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, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	64
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	6K x 8
Voltage - Supply (Vcc/Vdd)	1.72V ~ 5.5V
Data Converters	A/D 10x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	80-QFP
Supplier Device Package	80-QFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s12p128j0mqk

1.2.1 MC9S12P Family Comparison

Table 1 provides a summary of different members of the MC9S12P family and their proposed features. This information is intended to provide an understanding of the range of functionality offered by this microcontroller family.

Table 1. MC9S12P Family

Feature	MC9S12P32	MC9S12P64	MC9S12P96	MC9S12P128
CPU	CPU12-V1			
Flash memory (ECC)	32 Kbytes	64 Kbytes	96 Kbytes	128 Kbytes
Data flash (ECC)	4 Kbytes			
RAM	2 Kbytes	4 Kbytes	6 Kbytes	
MSCAN	1			
SCI	1			
SPI	1			
Timer	8 ch x 16-bit			
PWM	6 ch x 8-bit			
ADC	10 ch x 12-bit			
Frequency modulated PLL	Yes			
External oscillator (4 – 16 MHz Pierce with loop control)	Yes			
Internal 1 MHz RC oscillator	Yes			
Supply voltage	3.15 V – 5.5 V			
Execution speed	Static(1) – 32 MHz			
Package	80 QFP, 64 LQFP, 48 QFN			

1. P or D Flash erasing or programming requires a minimum bus frequency of 1MHz

1.2.2 Chip-Level Features

On-chip modules available within the family include the following features:

- S12 CPU core
- Up to 128 Kbyte on-chip flash with ECC
- 4 Kbyte data flash with ECC
- Up to 6 Kbyte on-chip SRAM
- Phase locked loop (IPLL) frequency multiplier with internal filter
- 4–16 MHz amplitude controlled Pierce oscillator
- 1 MHz internal RC oscillator
- Timer module (TIM) supporting input/output channels that provide a range of 16-bit input capture, output compare, counter, and pulse accumulator functions

1.9.1 Chip Configuration Summary

The different modes and the security state of the MCU affect the debug features (enabled or disabled).

The operating mode out of reset is determined by the state of the MODC signal during reset (see Table 1-10). The MODC bit in the MODE register shows the current operating mode and provides limited mode switching during operation. The state of the MODC signal is latched into this bit on the rising edge of $\overline{\text{RESET}}$.

Table 1-10. Chip Modes

Chip Modes	MODC
Normal single chip	1
Special single chip	0

1.9.1.1 Normal Single-Chip Mode

This mode is intended for normal device operation. The opcode from the on-chip memory is being executed after reset (requires the reset vector to be programmed correctly). The processor program is executed from internal memory.

1.9.1.2 Special Single-Chip Mode

This mode is used for debugging single-chip operation, boot-strapping, or security related operations. The background debug module BDM is active in this mode. The CPU executes a monitor program located in an on-chip ROM. BDM firmware waits for additional serial commands through the BKGD pin.

2.3.11 Ports A, B, E Reduced Drive Register (RDRIV)

Address 0x000D Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	0	0	0	RDPE	0	0	RDPB	RDPA
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 2-9. Ports ABEK Reduced Drive Register (RDRIV)

1. Read: Anytime
   Write: Anytime

Table 2-11. RDRIV Register Field Descriptions

Field	Description
4 RDPE	**Port E reduced drive**—Select reduced drive for output port This bit configures the drive strength of all associated port output pins as either full or reduced. If a pin is used as input this bit has no effect. The reduced drive function is independent of which function is being used on a particular pin. 1 Reduced drive selected (approx. 1/5 of the full drive strength) 0 Full drive strength enabled
1 RDPE	**Port B reduced drive**—Select reduced drive for output port This bit configures the drive strength of all associated port output pins as either full or reduced. If a pin is used as input this bit has no effect. The reduced drive function is independent of which function is being used on a particular pin. 1 Reduced drive selected (approx. 1/5 of the full drive strength) 0 Full drive strength enabled
0 RDPA	**Port A reduced drive**—Select reduced drive for output port This bit configures the drive strength of all associated port output pins as either full or reduced. If a pin is used as input this bit has no effect. The reduced drive function is independent of which function is being used on a particular pin. 1 Reduced drive selected (approx. 1/5 of the full drive strength) 0 Full drive strength enabled

2.3.23 Port T Routing Register (PTTRR)

Address 0x0247 Access: User read(1)

	7	6	5	4	3	2	1	0
R	0	0	PTTRR5	PTTRR4	0	0	0	PTTRR0
W								
Routing Option	—	—	PWM5	PWM4	—	—	—	PWM0
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 2-21. Port T Routing Register (PTTRR)

1. Read: Anytime
   Write: Anytime

This register configures the re-routing of PWM channels on alternative pins on Port T.

Table 2-20. Port T Routing Register Field Descriptions

Field	Description
5 PTTRR	**Port T data direction**— This register controls the routing of PWM channel 5. 1 PWM5 routed to PT5 0 PWM5 routed to PP5
4 PTTRR	**Port T data direction**— This register controls the routing of PWM channel 4. 1 PWM4 routed to PT4 0 PWM4 routed to PP4
0 PTTRR	**Port T data direction**— This register controls the routing of PWM channel 0. 1 PWM0 routed to PT0 0 PWM0 routed to PP0

Chapter 5
Background Debug Module (S12SBDMV1)

Revision History

Revision Number	Date		Summary of Changes
s12s_bdm.01.00.00	08.Feb.2006	General	First version of S12SBDMV1
s12s_bdm.01.00.02	09.Feb.2006	General	Updated register address information & Block Version
s12s_bdm.01.00.12	10.May.2006	5.3.2/5-134	Removed CLKSW bit and description
s12s_bdm.01.01.01	20.Sep.2007	General	Added conditional text for S12P family
1.02	08.Apr.2009	General	Minor text corrections following review
1.03	14.May.2009		Internal Conditional text only
1.04	30.Nov.2009		Internal Conditional text only

5.1 Introduction

This section describes the functionality of the background debug module (BDM) sub-block of the HCS12S core platform.

