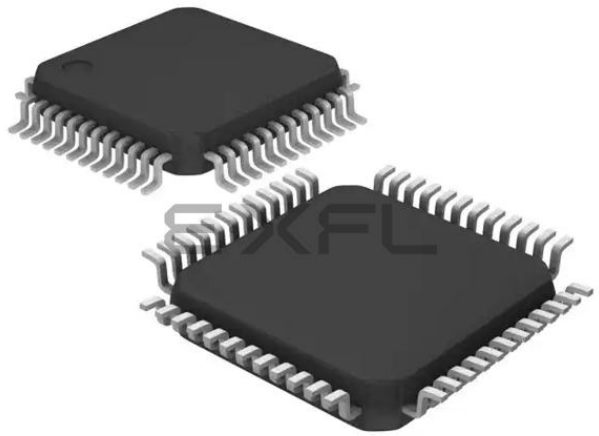

Welcome to E-XFL.COM

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	12V1
Core Size	16-Bit
Speed	25MHz
Connectivity	CANbus, IrDA, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	40
Program Memory Size	240KB (240K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	11K x 8
Voltage - Supply (Vcc/Vdd)	3.13V ~ 5.5V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s12g240f0clfr

Table 1-30. 64-Pin LQFP Pinout for S12GA192 and S12GA240

	Function <----lowest-----PRIORITY-----highest---->					Power Supply	Internal Pull Resistor	
Package Pin	**Pin**	**2nd Func.**	**3rd Func.**	**4th Func**	**5th Func**		**CTRL**	**Reset State**
57	PS5	MOSI0	—	—	—	V_{DDX}	PERS/PPSS	Up
58	PS6	SCK0	—	—	—	V_{DDX}	PERS/PPSS	Up
59	PS7	API_EXTCLK	ECLK	$\overline{SS0}$	—	V_{DDX}	PERS/PPSS	Up
60	PM0	RXCAN	—	—	—	V_{DDX}	PERM/PPSM	Disabled
61	PM1	TXCAN	—	—	—	V_{DDX}	PERM/PPSM	Disabled
62	PM2	RXD2	—	—	—	V_{DDX}	PERM/PPSM	Disabled
63	PM3	TXD2	—	—	—	V_{DDX}	PERM/PPSM	Disabled
64	PJ7	KWJ7	$\overline{SS2}$	—	—	V_{DDX}	PERJ/PPSJ	Up

[1] The regular I/O characteristics (see Section A.2, “I/O Characteristics”) apply if the EXTAL/XTAL function is disabled

2.3.9 Pins PM3-0

Table 2-13. Port M Pins PM3-0

PM3	• 64/100 LQFP: The SCI2 TXD signal is mapped to this pin when used with the SCI function. If the SCI2 TXD signal is enabled the I/O state will depend on the SCI2 configuration. • Signal priority: 64/100 LQFP: TXD2 > GPO
PM2	• 64/100 LQFP: The SCI2 RXD signal is mapped to this pin when used with the SCI function. If the SCI2 RXD signal is enabled the I/O state will be forced to be input. • Signal priority: 64/100 LQFP: RXD2 > GPO
PM1	• Except 20 TSSOP: The TXCAN signal is mapped to this pin when used with the CAN function. The enabled CAN forces the I/O state to be an output. • 32 LQFP: The SCI1 TXD signal is mapped to this pin when used with the SCI function. If the SCI1 TXD signal is enabled the I/O state will depend on the SCI1 configuration. • 48 LQFP: The SCI2 TXD signal is mapped to this pin when used with the SCI function. If the SCI2 TXD signal is enabled the I/O state will depend on the SCI2 configuration. • Signal priority: 32 LQFP: TXCAN > TXD1 > GPO 48 LQFP: TXCAN > TXD2 > GPO 64/100 LQFP: TXCAN > GPO
PM0	• Except 20 TSSOP: The RXCAN signal is mapped to this pin when used with the CAN function. The enabled CAN forces the I/O state to be an input. If CAN is active the selection of a pulldown device on the RXCAN input has no effect. • 32 LQFP: The SCI1 RXD signal is mapped to this pin when used with the SCI function. The enabled SCI1 RXD signal forces the I/O state to an input. • 48 LQFP: The SCI2 RXD signal is mapped to this pin when used with the SCI function. The enabled SCI2 RXD signal forces the I/O state to an input. • Signal priority: 32 LQFP: RXCAN > RXD1 > GPO 48 LQFP: RXCAN > RXD2 > GPO 64/100 LQFP: RXCAN > GPO

2.3.10 Pins PP7-0

Table 2-14. Port P Pins PP7-0

PP7-PP6	• 64/100 LQFP: The PWM channels 7 and 6 signal are mapped to these pins when used with the PWM function. The enabled PWM channel forces the I/O state to be an output. • 64/100 LQFP: Pin interrupts can be generated if enabled in input or output mode. • Signal priority: 64/100 LQFP: PWM > GPO
PP5-PP4	• 48/64/100 LQFP: The PWM channels 5 and 4 signal are mapped to these pins when used with the PWM function. The enabled PWM channel forces the I/O state to be an output. • 48/64/100 LQFP: Pin interrupts can be generated if enabled in input or output mode. • Signal priority: 48/64/100 LQFP: PWM > GPO

