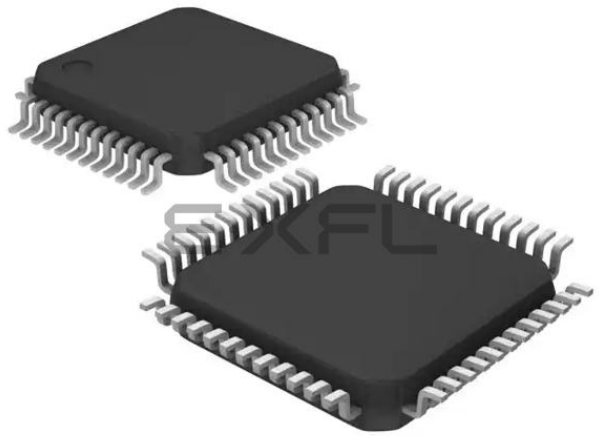


Welcome to E-XFL.COM

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	25MHz
Connectivity	CANbus, EBI/EMI, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	31
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°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/mc9s12c128mfae

Chapter 20
96 Kbyte Flash Module (S12FTS96KV1)

Chapter 21
128 Kbyte Flash Module (S12FTS128K1V1)

The security byte resides in a portion of the Flash array.

Check the Flash Block User Guide for more details on the security configuration.

1.5.2.2 Operation of the Secured Microcontroller

1.5.2.2.1 Normal Single Chip Mode

This will be the most common usage of the secured part. Everything will appear the same as if the part was not secured with the exception of BDM operation. The BDM operation will be blocked.

1.5.2.2.2 Executing from External Memory

The user may wish to execute from external space with a secured microcontroller. This is accomplished by resetting directly into expanded mode. The internal FLASH will be disabled. BDM operations will be blocked.

1.5.2.3 Unsecuring the Microcontroller

In order to unsecure the microcontroller, the internal FLASH must be erased. This can be done through an external program in expanded mode or via a sequence of BDM commands. Unsecuring is also possible via the Backdoor Key Access. Refer to Flash Block Guide for details.

Once the user has erased the FLASH, the part can be reset into special single chip mode. This invokes a program that verifies the erasure of the internal FLASH. Once this program completes, the user can erase and program the FLASH security bits to the unsecured state. This is generally done through the BDM, but the user could also change to expanded mode (by writing the mode bits through the BDM) and jumping to an external program (again through BDM commands). Note that if the part goes through a reset before the security bits are reprogrammed to the unsecure state, the part will be secured again.

1.5.3 Low-Power Modes

The microcontroller features three main low power modes. Consult the respective Block User Guide for information on the module behavior in stop, pseudo stop, and wait mode. An important source of information about the clock system is the Clock and Reset Generator User Guide (CRG).

1.5.3.1 Stop

Executing the CPU STOP instruction stops all clocks and the oscillator thus putting the chip in fully static mode. Wake up from this mode can be done via reset or external interrupts.

1.5.3.2 Pseudo Stop

This mode is entered by executing the CPU STOP instruction. In this mode the oscillator is still running and the real time interrupt (RTI) or watchdog (COP) sub module can stay active. Other peripherals are turned off. This mode consumes more current than the full stop mode, but the wake up time from this mode is significantly shorter.

Table 4-7. MODE Field Descriptions

Field	Description
7:5 MOD[C:A]	**Mode Select Bits** — These bits indicate the current operating mode. If MODA = 1, then MODC, MODB, and MODA are write never. If MODC = MODA = 0, then MODC, MODB, and MODA are writable with the exception that you cannot change to or from special peripheral mode If MODC = 1, MODB = 0, and MODA = 0, then MODC is write never. MODB and MODA are write once, except that you cannot change to special peripheral mode. From normal single-chip, only normal expanded narrow and normal expanded wide modes are available. See Table 4-8 and Table 4-16.
3 IVIS	**Internal Visibility (for both read and write accesses)** — This bit determines whether internal accesses generate a bus cycle that is visible on the external bus. Normal: write once Emulation: write never Special: write anytime 0 No visibility of internal bus operations on external bus. 1 Internal bus operations are visible on external bus.
1 EMK	**Emulate Port K** Normal: write once Emulation: write never Special: write anytime 0 PORTK and DDRK are in the memory map so port K can be used for general-purpose I/O. 1 If in any expanded mode, PORTK and DDRK are removed from the memory map. In single-chip modes, PORTK and DDRK are always in the map regardless of the state of this bit. In special peripheral mode, PORTK and DDRK are never in the map regardless of the state of this bit.
0 EME	**Emulate Port E** Normal and Emulation: write never Special: write anytime 0 PORTE and DDRE are in the memory map so port E can be used for general-purpose I/O. 1 If in any expanded mode or special peripheral mode, PORTE and DDRE are removed from the memory map. Removing the registers from the map allows the user to emulate the function of these registers externally. In single-chip modes, PORTE and DDRE are always in the map regardless of the state of this bit.

mode. Background debugging should not be used while the MCU is in special peripheral mode as internal bus conflicts between BDM and the external master can cause improper operation of both functions.

4.4.4 Internal Visibility

Internal visibility is available when the MCU is operating in expanded wide modes or emulation narrow mode. It is not available in single-chip, peripheral or normal expanded narrow modes. Internal visibility is enabled by setting the IVIS bit in the MODE register.