The background debug module (BDM) sub-block is a single-wire, background debug system implemented in on-chip hardware for minimal CPU intervention. All interfacing with the BDM is done via the BKGD pin.

The BDM has enhanced capability for maintaining synchronization between the target and host while allowing more flexibility in clock rates. This includes a sync signal to determine the communication rate and a handshake signal to indicate when an operation is complete. The system is backwards compatible to the BDM of the S12 family with the following exceptions:

- TAGGO command not supported by S12SBDM
- External instruction tagging feature is part of the DBG module
- S12SBDM register map and register content modified
- Family ID readable from BDM ROM at global address 0x3_FF0F in active BDM (value for devices with HCS12S core is 0xC2)
- Clock switch removed from BDM (CLKSW bit removed from BDMSTS register)

6.3 Memory Map and Registers

6.3.1 Module Memory Map

A summary of the registers associated with the DBG sub-block is shown in Figure 6-2. Detailed descriptions of the registers and bits are given in the subsections that follow.

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0020	DBGC1	R	ARM	0	0	BDM	DBGBRK	0	COMRV	
		W		TRIG						
0x0021	DBGSR	R	[1]TBF	0	0	0	0	SSF2	SSF1	SSF0
		W								
0x0022	DBGTCR	R	0	TSOURCE	0	0	TRCMOD		0	TALIGN
		W								
0x0023	DBGC2	R	0	0	0	0	0	0	ABCM	
		W								
0x0024	DBGTBH	R	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
		W								
0x0025	DBGTBL	R	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
		W								
0x0026	DBGCNT	R	[1] TBF	0	CNT					
		W								
0x0027	DBGSCRX	R	0	0	0	0	SC3	SC2	SC1	SC0
		W								
0x0027	DBGMFR	R	0	0	0	0	0	MC2	MC1	MC0
		W								
[2] 0x0028	DBGACTL	R	SZE	SZ	TAG	BRK	RW	RWE	NDB	COMPE
		W								
[3] 0x0028	DBGBCTL	R	SZE	SZ	TAG	BRK	RW	RWE	0	COMPE
		W								
[4] 0x0028	DBGCCTL	R	0	0	TAG	BRK	RW	RWE	0	COMPE
		W								
0x0029	DBGXAH	R	0	0	0	0	0	0	Bit 17	Bit 16
		W								
0x002A	DBGXAM	R	Bit 15	14	13	12	11	10	9	Bit 8
		W								
0x002B	DBGXAL	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x002C	DBGADH	R	Bit 15	14	13	12	11	10	9	Bit 8
		W								
0x002D	DBGADL	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								

Figure 6-2. Quick Reference to DBG Registers

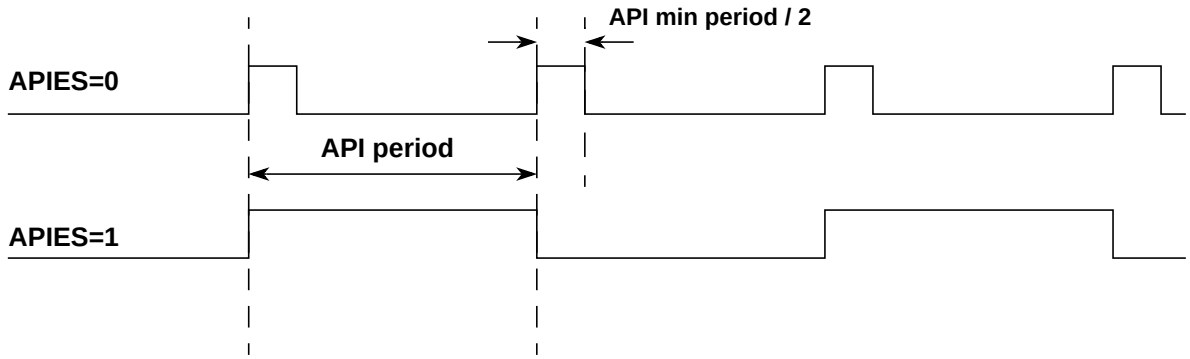
Table 7-11. CPMUCOP Field Descriptions

Field	Description
7 WCOP	**Window COP Mode Bit** — When set, a write to the CPMUARMCOP register must occur in the last 25% of the selected period. A write during the first 75% of the selected period generates a COP reset. As long as all writes occur during this window, $55 can be written as often as desired. Once $AA is written after the $55, the time-out logic restarts and the user must wait until the next window before writing to CPMUARMCOP. Table 7-12 shows the duration of this window for the seven available COP rates. 0 Normal COP operation 1 Window COP operation
6 RSBCK	**COP and RTI Stop in Active BDM Mode Bit** 0 Allows the COP and RTI to keep running in Active BDM mode. 1 Stops the COP and RTI counters whenever the part is in Active BDM mode.
5 WRTMASK	**Write Mask for WCOP and CR[2:0] Bit** — This write-only bit serves as a mask for the WCOP and CR[2:0] bits while writing the CPMUCOP register. It is intended for BDM writing the RSBCK without changing the content of WCOP and CR[2:0]. 0 Write of WCOP and CR[2:0] has an effect with this write of CPMUCOP 1 Write of WCOP and CR[2:0] has no effect with this write of CPMUCOP. (Does not count for “write once”.)
2–0 CR[2:0]	**COP Watchdog Timer Rate Select** — These bits select the COP time-out rate (see Table 7-12). Writing a nonzero value to CR[2:0] enables the COP counter and starts the time-out period. A COP counter time-out causes a System Reset. This can be avoided by periodically (before time-out) initializing the COP counter via the CPMUARMCOP register. While all of the following four conditions are true the CR[2:0], WCOP bits are ignored and the COP operates at highest time-out period (2^{24} cycles) in normal COP mode (Window COP mode disabled): 1) COP is enabled (CR[2:0] is not 000) 2) BDM mode active 3) RSBCK = 0 4) Operation in special mode

