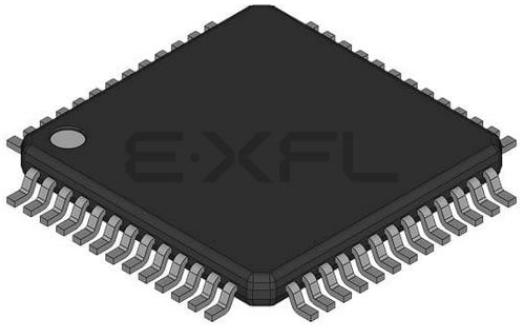

Welcome to E-XFL.COM

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	EBI/EMI, SCI, SPI
Peripherals	POR, PWM, WDT
Number of I/O	35
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K 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	52-LQFP
Supplier Device Package	52-QFP (10x10)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s12gc32mpbe

Chapter 3
Module Mapping Control (MMCV4) Block Description

Chapter 4
Multiplexed External Bus Interface (MEBIV3)

Table 1-9. Interrupt Vector Locations (continued)

Vector Address	Interrupt Source	CCR Mask	Local Enable	HPRIO Value to Elevate
0xFFDE, 0xFFDF	Standard timer overflow	I bit	TMSK2 (TOI)	0x00DE
0xFFDC, 0xFFDD	Pulse accumulator A overflow	I bit	PACTL (PAOVI)	0x00DC
0xFFDA, 0xFFDB	Pulse accumulator input edge	I bit	PACTL (PAI)	0x00DA
0xFFD8, 0xFFD9	SPI	I bit	SPICR1 (SPIE, SPTIE)	0x00D8
0xFFD6, 0xFFD7	SCI	I bit	SCICR2 (TIE, TCIE, RIE, ILIE)	0x00D6
0xFFD4, 0xFFD5	Reserved			
0xFFD2, 0xFFD3	ATD	I bit	ATDCTL2 (ASCIE)	0x00D2
0xFFD0, 0xFFD1	Reserved			
0xFFCE, 0xFFCF	Port J	I bit	PIEP (PIEP7-6)	0x00CE
0xFFCC, 0xFFCD	Reserved			
0xFFCA, 0xFFCB	Reserved			
0xFFC8, 0xFFC9	Reserved			
0xFFC6, 0xFFC7	CRG PLL lock	I bit	PLLCR (LOCKIE)	0x00C6
0xFFC4, 0xFFC5	CRG self clock mode	I bit	PLLCR (SCMIE)	0x00C4
0xFFBA to 0xFFC3	Reserved			
0xFFB8, 0xFFB9	FLASH	I bit	FCNFG (CCIE, CBEIE)	0x00B8
0xFFB6, 0xFFB7	CAN wake-up(1)	I bit	CANRIER (WUPIE)	0x00B6
0xFFB4, 0xFFB5	CAN errors[1]	I bit	CANRIER (CSCIE, OVRIE)	0x00B4
0xFFB2, 0xFFB3	CAN receive[1]	I bit	CANRIER (RXFIE)	0x00B2
0xFFB0, 0xFFB1	CAN transmit[1]	I bit	CANTIER (TXEIE[2:0])	0x00B0
0xFF90 to 0xFFAF	Reserved			
0xFF8E, 0xFF8F	Port P	I bit	PIEP (PIEP7-0)	0x008E
0xFF8C, 0xFF8D	Reserved			
0xFF8C, 0xFF8D	PWM Emergency Shutdown	I bit	PWMSDN(PWMIE)	0x008C
0xFF8A, 0xFF8B	VREG LVI	I bit	CTRL0 (LVIE)	0x008A
0xFF80 to 0xFF89	Reserved			

