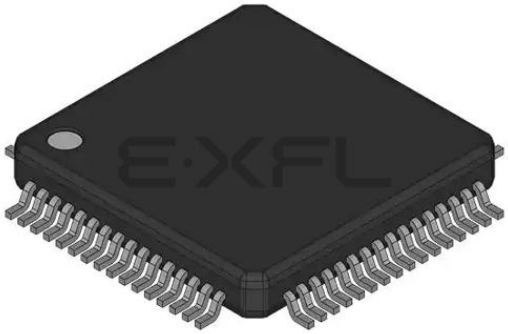


Welcome to E-XFL.COM

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	50
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=s9s12hy32j0mlh

1.7.2 Pin Assignment Overview

Table 1-6 provides a summary of which ports are available for each package option. Routing of pin functions is summarized in Table 1-7.

Table 1-6. Port Availability by Package Option

Port	100 LQFP	64 LQFP
Port AD/ADC Channels	8/8	6/6
Port A	8	4
Port B	8	4
Port H	8	4
Port P	8	4
Port R	8	4
Port S	8	4
Port T	8	8
Port U	8	8
Port V	8	4
Sum of Ports	80	50
I/O Power Pairs VDDM/VSSM	2/2	1/1
I/O Power Pairs VDDX/VSSX	1/1	1/1
I/O Power Pairs VDDA/VSSA(1)	1/1	1/1
VREG Power Pairs VDDR/VSS3	1/1	1/1
I/O Power Pair VSSPLL	1	0(2)
VLCD power	1	1

1. VRH/VRL are sharing with VDDA/VSSA pins
2. Double bond with VSS3 on 64LQFP package

Table 1-8. Pin-Out Summary[1] (Sheet 7 of 8)

Package Pin		Function						Power Supply	Internal Pull Resistor		Description
100 LQ FP	64 LQ FP	Pin	2nd Func.	3rd Func.	4th Func.	5th Func.	6th Func.		CTRL	Reset State	
78	—	PA4	FP33	—	—	—	—	V_{DDX}	PUCR	Down	Port A I/O, LCD Frontplane driver
79	—	PA5	FP34	—	—	—	—	V_{DDX}	PUCR	Down	Port A I/O, LCD Frontplane driver
80	—	PA6	FP35	—	—	—	—	V_{DDX}	PUCR	Down	Port A I/O, LCD Frontplane driver
81	—	PA7	FP36	—	—	—	—	V_{DDX}	PUCR	Down	Port A I/O, LCD Frontplane driver
82	—	PB1	FP37	—	—	—	—	V_{DDX}	PUCR	Down	Port B I/O, LCD Frontplane driver
83	—	PB2	FP38	—	—	—	—	V_{DDX}	PUCR	Down	Port B I/O, LCD Frontplane driver
84	—	PB3	FP39	—	—	—	—	V_{DDX}	PUCR	Down	Port B I/O, LCD Frontplane driver
85	51	PB4	BP0	—	—	—	—	V_{DDX}	PUCR	Down	Port B I/O, LCD Backplane driver
86	52	PB5	BP1	—	—	—	—	V_{DDX}	PUCR	Down	Port B I/O, LCD Backplane driver
87	53	PB6	BP2	—	—	—	—	V_{DDX}	PUCR	Down	Port B I/O, LCD Backplane driver
88	54	PB7	BP3	—	—	—	—	V_{DDX}	PUCR	Down	Port B I/O, LCD Backplane driver
89	55	VLCD	—	—	—	—	—	V_{DDX}	—	—	Voltage reference pin for the LCD driver.
90	56	BKGD	MODC	—	—	—	—	V_{DDX}	Always on	Up	Background debug, Mode selection pin
91	57	VSSA	VRL	—	—	—	—	—	—	—	—

2.3.4 Port B Data Register (PORTB)

Address 0x0001 (PRR) Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB7
Altern. Function	BP3	BP2	BP1	BP0	FP39	FP38	FP37	FP28
Reset	0	0	0	0	0	0	0	0

Figure 2-2. Port B Data Register (PORTB)

1 Read: Anytime. The data source is depending on the data direction value.
Write: Anytime

Table 2-5. PORTB Register Field Descriptions

Field	Description
7-0 PB	**Port B general purpose input/output data**—Data Register, LCD segment driver output The associated pin can be used as general purpose I/O when not used as alternative function. In general purpose output mode the register bit value is driven to the pin. If the associated data direction bit is set to 1, a read returns the value of the port register bit, otherwise the buffered pin input state is read. • The LCD segment driver output takes precedence over the general purpose I/O function if the related LCD segment is enabled.