Table 2-21. Block Register Map (G3) (continued)

Global Address Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x026D PPSJ	R	0	0	0	0	PPSJ3	PPSJ2	PPSJ1	PPSJ0
	W								
0x026E PIEJ	R	0	0	0	0	PIEJ3	PIEJ2	PIEJ1	PIEJ0
	W								
0x026F PIFJ	R	0	0	0	0	PIFJ3	PIFJ2	PIFJ1	PIFJ0
	W								
0x0270 PT0AD	R	0	0	0	0	PT0AD3	PT0AD2	PT0AD1	PT0AD0
	W								
0x0271 PT1AD	R	PT1AD7	PT1AD6	PT1AD5	PT1AD4	PT1AD3	PT1AD2	PT1AD1	PT1AD0
	W								
0x0272 PTI0AD	R	0	0	0	0	PTI0AD3	PTI0AD2	PTI0AD1	PTI0AD0
	W								
0x0273 PTI1AD	R	PTI1AD7	PTI1AD6	PTI1AD5	PTI1AD4	PTI1AD3	PTI1AD2	PTI1AD1	PTI1AD0
	W								
0x0274 DDR0AD	R	0	0	0	0	DDR0AD3	DDR0AD2	DDR0AD1	DDR0AD0
	W								
0x0275 DDR1AD	R	DDR1AD7	DDR1AD6	DDR1AD5	DDR1AD4	DDR1AD3	DDR1AD2	DDR1AD1	DDR1AD0
	W								
0x0276 Reserved	R	0	0	0	0	0	0	0	0
	W								
0x0277 Reserved	R	0	0	0	0	0	0	0	0
	W								
0x0278 PER0AD	R	0	0	0	0	PER0AD3	PER0AD2	PER0AD1	PER0AD0
	W								
0x0279 PER1AD	R	PER1AD7	PER1AD6	PER1AD5	PER1AD4	PER1AD3	PER1AD2	PER1AD1	PER1AD0
	W								
0x027A PPS0AD	R	0	0	0	0	PPS0AD3	PPS0AD2	PPS0AD1	PPS0AD0
	W								
0x027B PPS1AD	R	PPS1AD7	PPS1AD6	PPS1AD5	PPS1AD4	PPS1AD3	PPS1AD2	PPS1AD1	PPS1AD0
	W								

= Unimplemented or Reserved

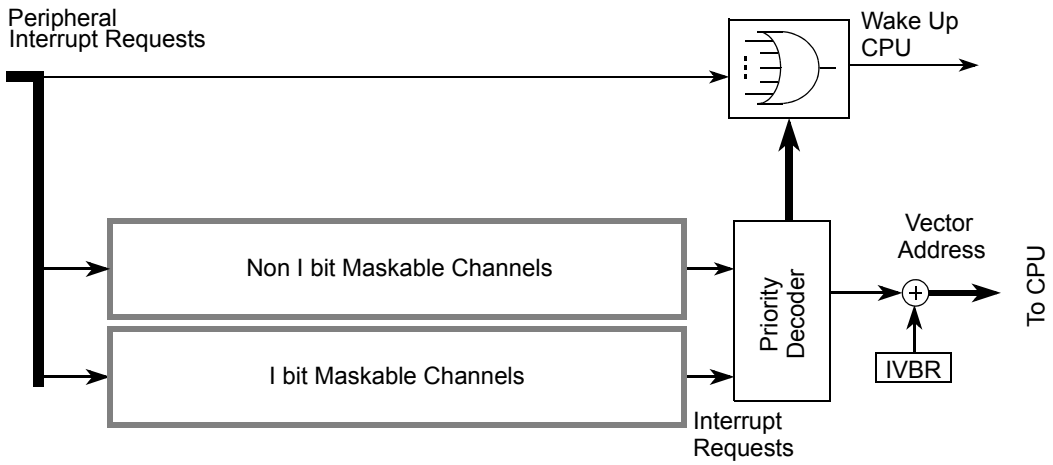

Figure 6-1. INT Block Diagram

6.2 External Signal Description

The INT module has no external signals.

6.3 Memory Map and Register Definition

This section provides a detailed description of all registers accessible in the INT module.

6.3.1 Register Descriptions

This section describes in address order all the INT registers and their individual bits.

6.3.1.1 Interrupt Vector Base Register (IVBR)

Address: 0x0120

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

Figure 6-2. Interrupt Vector Base Register (IVBR)

Read: Anytime

Write: Anytime

11.1.2 Modes of Operation

11.1.2.1 Conversion Modes

There is software programmable selection between performing **single** or **continuous conversion** on a **single channel** or **multiple channels**.