If an internal access is made while E, R/$\overline{W}$, and $\overline{LSTRB}$ are configured as bus control outputs and internal visibility is off (IVIS=0), E will remain low for the cycle, R/$\overline{W}$ will remain high, and address, data and the $\overline{LSTRB}$ pins will remain at their previous state.

When internal visibility is enabled (IVIS=1), certain internal cycles will be blocked from going external. During cycles when the BDM is selected, R/$\overline{W}$ will remain high, data will maintain its previous state, and address and $\overline{LSTRB}$ pins will be updated with the internal value. During CPU no access cycles when the BDM is not driving, R/$\overline{W}$ will remain high, and address, data and the $\overline{LSTRB}$ pins will remain at their previous state.

NOTE

When the system is operating in a secure mode, internal visibility is not available (i.e., IVIS = 1 has no effect). Also, the IPIPE signals will not be visible, regardless of operating mode. IPIPE1–IPIPE0 will display 0es if they are enabled. In addition, the MOD bits in the MODE control register cannot be written.

4.4.5 Low-Power Options

The MEBI does not contain any user-controlled options for reducing power consumption. The operation of the MEBI in low-power modes is discussed in the following subsections.

4.4.5.1 Operation in Run Mode

The MEBI does not contain any options for reducing power in run mode; however, the external addresses are conditioned to reduce power in single-chip modes. Expanded bus modes will increase power consumption.

4.4.5.2 Operation in Wait Mode

The MEBI does not contain any options for reducing power in wait mode.

4.4.5.3 Operation in Stop Mode

The MEBI will cease to function after execution of a CPU STOP instruction.

5.6.3 Interrupt Priority Decoder

The priority decoder evaluates all interrupts pending and determines their validity and priority. When the CPU requests an interrupt vector, the decoder will provide the vector for the highest priority interrupt request. Because the vector is not supplied until the CPU requests it, it is possible that a higher priority interrupt request could override the original exception 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 exception instead of the original request.

NOTE

Care must be taken to ensure that all exception requests remain active until the system begins execution of the applicable service routine; otherwise, the exception request may not be processed.

If for any reason the interrupt source is unknown (e.g., an interrupt request becomes inactive after the interrupt has been recognized but prior to the vector request), the vector address will default to that of the last valid interrupt that existed during the particular interrupt sequence. If the CPU requests an interrupt vector when there has never been a pending interrupt request, the INT will provide the software interrupt (SWI) vector address.

5.7 Exception Priority

The priority (from highest to lowest) and address of all exception vectors issued by the INT upon request by the CPU is shown in Table 5-5.

Table 5-5. Exception Vector Map and Priority

Vector Address	Source
0xFFFE–0xFFFF	System reset
0xFFFC–0xFFFD	Crystal monitor reset
0xFFFA–0xFFFB	COP reset
0xFFF8–0xFFF9	Unimplemented opcode trap
0xFFF6–0xFFF7	Software interrupt instruction (SWI) or BDM vector request
0xFFF4–0xFFF5	$\overline{\text{XIRQ}}$ signal
0xFFF2–0xFFF3	$\overline{\text{IRQ}}$ signal
0xFFF0–0xFF00	Device-specific I-bit maskable interrupt sources (priority in descending order)

9.3.2.4 CRG Flags Register (CRGFLG)

This register provides CRG status bits and flags.

Module Base + 0x0003

	7	6	5	4	3	2	1	0
R	RTIF	PORF	LVRF	LOCKIF	LOCK	TRACK	SCMIF	SCM
W								
Reset	0	Note 1	Note 2	0	0	0	0	0

1. PORF is set to 1 when a power-on reset occurs. Unaffected by system reset.
2. LVRF is set to 1 when a low-voltage reset occurs. Unaffected by system reset.

= Unimplemented or Reserved

Figure 9-7. CRG Flag Register (CRGFLG)

Read: anytime

Write: refer to each bit for individual write conditions

Table 9-2. CRGFLG Field Descriptions

Field	Description
7 RTIF	**Real-Time Interrupt Flag** — RTIF is set to 1 at the end of the RTI period. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (RTIE = 1), RTIF causes an interrupt request. 0 RTI time-out has not yet occurred. 1 RTI time-out has occurred.
6 PORF	**Power-on Reset Flag** — PORF is set to 1 when a power-on reset occurs. This flag can only be cleared by writing a 1. Writing a 0 has no effect. 0 Power-on reset has not occurred. 1 Power-on reset has occurred.
5 LVRF	**Low-Voltage Reset Flag** — If low voltage reset feature is not available (see the device overview chapter), LVRF always reads 0. LVRF is set to 1 when a low voltage reset occurs. This flag can only be cleared by writing a 1. Writing a 0 has no effect. 0 Low voltage reset has not occurred. 1 Low voltage reset has occurred.
4 LOCKIF	**PLL Lock Interrupt Flag** — LOCKIF is set to 1 when LOCK status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect.If enabled (LOCKIE = 1), LOCKIF causes an interrupt request. 0 No change in LOCK bit. 1 LOCK bit has changed.
3 LOCK	**Lock Status Bit** — LOCK reflects the current state of PLL lock condition. This bit is cleared in self-clock mode. Writes have no effect. 0 PLL VCO is not within the desired tolerance of the target frequency. 1 PLL VCO is within the desired tolerance of the target frequency.
2 TRACK	**Track Status Bit** — TRACK reflects the current state of PLL track condition. This bit is cleared in self-clock mode. Writes have no effect. 0 Acquisition mode status. 1 Tracking mode status.