Table 7-12. COP Watchdog Rates

CR2	CR1	CR0	COPCLK Cycles to time-out (COPCLK is either IRCCLK or OSCCLK depending on the COPOSCSEL bit)
0	0	0	COP disabled
0	0	1	2^{14}
0	1	0	2^{16}
0	1	1	2^{18}
1	0	0	2^{20}
1	0	1	2^{22}
1	1	0	2^{23}
1	1	1	2^{24}

Figure 7-19. Waveform selected on API_EXTCLK pin (APIEA=1, APIFE=1)



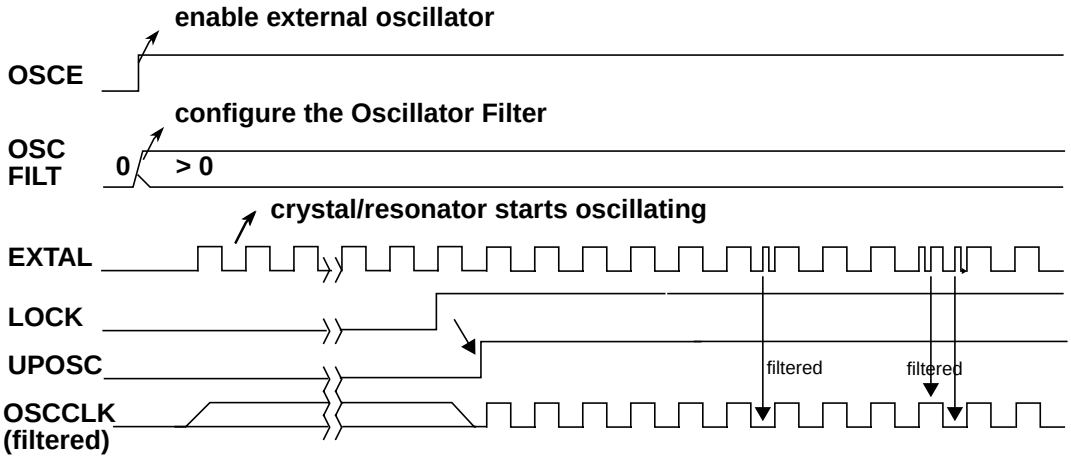
7.4.5.2 The Adaptive Oscillator Filter

A spike in the oscillator clock can disturb the function of the modules driven by this clock.

The adaptive Oscillator Filter includes two features:

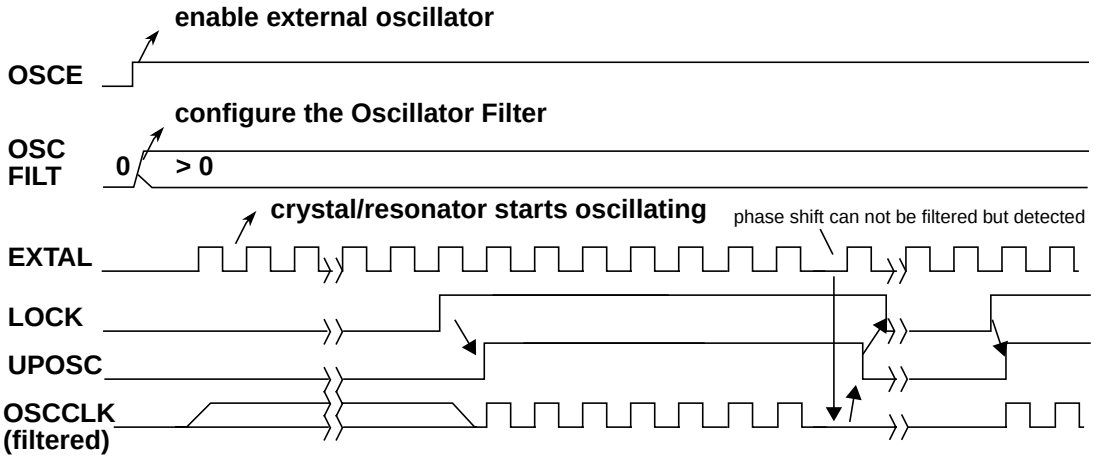
1. Filter noise (spikes) from the incoming external oscillator clock. The filter function is illustrated in Figure 7-35.

Figure 7-35. Noise filtered by the Adaptive Oscillator Filter



2. Detect severe noise disturbances on the external oscillator clock, which can not be filtered and indicate the critical situation to the software by clearing the UPOSC and LOCK status bit and setting the OSCIF and LOCKIF flag. An example for the detection of critical noise is illustrated in Figure 7-36.