11.1.2.2 MCU Operating Modes

- **Stop Mode**
  Entering Stop Mode aborts any conversion sequence in progress and if a sequence was aborted restarts it after exiting stop mode. This has the same effect/consequences as starting a conversion sequence with write to ATDCTL5. So after exiting from stop mode with a previously aborted sequence all flags are cleared etc.
- **Wait Mode**
  ADC10B8C behaves same in Run and Wait Mode. For reduced power consumption continuous conversions should be aborted before entering Wait mode.
- **Freeze Mode**
  In Freeze Mode the ADC10B8C will either continue or finish or stop converting according to the FRZ1 and FRZ0 bits. This is useful for debugging and emulation.

11.3.2.4 ATD Control Register 3 (ATDCTL3)

Writes to this register will abort current conversion sequence.

Module Base + 0x0003

	7	6	5	4	3	2	1	0
R W	DJM	S8C	S4C	S2C	S1C	FIFO	FRZ1	FRZ0
Reset	0	0	1	0	0	0	0	0

= Unimplemented or Reserved

Figure 11-6. ATD Control Register 3 (ATDCTL3)

Read: Anytime

Write: Anytime

Table 11-8. ATDCTL3 Field Descriptions

Field	Description
7 DJM	**Result Register Data Justification** — Result data format is always unsigned. This bit controls justification of conversion data in the result registers. 0 Left justified data in the result registers. 1 Right justified data in the result registers. Table 11-9 gives example ATD results for an input signal range between 0 and 5.12 Volts.
6–3 S8C, S4C, S2C, S1C	**Conversion Sequence Length** — These bits control the number of conversions per sequence. Table 11-10 shows all combinations. At reset, S4C is set to 1 (sequence length is 4). This is to maintain software continuity to HC12 family.
2 FIFO	**Result Register FIFO Mode** — If this bit is zero (non-FIFO mode), the A/D conversion results map into the result registers based on the conversion sequence; the result of the first conversion appears in the first result register (ATDDR0), the second result in the second result register (ATDDR1), and so on. If this bit is one (FIFO mode) the conversion counter is not reset at the beginning or end of a conversion sequence; sequential conversion results are placed in consecutive result registers. In a continuously scanning conversion sequence, the result register counter will wrap around when it reaches the end of the result register file. The conversion counter value (CC3-0 in ATDSTAT0) can be used to determine where in the result register file, the current conversion result will be placed. Aborting a conversion or starting a new conversion clears the conversion counter even if FIFO=1. So the first result of a new conversion sequence, started by writing to ATDCTL5, will always be place in the first result register (ATDDDR0). Intended usage of FIFO mode is continuos conversion (SCAN=1) or triggered conversion (ETRIG=1). Which result registers hold valid data can be tracked using the conversion complete flags. Fast flag clear mode may be useful in a particular application to track valid data. If this bit is one, automatic compare of result registers is always disabled, that is ADC10B8C will behave as if ACMPIE and all CPME[*n*] were zero. 0 Conversion results are placed in the corresponding result register up to the selected sequence length. 1 Conversion results are placed in consecutive result registers (wrap around at end).
1–0 FRZ[1:0]	**Background Debug Freeze Enable** — When debugging an application, it is useful in many cases to have the ATD pause when a breakpoint (Freeze Mode) is encountered. These 2 bits determine how the ATD will respond to a breakpoint as shown in Table 11-11. Leakage onto the storage node and comparator reference capacitors may compromise the accuracy of an immediately frozen conversion depending on the length of the freeze period.

12.3.2.9 ATD Status Register 2 (ATDSTAT2)

This read-only register contains the Conversion Complete Flags CCF[7:0].

Module Base + 0x000A

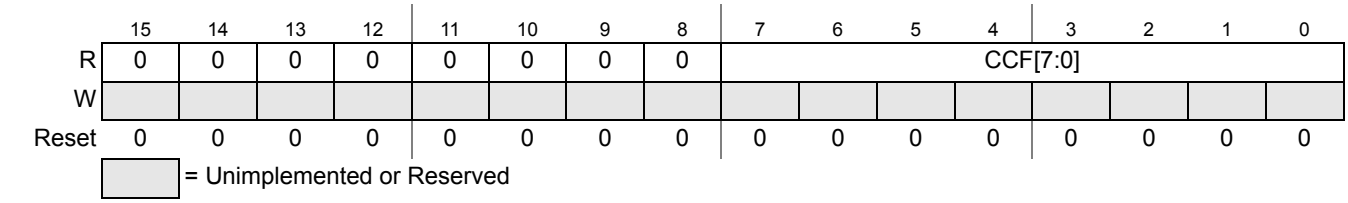

Figure 12-11. ATD Status Register 2 (ATDSTAT2)

Read: Anytime

Write: Anytime (for details see Table 12-18 below)