If the bus clock is generated from a PLL, it is recommended to select the oscillator clock rather than the bus clock due to jitter considerations, especially at the faster CAN bus rates.

For microcontrollers without a clock and reset generator (CRG), CANCLK is driven from the crystal oscillator (oscillator clock).

A programmable prescaler generates the time quanta (Tq) clock from CANCLK. A time quantum is the atomic unit of time handled by the MSCAN.

Eqn. 10-2

$$Tq = \frac{f_{CANCLK}}{(\text{Prescaler value})}$$

A bit time is subdivided into three segments as described in the Bosch CAN specification. (see Figure 10-43):

- SYNC_SEG: This segment has a fixed length of one time quantum. Signal edges are expected to happen within this section.
- Time Segment 1: This segment includes the PROP_SEG and the PHASE_SEG1 of the CAN standard. It can be programmed by setting the parameter TSEG1 to consist of 4 to 16 time quanta.
- Time Segment 2: This segment represents the PHASE_SEG2 of the CAN standard. It can be programmed by setting the TSEG2 parameter to be 2 to 8 time quanta long.

Eqn. 10-3

$$\text{Bit Rate} = \frac{f_{Tq}}{(\text{number of Time Quanta})}$$

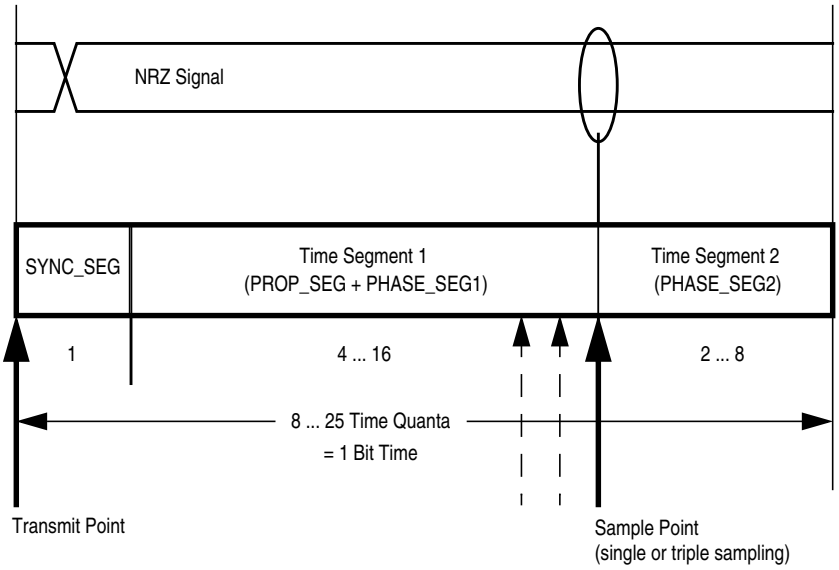


Figure 10-43. Segments within the Bit Time

Table 12-5. PWMPRCLK Field Descriptions

Field	Description
6:5 PCKB[2:0]	**Prescaler Select for Clock B** — Clock B is 1 of two clock sources which can be used for channels 2 or 3. These three bits determine the rate of clock B, as shown in Table 12-6.
2:0 PCKA[2:0]	**Prescaler Select for Clock A** — Clock A is 1 of two clock sources which can be used for channels 0, 1, 4, or 5. These three bits determine the rate of clock A, as shown in Table 12-7.

Table 12-6. Clock B Prescaler Selects

PCKB2	PCKB1	PCKB0	Value of Clock B
0	0	0	Bus Clock
0	0	1	Bus Clock / 2
0	1	0	Bus Clock / 4
0	1	1	Bus Clock / 8
1	0	0	Bus Clock / 16
1	0	1	Bus Clock / 32
1	1	0	Bus Clock / 64
1	1	1	Bus Clock / 128

Table 12-7. Clock A Prescaler Selects

PCKA2	PCKA1	PCKA0	Value of Clock A
0	0	0	Bus Clock
0	0	1	Bus Clock / 2
0	1	0	Bus Clock / 4
0	1	1	Bus Clock / 8
1	0	0	Bus Clock / 16
1	0	1	Bus Clock / 32
1	1	0	Bus Clock / 64
1	1	1	Bus Clock / 128

12.3.2.5 PWM Center Align Enable Register (PWMCAE)

The PWMCAE register contains six control bits for the selection of center aligned outputs or left aligned outputs for each PWM channel. If the CAEx bit is set to a 1, the corresponding PWM output will be center aligned. If the CAEx bit is cleared, the corresponding PWM output will be left aligned. Reference Section 12.4.2.5, “Left Aligned Outputs,” and Section 12.4.2.6, “Center Aligned Outputs,” for a more detailed description of the PWM output modes.

Module Base + 0x0005

	7	6	5	4	3	2	1	0
R	0	CON45	CON23	CON01	PSWAI	PFRZ	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 12-8. PWM Control Register (PWMCTL)

Read: anytime

Write: anytime

There are three control bits for concatenation, each of which is used to concatenate a pair of PWM channels into one 16-bit channel. When channels 4 and 5 are concatenated, channel 4 registers become the high-order bytes of the double-byte channel. When channels 2 and 3 are concatenated, channel 2 registers become the high-order bytes of the double-byte channel. When channels 0 and 1 are concatenated, channel 0 registers become the high-order bytes of the double-byte channel.

