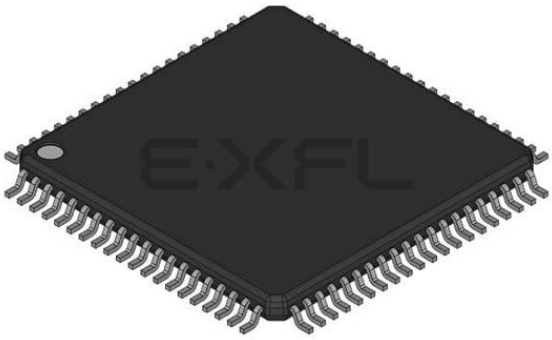

Welcome to E-XFL.COM

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	60
Program Memory Size	64KB (64K 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 ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	80-QFP
Supplier Device Package	80-QFP (14x14)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s12c64vfue

Chapter 5
Interrupt (INTV1) Block Description

Chapter 6
Background Debug Module (BDMV4) Block Description

Chapter 7
Debug Module (DBGV1) Block Description

Chapter 8
Analog-to-Digital Converter (ATD10B8C) Block Description

0x0034–0x003F CRG (Clock and Reset Generator)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0034	SYNR	Read:	0	0	SYN5	SYN4	SYN3	SYN2	SYN1	SYN0
		Write:								
0x0035	REFDV	Read:	0	0	0	0	REFDV3	REFDV2	REFDV1	REFDV0
		Write:								
0x0036	CTFLG TEST ONLY	Read:	TOUT7	TOUT6	TOUT5	TOUT4	TOUT3	TOUT2	TOUT1	TOUT0
		Write:								
0x0037	CRGFLG	Read:	RTIF	PORF	LVRF	LOCKIF	LOCK	TRACK	SCMIF	SCM
		Write:								
0x0038	CRGINT	Read:	RTIE	0	0	LOCKIE	0	0	SCMIE	0
		Write:								
0x0039	CLKSEL	Read:	PLLSEL	PSTP	SYSWAI	ROAWAI	PLLWAI	CWAI	RTIWAI	COPWAI
		Write:								
0x003A	PLLCTL	Read:	CME	PLLON	AUTO	ACQ	0	PRE	PCE	SCME
		Write:								
0x003B	RTICTL	Read:	0	RTR6	RTR5	RTR4	RTR3	RTR2	RTR1	RTR0
		Write:								
0x003C	COPCTL	Read:	WCOP	RSBCK	0	0	0	CR2	CR1	CR0
		Write:								
0x003D	FORBYP TEST ONLY	Read:	RTIBYP	COPBYP	0	PLLBYP	0	0	FCM	0
		Write:								
0x003E	CTCTL TEST ONLY	Read:	TCTL7	TCTL6	TCTL5	TCTL4	TCLT3	TCTL2	TCTL1	TCTL0
		Write:								
0x003F	ARMCOP	Read:	0	0	0	0	0	0	0	0
		Write:	Bit 7	6	5	4	3	2	1	Bit 0

0x0040-0x006F TIM

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0040	TIOS	Read:	IOS7	IOS6	IOS5	IOS4	IOS3	IOS2	IOS1	IOS0
		Write:								
0x0041	CFORC	Read:	0	0	0	0	0	0	0	0
		Write:	FOC7	FOC6	FOC5	FOC4	FOC3	FOC2	FOC1	FOC0
0x0042	OC7M	Read:	OC7M7	OC7M6	OC7M5	OC7M4	OC7M3	OC7M2	OC7M1	OC7M0
		Write:								
0x0043	OC7D	Read:	OC7D7	OC7D6	OC7D5	OC7D4	OC7D3	OC7D2	OC7D1	OC7D0
		Write:								
0x0044	TCNT (hi)	Read:	Bit 15	14	13	12	11	10	9	Bit 8
		Write:								
0x0045	TCNT (lo)	Read:	Bit 7	6	5	4	3	2	1	Bit 0
		Write:								
0x0046	TSCR1	Read:	TEN	TSWAI	TSFRZ	TFFCA	0	0	0	0
		Write:								
0x0047	TTOV	Read:	TOV7	TOV6	TOV5	TOV4	TOV3	TOV2	TOV1	TOV0
		Write:								