1. Not available on MC9S12GC Family members

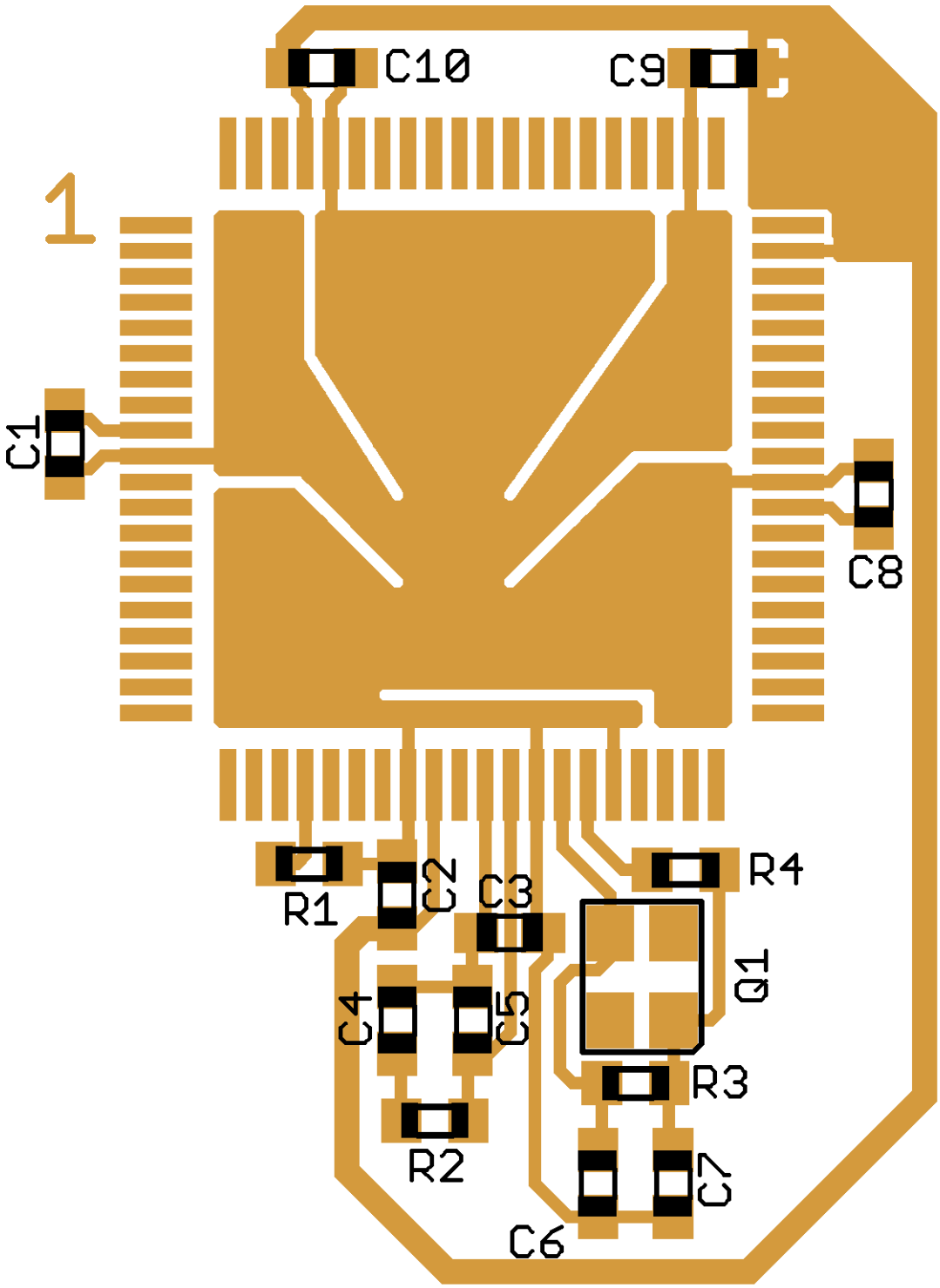

Figure 1-20. Recommended PCB Layout for 80QFP Pierce Oscillator

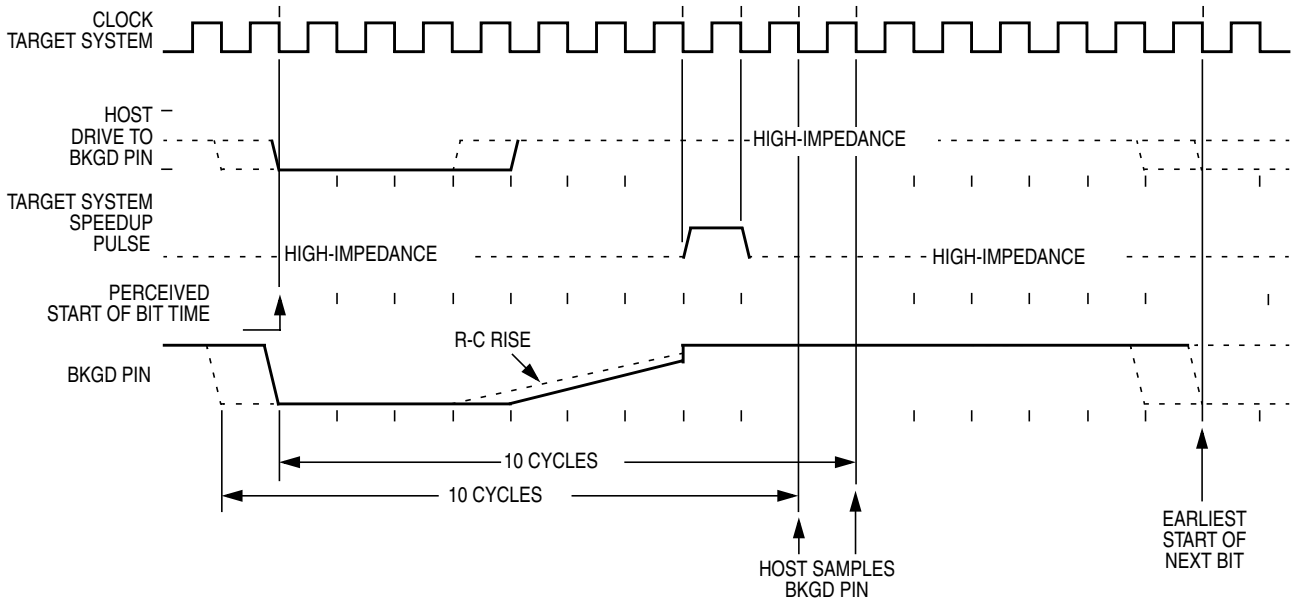

Figure 6-8. BDM Target-to-Host Serial Bit Timing (Logic 1)

Figure 6-9 shows the host receiving a logic 0 from the target. Because the host is asynchronous to the target, there is up to a one clock-cycle delay from the host-generated falling edge on BKGD to the start of the bit time as perceived by the target. The host initiates the bit time but the target finishes it. Because the target wants the host to receive a logic 0, it drives the BKGD pin low for 13 target clock cycles then briefly drives it high to speed up the rising edge. The host samples the bit level about 10 target clock cycles after starting the bit time.