Reference Section 12.4.2.7, “PWM 16-Bit Functions,” for a more detailed description of the concatenation PWM function.

NOTE

Change these bits only when both corresponding channels are disabled.

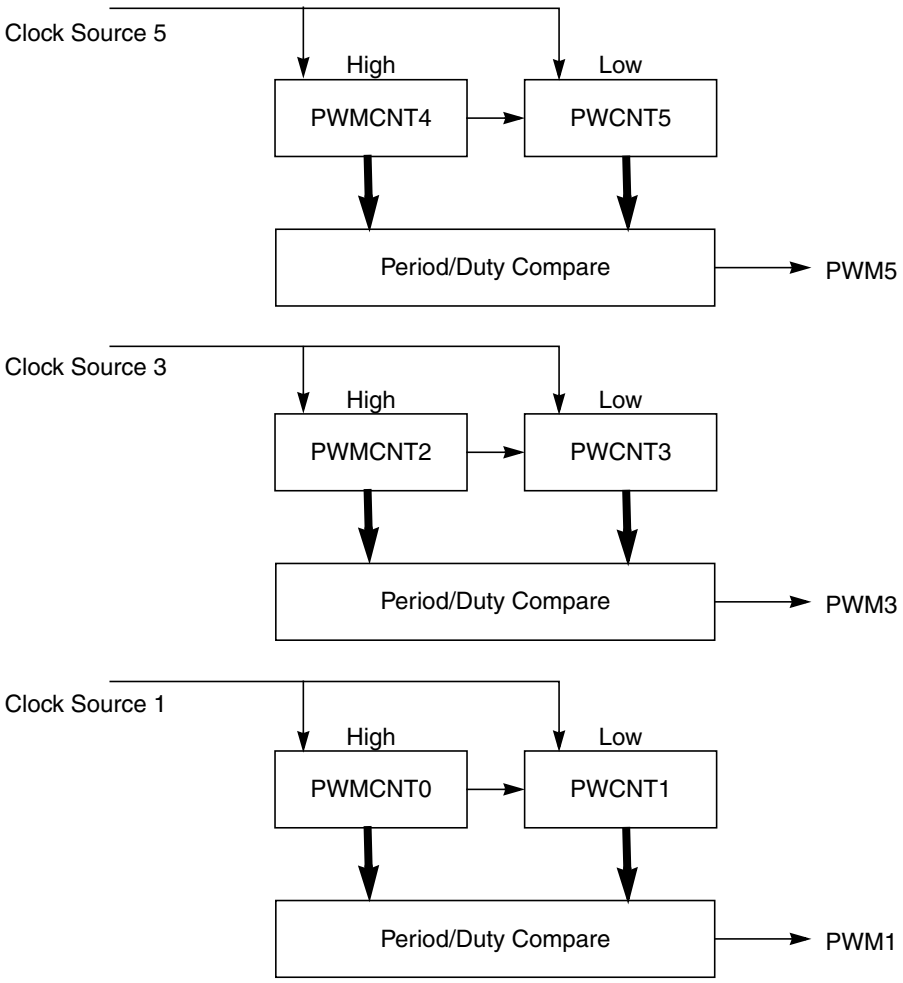


Figure 12-40. PWM 16-Bit Mode

When using the 16-bit concatenated mode, the clock source is determined by the low-order 8-bit channel clock select control bits. That is channel 5 when channels 4 and 5 are concatenated, channel 3 when channels 2 and 3 are concatenated, and channel 1 when channels 0 and 1 are concatenated. The resulting PWM is output to the pins of the corresponding low-order 8-bit channel as also shown in Figure 12-40. The polarity of the resulting PWM output is controlled by the PPOLx bit of the corresponding low-order 8-bit channel as well.

After concatenated mode is enabled (CONxx bits set in PWMCTL register), enabling/disabling the corresponding 16-bit PWM channel is controlled by the low-order PWMEx bit. In this case, the high-order bytes PWMEx bits have no effect and their corresponding PWM output is disabled.

In concatenated mode, writes to the 16-bit counter by using a 16-bit access or writes to either the low or high-order byte of the counter will reset the 16-bit counter. Reads of the 16-bit counter must be made by 16-bit access to maintain data coherency.

Either left aligned or center aligned output mode can be used in concatenated mode and is controlled by the low-order CAEx bit. The high-order CAEx bit has no effect.

Table 13-3. Loop Functions

LOOPS	RSRC	Function
0	x	Normal operation
1	0	Loop mode with Rx input internally connected to Tx output
1	1	Single-wire mode with Rx input connected to TXD

13.3.2.3 SCI Control Register 2 (SCICR2)

Module Base + 0x_0003

	7	6	5	4	3	2	1	0
R W	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
Reset	0	0	0	0	0	0	0	0

Figure 13-5. SCI Control Register 2 (SCICR2)