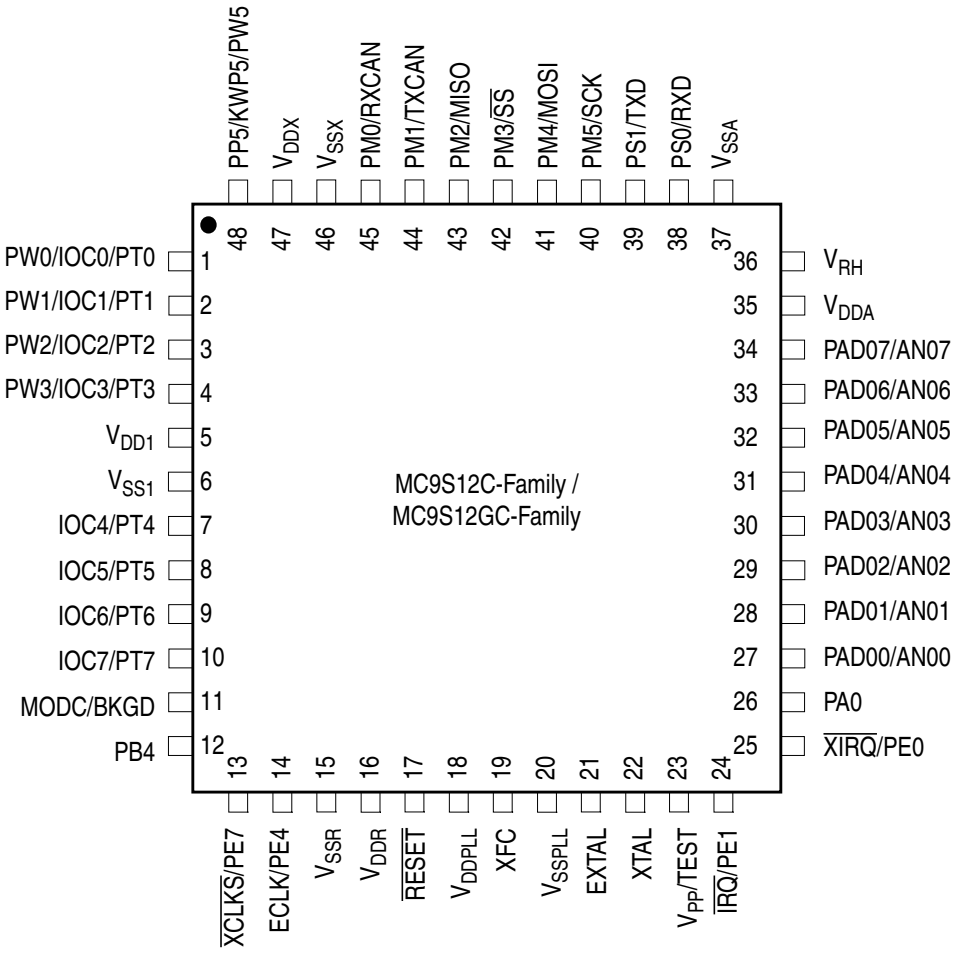

Figure 1-9. Pin Assignments in 48-Pin LQFP

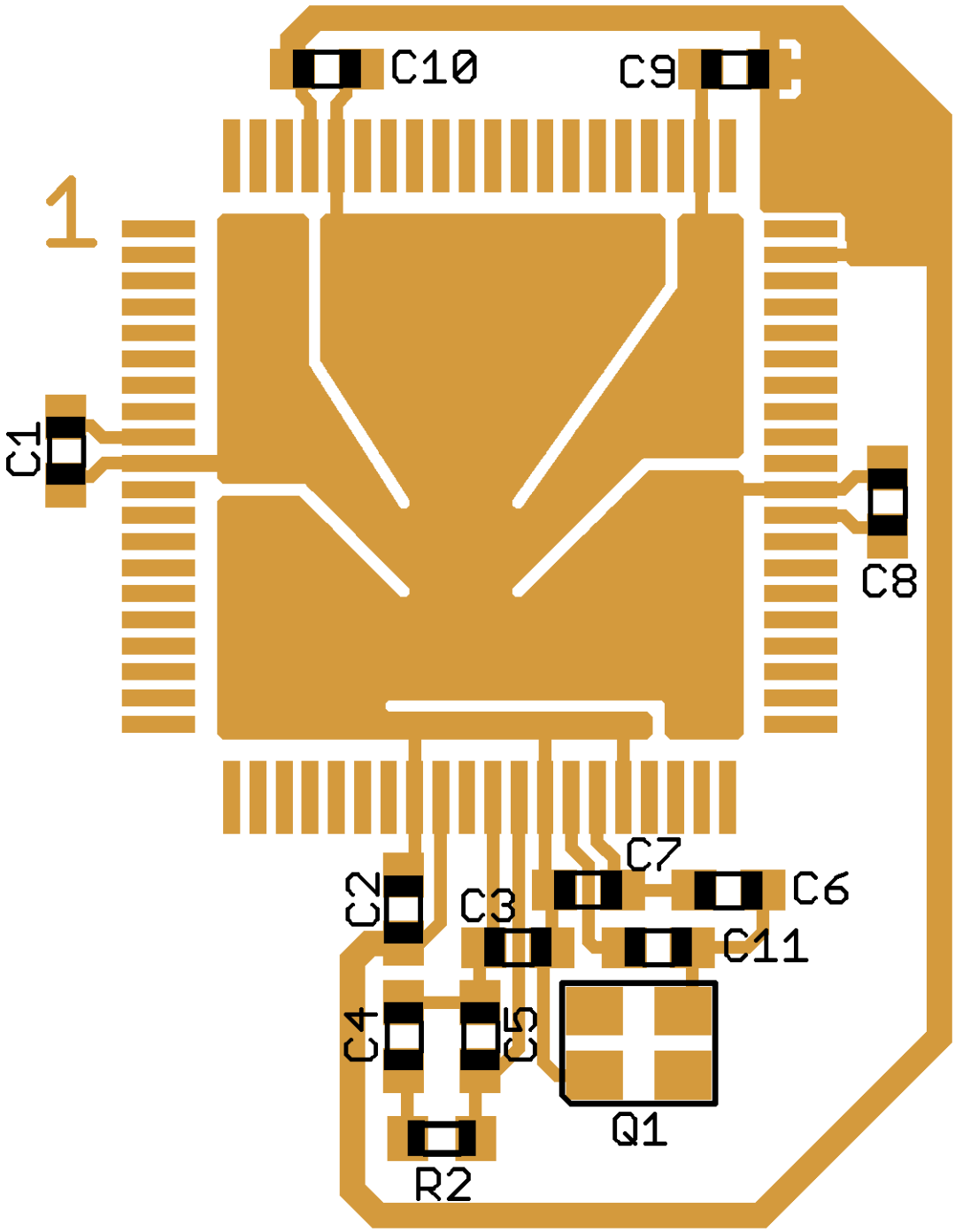

Figure 1-17. Recommended PCB Layout (80 QFP) Colpitts Oscillator

2.3.2.3.6 Port M Polarity Select Register (PPSM)

Module Base + 0x0015

	7	6	5	4	3	2	1	0
R	0	0	PPSM5	PPSM4	PPSM3	PPSM2	PPSM1	PPSM0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 2-22. Port M Polarity Select Register (PPSM)

Read: Anytime.

Write: Anytime.

Table 2-20. PPSM Field Descriptions

Field	Description
5–0 PPSM[5:0]	**Polarity Select Port M** — This register selects whether a pull-down or a pull-up device is connected to the pin. 0 A pull-up device is connected to the associated port M pin, if enabled by the associated bit in register PERM and if the port is used as input or as wired-or output. 1 A pull-down device is connected to the associated port M pin, if enabled by the associated bit in register PERM and if the port is used as input.

2.3.2.3.7 Port M Wired-OR Mode Register (WOMM)

Module Base + 0x0016

	7	6	5	4	3	2	1	0
R	0	0	WOMM5	WOMM4	WOMM3	WOMM2	WOMM1	WOMM0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 2-23. Port M Wired-OR Mode Register (WOMM)

Read: Anytime.

Write: Anytime.