2.3.5 Port A Data Direction Register (DDRA)

Address 0x0002 (PRR) Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
Reset	0	0	0	0	0	0	0	0

Figure 2-3. Port A Data Direction Register (DDRA)

1 Read: Anytime
Write: Anytime

Table 2-18. Port T Routing Register Field Descriptions

Field	Description
5 PTTRR	**Port T data direction**— This register controls the routing of IOC0_7. 0 IOC0_7 routed to PT7 1 IOC0_7 routed to PR1
4 PTTRR	**Port T data direction**— This register controls the routing of IOC0_6. 0 IOC0_6 routed to PT6 1 IOC0_6 routed to PR0
1 PTTRR	**Port T data direction**— This register controls the routing of IOC1_7. 0 IOC1_7routed to PT3 1 IOC1_7 routed to PR3
0 PTTRR	**Port T data direction**— This register controls the routing of IOC1_6. 0 IOC1_6 routed to PT2 1 IOC1_6 routed to PR2

2.3.22 Port S Data Register (PTS)

Address 0x0248 Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	PTS7	PTS6	PTS5	PTS4	PTS3	PTS2	PTS1	PTS0
	PWM3	PWM2	PWM1	PWM0	—	—	PWM7	PWM6
	SDA	—	—	SCL	—	—	—	—
Altern. Function	$\overline{SS}$	SCK	MOSI	MISO	TXCAN	RXCAN	TXD	RXD
Reset	0	0	0	0	0	0	0	0

Figure 2-20. Port S Data Register (PTS)

1 Read: Anytime The data source is depending on the data direction value.
Write: Anytime

Table 2-70. DDRV Register Field Descriptions (continued)

Field	Description
1 DDRV	**Port V data direction**— If the Motor driver PWM output is enabled, it will force the I/O state to be output Else if SPI is routing to PV and SPI is enabled, SPI will determine the I/O state Else if PWM5 is routing to PV, it will force I/O state to be output Else if SPI is routing to PV and SPI is enabled, SPI will determine the I/O state. 1 Associated pin is configured as output. 0 Associated pin is configured as input.
0 DDRV	**Port V data direction**— If the Motor driver PWM output is enabled, it will force the I/O state to be output Else if corresponding TIM1 output compare channel is enabled, it will be forced as output Else if IIC is routing to PV and IIC is enabled, it will force the I/O state to be open drain output, also the input buffer is enabled Else if PWM4 is routing to PV, it will force I/O state to be output Else if SPI is routing to PV and SPI is enabled, SPI will determine the I/O state. 1 Associated pin is configured as output. 0 Associated pin is configured as input.

NOTE

Due to internal synchronization circuits, it can take up to 2 bus clock cycles until the correct value is read on PTV or PTIV registers, when changing the DDRV register.

2.3.84 PIM Reserved Registers

Address 0x029B Access: User read[1]

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved u = Unaffected by reset

Figure 2-82. PIM Reserved Registers

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

Table 4-3. IVBR Field Descriptions

Field	Description
7–0 IVB_ADDR[7:0]	**Interrupt Vector Base Address Bits** — These bits represent the upper byte of all vector addresses. Out of reset these bits are set to 0xFF (that means vectors are located at 0xFF80–0xFFFE) to ensure compatibility to HCS12. **Note:** A system reset will initialize the interrupt vector base register with “0xFF” before it is used to determine the reset vector address. Therefore, changing the IVBR has no effect on the location of the three reset vectors (0xFFFA–0xFFFE). **Note:** If the BDM is active (that means the CPU is in the process of executing BDM firmware code), the contents of IVBR are ignored and the upper byte of the vector address is fixed as “0xFF”. This is done to enable handling of all non-maskable interrupts in the BDM firmware.

4.4 Functional Description

The INT module processes all exception requests to be serviced by the CPU module. These exceptions include interrupt vector requests and reset vector requests. Each of these exception types and their overall priority level is discussed in the subsections below.