Figure 7-36. Critical noise detected by the Adaptive Oscillator Filter



NOTE

If the LOCK bit is clear due to severe noise disturbance on the external oscillator clock the PLLCLK is derived from the VCO clock (with its actual frequency) divided by four (see also Section 7.3.2.3, “S12CPMU Post Divider Register (CPMUPOSTDIV))

Module Base + 0x0006 — Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	TXE2	TXE1	TXE0
W								
Reset:	0	0	0	0	0	1	1	1

= Unimplemented

Figure 8-10. MSCAN Transmitter Flag Register (CANTFLG)

1. Read: Anytime
   Write: Anytime when not in initialization mode; write of 1 clears flag, write of 0 is ignored

NOTE

The CANTFLG register is held in the reset state when the initialization mode is active (INITRQ = 1 and INITAK = 1). This register is writable when not in initialization mode (INITRQ = 0 and INITAK = 0).

Table 8-13. CANTFLG Register Field Descriptions

Field	Description
2-0 TXE[2:0]	**Transmitter Buffer Empty** — This flag indicates that the associated transmit message buffer is empty, and thus not scheduled for transmission. The CPU must clear the flag after a message is set up in the transmit buffer and is due for transmission. The MSCAN sets the flag after the message is sent successfully. The flag is also set by the MSCAN when the transmission request is successfully aborted due to a pending abort request (see Section 8.3.2.9, "MSCAN Transmitter Message Abort Request Register (CANTARQ)"). If not masked, a transmit interrupt is pending while this flag is set. Clearing a TXEx flag also clears the corresponding ABTAKx (see Section 8.3.2.10, "MSCAN Transmitter Message Abort Acknowledge Register (CANTAAK)"). When a TXEx flag is set, the corresponding ABTRQx bit is cleared (see Section 8.3.2.9, "MSCAN Transmitter Message Abort Request Register (CANTARQ)"). When listen-mode is active (see Section 8.3.2.2, "MSCAN Control Register 1 (CANCTL1)") the TXEx flags cannot be cleared and no transmission is started. Read and write accesses to the transmit buffer will be blocked, if the corresponding TXEx bit is cleared (TXEx = 0) and the buffer is scheduled for transmission. 0 The associated message buffer is full (loaded with a message due for transmission) 1 The associated message buffer is empty (not scheduled)

8.3.2.8 MSCAN Transmitter Interrupt Enable Register (CANTIER)

This register contains the interrupt enable bits for the transmit buffer empty interrupt flags.

Module Base + 0x0007 — Access: User read/write(1)

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	TXEIE2	TXEIE1	TXEIE0
W								
Reset:	0	0	0	0	0	0	0	0

= Unimplemented

Figure 8-11. MSCAN Transmitter Interrupt Enable Register (CANTIER)

8.4.4.2 Special System Operating Modes

The MSCAN module behaves as described within this specification in all special system operating modes. Write restrictions which exist on specific registers in normal modes are lifted for test purposes in special modes.

8.4.4.3 Emulation Modes

In all emulation modes, the MSCAN module behaves just like in normal system operating modes as described within this specification.

8.4.4.4 Listen-Only Mode

In an optional CAN bus monitoring mode (listen-only), the CAN node is able to receive valid data frames and valid remote frames, but it sends only "recessive" bits on the CAN bus. In addition, it cannot start a transmission.

If the MAC sub-layer is required to send a "dominant" bit (ACK bit, overload flag, or active error flag), the bit is rerouted internally so that the MAC sub-layer monitors this "dominant" bit, although the CAN bus may remain in recessive state externally.

8.4.4.5 MSCAN Initialization Mode

The MSCAN enters initialization mode when it is enabled (CANE=1).

When entering initialization mode during operation, any on-going transmission or reception is immediately aborted and synchronization to the CAN bus is lost, potentially causing CAN protocol violations. To protect the CAN bus system from fatal consequences of violations, the MSCAN immediately drives TXCAN into a recessive state.

NOTE

The user is responsible for ensuring that the MSCAN is not active when initialization mode is entered. The recommended procedure is to bring the MSCAN into sleep mode (SLPRQ = 1 and SLPAK = 1) before setting the INITRQ bit in the CANCTL0 register. Otherwise, the abort of an on-going message can cause an error condition and can impact other CAN bus devices.

In initialization mode, the MSCAN is stopped. However, interface registers remain accessible. This mode is used to reset the CANCTL0, CANRFLG, CANRIER, CANTFLG, CANTIER, CANTARQ, CANTAAK, and CANTBSEL registers to their default values. In addition, the MSCAN enables the configuration of the CANBTR0, CANBTR1 bit timing registers; CANIDAC; and the CANIDAR, CANIDMR message filters. See Section 8.3.2.1, "MSCAN Control Register 0 (CANCTL0)," for a detailed description of the initialization mode.