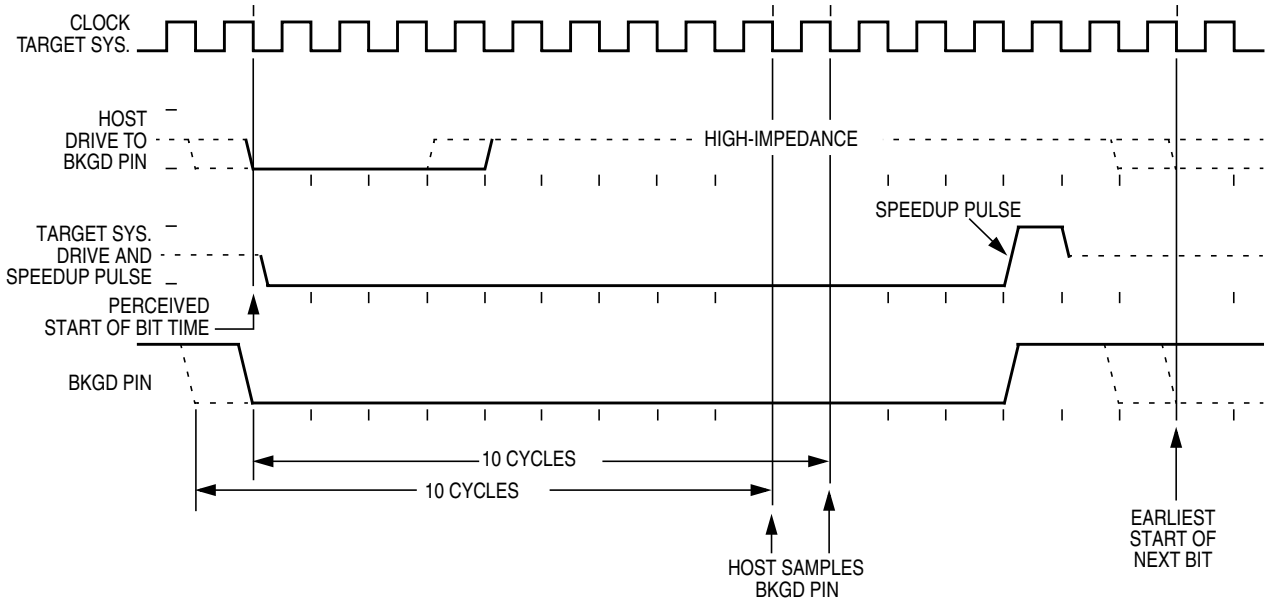

Figure 6-9. BDM Target-to-Host Serial Bit Timing (Logic 0)

Chapter 7
Debug Module (DBGV1) Block Description

7.1 Introduction

This section describes the functionality of the debug (DBG) sub-block of the HCS12 core platform.

The DBG module is designed to be fully compatible with the existing BKP_HCS12_A module (BKP mode) and furthermore provides an on-chip trace buffer with flexible triggering capability (DBG mode). The DBG module provides for non-intrusive debug of application software. The DBG module is optimized for the HCS12 16-bit architecture.

7.1.1 Features

The DBG module in BKP mode includes these distinctive features:

- Full or dual breakpoint mode
  - Compare on address and data (full)
  - Compare on either of two addresses (dual)
- BDM or SWI breakpoint
  - Enter BDM on breakpoint (BDM)
  - Execute SWI on breakpoint (SWI)
- Tagged or forced breakpoint
  - Break just before a specific instruction will begin execution (TAG)
  - Break on the first instruction boundary after a match occurs (Force)
- Single, range, or page address compares
  - Compare on address (single)
  - Compare on address 256 byte (range)
  - Compare on any 16K page (page)
- At forced breakpoints compare address on read or write
- High and/or low byte data compares
- Comparator C can provide an additional tag or force breakpoint (enhancement for BKP mode)