4.4.1 S12S Exception Requests

The CPU handles both reset requests and interrupt requests. A priority decoder is used to evaluate the priority of pending interrupt requests.

4.4.2 Interrupt Prioritization

The INT module contains a priority decoder to determine the priority for all interrupt requests pending for the CPU. If more than one interrupt request is pending, the interrupt request with the higher vector address wins the prioritization.

The following conditions must be met for an I bit maskable interrupt request to be processed.

1. The local interrupt enabled bit in the peripheral module must be set.
2. The I bit in the condition code register (CCR) of the CPU must be cleared.
3. There is no SWI, TRAP, or X bit maskable request pending.

NOTE

All non I bit maskable interrupt requests always have higher priority than the I bit maskable interrupt requests. If the X bit in the CCR is cleared, it is possible to interrupt an I bit maskable interrupt by an X bit maskable interrupt. It is possible to nest non maskable interrupt requests, for example by nesting SWI or TRAP calls.

Since an interrupt vector is only supplied at the time when the CPU requests it, it is possible that a higher priority interrupt request could override the original interrupt request that caused the CPU to request the vector. In this case, the CPU will receive the highest priority vector and the system will process this interrupt request first, before the original interrupt request is processed.

- Hardware handshake protocol to increase the performance of the serial communication
- Active out of reset in special single chip mode
- Nine hardware commands using free cycles, if available, for minimal CPU intervention
- Hardware commands not requiring active BDM
- 14 firmware commands execute from the standard BDM firmware lookup table
- Software control of BDM operation during wait mode
- When secured, hardware commands are allowed to access the register space in special single chip mode, if the Flash erase tests fail.
- Family ID readable from BDM ROM at global address 0x3_FF0F in active BDM (value for devices with HCS12S core is 0xC2)
- BDM hardware commands are operational until system stop mode is entered

5.1.2 Modes of Operation

BDM is available in all operating modes but must be enabled before firmware commands are executed. Some systems may have a control bit that allows suspending the function during background debug mode.

5.1.2.1 Regular Run Modes

All of these operations refer to the part in run mode and not being secured. The BDM does not provide controls to conserve power during run mode.

- Normal modes

  General operation of the BDM is available and operates the same in all normal modes.
- Special single chip mode

  In special single chip mode, background operation is enabled and active out of reset. This allows programming a system with blank memory.

5.1.2.2 Secure Mode Operation

If the device is in secure mode, the operation of the BDM is reduced to a small subset of its regular run mode operation. Secure operation prevents access to Flash other than allowing erasure. For more information please see Section 5.4.1, “Security”.

5.1.2.3 Low-Power Modes

The BDM can be used until stop mode is entered. When CPU is in wait mode all BDM firmware commands as well as the hardware BACKGROUND command cannot be used and are ignored. In this case the CPU can not enter BDM active mode, and only hardware read and write commands are available. Also the CPU can not enter a low power mode (stop or wait) during BDM active mode.

In stop mode the BDM clocks are stopped. When BDM clocks are disabled and stop mode is exited, the BDM clocks will restart and BDM will have a soft reset (clearing the instruction register, any command in progress and disable the ACK function). The BDM is now ready to receive a new command.

Table 5-2. BDM Memory Map

Global Address	Module	Size (Bytes)
0x3_FF0C–0x3_FF0E	BDM firmware ROM	3
0x3_FF0F	Family ID (part of BDM firmware ROM)	1
0x3_FF10–0x3_FFFF	BDM firmware ROM	240

5.3.2 Register Descriptions

A summary of the registers associated with the BDM is shown in Figure 5-2. Registers are accessed by host-driven communications to the BDM hardware using READ_BD and WRITE_BD commands.