Table 2-21. WOMM Field Descriptions

Field	Description
5–0 WOMM[5:0]	**Wired-OR Mode Port M** — This register configures the output pins as wired-or. If enabled the output is driven active low only (open-drain). A logic level of “1” is not driven. This bit has no influence on pins used as inputs. 0 Output buffers operate as push-pull outputs. 1 Output buffers operate as open-drain outputs.

4.3.2.5 Reserved Registers

Module Base + 0x0004
Starting address location affected by INITRG register setting.

	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 4-6. Reserved Register

Module Base + 0x0005
Starting address location affected by INITRG register setting.

	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 4-7. Reserved Register

Module Base + 0x0006
Starting address location affected by INITRG register setting.

	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 4-8. Reserved Register

Module Base + 0x0007
Starting address location affected by INITRG register setting.

	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 4-9. Reserved Register

Chapter 5
Interrupt (INTV1) Block Description

5.1 Introduction

This section describes the functionality of the interrupt (INT) sub-block of the S12 core platform.

A block diagram of the interrupt sub-block is shown in Figure 5-1.

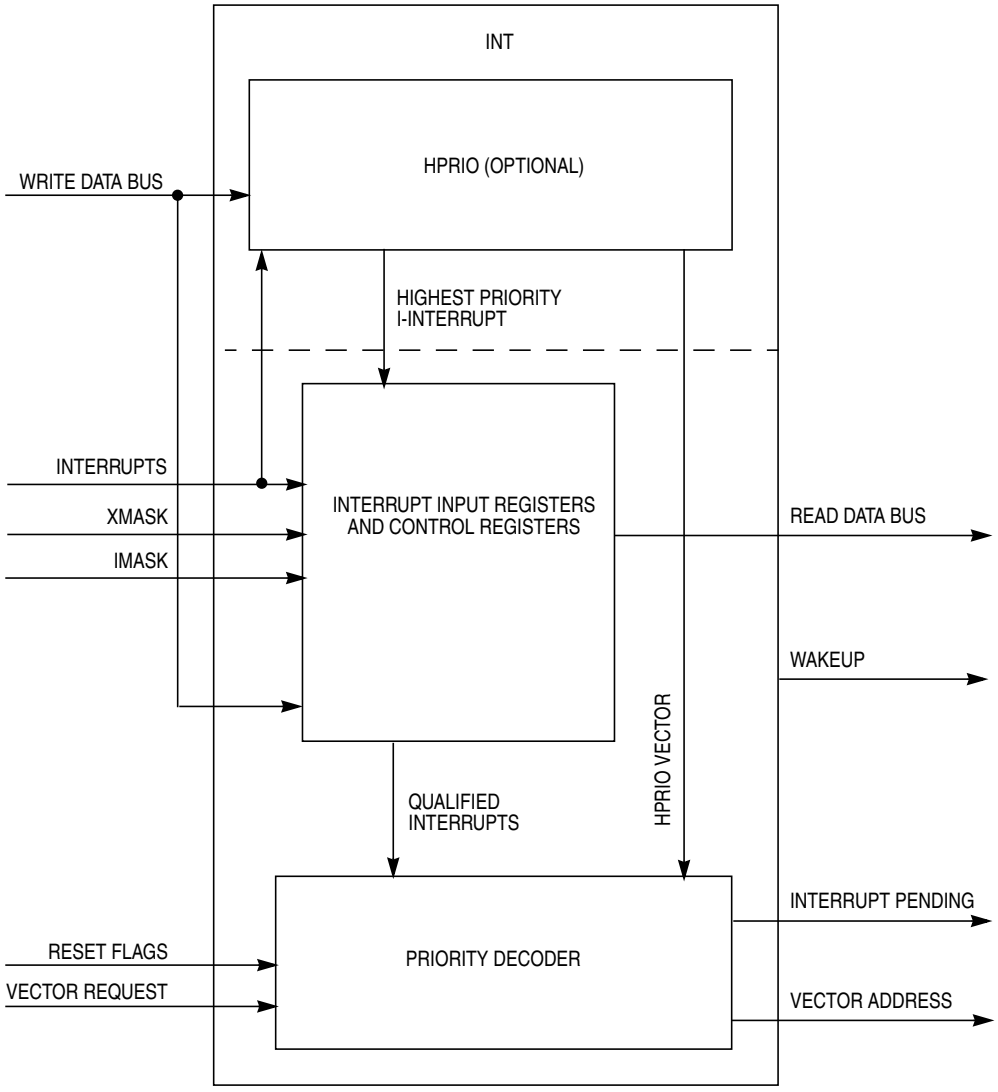

Figure 5-1. INTV1 Block Diagram

1. The DBG module is designed for backwards compatibility to existing BKP modules. Register and bit names have changed from the BKP module. This column shows the DBG register name, as well as the BKP register name for reference.
2. Comparator C can be used to enhance the BKP mode by providing a third breakpoint.