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
Left Justified Result Data										
0x0010	ATDDR0H	R	BIT 9 MSB BIT 7 MSB	BIT 8 BIT 6	BIT 7 BIT 5	BIT 6 BIT 4	BIT 5 BIT 3	BIT 4 BIT 2	BIT 3 BIT 1	BIT 2 BIT 0
		W								
0x0011	ATDDR0L	R	BIT 1 u	BIT 0 u	0 0	0 0	0 0	0 0	0 0	0 0
		W								
0x0012	ATDDR1H	R	BIT 9 MSB BIT 7 MSB	BIT 8 BIT 6	BIT 7 BIT 5	BIT 6 BIT 4	BIT 5 BIT 3	BIT 4 BIT 2	BIT 3 BIT 1	BIT 2 BIT 0
		W								
0x0013	ATDDR1L	R	BIT 1 u	BIT 0 u	0 0	0 0	0 0	0 0	0 0	0 0
		W								
0x0014	ATDDR2H	R	BIT 9 MSB BIT 7 MSB	BIT 8 BIT 6	BIT 7 BIT 5	BIT 6 BIT 4	BIT 5 BIT 3	BIT 4 BIT 2	BIT 3 BIT 1	BIT 2 BIT 0
		W								
0x0015	ATDDR2L	R	BIT 1 u	BIT 0 u	0 0	0 0	0 0	0 0	0 0	0 0
		W								
0x0016	ATDDR3H	R	BIT 9 MSB BIT 7 MSB	BIT 8 BIT 6	BIT 7 BIT 5	BIT 6 BIT 4	BIT 5 BIT 3	BIT 4 BIT 2	BIT 3 BIT 1	BIT 2 BIT 0
		W								
0x0017	ATDDR3L	R	BIT 1 u	BIT 0 u	0 0	0 0	0 0	0 0	0 0	0 0
		W								
0x0018	ATDDR4H	R	BIT 9 MSB BIT 7 MSB	BIT 8 BIT 6	BIT 7 BIT 5	BIT 6 BIT 4	BIT 5 BIT 3	BIT 4 BIT 2	BIT 3 BIT 1	BIT 2 BIT 0
		W								
0x0019	ATDDR4L	R	BIT 1 u	BIT 0 u	0 0	0 0	0 0	0 0	0 0	0 0
		W								
0x001A	ATDDR5H	R	BIT 9 MSB BIT 7 MSB	BIT 8 BIT 6	BIT 7 BIT 5	BIT 6 BIT 4	BIT 5 BIT 3	BIT 4 BIT 2	BIT 3 BIT 1	BIT 2 BIT 0
		W								
0x001B	ATDDR5L	R	BIT 1 u	BIT 0 u	0 0	0 0	0 0	0 0	0 0	0 0
		W								
0x001C	ATDDR6H	R	BIT 9 MSB BIT 7 MSB	BIT 8 BIT 6	BIT 7 BIT 5	BIT 6 BIT 4	BIT 5 BIT 3	BIT 4 BIT 2	BIT 3 BIT 1	BIT 2 BIT 0
		W								
0x001D	ATDDR6L	R	BIT 1 u	BIT 0 u	0 0	0 0	0 0	0 0	0 0	0 0
		W								

= Unimplemented or Reserved

Figure 8-2. ATD Register Summary (Sheet 2 of 4)

8.5.1.3 Step 3

Configure how many conversions you want to perform in one sequence and define other settings in ATDCTL3.

Example: Write S4C=1 to do 4 conversions per sequence.

8.5.1.4 Step 4

Configure resolution, sampling time and ATD clock speed in ATDCTL4.

Example: Use default for resolution and sampling time by leaving SRES8, SMP1 and SMP0 clear. For a bus clock of 40MHz write 9 to PR4-0, this gives an ATD clock of 0.5*40MHz/(9+1) = 2MHz which is within the allowed range for f_{ATDCLK}.

8.5.1.5 Step 5

Configure starting channel, single/multiple channel, continuous or single sequence and result data format in ATDCTL5. Writing ATDCTL5 will start the conversion, so make sure your write ATDCTL5 in the last step.

Example: Leave CC,CB,CA clear to start on channel AN0. Write MULT=1 to convert channel AN0 to AN3 in a sequence (4 conversion per sequence selected in ATDCTL3).

8.5.2 Aborting an A/D conversion

8.5.2.1 Step 1

Disable the ATD Interrupt by writing ASCIE=0 in ATDCTL2. This will also abort any ongoing conversion sequence.

It is important to clear the interrupt enable at this point, prior to step 3, as depending on the device clock gating it may not always be possible to clear it or the SCF flag once the module is disabled (ADPU=0).

8.5.2.2 Step 2

Clear the SCF flag by writing a 1 in ATDSTAT0.

(Remaining flags will be cleared with the next start of a conversions, but SCF flag should be cleared to avoid SCF interrupt.)

8.5.2.3 Step 3

Power down ATD by writing ADPU=0 in ATDCTL2.

8.6 Resets

At reset the ATD10B8C is in a power down state. The reset state of each individual bit is listed within Section 8.3.2, “Register Descriptions” which details the registers and their bit-field.

9.1.2 Modes of Operation

This subsection lists and briefly describes all operating modes supported by the CRG.

- Run mode

  All functional parts of the CRG are running during normal run mode. If RTI or COP functionality is required the individual bits of the associated rate select registers (COPCTL, RTICTL) have to be set to a nonzero value.

- Wait mode

  This mode allows to disable the system and core clocks depending on the configuration of the individual bits in the CLKSEL register.

- Stop mode

  Depending on the setting of the PSTP bit, stop mode can be differentiated between full stop mode (PSTP = 0) and pseudo-stop mode (PSTP = 1).

  — Full stop mode

  The oscillator is disabled and thus all system and core clocks are stopped. The COP and the RTI remain frozen.

  — Pseudo-stop mode

  The oscillator continues to run and most of the system and core clocks are stopped. If the respective enable bits are set the COP and RTI will continue to run, else they remain frozen.