Global Address	Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x3_FF00	Reserved	R	X	X	X	X	X	X	0	0
		W								
0x3_FF01	BDMSTS	R	ENBDM	BDMACT	0	SDV	TRACE	0	UNSEC	0
		W								
0x3_FF02	Reserved	R	X	X	X	X	X	X	X	X
		W								
0x3_FF03	Reserved	R	X	X	X	X	X	X	X	X
		W								
0x3_FF04	Reserved	R	X	X	X	X	X	X	X	X
		W								
0x3_FF05	Reserved	R	X	X	X	X	X	X	X	X
		W								
0x3_FF06	BDMCCR	R	CCR7	CCR6	CCR5	CCR4	CCR3	CCR2	CCR1	CCR0
		W								
0x3_FF07	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x3_FF08	BDMPPR	R	BPAE	0	0	0	BPP3	BPP2	BPP1	BPP0
		W								

= Unimplemented, Reserved — = Implemented (do not alter)

X = Indeterminate — 0 = Always read zero

Figure 5-2. BDM Register Summary

Table 5-5. Hardware Commands (continued)

Command	Opcode (hex)	Data	Description
READ_BD_BYTE	E4	16-bit address 16-bit data out	Read from memory with standard BDM firmware lookup table in map. Odd address data on low byte; even address data on high byte.
READ_BD_WORD	EC	16-bit address 16-bit data out	Read from memory with standard BDM firmware lookup table in map. Must be aligned access.
READ_BYTE	E0	16-bit address 16-bit data out	Read from memory with standard BDM firmware lookup table out of map. Odd address data on low byte; even address data on high byte.
READ_WORD	E8	16-bit address 16-bit data out	Read from memory with standard BDM firmware lookup table out of map. Must be aligned access.
WRITE_BD_BYTE	C4	16-bit address 16-bit data in	Write to memory with standard BDM firmware lookup table in map. Odd address data on low byte; even address data on high byte.
WRITE_BD_WORD	CC	16-bit address 16-bit data in	Write to memory with standard BDM firmware lookup table in map. Must be aligned access.
WRITE_BYTE	C0	16-bit address 16-bit data in	Write to memory with standard BDM firmware lookup table out of map. Odd address data on low byte; even address data on high byte.
WRITE_WORD	C8	16-bit address 16-bit data in	Write to memory with standard BDM firmware lookup table out of map. Must be aligned access.

NOTE:
If enabled, ACK will occur when data is ready for transmission for all BDM READ commands and will occur after the write is complete for all BDM WRITE commands.

5.4.4 Standard BDM Firmware Commands

BDM firmware commands are used to access and manipulate CPU resources. The system must be in active BDM to execute standard BDM firmware commands, see Section 5.4.2, “Enabling and Activating BDM”. Normal instruction execution is suspended while the CPU executes the firmware located in the standard BDM firmware lookup table. The hardware command BACKGROUND is the usual way to activate BDM.

As the system enters active BDM, the standard BDM firmware lookup table and BDM registers become visible in the on-chip memory map at 0x3_FF00–0x3_FFFF, and the CPU begins executing the standard BDM firmware. The standard BDM firmware watches for serial commands and executes them as they are received.

The firmware commands are shown in Table 5-6.

NOTE

When program control returns from a tagged breakpoint using an RTI or BDM GO command without program counter modification it returns to the instruction whose tag generated the breakpoint. To avoid a repeated breakpoint at the same location reconfigure the DBG module in the SWI routine, if configured for an SWI breakpoint, or over the BDM interface by executing a TRACE command before the GO to increment the program flow past the tagged instruction.

6.5 Application Information

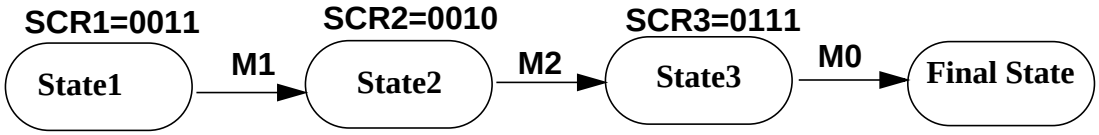
6.5.1 State Machine scenarios

Defining the state control registers as SCR1,SCR2, SCR3 and M0,M1,M2 as matches on channels 0,1,2 respectively. SCR encoding supported by S12SDBGV1 are shown in black. SCR encoding supported only in S12SDBGV2 are shown in red. For backwards compatibility the new scenarios use a 4th bit in each SCR register. Thus the existing encoding for SCRx[2:0] is not changed.

6.5.2 Scenario 1

A trigger is generated if a given sequence of 3 code events is executed.

Figure 6-27. Scenario 1



Scenario 1 is possible with S12SDBGV1 SCR encoding

6.5.3 Scenario 2

A trigger is generated if a given sequence of 2 code events is executed.