7.3.2.1 Debug Control Register 1 (DBGC1)

NOTE

All bits are used in DBG mode only.

Module Base + 0x0020
Starting address location affected by INITRG register setting.

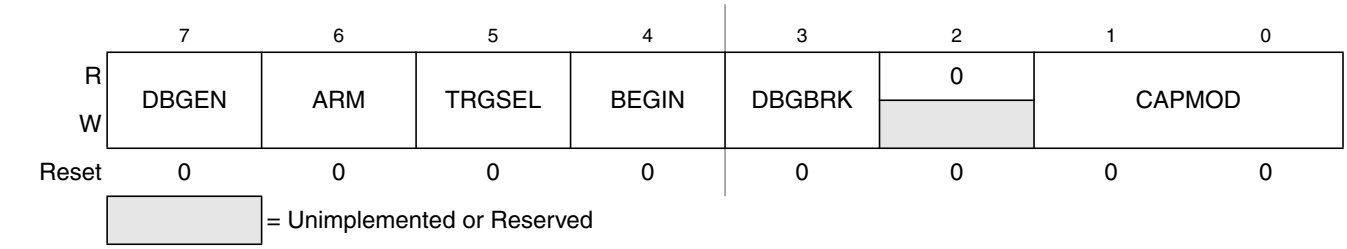

Figure 7-4. Debug Control Register (DBGC1)

NOTE

This register cannot be written if BKP mode is enabled (BKABEN in DBGC2 is set).

Table 7-3. DBGC1 Field Descriptions

Field	Description
7 DBGEN	**DBG Mode Enable Bit** — The DBGEN bit enables the DBG module for use in DBG mode. This bit cannot be set if the MCU is in secure mode. 0 DBG mode disabled 1 DBG mode enabled
6 ARM	**Arm Bit** — The ARM bit controls whether the debugger is comparing and storing data in the trace buffer. See Section 7.4.2.4, “Arming the DBG Module,” for more information. 0 Debugger unarmed 1 Debugger armed **Note:** This bit cannot be set if the DBGEN bit is not also being set at the same time. For example, a write of 01 to DBGEN[7:6] will be interpreted as a write of 00.
5 TRGSEL	**Trigger Selection Bit** — The TRGSEL bit controls the triggering condition for comparators A and B in DBG mode. It serves essentially the same function as the TAGAB bit in the DBGC2 register does in BKP mode. See Section 7.4.2.1.2, “Trigger Selection,” for more information. TRGSEL may also determine the type of breakpoint based on comparator A and B if enabled in DBG mode (DBGBRK = 1). Please refer to Section 7.4.3.1, “Breakpoint Based on Comparator A and B.” 0 Trigger on any compare address match 1 Trigger before opcode at compare address gets executed (tagged-type)
4 BEGIN	**Begin/End Trigger Bit** — The BEGIN bit controls whether the trigger begins or ends storing of data in the trace buffer. See Section 7.4.2.8.1, “Storing with Begin-Trigger,” and Section 7.4.2.8.2, “Storing with End-Trigger,” for more details. 0 Trigger at end of stored data 1 Trigger before storing data

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x001C	ATDDR6H	R	0 0	0 0	0 0	0 0	0 0	0 0	BIT 9 MSB 0	BIT 8 0
		W								
0x001D	ATDDR6L	R	BIT 7 BIT 7 MSB	BIT 6 BIT 6	BIT 5 BIT 5	BIT 4 BIT 4	BIT 3 BIT 3	BIT 2 BIT 2	BIT 1 BIT 1	BIT 0 BIT 0
		W								
0x001E	ATDDR7H	R	0 0	0 0	0 0	0 0	0 0	0 0	BIT 9 MSB 0	BIT 8 0
		W								
0x001F	ATDDR7L	R	BIT 7 BIT 7 MSB	BIT 6 BIT 6	BIT 5 BIT 5	BIT 4 BIT 4	BIT 3 BIT 3	BIT 2 BIT 2	BIT 1 BIT 1	BIT 0 BIT 0
		W								

= Unimplemented or Reserved

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

NOTE

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

8.3.2.7 ATD Status Register 0 (ATDSTAT0)

This read-only register contains the sequence complete flag, overrun flags for external trigger and FIFO mode, and the conversion counter.