- Self-clock mode

  Self-clock mode will be entered if the clock monitor enable bit (CME) and the self-clock mode enable bit (SCME) are both asserted and the clock monitor in the oscillator block detects a loss of clock. As soon as self-clock mode is entered the CRGV4 starts to perform a clock quality check. Self-clock mode remains active until the clock quality check indicates that the required quality of the incoming clock signal is met (frequency and amplitude). Self-clock mode should be used for safety purposes only. It provides reduced functionality to the MCU in case a loss of clock is causing severe system conditions.

9.1.3 Block Diagram

Figure 9-1 shows a block diagram of the CRGV4.

Module Base + 0x00X3

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

= Unused; always read 'x'

Figure 10-32. Identifier Register 3 — Standard Mapping

10.3.3.2 Data Segment Registers (DSR0-7)

The eight data segment registers, each with bits DB[7:0], contain the data to be transmitted or received. The number of bytes to be transmitted or received is determined by the data length code in the corresponding DLR register.

Module Base + 0x0004 (DSR0)
0x0005 (DSR1)
0x0006 (DSR2)
0x0007 (DSR3)
0x0008 (DSR4)
0x0009 (DSR5)
0x000A (DSR6)
0x000B (DSR7)

	7	6	5	4	3	2	1	0
R W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Reset:	x	x	x	x	x	x	x	x

Figure 10-33. Data Segment Registers (DSR0–DSR7) — Extended Identifier Mapping

Table 10-30. DSR0–DSR7 Register Field Descriptions

Field	Description
7:0 DB[7:0]	**Data bits 7:0**

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

In master mode, with slave select output enabled the $\overline{SS}$ line is always deasserted and reasserted between successive transfers for at least minimum idle time.

14.4.3.3 CPHA = 1 Transfer Format

Some peripherals require the first SCK edge before the first data bit becomes available at the data out pin, the second edge clocks data into the system. In this format, the first SCK edge is issued by setting the CPHA bit at the beginning of the 8-cycle transfer operation.

The first edge of SCK occurs immediately after the half SCK clock cycle synchronization delay. This first edge commands the slave to transfer its first data bit to the serial data input pin of the master.

A half SCK cycle later, the second edge appears on the SCK pin. This is the latching edge for both the master and slave.

When the third edge occurs, the value previously latched from the serial data input pin is shifted into the LSB or MSB of the SPI shift register, depending on LSBFE bit. After this edge, the next bit of the master data is coupled out of the serial data output pin of the master to the serial input pin on the slave.

This process continues for a total of 16 edges on the SCK line with data being latched on even numbered edges and shifting taking place on odd numbered edges.

Data reception is double buffered, data is serially shifted into the SPI shift register during the transfer and is transferred to the parallel SPI Data Register after the last bit is shifted in.

After the 16th SCK edge:

- Data that was previously in the SPI Data Register of the master is now in the data register of the slave, and data that was in the data register of the slave is in the master.
- The SPIF flag bit in SPISR is set indicating that the transfer is complete.

Figure 14-10 shows two clocking variations for CPHA = 1. The diagram may be interpreted as a master or slave timing diagram because the SCK, MISO, and MOSI pins are connected directly between the master and the slave. The MISO signal is the output from the slave, and the MOSI signal is the output from the master. The $\overline{SS}$ line is the slave select input to the slave. The $\overline{SS}$ pin of the master must be either high or reconfigured as a general-purpose output not affecting the SPI.

The $\overline{SS}$ line can remain active low between successive transfers (can be tied low at all times). This format is sometimes preferred in systems having a single fixed master and a single slave that drive the MISO data line.

- Back-to-back transfers in master mode

In master mode, if a transmission has completed and a new data byte is available in the SPI Data Register, this byte is send out immediately without a trailing and minimum idle time.

The SPI interrupt request flag (SPIF) is common to both the master and slave modes. SPIF gets set one half SCK cycle after the last SCK edge.

15.3.2.15 16-Bit Pulse Accumulator Control Register (PACTL)

Module Base + 0x0020

	7	6	5	4	3	2	1	0
R	0	PAEN	PAMOD	PEDGE	CLK1	CLK0	PAOVI	PAI
W								
Reset	0	0	0	0	0	0	0	0

Unimplemented or Reserved

Figure 15-24. 16-Bit Pulse Accumulator Control Register (PACTL)

When PAEN is set, the PACT is enabled.The PACT shares the input pin with IOC7.