Table 12-18. ATDSTAT2 Field Descriptions

Field	Description
7–0 CCF[7:0]	**Conversion Complete Flag *n* (*n*= 7, 6, 5, 4, 3, 2, 1, 0) (*n conversion number, NOT channel number!)*—** A conversion complete flag is set at the end of each conversion in a sequence. The flags are associated with the conversion position in a sequence (and also the result register number). Therefore in non-fifo mode, CCF[4] is set when the fifth conversion in a sequence is complete and the result is available in result register ATDDR4; CCF[5] is set when the sixth conversion in a sequence is complete and the result is available in ATDDR5, and so forth. If automatic compare of conversion results is enabled (CMPE[*n*]=1 in ATDCMPE), the conversion complete flag is only set if comparison with ATDDR*n* is true. If ACMPIE=1 a compare interrupt will be requested. In this case, as the ATDDR*n* result register is used to hold the compare value, the result will not be stored there at the end of the conversion but is lost. A flag CCF[*n*] is cleared when one of the following occurs: A) Write to ATDCTL5 (a new conversion sequence is started) B) If AFFC=0, write “1” to CCF[*n*] C) If AFFC=1 and CMPE[*n*]=0, read of result register ATDDR*n* D) If AFFC=1 and CMPE[*n*]=1, write to result register ATDDR*n* In case of a concurrent set and clear on CCF[*n*]: The clearing by method A) will overwrite the set. The clearing by methods B) or C) or D) will be overwritten by the set. 0 Conversion number *n* not completed or successfully compared 1 If (CMPE[*n*]=0): Conversion number *n* has completed. Result is ready in ATDDR*n*. If (CMPE[*n*]=1): Compare for conversion result number *n* with compare value in ATDDR*n*, using compare operator CMPGT[*n*] is true. (No result available in ATDDR*n*)

edge or level sensitive with polarity control. Table 13-23 gives a brief description of the different combinations of control bits and their effect on the external trigger function.

In order to avoid maybe false trigger events please enable the external digital input via ATDDIEN register first and in the following enable the external trigger mode by bit ETRIGE.

Table 13-23. External Trigger Control Bits

ETRIGLE	ETRIGP	ETRIGE	SCAN	Description
X	X	0	0	Ignores external trigger. Performs one conversion sequence and stops.
X	X	0	1	Ignores external trigger. Performs continuous conversion sequences.
0	0	1	X	Trigger falling edge sensitive. Performs one conversion sequence per trigger.
0	1	1	X	Trigger rising edge sensitive. Performs one conversion sequence per trigger.
1	0	1	X	Trigger low level sensitive. Performs continuous conversions while trigger level is active.
1	1	1	X	Trigger high level sensitive. Performs continuous conversions while trigger level is active.

In either level or edge sensitive modes, the first conversion begins when the trigger is received.

Once ETRIGE is enabled a conversion must be triggered externally after writing to ATDCTL5 register.

During a conversion in edge sensitive mode, if additional trigger events are detected the overrun error flag ETORF is set.

If level sensitive mode is active and the external trigger de-asserts and later asserts again during a conversion sequence, this does not constitute an overrun. Therefore, the flag is not set. If the trigger is left active in level sensitive mode when a sequence is about to complete, another sequence will be triggered immediately.

13.4.2.2 General-Purpose Digital Port Operation

Each ATD input pin can be switched between analog or digital input functionality. An analog multiplexer makes each ATD input pin selected as analog input available to the A/D converter.

The pad of the ATD input pin is always connected to the analog input channel of the analog mulitplexer.

Each pad input signal is buffered to the digital port register.

This buffer can be turned on or off with the ATDDIEN register for each ATD input pin.
This is important so that the buffer does not draw excess current when an ATD input pin is selected as analog input to the ADC10B12C.

15.2 Signal Description

This section lists all inputs to the ADC10B16C block.

15.2.1 Detailed Signal Descriptions

15.2.1.1 AN*x* (*x* = 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0)

This pin serves as the analog input Channel *x*. It can also be configured as digital port or external trigger for the ATD conversion.

15.2.1.2 ETRIG3, ETRIG2, ETRIG1, ETRIG0

These inputs can be configured to serve as an external trigger for the ATD conversion.

Refer to device specification for availability and connectivity of these inputs!

15.2.1.3 VRH, VRL

VRH is the high reference voltage, VRL is the low reference voltage for ATD conversion.

15.2.1.4 VDDA, VSSA

These pins are the power supplies for the analog circuitry of the ADC10B16C block.

15.3 Memory Map and Register Definition

This section provides a detailed description of all registers accessible in the ADC10B16C.

15.3.1 Module Memory Map

Figure 15-2 gives an overview on all ADC10B16C registers.

NOTE

Register Address = Base Address + Address Offset, where the Base Address is defined at the MCU level and the Address Offset is defined at the module level.

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000	ATDCTL0	R W	Reserved	0	0	0	WRAP3	WRAP2	WRAP1	WRAP0
0x0001	ATDCTL1	R W	ETRIGSEL	SRES1	SRES0	SMP_DIS	ETRIGCH3	ETRIGCH2	ETRIGCH1	ETRIGCH0
0x0002	ATDCTL2	R W	0	AFFC	Reserved	ETRIGLE	ETRIGP	ETRIGE	ASCIE	ACMPIE

= Unimplemented or Reserved

Figure 15-2. ADC10B16C Register Summary (Sheet 1 of 3)

15.3.2 Register Descriptions

This section describes in address order all the ADC10B16C registers and their individual bits.