Module Base + 0x0006

	7	6	5	4	3	2	1	0
R	SCF	0	ETORF	FIFOR	0	CC2	CC1	CC0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 8-9. ATD Status Register 0 (ATDSTAT0)

Read: Anytime

Write: Anytime (no effect on (CC2, CC1, CC0))

Table 8-13. ATDSTAT0 Field Descriptions

Field	Description
7 SCF	**Sequence Complete Flag** — This flag is set upon completion of a conversion sequence. If conversion sequences are continuously performed (SCAN = 1), the flag is set after each one is completed. This flag is cleared when one of the following occurs: A) Write “1” to SCF B) Write to ATDCTL5 (a new conversion sequence is started) C) If AFFC=1 and read of a result register 0 Conversion sequence not completed 1 Conversion sequence has completed
5 ETORF	**External Trigger Overrun Flag** — While in edge trigger mode (ETRIGLE = 0), if additional active edges are detected while a conversion sequence is in process the overrun flag is set. This flag is cleared when one of the following occurs: A) Write “1” to ETORF B) Write to ATDCTL2, ATDCTL3 or ATDCTL4 (a conversion sequence is aborted) C) Write to ATDCTL5 (a new conversion sequence is started) 0 No External trigger over run error has occurred 1 External trigger over run error has occurred

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.

9.3.2 Register Descriptions

This section describes in address order all the CRGV4 registers and their individual bits.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 SYNR	R	0	0	SYN5	SYN4	SYN3	SYN2	SYN1	SYN0
	W								
0x0001 REFDV	R	0	0	0	0	REFDV3	REFDV2	REFDV1	REFDV0
	W								
0x0002 CTFLG	R	0	0	0	0	0	0	0	0
	W								
0x0003 CRGFLG	R	RTIF	PORF	LVRF	LOCKIF	LOCK	TRACK	SCMIF	SCM
	W								
0x0004 CRGINT	R	RTIE	0	0	LOCKIE	0	0	SCMIE	0
	W								
0x0005 CLKSEL	R	PLLSEL	PSTP	SYSWAI	ROAWAI	PLLWAI	CWAI	RTIWAI	COPWAI
	W								
0x0006 PLLCTL	R	CME	PLLON	AUTO	ACQ	0	PRE	PCE	SCME
	W								
0x0007 RTICTL	R	0	RTR6	RTR5	RTR4	RTR3	RTR2	RTR1	RTR0
	W								
0x0008 COPCTL	R	WCOP	RSBCK	0	0	0	CR2	CR1	CR0
	W								
0x0009 FORBYP	R	0	0	0	0	0	0	0	0
	W								
0x000A CTCTL	R	0	0	0	0	0	0	0	0
	W								

= Unimplemented or Reserved

Figure 9-3. CRG Register Summary

10.3.2.16 MSCAN Identifier Acceptance Registers (CANIDAR0-7)

On reception, each message is written into the background receive buffer. The CPU is only signalled to read the message if it passes the criteria in the identifier acceptance and identifier mask registers (accepted); otherwise, the message is overwritten by the next message (dropped).

The acceptance registers of the MSCAN are applied on the IDR0–IDR3 registers (see Section 10.3.3.1, "Identifier Registers (IDR0–IDR3)") of incoming messages in a bit by bit manner (see Section 10.4.3, "Identifier Acceptance Filter").

For extended identifiers, all four acceptance and mask registers are applied. For standard identifiers, only the first two (CANIDAR0/1, CANIDMR0/1) are applied.