Read: Any time

Write: Any time

Table 15-19. PACTL Field Descriptions

Field	Description
6 PAEN	**Pulse Accumulator System Enable** — PAEN is independent from TEN. With timer disabled, the pulse accumulator can function unless pulse accumulator is disabled. 0 16-Bit Pulse Accumulator system disabled. 1 Pulse Accumulator system enabled.
5 PAMOD	**Pulse Accumulator Mode** — This bit is active only when the Pulse Accumulator is enabled (PAEN = 1). See Table 15-20. 0 Event counter mode. 1 Gated time accumulation mode.
4 PEDGE	**Pulse Accumulator Edge Control** — This bit is active only when the Pulse Accumulator is enabled (PAEN = 1). For PAMOD bit = 0 (event counter mode). See Table 15-20. 0 Falling edges on IOC7 pin cause the count to be incremented. 1 Rising edges on IOC7 pin cause the count to be incremented. For PAMOD bit = 1 (gated time accumulation mode). 0 IOC7 input pin high enables M (bus clock) divided by 64 clock to Pulse Accumulator and the trailing falling edge on IOC7 sets the PAIF flag. 1 IOC7 input pin low enables M (bus clock) divided by 64 clock to Pulse Accumulator and the trailing rising edge on IOC7 sets the PAIF flag.
3:2 CLK[1:0]	**Clock Select Bits —** Refer to Table 15-21.
1 PAOVI	**Pulse Accumulator Overflow Interrupt Enable** 0 Interrupt inhibited. 1 Interrupt requested if PAOVF is set.
0 PAI	**Pulse Accumulator Input Interrupt Enable** 0 Interrupt inhibited. 1 Interrupt requested if PAIF is set.

17.1.3 Modes of Operation

See Section 17.4.2, “Operating Modes” for a description of the Flash module operating modes. For program and erase operations, refer to Section 17.4.1, “Flash Command Operations”.

17.1.4 Block Diagram

Figure 17-1 shows a block diagram of the FTS16K module.

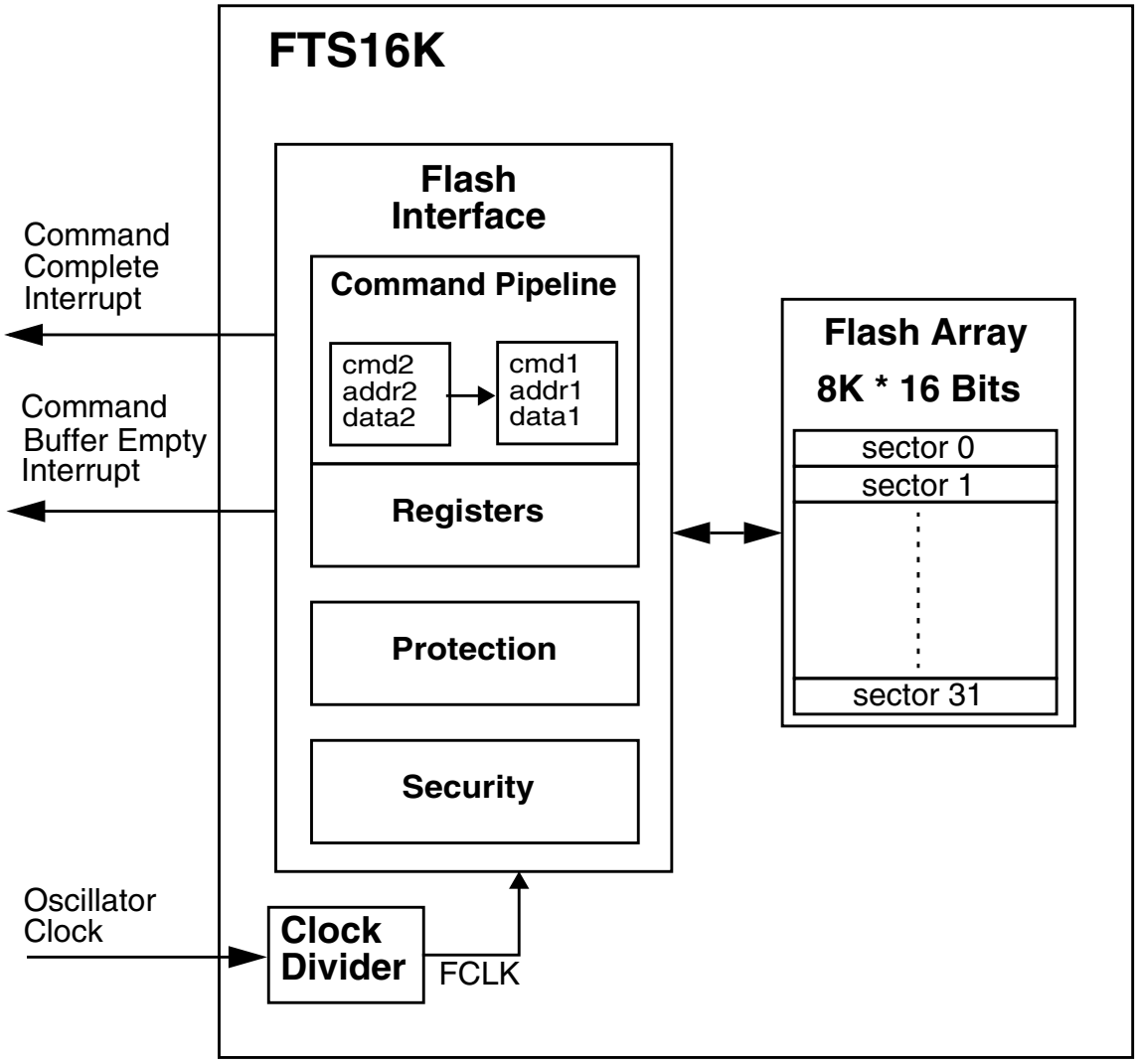

Figure 17-1. FTS16K Block Diagram

17.2 External Signal Description

The FTS16K module contains no signals that connect off-chip.