15.3.2.1 ATD Control Register 0 (ATDCTL0)

Writes to this register will abort current conversion sequence.

Module Base + 0x0000

	7	6	5	4	3	2	1	0
R	Reserved	0	0	0	WRAP3	WRAP2	WRAP1	WRAP0
W								
Reset	0	0	0	0	1	1	1	1

= Unimplemented or Reserved

Figure 15-3. ATD Control Register 0 (ATDCTL0)

Read: Anytime

Write: Anytime, in special modes always write 0 to Reserved Bit 7.

Table 15-1. ATDCTL0 Field Descriptions

Field	Description
3-0 WRAP[3-0]	**Wrap Around Channel Select Bits** — These bits determine the channel for wrap around when doing multi-channel conversions. The coding is summarized in Table 15-2.

Table 15-2. Multi-Channel Wrap Around Coding

WRAP3	WRAP2	WRAP1	WRAP0	Multiple Channel Conversions (MULT = 1) Wraparound to AN0 after Converting
0	0	0	0	Reserved[1]
0	0	0	1	AN1
0	0	1	0	AN2
0	0	1	1	AN3
0	1	0	0	AN4
0	1	0	1	AN5
0	1	1	0	AN6
0	1	1	1	AN7
1	0	0	0	AN8
1	0	0	1	AN9
1	0	1	0	AN10
1	0	1	1	AN11
1	1	0	0	AN12
1	1	0	1	AN13
1	1	1	0	AN14
1	1	1	1	AN15

The MSCAN then schedules the message for transmission and signals the successful transmission of the buffer by setting the associated TXE flag. A transmit interrupt (see Section 18.4.7.2, “Transmit Interrupt”) is generated[1] when TXEx is set and can be used to drive the application software to re-load the buffer.

If more than one buffer is scheduled for transmission when the CAN bus becomes available for arbitration, the MSCAN uses the local priority setting of the three buffers to determine the prioritization. For this purpose, every transmit buffer has an 8-bit local priority field (PRIO). The application software programs this field when the message is set up. The local priority reflects the priority of this particular message relative to the set of messages being transmitted from this node. The lowest binary value of the PRIO field is defined to be the highest priority. The internal scheduling process takes place whenever the MSCAN arbitrates for the CAN bus. This is also the case after the occurrence of a transmission error.

When a high priority message is scheduled by the application software, it may become necessary to abort a lower priority message in one of the three transmit buffers. Because messages that are already in transmission cannot be aborted, the user must request the abort by setting the corresponding abort request bit (ABTRQ) (see Section 18.3.2.9, “MSCAN Transmitter Message Abort Request Register (CANTARQ)”.) The MSCAN then grants the request, if possible, by:

1. Setting the corresponding abort acknowledge flag (ABTAK) in the CANTAAK register.
2. Setting the associated TXE flag to release the buffer.
3. Generating a transmit interrupt. The transmit interrupt handler software can determine from the setting of the ABTAK flag whether the message was aborted (ABTAK = 1) or sent (ABTAK = 0).

18.4.2.3 Receive Structures

The received messages are stored in a five stage input FIFO. The five message buffers are alternately mapped into a single memory area (see Figure 18-39). The background receive buffer (RxBG) is exclusively associated with the MSCAN, but the foreground receive buffer (RxFG) is addressable by the CPU (see Figure 18-39). This scheme simplifies the handler software because only one address area is applicable for the receive process.

All receive buffers have a size of 15 bytes to store the CAN control bits, the identifier (standard or extended), the data contents, and a time stamp, if enabled (see Section 18.3.3, “Programmer’s Model of Message Storage”).

The receiver full flag (RXF) (see Section 18.3.2.5, “MSCAN Receiver Flag Register (CANRFLG)”) signals the status of the foreground receive buffer. When the buffer contains a correctly received message with a matching identifier, this flag is set.

On reception, each message is checked to see whether it passes the filter (see Section 18.4.3, “Identifier Acceptance Filter”) and simultaneously is written into the active RxBG. After successful reception of a valid message, the MSCAN shifts the content of RxBG into the receiver FIFO, sets the RXF flag, and generates a receive interrupt[2] (see Section 18.4.7.3, “Receive Interrupt”) to the CPU. The user’s receive handler must read the received message from the RxFG and then reset the RXF flag to acknowledge the interrupt and to release the foreground buffer. A new message, which can follow immediately after the IFS field of the CAN frame, is received into the next available RxBG. If the MSCAN receives an invalid

1. The transmit interrupt occurs only if not masked. A polling scheme can be applied on TXEx also.
2. The receive interrupt occurs only if not masked. A polling scheme can be applied on RXF also.