Module Base + 0x0010 (CANIDAR0)
0x0011 (CANIDAR1)
0x0012 (CANIDAR2)
0x0013 (CANIDAR3)

	7	6	5	4	3	2	1	0
R/W	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
Reset	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R/W	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
Reset	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R/W	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
Reset	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R/W	AC7	AC6	AC5	AC4	AC3	AC2	AC1	AC0
Reset	0	0	0	0	0	0	0	0

Figure 10-19. MSCAN Identifier Acceptance Registers (First Bank) — CANIDAR0–CANIDAR3

Read: Anytime

Write: Anytime in initialization mode (INITRQ = 1 and INITAK = 1)

Table 10-19. CANIDAR0–CANIDAR3 Register Field Descriptions

Field	Description
7:0 AC[7:0]	**Acceptance Code Bits** — AC[7:0] comprise a user-defined sequence of bits with which the corresponding bits of the related identifier register (IDRn) of the receive message buffer are compared. The result of this comparison is then masked with the corresponding identifier mask register.

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.

Table 15-5. OC7M Field Descriptions

Field	Description
7:0 OC7M[7:0]	**Output Compare 7 Mask** — Setting the OC7Mx (x ranges from 0 to 6) will set the corresponding port to be an output port when the corresponding TIOSx (x ranges from 0 to 6) bit is set to be an output compare. **Note:** A successful channel 7 output compare overrides any channel 6:0 compares. For each OC7M bit that is set, the output compare action reflects the corresponding OC7D bit.

15.3.2.4 Output Compare 7 Data Register (OC7D)

Module Base + 0x0003

	7	6	5	4	3	2	1	0
R W	OC7D7	OC7D6	OC7D5	OC7D4	OC7D3	OC7D2	OC7D1	OC7D0
Reset	0	0	0	0	0	0	0	0

Figure 15-9. Output Compare 7 Data Register (OC7D)

Read: Anytime

Write: Anytime

Table 15-6. OC7D Field Descriptions

Field	Description
7:0 OC7D[7:0]	**Output Compare 7 Data** — A channel 7 output compare can cause bits in the output compare 7 data register to transfer to the timer port data register depending on the output compare 7 mask register.

15.3.2.5 Timer Count Register (TCNT)

Module Base + 0x0004

	15	14	13	12	11	10	9	9
R W	TCNT15	TCNT14	TCNT13	TCNT12	TCNT11	TCNT10	TCNT9	TCNT8
Reset	0	0	0	0	0	0	0	0

Figure 15-10. Timer Count Register High (TCNTH)

Module Base + 0x0005

	7	6	5	4	3	2	1	0
R W	TCNT7	TCNT6	TCNT5	TCNT4	TCNT3	TCNT2	TCNT1	TCNT0
Reset	0	0	0	0	0	0	0	0

Figure 15-11. Timer Count Register Low (TCNTL)

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

Table 15-17. TRLG1 Field Descriptions

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

15.3.2.13 Main Timer Interrupt Flag 2 (TFLG2)

Module Base + 0x000F

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

Unimplemented or Reserved

Figure 15-21. Main Timer Interrupt Flag 2 (TFLG2)

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

Read: Anytime

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

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

Table 15-18. TRLG2 Field Descriptions

Field	Description
7 TOF	**Timer Overflow Flag** — Set when 16-bit free-running timer overflows from 0xFFFF to 0x0000. Clearing this bit requires writing a one to bit 7 of TFLG2 register while TEN bit of TSCR1 is set to one. (See also TCRE control bit explanation.)

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.

addresses sequentially staring with 0xFF00-0xFF01 and ending with 0xFF06–0xFF07. The values 0x0000 and 0xFFFF are not permitted as keys. When the KEYACC bit is set, reads of the Flash array will return invalid data.

The user code stored in the Flash array must have a method of receiving the backdoor key from an external stimulus. This external stimulus would typically be through one of the on-chip serial ports.