Figure 6-28. Scenario 2a

NOTE

The Adaptive Oscillator Filter uses the VCO clock as a reference to continuously qualify the external oscillator clock. Because of this, the PLL is always active and a valid PLL configuration is required for the system to work properly. Furthermore, the Adaptive Oscillator Filter is used to determine the status of the external oscillator (reflected in the UPOSC bit). Since this function also relies on the VCO clock, loosing PLL lock status (LOCK=0, except for entering Pseudo Stop Mode) means loosing the oscillator status information as well (UPOSC=0).

Write: Anytime

11.3.2.15 PWM Shutdown Register (PWMSDN)

The PWMSDN register provides for the shutdown functionality of the PWM module in the emergency cases. For proper operation, channel 7 must be driven to the active level for a minimum of two bus clocks.

Module Base + 0x0024

	7	6	5	4	3	2	1	0
R	PWMIF	PWMIE	0	PWMLVL	0	PWM7IN	PWM7INL	PWM7ENA
W			PWMRSTRT					
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 11-17. PWM Shutdown Register (PWMSDN)

Read: Anytime

Write: Anytime

Table 11-9. PWMSDN Field Descriptions

Field	Description
7 PWMIF	**PWM Interrupt Flag** — Any change from passive to asserted (active) state or from active to passive state will be flagged by setting the PWMIF flag = 1. The flag is cleared by writing a logic 1 to it. Writing a 0 has no effect. 0 No change on PWM7IN input. 1 Change on PWM7IN input
6 PWMIE	**PWM Interrupt Enable** — If interrupt is enabled an interrupt to the CPU is asserted. 0 PWM interrupt is disabled. 1 PWM interrupt is enabled.
5 PWMRSTRT	**PWM Restart** — The PWM can only be restarted if the PWM channel input 7 is de-asserted. After writing a logic 1 to the PWMRSTRT bit (trigger event) the PWM channels start running after the corresponding counter passes next “counter == 0” phase. Also, if the PWM7ENA bit is reset to 0, the PWM do not start before the counter passes $00. The bit is always read as “0”.
4 PWMLVL	**PWM Shutdown Output Level** If active level as defined by the PWM7IN input, gets asserted all enabled PWM channels are immediately driven to the level defined by PWMLVL. 0 PWM outputs are forced to 0 1 Outputs are forced to 1.
2 PWM7IN	**PWM Channel 7 Input Status** — This reflects the current status of the PWM7 pin.
1 PWM7INL	**PWM Shutdown Active Input Level for Channel 7** — If the emergency shutdown feature is enabled (PWM7ENA = 1), this bit determines the active level of the PWM7channel. 0 Active level is low 1 Active level is high
0 PWM7ENA	**PWM Emergency Shutdown Enable** — If this bit is logic 1, the pin associated with channel 7 is forced to input and the emergency shutdown feature is enabled. All the other bits in this register are meaningful only if PWM7ENA = 1. 0 PWM emergency feature disabled. 1 PWM emergency feature is enabled.

Table 13-9. SPIF Interrupt Flag Clearing Sequence

XFRW Bit	SPIF Interrupt Flag Clearing Sequence		
0	Read SPISR with SPIF == 1	**then**	Read SPIDRL
1	Read SPISR with SPIF == 1	**then**	Byte Read SPIDRL [1]
			or
			Byte Read SPIDRH [2] then Byte Read SPIDRL
			or
			Word Read (SPIDRH:SPIDRL)

1 Data in SPIDRH is lost in this case.

2 SPIDRH can be read repeatedly without any effect on SPIF. SPIF Flag is cleared only by the read of SPIDRL after reading SPISR with SPIF == 1.