Read: Anytime

Write: Anytime

Table 13-4. SCICR2 Field Descriptions

Field	Description
7 TIE	**Transmitter Interrupt Enable Bit** — TIE enables the transmit data register empty flag, TDRE, to generate interrupt requests. 0 TDRE interrupt requests disabled 1 TDRE interrupt requests enabled
6 TCIE	**Transmission Complete Interrupt Enable Bit** — TCIE enables the transmission complete flag, TC, to generate interrupt requests. 0 TC interrupt requests disabled 1 TC interrupt requests enabled
5 RIE	**Receiver Full Interrupt Enable Bit** — RIE enables the receive data register full flag, RDRF, or the overrun flag, OR, to generate interrupt requests. 0 RDRF and OR interrupt requests disabled 1 RDRF and OR interrupt requests enabled
4 ILIE	**Idle Line Interrupt Enable Bit** — ILIE enables the idle line flag, IDLE, to generate interrupt requests. 0 IDLE interrupt requests disabled 1 IDLE interrupt requests enabled
3 TE	**Transmitter Enable Bit** — TE enables the SCI transmitter and configures the TXD pin as being controlled by the SCI. The TE bit can be used to queue an idle preamble. 0 Transmitter disabled 1 Transmitter enabled
2 RE	**Receiver Enable Bit** — RE enables the SCI receiver. 0 Receiver disabled 1 Receiver enabled

14.1.3 Block Diagram

Figure 14-1 gives an overview on the SPI architecture. The main parts of the SPI are status, control, and data registers, shifter logic, baud rate generator, master/slave control logic, and port control logic.

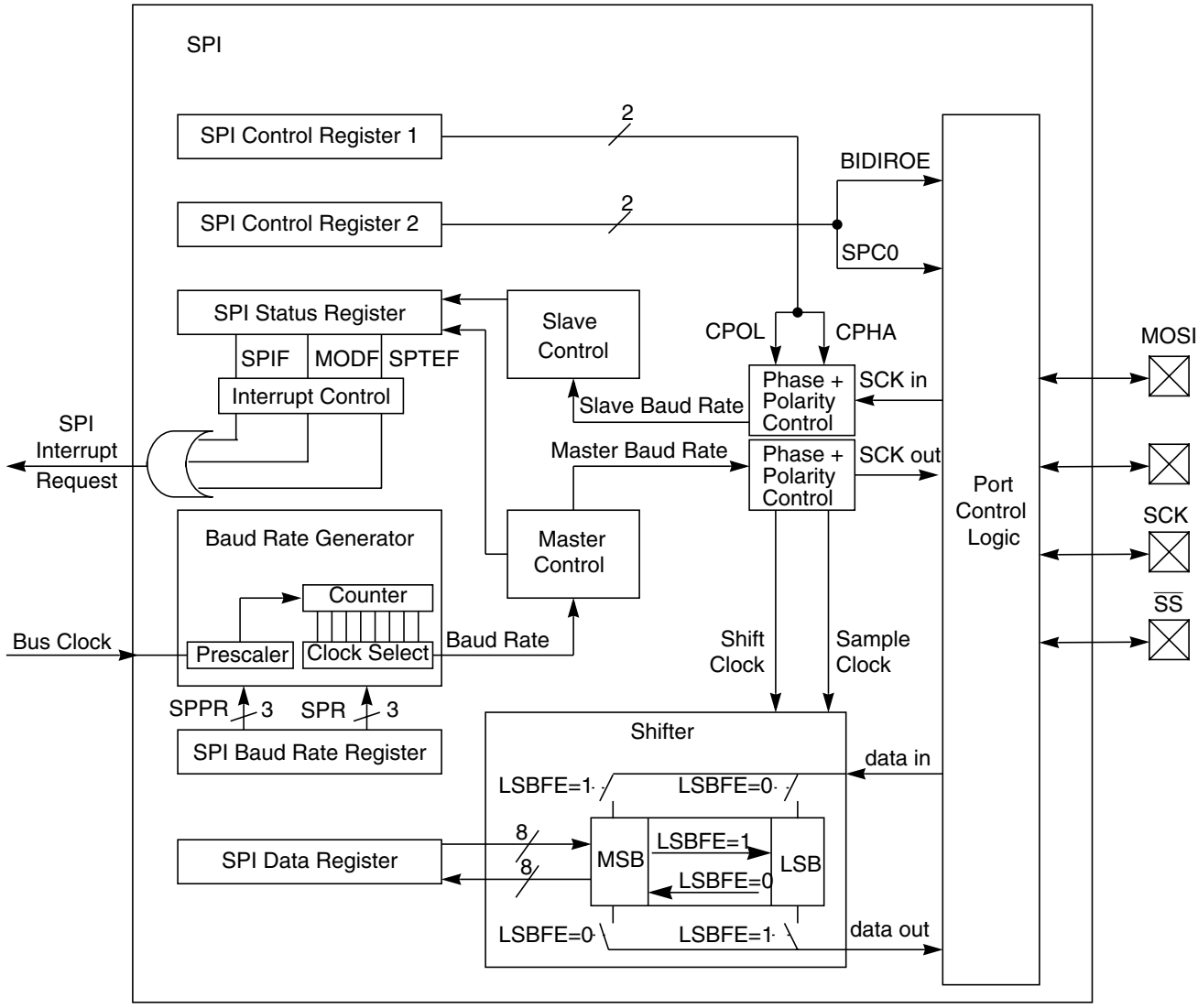


Figure 14-1. SPI Block Diagram

14.2 External Signal Description

This section lists the name and description of all ports including inputs and outputs that do, or may, connect off chip. The SPIV3 module has a total of four external pins.

14.2.1 MOSI — Master Out/Slave In Pin

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

14.4.1 Master Mode

The SPI operates in master mode when the MSTR bit is set. Only a master SPI module can initiate transmissions. A transmission begins by writing to the master SPI Data Register. If the shift register is empty, the byte immediately transfers to the shift register. The byte begins shifting out on the MOSI pin under the control of the serial clock.