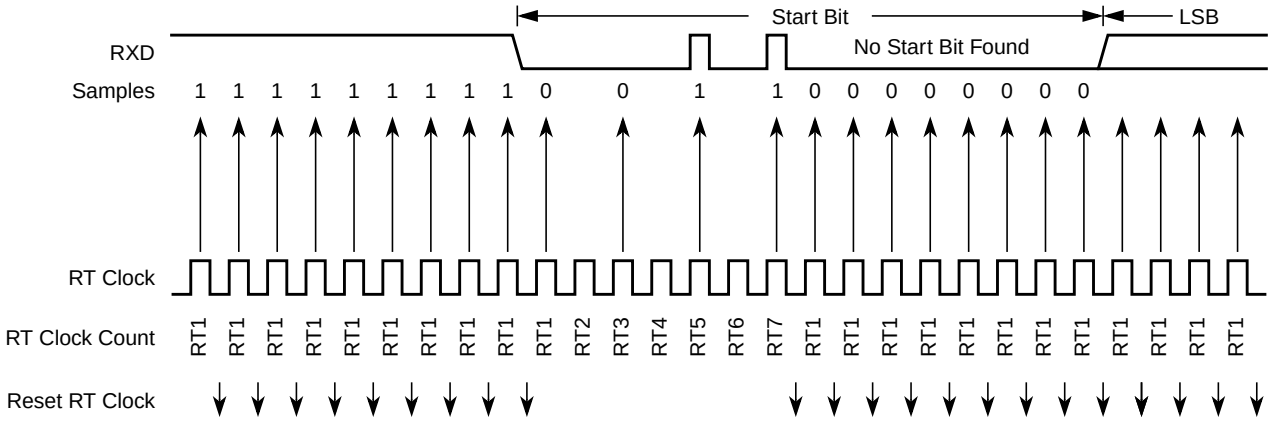


Figure 11-26. Start Bit Search Example 5

In Figure 11-27, a noise burst makes the majority of data samples RT8, RT9, and RT10 high. This sets the noise flag but does not reset the RT clock. In start bits only, the RT8, RT9, and RT10 data samples are ignored.

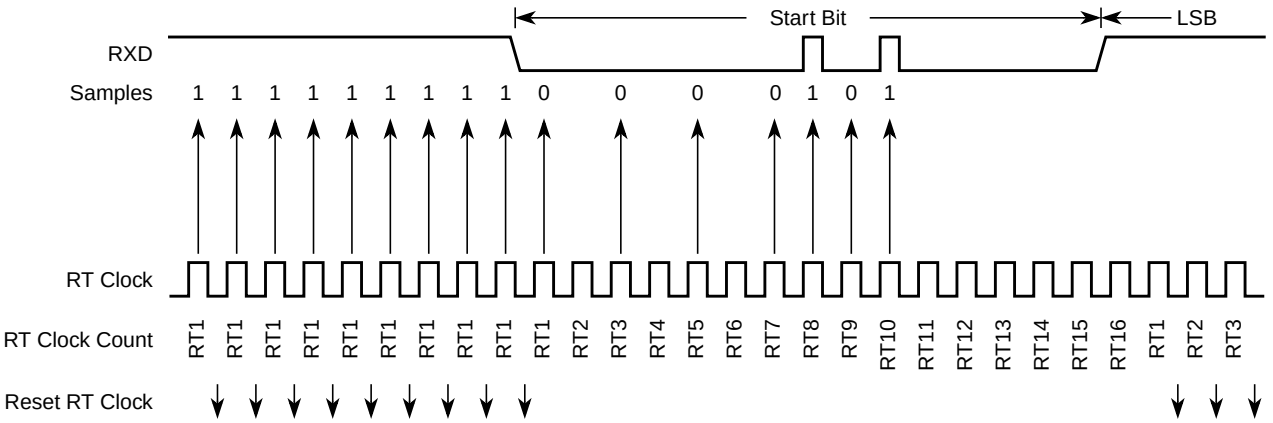


Figure 11-27. Start Bit Search Example 6

11.4.6.4 Framing Errors

If the data recovery logic does not detect a logic 1 where the stop bit should be in an incoming frame, it sets the framing error flag, FE, in SCI status register 1 (SCISR1). A break character also sets the FE flag because a break character has no stop bit. The FE flag is set at the same time that the RDRF flag is set.

11.4.6.5 Baud Rate Tolerance

A transmitting device may be operating at a baud rate below or above the receiver baud rate. Accumulated bit time misalignment can cause one of the three stop bit data samples (RT8, RT9, and RT10) to fall outside the actual stop bit. A noise error will occur if the RT8, RT9, and RT10 samples are not all the same logical values. A framing error will occur if the receiver clock is misaligned in such a way that the majority of the RT8, RT9, and RT10 stop bit samples are a logic zero.

phrase containing the Flash security byte during the reset sequence, all bits in the FSEC register will be set to leave the Flash module in a secured state with backdoor key access disabled.

Table 13-8. FSEC Field Descriptions

Field	Description
7–6 KEYEN[1:0]	**Backdoor Key Security Enable Bits** — The KEYEN[1:0] bits define the enabling of backdoor key access to the Flash module as shown in Table 13-9.
5–2 RNV[5:2}	**Reserved Nonvolatile Bits** — The RNV bits should remain in the erased state for future enhancements.
1–0 SEC[1:0]	**Flash Security Bits** — The SEC[1:0] bits define the security state of the MCU as shown in Table 13-10. If the Flash module is unsecured using backdoor key access, the SEC bits are forced to 10.

Table 13-9. Flash KEYEN States

KEYEN[1:0]	Status of Backdoor Key Access
00	DISABLED
01	DISABLED(1)
10	ENABLED
11	DISABLED

1. Preferred KEYEN state to disable backdoor key access.

Table 13-10. Flash Security States

SEC[1:0]	Status of Security
00	SECURED
01	SECURED(1)
10	UNSECURED
11	SECURED

1. Preferred SEC state to set MCU to secured state.

The security function in the Flash module is described in Section 13.5.

13.3.2.3 Flash CCOB Index Register (FCCOBIX)

The FCCOBIX register is used to index the FCCOB register for Flash memory operations.

Offset Module Base + 0x0002

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	CCOBIX[2:0]		
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 13-7. FCCOB Index Register (FCCOBIX)

CCOBIX bits are readable and writable while remaining bits read 0 and are not writable.