Table 13-10. SPTEF Interrupt Flag Clearing Sequence

XFRW Bit	SPTEF Interrupt Flag Clearing Sequence		
0	Read SPISR with SPTEF == 1	**then**	Write to SPIDRL [1]
1	Read SPISR with SPTEF == 1	**then**	Byte Write to SPIDRL [12]
			or
			Byte Write to SPIDRH [13] then Byte Write to SPIDRL [1]
			or
			Word Write to (SPIDRH:SPIDRL) [1]

1 Any write to SPIDRH or SPIDRL with SPTEF == 0 is effectively ignored.

2 Data in SPIDRH is undefined in this case.

3 SPIDRH can be written repeatedly without any effect on SPTEF. SPTEF Flag is cleared only by writing to SPIDRL after reading SPISR with SPTEF == 1.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0006 TSCR1	R	TEN	TSWAI	TSFRZ	TFFCA	PRNT	0	0	0
	W								
0x0007 TTOV	R	TOV7	TOV6	TOV5	TOV4	TOV3	TOV2	TOV1	TOV0
	W								
0x0008 TCTL1	R	OM7	OL7	OM6	OL6	OM5	OL5	OM4	OL4
	W								
0x0009 TCTL2	R	OM3	OL3	OM2	OL2	OM1	OL1	OM0	OL0
	W								
0x000A TCTL3	R	EDG7B	EDG7A	EDG6B	EDG6A	EDG5B	EDG5A	EDG4B	EDG4A
	W								
0x000B TCTL4	R	EDG3B	EDG3A	EDG2B	EDG2A	EDG1B	EDG1A	EDG0B	EDG0A
	W								
0x000C TIE	R	C7I	C6I	C5I	C4I	C3I	C2I	C1I	C0I
	W								
0x000D TSCR2	R	TOI	0	0	0	TCRE	PR2	PR1	PR0
	W								
0x000E TFLG1	R	C7F	C6F	C5F	C4F	C3F	C2F	C1F	C0F
	W								
0x000F TFLG2	R	TOF	0	0	0	0	0	0	0
	W								
0x0010–0x001F TCxH–TCxL	R	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
	W								
	R	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	W								
0x0020 PACTL	R	0	PAEN	PAMOD	PEDGE	CLK1	CLK0	PAOVI	PAI
	W								
0x0021 PAFLG	R	0	0	0	0	0	0	PAOVF	PAIF
	W								
0x0022 PACNTH	R	PACNT15	PACNT14	PACNT13	PACNT12	PACNT11	PACNT10	PACNT9	PACNT8
	W								
0x0023 PACNTL	R	PACNT7	PACNT6	PACNT5	PACNT4	PACNT3	PACNT2	PACNT1	PACNT0
	W								
0x0024–0x002B Reserved	R								
	W								

= Unimplemented or Reserved

Figure 14-5. TIM16B8CV2 Register Summary (Sheet 2 of 3)

NOTE

The newly selected prescale factor will not take effect until the next synchronized edge where all prescale counter stages equal zero.

14.3.2.12 Main Timer Interrupt Flag 1 (TFLG1)

Module Base + 0x000E

	7	6	5	4	3	2	1	0
R W	C7F	C6F	C5F	C4F	C3F	C2F	C1F	C0F
Reset	0	0	0	0	0	0	0	0

Figure 14-20. Main Timer Interrupt Flag 1 (TFLG1)

Read: Anytime

Write: Used in the clearing mechanism (set bits cause corresponding bits to be cleared). Writing a zero will not affect current status of the bit.

Table 14-16. TRLG1 Field Descriptions

Field	Description
7:0 C[7:0]F	**Input Capture/Output Compare Channel “x” Flag** — These flags are set when an input capture or output compare event occurs. Clearing requires writing a one to the corresponding flag bit while TEN or PAEN is set to one. When TFFCA bit in TSCR register is set, a read from an input capture or a write into an output compare channel (0x0010–0x001F) will cause the corresponding channel flag CxF to be cleared.

14.3.2.13 Main Timer Interrupt Flag 2 (TFLG2)

Module Base + 0x000F

	7	6	5	4	3	2	1	0
R	TOF	0	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

Unimplemented or Reserved

Figure 14-21. Main Timer Interrupt Flag 2 (TFLG2)

TFLG2 indicates when interrupt conditions have occurred. To clear a bit in the flag register, write the bit to one while TEN bit of TSCR1 or PAEN bit of PACTL is set to one.

Read: Anytime

Write: Used in clearing mechanism (set bits cause corresponding bits to be cleared).

Any access to TCNT will clear TFLG2 register if the TFFCA bit in TSCR register is set.

15.3.2.9.1 P-Flash Protection Restrictions

The general guideline is that P-Flash protection can only be added and not removed. Table 15-20 specifies all valid transitions between P-Flash protection scenarios. Any attempt to write an invalid scenario to the FPROT register will be ignored. The contents of the FPROT register reflect the active protection scenario. See the FPHS and FPLS bit descriptions for additional restrictions.