- S-clock

  The SPR2, SPR1, and SPR0 baud rate selection bits in conjunction with the SPPR2, SPPR1, and SPPR0 baud rate preselection bits in the SPI Baud Rate register control the baud rate generator and determine the speed of the transmission. The SCK pin is the SPI clock output. Through the SCK pin, the baud rate generator of the master controls the shift register of the slave peripheral.
- MOSI and MISO Pins

  In master mode, the function of the serial data output pin (MOSI) and the serial data input pin (MISO) is determined by the SPC0 and BIDIROE control bits.
- $\overline{SS}$ Pin

  If MODFEN and SSOE bit are set, the $\overline{SS}$ pin is configured as slave select output. The $\overline{SS}$ output becomes low during each transmission and is high when the SPI is in idle state.

  If MODFEN is set and SSOE is cleared, the $\overline{SS}$ pin is configured as input for detecting mode fault error. If the $\overline{SS}$ input becomes low this indicates a mode fault error where another master tries to drive the MOSI and SCK lines. In this case, the SPI immediately switches to slave mode, by clearing the MSTR bit and also disables the slave output buffer MISO (or SISO in bidirectional mode). So the result is that all outputs are disabled and SCK, MOSI and MISO are inputs. If a transmission is in progress when the mode fault occurs, the transmission is aborted and the SPI is forced into idle state.

This mode fault error also sets the mode fault (MODF) flag in the SPI Status Register (SPISR). If the SPI interrupt enable bit (SPIE) is set when the MODF flag gets set, then an SPI interrupt sequence is also requested.

When a write to the SPI Data Register in the master occurs, there is a half SCK-cycle delay. After the delay, SCK is started within the master. The rest of the transfer operation differs slightly, depending on the clock format specified by the SPI clock phase bit, CPHA, in SPI Control Register 1 (see Section 14.4.3, “Transmission Formats”).

NOTE

A change of the bits CPOL, CPHA, SSOE, LSBFE, MODFEN, SPC0, BIDIROE with SPC0 set, SPPR2–SPPR0 and SPR2–SPR0 in master mode will abort a transmission in progress and force the SPI into idle state. The remote slave cannot detect this, therefore the master has to ensure that the remote slave is set back to idle state.