Table 21-3. SPICR2 Field Descriptions

Field	Description
6 XFRW	**Transfer Width —** This bit is used for selecting the data transfer width. If 8-bit transfer width is selected, SPIDRL becomes the dedicated data register and SPIDRH is unused. If 16-bit transfer width is selected, SPIDRH and SPIDRL form a 16-bit data register. Please refer to Section 21.3.2.4, “SPI Status Register (SPISR) for information about transmit/receive data handling and the interrupt flag clearing mechanism. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state. 0 8-bit Transfer Width (n = 8)[1] 1 16-bit Transfer Width (n = 16)[1]
4 MODFEN	**Mode Fault Enable Bit** — This bit allows the MODF failure to be detected. If the SPI is in master mode and MODFEN is cleared, then the $\overline{SS}$ port pin is not used by the SPI. In slave mode, the $\overline{SS}$ is available only as an input regardless of the value of MODFEN. For an overview on the impact of the MODFEN bit on the $\overline{SS}$ port pin configuration, refer to Table 21-2. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state. 0 $\overline{SS}$ port pin is not used by the SPI. 1 $\overline{SS}$ port pin with MODF feature.
3 BIDIROE	**Output Enable in the Bidirectional Mode of Operation** — This bit controls the MOSI and MISO output buffer of the SPI, when in bidirectional mode of operation (SPC0 is set). In master mode, this bit controls the output buffer of the MOSI port, in slave mode it controls the output buffer of the MISO port. In master mode, with SPC0 set, a change of this bit will abort a transmission in progress and force the SPI into idle state. 0 Output buffer disabled. 1 Output buffer enabled.
1 SPISWAI	**SPI Stop in Wait Mode Bit** — This bit is used for power conservation while in wait mode. 0 SPI clock operates normally in wait mode. 1 Stop SPI clock generation when in wait mode.
0 SPC0	**Serial Pin Control Bit 0** — This bit enables bidirectional pin configurations as shown in Table 21-4. In master mode, a change of this bit will abort a transmission in progress and force the SPI system into idle state.

[1] n is used later in this document as a placeholder for the selected transfer width.

Table 21-4. Bidirectional Pin Configurations

Pin Mode	SPC0	BIDIROE	MISO	MOSI
Master Mode of Operation				
Normal	0	X	Master In	Master Out
Bidirectional	1	0	MISO not used by SPI	Master In
		1		Master I/O
Slave Mode of Operation				
Normal	0	X	Slave Out	Slave In
Bidirectional	1	0	Slave In	MOSI not used by SPI
		1	Slave I/O	