Table 13-14. FSTAT Field Descriptions (continued)

Field	Description
3 MGBUSY	**Memory Controller Busy Flag** — The MGBUSY flag reflects the active state of the Memory Controller. 0 Memory Controller is idle 1 Memory Controller is busy executing a Flash command (CCIF = 0)
2 RSVD	**Reserved Bit** — This bit is reserved and always reads 0.
1–0 MGSTAT[1:0]	**Memory Controller Command Completion Status Flag** — One or more MGSTAT flag bits are set if an error is detected during execution of a Flash command or during the Flash reset sequence. See Section 13.4.5, “Flash Command Description,” and Section 13.6, “Initialization” for details.

13.3.2.8 Flash Error Status Register (FERSTAT)

The FERSTAT register reflects the error status of internal Flash operations.

Offset: Module Base + 0x0007

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

= Unimplemented or Reserved

Figure 13-12. Flash Error Status Register (FERSTAT)

All flags in the FERSTAT register are readable and only writable to clear the flag.

Table 13-15. FERSTAT Field Descriptions

Field	Description
1 DFDIF	**Double Bit Fault Detect Interrupt Flag** — The setting of the DFDIF flag indicates that a double bit fault was detected in the stored parity and data bits during a Flash array read operation or that a Flash array read operation was attempted on a Flash block that was under a Flash command operation.(1) The DFDIF flag is cleared by writing a 1 to DFDIF. Writing a 0 to DFDIF has no effect on DFDIF. 0 No double bit fault detected 1 Double bit fault detected or an invalid Flash array read operation attempted
0 SFDIF	**Single Bit Fault Detect Interrupt Flag** — With the IGNSF bit in the FCNFG register clear, the SFDIF flag indicates that a single bit fault was detected in the stored parity and data bits during a Flash array read operation or that a Flash array read operation was attempted on a Flash block that was under a Flash command operation.[1] The SFDIF flag is cleared by writing a 1 to SFDIF. Writing a 0 to SFDIF has no effect on SFDIF. 0 No single bit fault detected 1 Single bit fault detected and corrected or an invalid Flash array read operation attempted

1. The single bit fault and double bit fault flags are mutually exclusive for parity errors (an ECC fault occurrence can be either single fault or double fault but never both). A simultaneous access collision (read attempted while command running) is indicated when both SFDIF and DFDIF flags are high.

13.3.2.9 P-Flash Protection Register (FPROT)

The FPROT register defines which P-Flash sectors are protected against program and erase operations.

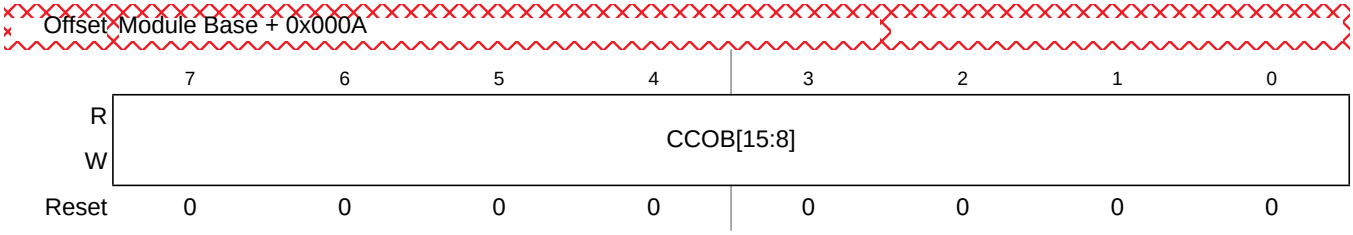


Figure 13-16. Flash Common Command Object High Register (FCCOBHI)

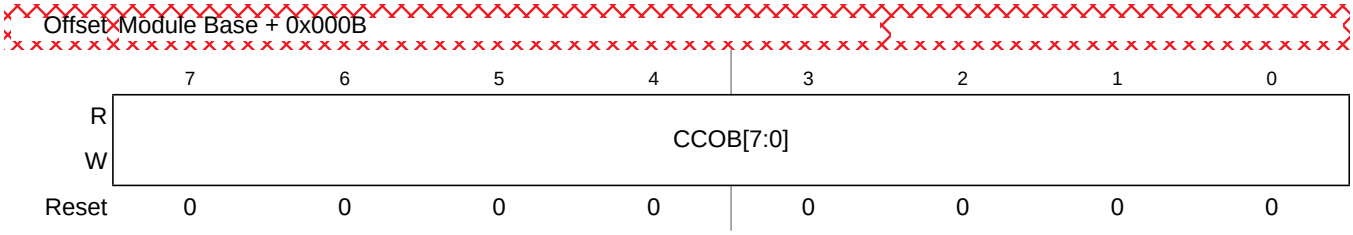


Figure 13-17. Flash Common Command Object Low Register (FCCOBLO)

13.3.2.11.1 FCCOB - NVM Command Mode

NVM command mode uses the indexed FCCOB register to provide a command code and its relevant parameters to the Memory Controller. The user first sets up all required FCCOB fields and then initiates the command's execution by writing a 1 to the CCIF bit in the FSTAT register (a 1 written by the user clears the CCIF command completion flag to 0). When the user clears the CCIF bit in the FSTAT register all FCCOB parameter fields are locked and cannot be changed by the user until the command completes (as evidenced by the Memory Controller returning CCIF to 1). Some commands return information to the FCCOB register array.

The generic format for the FCCOB parameter fields in NVM command mode is shown in Table 13-23. The return values are available for reading after the CCIF flag in the FSTAT register has been returned to 1 by the Memory Controller. Writes to the unimplemented parameter fields (CCOBIX = 110 and CCOBIX = 111) are ignored with reads from these fields returning 0x0000.