17.3 Memory Map and Registers

This section describes the FTS16K memory map and registers.

17.3.1 Module Memory Map

The FTS16K memory map is shown in Figure 17-2. The HCS12 architecture places the Flash array addresses between 0xC000 and 0xFFFF. The content of the HCS12 Core PPAGE register is used to map the logical page ranging from address 0x8000 to 0xBFFF to a physical 16K byte page in the Flash array memory.[1] The FPROT register (see Section 17.3.2.5) can be set to globally protect the entire Flash array or one growing downward from the Flash array end address. The higher address area is mainly targeted to hold the boot loader code since it covers the vector space. Default protection settings as well as security information that allows the MCU to restrict access to the Flash module are stored in the Flash configuration field described in Table 17-1.

Table 17-1. Flash Configuration Field

Flash Address	Size (bytes)	Description
0xFF00–0xFF07	8	Backdoor Key to unlock security
0xFF08–0xFF0C	5	Reserved
0xFF0D	1	Flash Protection byte Refer to Section 17.3.2.5, “Flash Protection Register (FPROT)”
0xFF0E	1	Reserved
0xFF0F	1	Flash Security/Options byte Refer to Section 17.3.2.2, “Flash Security Register (FSEC)”

1. By placing 0x3F in the HCS12 Core PPAGE register, the 16 Kbyte page can be seen twice in the MCU memory map.

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 18-17. RESERVED3

All bits read 0 and are not writable.

18.3.2.12 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 18-18. RESERVED4

All bits read 0 and are not writable.

18.3.2.13 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 18-19. RESERVED5

All bits read 0 and are not writable.

18.3.2.14 RESERVED6

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

19.4.1.2 Command Write Sequence

The Flash command controller is used to supervise the command write sequence to execute program, erase, and erase verify algorithms.

Before starting a command write sequence, the ACCERR and PVIOL flags in the FSTAT register must be clear and the CBEIF flag should be tested to determine the state of the address, data, and command buffers. If the CBEIF flag is set, indicating the buffers are empty, a new command write sequence can be started. If the CBEIF flag is clear, indicating the buffers are not available, a new command write sequence will overwrite the contents of the address, data, and command buffers.

A command write sequence consists of three steps which must be strictly adhered to with writes to the Flash module not permitted between the steps. However, Flash register and array reads are allowed during a command write sequence. The basic command write sequence is as follows:

1. Write to a valid address in the Flash array memory.
2. Write a valid command to the FCMD register.
3. Clear the CBEIF flag in the FSTAT register by writing a 1 to CBEIF to launch the command.

The address written in step 1 will be stored in the FADDR registers and the data will be stored in the FDATA registers. When the CBEIF flag is cleared in step 3, the CCIF flag is cleared by the Flash command controller indicating that the command was successfully launched. For all command write sequences, the CBEIF flag will set after the CCIF flag is cleared indicating that the address, data, and command buffers are ready for a new command write sequence to begin. A buffered command will wait for the active operation to be completed before being launched. Once a command is launched, the completion of the command operation is indicated by the setting of the CCIF flag in the FSTAT register. The CCIF flag will set upon completion of all active and buffered commands.

21.4.1.2 Command Write Sequence

The Flash command controller is used to supervise the command write sequence to execute program, erase, and erase verify algorithms.

Before starting a command write sequence, the ACCERR and PVIOL flags in the FSTAT register must be clear and the CBEIF flag should be tested to determine the state of the address, data, and command buffers. If the CBEIF flag is set, indicating the buffers are empty, a new command write sequence can be started. If the CBEIF flag is clear, indicating the buffers are not available, a new command write sequence will overwrite the contents of the address, data, and command buffers.

A command write sequence consists of three steps which must be strictly adhered to with writes to the Flash module not permitted between the steps. However, Flash register and array reads are allowed during a command write sequence. The basic command write sequence is as follows:

1. Write to a valid address in the Flash array memory.
2. Write a valid command to the FCMD register.
3. Clear the CBEIF flag in the FSTAT register by writing a 1 to CBEIF to launch the command.

The address written in step 1 will be stored in the FADDR registers and the data will be stored in the FDATA registers. When the CBEIF flag is cleared in step 3, the CCIF flag is cleared by the Flash command controller indicating that the command was successfully launched. For all command write sequences, the CBEIF flag will set after the CCIF flag is cleared indicating that the address, data, and command buffers are ready for a new command write sequence to begin. A buffered command will wait for the active operation to be completed before being launched. Once a command is launched, the completion of the command operation is indicated by the setting of the CCIF flag in the FSTAT register. The CCIF flag will set upon completion of all active and buffered commands.

A.1.7 Operating Conditions

This chapter describes the operating conditions of the devices. Unless otherwise noted those conditions apply to all the following data.

