The user code stored in the P-Flash memory must have a method of receiving the backdoor keys from an external stimulus. This external stimulus would typically be through one of the on-chip serial ports.

If the KEYEN[1:0] bits are in the enabled state (see Section 17.3.2.2), the MCU can be unsecured by the backdoor key access sequence described below:

1. Follow the command sequence for the Verify Backdoor Access Key command as explained in Section 17.4.5.11
2. If the Verify Backdoor Access Key command is successful, the MCU is unsecured and the SEC[1:0] bits in the FSEC register are forced to the unsecure state of 10

The Verify Backdoor Access Key command is monitored by the Memory Controller and an illegal key will prohibit future use of the Verify Backdoor Access Key command. A reset of the MCU is the only method to re-enable the Verify Backdoor Access Key command. The security as defined in the Flash security byte (0x3_FF0F) is not changed by using the Verify Backdoor Access Key command sequence. The backdoor keys stored in addresses 0x3_FF00-0x3_FF07 are unaffected by the Verify Backdoor Access Key command sequence. The Verify Backdoor Access Key command sequence has no effect on the program and erase protections defined in the Flash protection register, FPROT.

After the backdoor keys have been correctly matched, the MCU will be unsecured. After the MCU is unsecured, the sector containing the Flash security byte can be erased and the Flash security byte can be reprogrammed to the unsecure state, if desired. In the unsecure state, the user has full control of the contents of the backdoor keys by programming addresses 0x3_FF00-0x3_FF07 in the Flash configuration field.

17.5.2 Unsecuring the MCU in Special Single Chip Mode using BDM

A secured MCU can be unsecured in special single chip mode by using the following method to erase the P-Flash and D-Flash memory:

1. Reset the MCU into special single chip mode
2. Delay while the BDM executes the Erase Verify All Blocks command write sequence to check if the P-Flash and D-Flash memories are erased
3. Send BDM commands to disable protection in the P-Flash and D-Flash memory
4. Execute the Erase All Blocks command write sequence to erase the P-Flash and D-Flash memory
5. After the CCIF flag sets to indicate that the Erase All Blocks operation has completed, reset the MCU into special single chip mode
6. Delay while the BDM executes the Erase Verify All Blocks command write sequence to verify that the P-Flash and D-Flash memory are erased

If the P-Flash and D-Flash memory are verified as erased, the MCU will be unsecured. All BDM commands will now be enabled and the Flash security byte may be programmed to the unsecure state by continuing with the following steps:

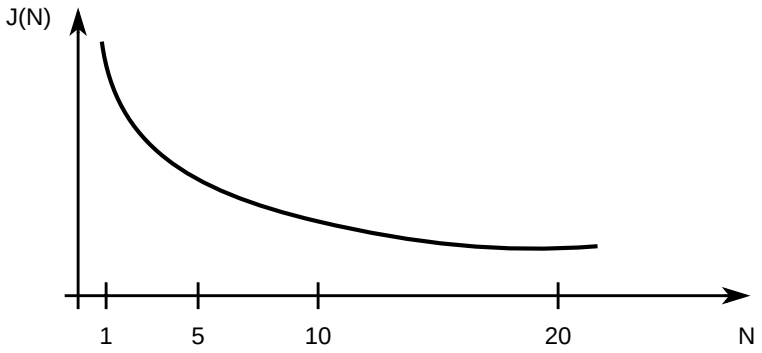

Figure A-3. Maximum Bus Clock Jitter Approximation

NOTE

On timers and serial modules a prescaler will eliminate the effect of the jitter to a large extent.

A.6 Electrical Characteristics for the PLL

Table A-18. PLL Characteristics

Conditions are shown in Table A-4 unless otherwise noted							
Num	**C**	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	D	VCO frequency during system reset	f_{VCORST}	8		32	MHz
2	C	VCO locking range	f_{VCO}	32		64	MHz
3	C	Reference Clock	f_{REF}	1		16	MHz
4	D	Lock Detection	$\|\Delta_{Lock}\|$	0		1.5	%(1)
6	D	Un-Lock Detection	$\|\Delta_{unl}\|$	0.5		2.5	%1
7	C	Time to lock	t_{lock}			150 + $256/f_{REF}$	µs
8	C	Jitter fit parameter 1(2)	j_1			1.2	%

1. % deviation from target frequency

2. f_{REF} = 1MHz, f_{BUS} = 32MHz equivalent f_{PLL} = 64MHz, REFRQ=00, SYNDIV=$1F, VCOFRQ=01, POSTDIV=$00

A.7 Electrical Characteristics for the IRC1M

Table A-19. IRC1M Characteristics

Conditions are shown in Table A-4 unless otherwise noted							
Num	**C**	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	P	Junction Temperature -40°C to 150°C Internal Reference Frequency, factory trimmed	f_{IRC1M_TRIM}	0.98	1	1.02	MHz
2	P	Junction Temperature Range -40°C to 105°C Internal Reference Frequency, factory trimmed	f_{IRC1M_TRIM}	0.985	1	1.015	MHz

Appendix C
Package Information

C.1 100-Pin LQFP Mechanical Dimensions

Figure C-1. 100-pin LQFP (case no. 983) - page 1

0x0010–0x0017 Module Mapping Control (MMC) Map 2 of 2

Address	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x00010	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0011	DIRECT	R	DP15	DP14	DP13	DP12	DP11	DP10	DP9	DP8
		W								
0x0012	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0013	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0014	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0015	PPAGE	R	0	0	0	0	PIX3	PIX2	PIX1	PIX0
		W								
0x0016-0x0017	Reserved	R	0	0	0	0	0	0	0	0
		W								

0x0018–0x001B Miscellaneous Peripheral

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0018	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0019	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x001A	PARTIDH(1)	R	0	0	0	1	1	0	1	0
		W								
0x001B	PARTIDL[1]	R	1	0	0	0	0	0	0	0
		W								

1. Refer to Part ID assignments in the device description section for a full list of MC9S12HY/HAPart ID values.

0x001C–0x001F Port Integration Module (PIM) Map 3 of 6

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x001C	ECLKCTL	R	NECLK	0	DIV16	EDIV4	EDIV3	EDIV2	EDIV1	EDIV0
		W								
0x001D	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x001E	IRQCR	R	IRQE	IRQEN	XIRQEN	0	0	0	0	0
		W								
0x001F	Reserved	R	0	0	0	0	0	0	0	0
		W								