Table 13-23 shows the generic Flash command format. The high byte of the first word in the CCOB array contains the command code, followed by the parameters for this specific Flash command. For details on the FCCOB settings required by each command, see the Flash command descriptions in Section 13.4.5.

Table 13-23. FCCOB - NVM Command Mode (Typical Usage)

CCOBIX[2:0]	Byte	FCCOB Parameter Fields (NVM Command Mode)
000	HI	FCMD[7:0] defining Flash command
	LO	6'h0, Global address [17:16]
001	HI	Global address [15:8]
	LO	Global address [7:0]
010	HI	Data 0 [15:8]
	LO	Data 0 [7:0]

14.3.2.18 Output Compare Pin Disconnect Register(OCPD)

Module Base + 0x002C

	7	6	5	4	3	2	1	0
R W	OCPD7	OCPD6	OCPD5	OCPD4	OCPD3	OCPD2	OCPD1	OCPD0
Reset	0	0	0	0	0	0	0	0

Figure 14-28. Ouput Compare Pin Disconnect Register (OCPD)

Read: Anytime

Write: Anytime

All bits reset to zero.

Table 14-21. OCPD Field Description

Field	Description
OCPD[7:0}	**Output Compare Pin Disconnect Bits** 0 Enables the timer channel port. Ouptut Compare action will occur on the channel pin. These bits do not affect the input capture or pulse accumulator functions 1 Disables the timer channel port. Output Compare action will not occur on the channel pin, but the output compare flag still become set .

14.3.2.19 Precision Timer Prescaler Select Register (PTPSR)

Module Base + 0x002E

	7	6	5	4	3	2	1	0
R W	PTPS7	PTPS6	PTPS5	PTPS4	PTPS3	PTPS2	PTPS1	PTPS0
Reset	0	0	0	0	0	0	0	0

Figure 14-29. Precision Timer Prescaler Select Register (PTPSR)

Read: Anytime

Write: Anytime

All bits reset to zero.

The pulse accumulator counter register reflect the number of pulses from the divided-by-64 clock since the last reset.

NOTE

The timer prescaler generates the divided-by-64 clock. If the timer is not active, there is no divided-by-64 clock.

14.5 Resets

The reset state of each individual bit is listed within Section 14.3, “Memory Map and Register Definition” which details the registers and their bit fields.

14.6 Interrupts

This section describes interrupts originated by the TIM16B8CV2 block. Table 14-24 lists the interrupts generated by the TIM16B8CV2 to communicate with the MCU.

Table 14-24. TIM16B8CV1 Interrupts

Interrupt	Offset (1)	Vector[1]	Priority[1]	Source	Description
C[7:0]F	—	—	—	Timer Channel 7–0	Active high timer channel interrupts 7–0
PAOVI	—	—	—	Pulse Accumulator Input	Active high pulse accumulator input interrupt
PAOVF	—	—	—	Pulse Accumulator Overflow	Pulse accumulator overflow interrupt
TOF	—	—	—	Timer Overflow	Timer Overflow interrupt

1. Chip Dependent.

The TIM16B8CV2 uses a total of 11 interrupt vectors. The interrupt vector offsets and interrupt numbers are chip dependent.

14.6.1 Channel [7:0] Interrupt (C[7:0]F)

This active high outputs will be asserted by the module to request a timer channel 7 – 0 interrupt to be serviced by the system controller.

14.6.2 Pulse Accumulator Input Interrupt (PAOVI)

This active high output will be asserted by the module to request a timer pulse accumulator input interrupt to be serviced by the system controller.

14.6.3 Pulse Accumulator Overflow Interrupt (PAOVF)

This active high output will be asserted by the module to request a timer pulse accumulator overflow interrupt to be serviced by the system controller.

A.1.10 Supply Currents

This section describes the current consumption characteristics of the device as well as the conditions for the measurements.

A.1.10.1 Measurement Conditions

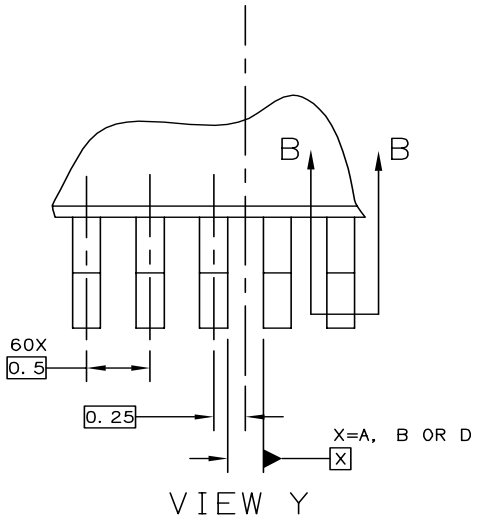
Run current is measured on VDDR pin. It does not include the current to drive external loads. Unless otherwise noted the currents are measured in special single chip mode and the CPU code is executed from RAM. For Run and Wait current measurements PLL is on and the reference clock is the IRC1M trimmed to 1MHz. The bus frequency is 32MHz and the CPU frequency is 64MHz. Table A-8., Table A-9. and Table A-10. show the configuration of the CPMU module and the peripherals for Run, Wait and Stop current measurement.