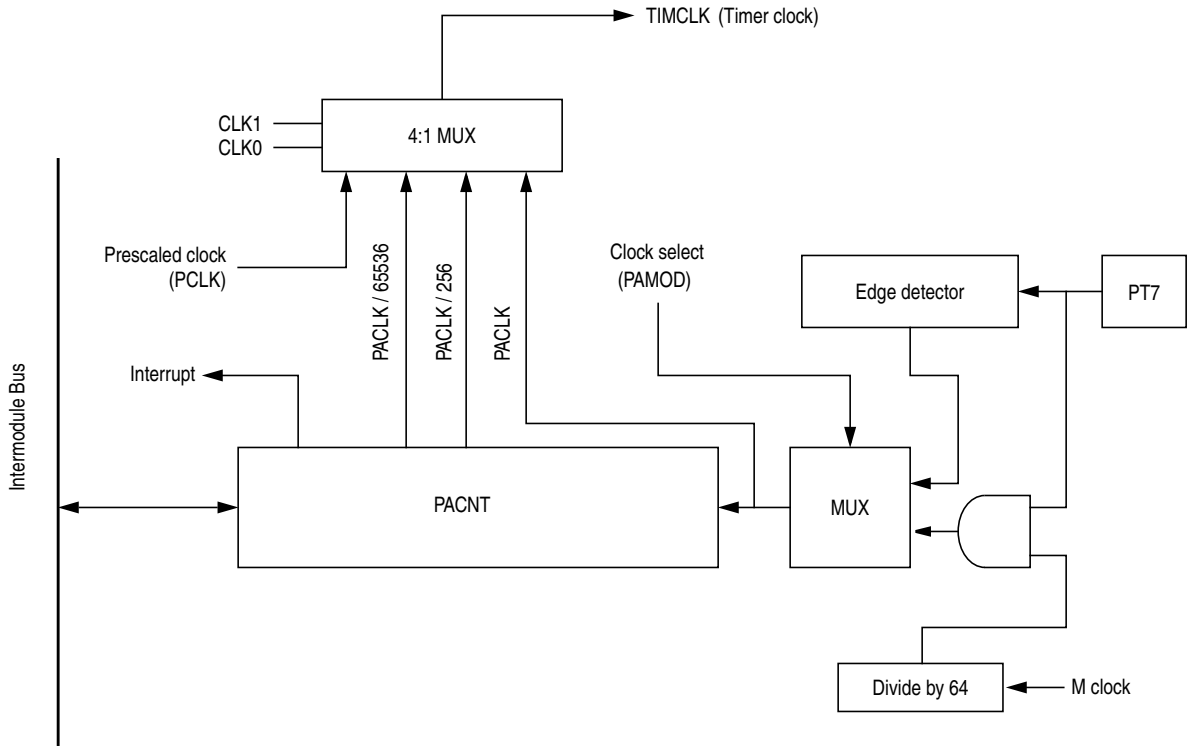


Figure 15-2. 16-Bit Pulse Accumulator Block Diagram

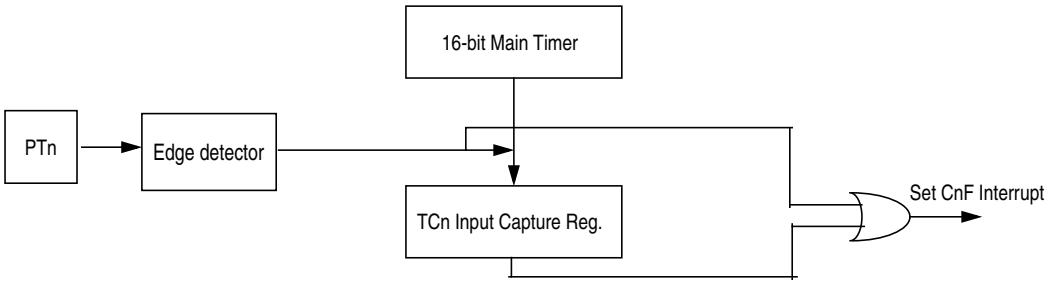


Figure 15-3. Interrupt Flag Setting

15.3.2.9 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 15-16. 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 15-17. Timer Control Register 4 (TCTL4)

Read: Anytime

Write: Anytime.

Table 15-12. TCTL3/TCTL4 Field Descriptions

Field	Description
7:0 EDGnB EDGnA	**Input Capture Edge Control** — These eight pairs of control bits configure the input capture edge detector circuits.

Table 15-13. Edge Detector Circuit Configuration

EDGnB	EDGnA	Configuration
0	0	Capture disabled
0	1	Capture on rising edges only
1	0	Capture on falling edges only
1	1	Capture on any edge (rising or falling)

18.1.3 Modes of Operation

See Section 18.4.2, “Operating Modes” for a description of the Flash module operating modes. For program and erase operations, refer to Section 18.4.1, “Flash Command Operations”.

18.1.4 Block Diagram

Figure 18-1 shows a block diagram of the FTS32K module.

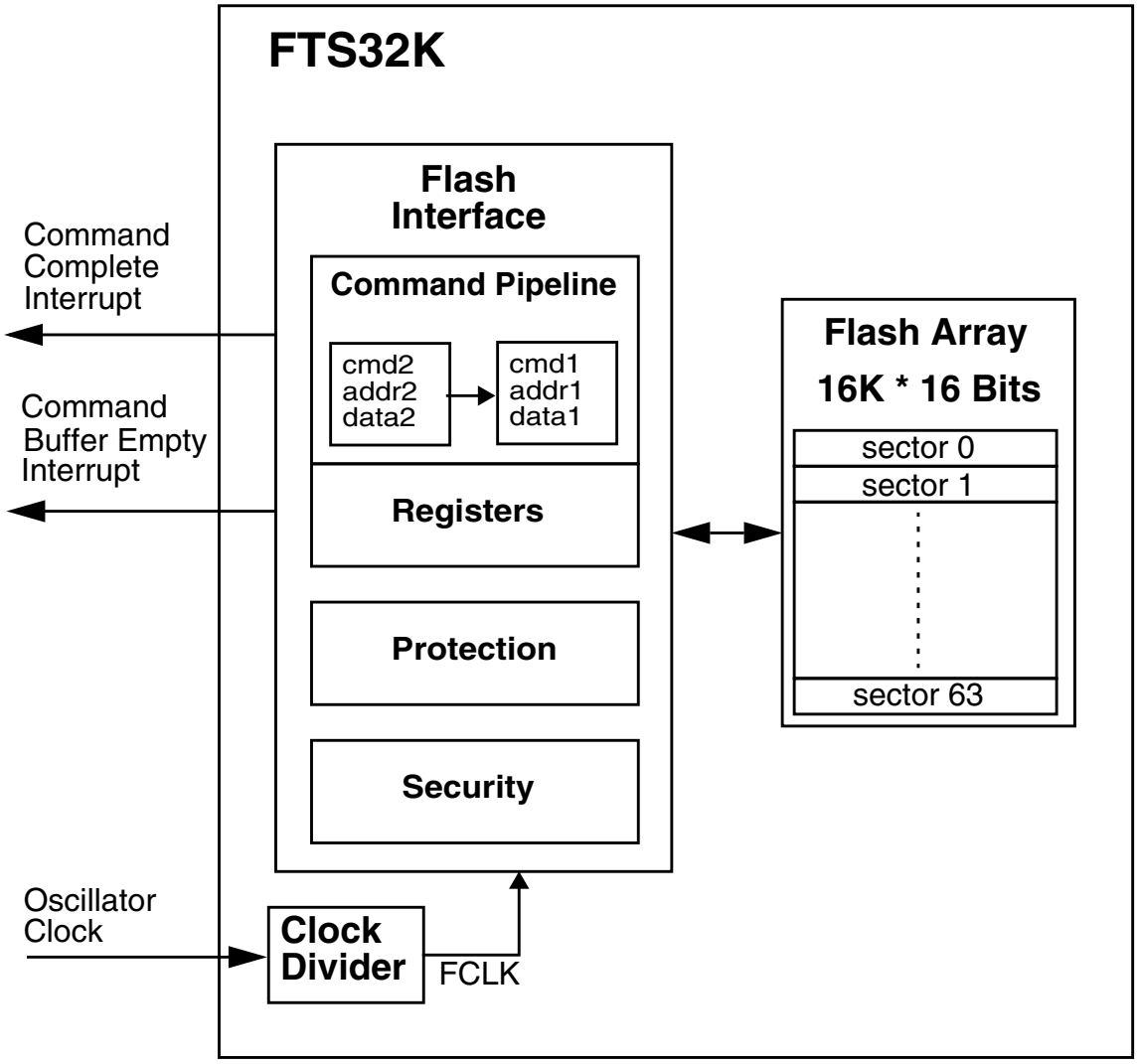


Figure 18-1. FTS32K Block Diagram

18.2 External Signal Description

The FTS32K module contains no signals that connect off-chip.