Table 15-20. P-Flash Protection Scenario Transitions

From Protection Scenario	To Protection Scenario[1]							
	0	1	2	3	4	5	6	7
0	X	X	X	X				
1		X		X				
2			X	X				
3				X				
4				X	X			
5			X	X	X	X		
6		X		X	X		X	
7	X	X	X	X	X	X	X	X

1 Allowed transitions marked with X, see Figure 15-14 for a definition of the scenarios.

15.3.2.10 D-Flash Protection Register (DFPROT)

The DFPROT register defines which D-Flash sectors are protected against program and erase operations.

Offset Module Base + 0x0009

	7	6	5	4	3	2	1	0
R	DPOPEN	0	0	0	DPS[3:0]			
W								
Reset	F	0	0	0	F	F	F	F

= Unimplemented or Reserved

Figure 15-15. D-Flash Protection Register (DFPROT)

The (unreserved) bits of the DFPROT register are writable with the restriction that protection can be added but not removed. Writes must increase the DPS value and the DPOPEN bit can only be written from 1 (protection disabled) to 0 (protection enabled). If the DPOPEN bit is set, the state of the DPS bits is irrelevant.

During the reset sequence, the DFPROT register is loaded with the contents of the D-Flash protection byte in the Flash configuration field at global address 0x3_FF0D located in P-Flash memory (see Table 15-3) as indicated by reset condition F in Figure 15-15. To change the D-Flash protection that will be loaded during the reset sequence, the P-Flash sector containing the D-Flash protection byte must be unprotected, then the D-Flash protection byte must be programmed. If a double bit fault is detected while reading the

(CCIF=0) the user must not write to any Flash module register. The CCIF flag will set after the Erase All Blocks operation has completed.

Table 15-44. Erase All Blocks 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 15-27)
	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[1]
	MGSTAT0	Set if any non-correctable errors have been encountered during the verify operation[1]

1 As found in the memory map for FTMRC64K1.

15.4.5.8 Erase Flash Block Command

The Erase Flash Block operation will erase all addresses in a P-Flash or D-Flash block.

Table 15-45. Erase Flash Block Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x09	Global address [17:16] to identify Flash block
001	Global address [15:0] in Flash block to be erased	

Upon clearing CCIF to launch the Erase Flash Block command, the Memory Controller will erase the selected Flash block and verify that it is erased. The CCIF flag will set after the Erase Flash Block operation has completed.

Table 15-46. Erase Flash Block 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 15-27)
		Set if an invalid global address [17:16] is supplied[1]
		Set if the supplied P-Flash address is not phrase-aligned or if the D-Flash address is not word-aligned
	FPVIOL	Set if an area of the selected Flash block is protected
	MGSTAT1	Set if any errors have been encountered during the verify operation[2]
	MGSTAT0	Set if any non-correctable errors have been encountered during the verify operation[2]

1 As defined by the memory map for FTMRC64K1.

2 As found in the memory map for FTMRC64K1.

Offset Module Base + 0x0008

	7	6	5	4	3	2	1	0
R	FPOPEN	RNV6	FPHDIS	FPHS[1:0]		FPLDIS	FPLS[1:0]	
W								
Reset	F	F	F	F	F	F	F	F

= Unimplemented or Reserved

Figure 16-13. Flash Protection Register (FPROT)

The (unreserved) bits of the FPROT register are writable with the restriction that the size of the protected region can only be increased (see Section 16.3.2.9.1, “P-Flash Protection Restrictions,” and Table 16-20).

During the reset sequence, the FPROT register is loaded with the contents of the P-Flash protection byte in the Flash configuration field at global address 0x3_FF0C located in P-Flash memory (see Table 16-3) as indicated by reset condition ‘F’ in Figure 16-13. To change the P-Flash protection that will be loaded during the reset sequence, the upper sector of the P-Flash memory must be unprotected, then the P-Flash protection byte must be reprogrammed. If a double bit fault is detected while reading the P-Flash phrase containing the P-Flash protection byte during the reset sequence, the FPOPEN bit will be cleared and remaining bits in the FPROT register will be set to leave the P-Flash memory fully protected.