Table A-8. CPMU Configuration for Pseudo Stop Current Measurement

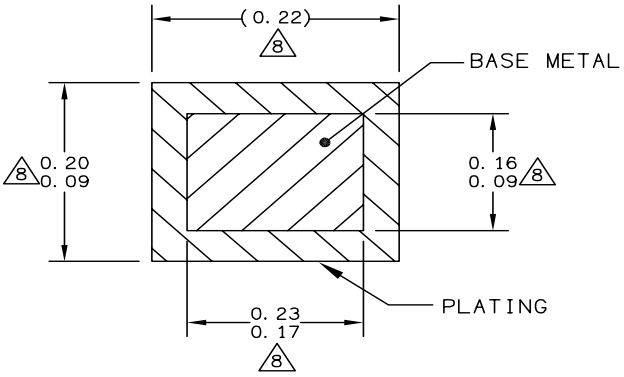
CPMU REGISTER	Bit settings/Conditions
CPMUCLKS	PLLSEL=0, PSTP=1, PRE=PCE=RTIOSCSEL=COPOSCSEL=1
CPMUOSC	OSCE=1, External Square wave on EXTAL f_{EXTAL}=16MHz, V_{IH}= 1.8V, V_{IL}=0V
CPMURTI	RTDEC=0, RTR[6:4]=111, RTR[3:0]=1111;
CPMUCOP	WCOP=1, CR[2:0]=111

Table A-9. CPUM Configuration for Run/Wait and Full Stop Current Measurement

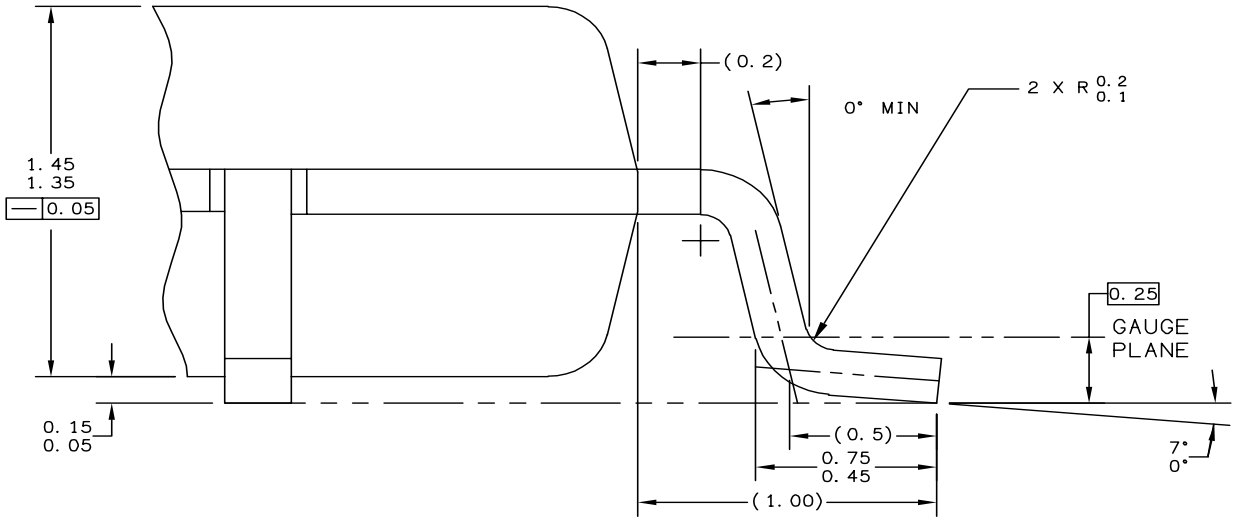
CPMU REGISTER	Bit settings/Conditions
CPMUSYNR	VCOFRQ[1:0]=01,SYNDIV[5:0] = 32
CPMUPOSTDIV	POSTDIV[4:0]=0,
CPMUCLKS	PLLSEL=1
CPMUOSC	OSCE=0, Reference clock for PLL is f_{ref}=f_{irc1m} trimmed to 1MHz
API settings for STOP current measurement	
CPMUAPICTL	APIEA=0, APIFE=1, APIE=0
CPMUAPITR	trimmed to 10Khz
CPMUAPIRH/RL	set to $FFFF



VIEW Y



SECTION B–B



VIEW AA

© FREESCALE SEMICONDUCTOR, INC. ALL RIGHTS RESERVED.	**MECHANICAL OUTLINE**	PRINT VERSION NOT TO SCALE
TITLE: 64LD LQFP, 10 X 10 X 1.4 PKG, 0.5 PITCH, CASE OUTLINE	DOCUMENT NO: 98ASS23234W	REV: E
	CASE NUMBER: 840F–02	11 AUG 2006
	STANDARD: JEDEC MS–026 BCD	

0x00A0-0x00C7 Pulse Width Modulator 6-Channels (PWM) Map

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00A9	PWMSCLB	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00AA	PWMSCNTA	R	0	0	0	0	0	0	0	0
		W								
0x00AB	PWMSCNTB	R	0	0	0	0	0	0	0	0
		W								
0x00AC	PWMCNT0	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00AD	PWMCNT1	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00AE	PWMCNT2	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00AF	PWMCNT3	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00B0	PWMCNT4	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00B1	PWMCNT5	R	Bit 7	6	5	4	3	2	1	Bit 0
		W	0	0	0	0	0	0	0	0
0x00B2	PWMPER0	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00B3	PWMPER1	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00B4	PWMPER2	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00B5	PWMPER3	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00B6	PWMPER4	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00B7	PWMPER5	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00B8	PWMDTY0	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00B9	PWMDTY1	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00BA	PWMDTY2	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00BB	PWMDTY3	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00BC	PWMDTY4	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00BD	PWMDTY5	R	Bit 7	6	5	4	3	2	1	Bit 0
		W								
0x00BE	PWMSDN	R	PWMIF	PWMIE	0	PWMLVL	0	PWM5IN	PWM5INL	PWM5 ENA
		W			PWRSTRT					
0x00BF-0x00C7	Reserved	R	0	0	0	0	0	0	0	0
		W								