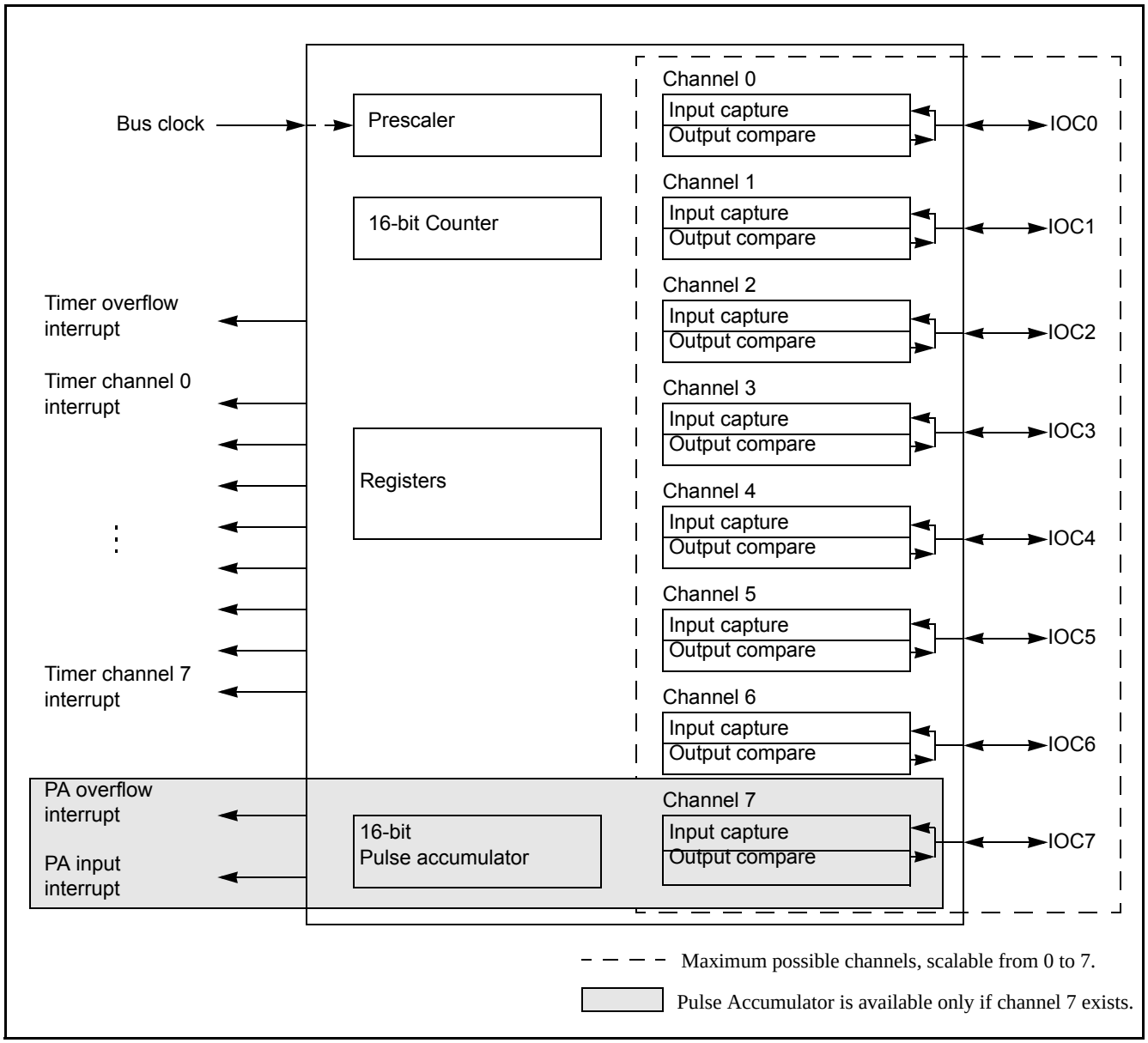

Figure 23-1. TIM16B8CV3 Block Diagram

Note: To enable output action using the OM7 and OL7 bits on the timer port,the corresponding bit OC7M7 in the OC7M register must also be cleared. The settings for these bits can be seen inTable 23-10.

Table 23-10. The OC7 and OCx event priority

OC7M7=0				OC7M7=1			
OC7Mx=1		OC7Mx=0		OC7Mx=1		OC7Mx=0	
TC7=TCx	TC7>TCx	TC7=TCx	TC7>TCx	TC7=TCx	TC7>TCx	TC7=TCx	TC7>TCx
IOCx=OC7Dx IOC7=OM7/OL7	IOCx=OC7Dx +OMx/OLx IOC7=OM7/OL7	IOCx=OMx/OLx IOC7=OM7/OL7		IOCx=OC7Dx IOC7=OC7D7	IOCx=OC7Dx +OMx/OLx IOC7=OC7D7	IOCx=OMx/OLx IOC7=OC7D7	

Note: in Table 23-10, the IOS7 and IOSx should be set to 1

IOSx is the register TIOS bit x,

OC7Mx is the register OC7M bit x,

TCx is timer Input Capture/Output Compare register,

IOCx is channel x,

OMx/OLx is the register TCTL1/TCTL2,

OC7Dx is the register OC7D bit x.

IOCx = OC7Dx+ OMx/OLx, means that both OC7 event and OCx event will change channel x value.

23.3.2.9 Timer Control Register 3/Timer Control Register 4 (TCTL3 and TCTL4)

Module Base + 0x000A

	7	6	5	4	3	2	1	0
R W	EDG7B	EDG7A	EDG6B	EDG6A	EDG5B	EDG5A	EDG4B	EDG4A
Reset	0	0	0	0	0	0	0	0

Figure 23-16. Timer Control Register 3 (TCTL3)

Module Base + 0x000B

	7	6	5	4	3	2	1	0
R W	EDG3B	EDG3A	EDG2B	EDG2A	EDG1B	EDG1A	EDG0B	EDG0A
Reset	0	0	0	0	0	0	0	0

Figure 23-17. Timer Control Register 4 (TCTL4)

Read: Anytime

Table 24-26. P-Flash Commands

FCMD	Command	Function on P-Flash Memory
0x02	Erase Verify Block	Verify that a P-Flash block is erased.
0x03	Erase Verify P-Flash Section	Verify that a given number of words starting at the address provided are erased.
0x04	Read Once	Read a dedicated 64 byte field in the nonvolatile information register in P-Flash block that was previously programmed using the Program Once command.
0x06	Program P-Flash	Program a phrase in a P-Flash block.
0x07	Program Once	Program a dedicated 64 byte field in the nonvolatile information register in P-Flash block that is allowed to be programmed only once.
0x08	Erase All Blocks	Erase all P-Flash (and EEPROM) blocks. An erase of all Flash blocks is only possible when the FPLDIS, FPHDIS, and FPOPEN bits in the FPROT register and the DPOPEN bit in the EEPROT register are set prior to launching the command.
0x09	Erase Flash Block	Erase a P-Flash (or EEPROM) block. An erase of the full P-Flash block is only possible when FPLDIS, FPHDIS and FPOPEN bits in the FPROT register are set prior to launching the command.
0x0A	Erase P-Flash Sector	Erase all bytes in a P-Flash sector.
0x0B	Unsecure Flash	Supports a method of releasing MCU security by erasing all P-Flash (and EEPROM) blocks and verifying that all P-Flash (and EEPROM) blocks are erased.
0x0C	Verify Backdoor Access Key	Supports a method of releasing MCU security by verifying a set of security keys.
0x0D	Set User Margin Level	Specifies a user margin read level for all P-Flash blocks.
0x0E	Set Field Margin Level	Specifies a field margin read level for all P-Flash blocks (special modes only).