Trying to alter data in any protected area in the P-Flash memory will result in a protection violation error and the FPVIOL bit will be set in the FSTAT register. The block erase of a P-Flash block is not possible if any of the P-Flash sectors contained in the same P-Flash block are protected.

Table 16-16. FPROT Field Descriptions

Field	Description
6 RNV[6]	**Reserved Nonvolatile Bit** — The RNV bit should remain in the erased state for future enhancements.
5 FPHDIS	**Flash Protection Higher Address Range Disable** — The FPHDIS bit determines whether there is a protected/unprotected area in a specific region of the P-Flash memory ending with global address 0x3_FFFF. 0 Protection/Unprotection enabled 1 Protection/Unprotection disabled
4–3 FPHS[1:0]	**Flash Protection Higher Address Size** — The FPHS bits determine the size of the protected/unprotected area in P-Flash memory as shown inTable 16-18. The FPHS bits can only be written to while the FPHDIS bit is set.

Table 16-17. P-Flash Protection Function

FPOPEN	FPHDIS	FPLDIS	Function[1]
1	1	1	No P-Flash Protection
1	1	0	Protected Low Range
1	0	1	Protected High Range
1	0	0	Protected High and Low Ranges
0	1	1	Full P-Flash Memory Protected
0	1	0	Unprotected Low Range
0	0	1	Unprotected High Range
0	0	0	Unprotected High and Low Ranges

return CCIF to 1 and the FCCOB register will be used to communicate any results. The flow for a generic command write sequence is shown in Figure 17-26.

A.2 ATD Characteristics

This section describes the characteristics of the analog-to-digital converter.

A.2.1 ATD Operating Characteristics

The Table A-12 and Table A-13 show conditions under which the ATD operates.

The following constraints exist to obtain full-scale, full range results:

V_{SSA} (V_{RL})≤ V_{IN} ≤ V_{DDA}(V_{RH}).

This constraint exists since the sample buffer amplifier can not drive beyond the power supply levels that it ties to. If the input level goes outside of this range it will effectively be clipped.

Table A-12. ATD Operating Characteristics

Conditions are shown in Table A-4 unless otherwise noted, supply voltage 3.13 V < V_{DDA} < 5.5 V							
Num	**C**	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	D	Voltage difference V_{DDX}/$V_{DDM}$to V_{DDA}	Δ_{VDDX}	–0.1	0	0.1	V
2	D	Voltage difference V_{SSX}/V_{SSM} to V_{SSA}	Δ_{VSSX}	–0.1	0	0.1	V
3	C	Differential reference voltage[1]	V_{RH}-V_{RL}	3.13	5.0	5.5	V
4	C	ATD Clock Frequency (derived from bus clock via the prescaler bus)	f_{ATDCLk}	0.25		8.0	MHz
5	P	ATD Clock Frequency in Stop mode (internal generated temperature and voltage dependent clock, ICLK)		0.6	1	1.7	MHz
6	D	ADC conversion in stop, recovery time[2]	$t_{ATDSTPRCV}$	—	—	1.5	us
7	D	ATD Conversion Period[3] 10 bit resolution: 8 bit resolution:	N_{CONV10} N_{CONV8}	19 17		41 39	ATD clock Cycles

1. Full accuracy is not guaranteed when differential voltage is less than 4.50 V
2. When converting in Stop Mode (ICLKSTP=1) an ATD Stop Recovery time tATDSTPRCV is required to switch back to bus clock based ATDCLK when leaving Stop Mode. Do not access ATD registers during this time.
3. The minimum time assumes a sample time of 4 ATD clock cycles. The maximum time assumes a sample time of 24 ATD clock cycles and the discharge feature (SMP_DIS) enabled, which adds 2 ATD clock cycles.

A.2.2 Factors Influencing Accuracy

Source resistance, source capacitance and current injection have an influence on the accuracy of the ATD. A further factor is that PortAD pins that are configured as output drivers switching.

A.2.2.1 Port AD Output Drivers Switching

PortAD output drivers switching can adversely affect the ATD accuracy whilst converting the analog voltage on other PortAD pins because the output drivers are supplied from the VDDA/VSSA ATD supply pins. Although internal design measures are implemented to minimize the affect of output driver noise, it