NOTE

Instead of specifying ambient temperature all parameters are specified for the more meaningful silicon junction temperature. For power dissipation calculations refer to Section A.1.8, “Power Dissipation and Thermal Characteristics”

Table A-4. Operating Conditions

Rating	Symbol	Min	Typ	Max	Unit
I/O, Regulator and Analog Supply Voltage	V_{DD5}	2.97	5	5.5	V
Digital Logic Supply Voltage (1)	V_{DD}	2.35	2.5	2.75	V
PLL Supply Voltage 1	V_{DDPLL}	2.35	2.5	2.75	V
Voltage Difference V_{DDX} to V_{DDA}	Δ_{VDDX}	–0.1	0	0.1	V
Voltage Difference V_{SSX} to V_{SSR} and V_{SSA}	Δ_{VSSX}	–0.1	0	0.1	V
Bus Frequency	f_{bus} (2)	0.25	—	25	MHz
Bus Frequency	f_{bus} (3)	0.25	—	16	MHz
Operating Junction Temperature Range	T_J	–40	—	140	°C

1. The device contains an internal voltage regulator to generate the logic and PLL supply out of the I/O supply. The operating conditions apply when this regulator is disabled and the device is powered from an external source.
   Using an external regulator, with the internal voltage regulator disabled, an external LVR must be provided.
2. Some blocks e.g. ATD (conversion) and NVMs (program/erase) require higher bus frequencies for proper operation.
3. Some blocks e.g. ATD (conversion) and NVMs (program/erase) require higher bus frequencies for proper operation.

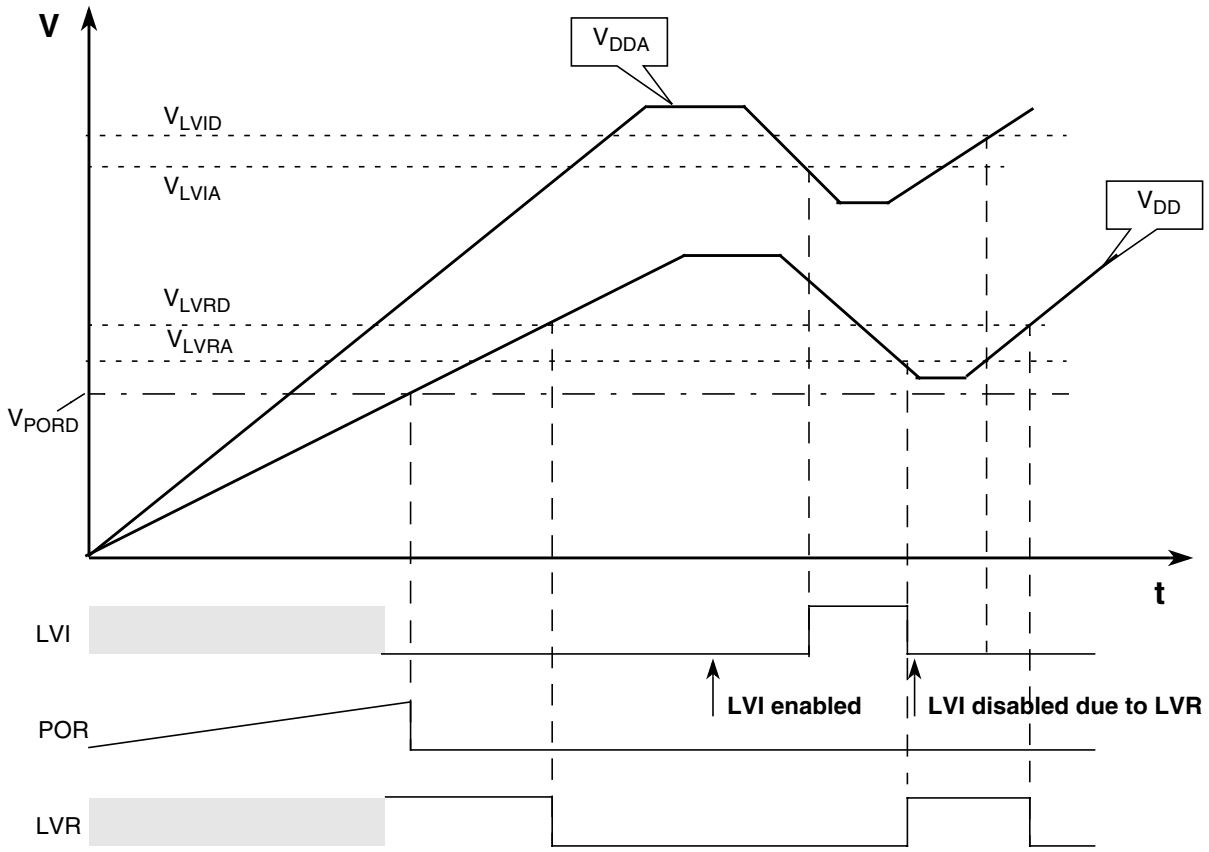

Figure A-10. Voltage Regulator — Chip Power-up and Voltage Drops (not scaled)

A.7.3 Output Loads

A.7.3.1 Resistive Loads

The on-chip voltage regulator is intended to supply the internal logic and oscillator circuits allows no external DC loads.

A.7.3.2 Capacitive Loads

The capacitive loads are specified in Table A-24. Ceramic capacitors with X7R dielectricum are required.

Table A-24. Voltage Regulator — Capacitive Loads

Num	Characteristic	Symbol	Min	Typical	Max	Unit
1	V_{DD} external capacitive load	C_{DDext}	400	440	12000	nF
2	V_{DDPLL} external capacitive load	$C_{DDPLLext}$	90	220	5000	nF