24.4.4.5 EEPROM Commands

Table 24-27 summarizes the valid EEPROM commands along with the effects of the commands on the EEPROM block.

Table 24-27. EEPROM Commands

FCMD	Command	Function on EEPROM Memory
0x01	Erase Verify All Blocks	Verify that all EEPROM (and P-Flash) blocks are erased.
0x02	Erase Verify Block	Verify that the EEPROM block is erased.

25.3.2.9.1 P-Flash Protection Restrictions

The general guideline is that P-Flash protection can only be added and not removed. Table 25-21 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 25-21. 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 25-14 for a definition of the scenarios.

25.3.2.10 EEPROM Protection Register (EEPROT)

The EEPROT register defines which EEPROM sectors are protected against program and erase operations.

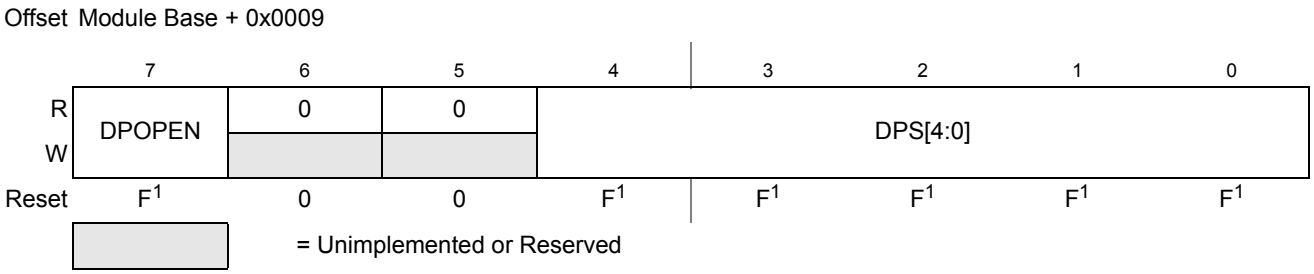

Figure 25-15. EEPROM Protection Register (EEPROT)

1 Loaded from IFR Flash configuration field, during reset sequence.

The (unreserved) bits of the EEPROT 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.

25.3.2.11 Flash Common Command Object Register (FCCOB)

The FCCOB is an array of six words addressed via the CCOBIX index found in the FCCOBIX register. Byte wide reads and writes are allowed to the FCCOB register.

Offset Module Base + 0x000A

	7	6	5	4	3	2	1	0
R W	CCOB[15:8]							
Reset	0	0	0	0	0	0	0	0

Figure 25-16. Flash Common Command Object High Register (FCCOBHI)

Offset Module Base + 0x000B

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

Figure 25-17. Flash Common Command Object Low Register (FCCOBLO)

25.3.2.11.1 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 25-24. 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 25-24 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 25.4.6.

Table 25-24. 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]

Table 25-46. 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 25-47. 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 25-27)
		Set if an invalid global address [17:16] is supplied
		Set if the supplied P-Flash address is not phrase-aligned or if the EEPROM 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
	MGSTAT0	Set if any non-correctable errors have been encountered during the verify operation

25.4.6.9 Erase P-Flash Sector Command

The Erase P-Flash Sector operation will erase all addresses in a P-Flash sector.

Table 25-48. Erase P-Flash Sector Command FCCOB Requirements

CCOBIX[2:0]	FCCOB Parameters	
000	0x0A	Global address [17:16] to identify P-Flash block to be erased
001	Global address [15:0] anywhere within the sector to be erased. Refer to Section 25.1.2.1 for the P-Flash sector size.	

Upon clearing CCIF to launch the Erase P-Flash Sector command, the Memory Controller will erase the selected Flash sector and then verify that it is erased. The CCIF flag will be set after the Erase P-Flash Sector operation has completed.

28.3.2.2 Flash Security Register (FSEC)

The FSEC register holds all bits associated with the security of the MCU and Flash module.

Offset Module Base + 0x0001

	7	6	5	4	3	2	1	0
R	KEYEN[1:0]		RNV[5:2]				SEC[1:0]	
W								
Reset	F[1]	F[1]	F[1]	F[1]	F[1]	F[1]	F[1]	F[1]

= Unimplemented or Reserved

Figure 28-6. Flash Security Register (FSEC)

1 Loaded from IFR Flash configuration field, during reset sequence.

All bits in the FSEC register are readable but not writable.

During the reset sequence, the FSEC register is loaded with the contents of the Flash security byte in the Flash configuration field at global address 0x3_FF0F located in P-Flash memory (see Table 28-4) as indicated by reset condition F in Figure 28-6. If a double bit fault is detected while reading the P-Flash 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 28-9. 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 28-10.
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 28-11. If the Flash module is unsecured using backdoor key access, the SEC bits are forced to 10.

Table 28-10. 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.