18.4.1.3 Valid Flash Commands

Table 18-16 summarizes the valid Flash commands along with the effects of the commands on the Flash array.

Table 18-16. Valid Flash Commands

FCMD	Meaning	Function on Flash Array
0x05	Erase Verify	Verify all bytes in the Flash array are erased. If the Flash array is erased, the BLANK bit will set in the FSTAT register upon command completion.
0x20	Program	Program a word (2 bytes) in the Flash array.
0x40	Sector Erase	Erase all 512 bytes in a sector of the Flash array.
0x41	Mass Erase	Erase all bytes in the Flash array. A mass erase of the full Flash array is only possible when FPLDIS, FPHDIS, and FPOPEN bits in the FPROT register are set prior to launching the command.

CAUTION

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

19.3.2.11 RESERVED3

This register is reserved for factory testing and is not accessible to the user.

Module Base + 0x000C

	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

Figure 19-20. RESERVED3

All bits read 0 and are not writable.

19.3.2.12 RESERVED4

This register is reserved for factory testing and is not accessible to the user.

Module Base + 0x000D

	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

Figure 19-21. RESERVED4

All bits read 0 and are not writable.

19.3.2.13 RESERVED5

This register is reserved for factory testing and is not accessible to the user.

Module Base + 0x000E

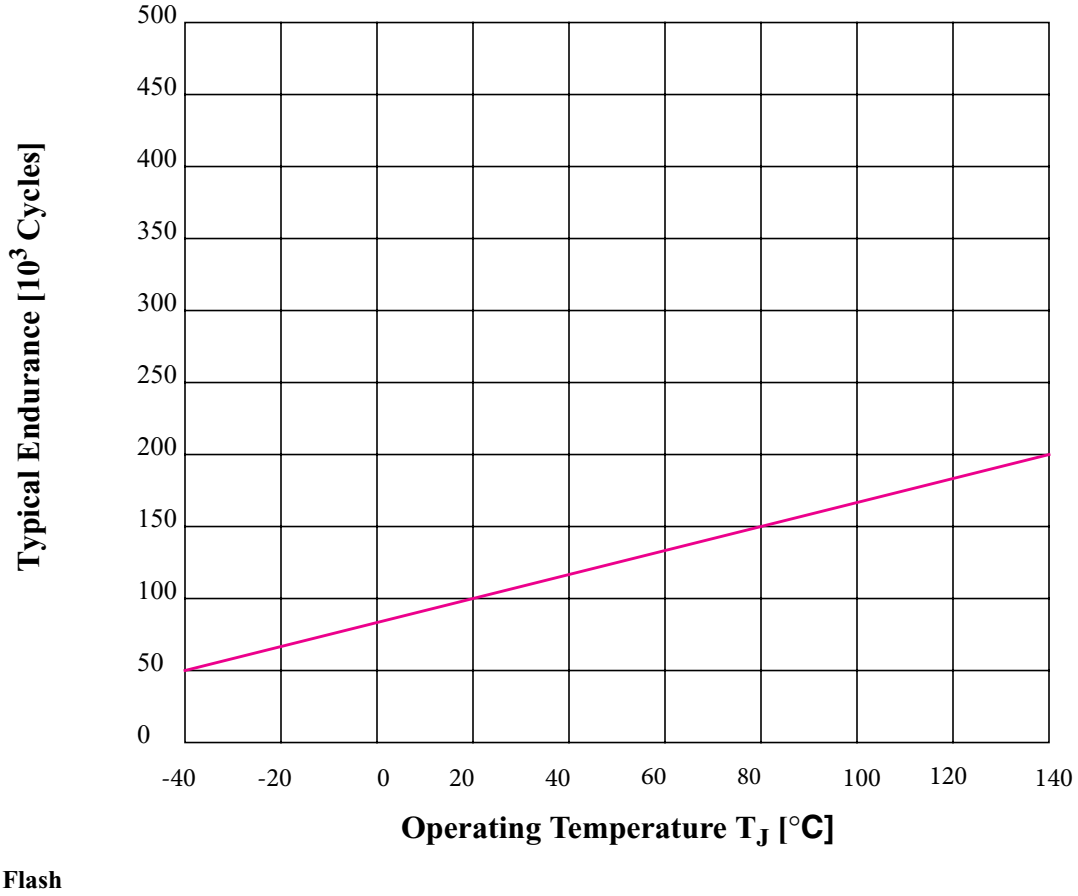
	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 19-22. RESERVED5

All bits read 0 and are not writable.

Figure A-5. Typical Endurance vs Temperature



A.6 SPI

This section provides electrical parametrics and ratings for the SPI.

In Table A-20 the measurement conditions are listed.

Table A-20. Measurement Conditions

Description	Value	Unit
Drive mode	Full drive mode	—
Load capacitance $C_{LOAD,}$ on all outputs	50	pF
Thresholds for delay measurement points	(20% / 80%) V_{DDX}	V

A.6.1 Master Mode

In Figure A-6 the timing diagram for master mode with transmission format CPHA=0 is depicted.