If KEYEN[1:0] = 1:0 in the FSEC register, the MCU can be unsecured by the backdoor key access sequence described below:

1. Set the KEYACC bit in the FCNFG register
2. Write the correct four 16-bit words to Flash addresses 0xFF00–0xFF07 sequentially starting with 0xFF00
3. Clear the KEYACC bit in the FCNFG register
4. If all four 16-bit words match the backdoor key stored in Flash addresses 0xFF00–0xFF07, the MCU is unsecured and bits SEC[1:0] in the FSEC register are forced to the unsecure state of 1:0

The backdoor key access sequence is monitored by the internal security state machine. An illegal operation during the backdoor key access sequence will cause the security state machine to lock, leaving the MCU in the secured state. A reset of the MCU will cause the security state machine to exit the lock state and allow a new backdoor key access sequence to be attempted. The following illegal operations will lock the security state machine:

1. If any of the four 16-bit words does not match the backdoor key programmed in the Flash array
2. If the four 16-bit words are written in the wrong sequence
3. If more than four 16-bit words are written
4. If any of the four 16-bit words written are 0x0000 or 0xFFFF
5. If the KEYACC bit does not remain set while the four 16-bit words are written

After the backdoor key access sequence has been correctly matched, the MCU will be unsecured. The Flash security byte can be programmed to the unsecure state, if desired.

In the unsecure state, the user has full control of the contents of the four word backdoor key by programming bytes 0xFF00–0xFF07 of the Flash configuration field.

The security as defined in the Flash security/options byte at address 0xFF0F is not changed by using the backdoor key access sequence to unsecure. The backdoor key stored in addresses 0xFF00–0xFF07 is unaffected by the backdoor key access sequence. After the next reset sequence, the security state of the Flash module is determined by the Flash security/options byte at address 0xFF0F. The backdoor key access sequence has no effect on the program and erase protection defined in the FPROT register.

It is not possible to unsecure the MCU in special single chip mode by executing the backdoor key access sequence in background debug mode.

A.2.5 ATD Accuracy (3.3V Range)

Table A-13 specifies the ATD conversion performance excluding any errors due to current injection, input capacitance and source resistance.

Table A-13. ATD Conversion Performance

Conditions are shown in Table A-4 unless otherwise noted
$V_{REF} = V_{RH} - V_{RL}$ = 3.328V. Resulting to one 8 bit count = 13mV and one 10 bit count = 3.25mV
f_{ATDCLK} = 2.0MHz

Num	C	Rating	Symbol	Min	Typ	Max	Unit
1	P	10-Bit Resolution	LSB	—	3.25	—	mV
2	P	10-Bit Differential Nonlinearity	DNL	–1.5	—	1.5	Counts
3	P	10-Bit Integral Nonlinearity	INL	–3.5	±1.5	3.5	Counts
4	P	10-Bit Absolute Error(1)	AE	–5	±2.5	5	Counts
5	P	8-Bit Resolution	LSB	—	13	—	mV
6	P	8-Bit Differential Nonlinearity	DNL	–0.5	—	0.5	Counts
7	P	8-Bit Integral Nonlinearity	INL	–1.5	±1	1.5	Counts
8	P	8-Bit Absolute Error[1]	AE	–2.0	±1.5	2.0	Counts

1. These values include the quantization error which is inherently 1/2 count for any A/D converter.

For the following definitions see also Figure A-1.

Differential Non-Linearity (DNL) is defined as the difference between two adjacent switching steps.

$$DNL(i) = \frac{V_i - V_{i-1}}{1LSB} - 1$$

The Integral Non-Linearity (INL) is defined as the sum of all DNLs:

$$INL(n) = \sum_{i=1}^{n} DNL(i) = \frac{V_n - V_0}{1LSB} - n$$

A.3 MSCAN

Table A-14. MSCAN Wake-up Pulse Characteristics

Conditions are shown in Table A-4 unless otherwise noted							
Num	**C**	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	P	MSCAN Wake-up dominant pulse filtered	t_{WUP}	—	—	2	us
2	P	MSCAN Wake-up dominant pulse pass	t_{WUP}	5	—	—	us

A.4 Reset, Oscillator and PLL

This section summarizes the electrical characteristics of the various startup scenarios for Oscillator and Phase-Locked-Loop (PLL).

A.4.1 Startup

Table A-15 summarizes several startup characteristics explained in this section. Detailed description of the startup behavior can be found in the Clock and Reset Generator (CRG) Block User Guide.

Table A-15. Startup Characteristics

Conditions are shown in Table A-4 unless otherwise noted							
Num	**C**	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	T	POR release level	V_{PORR}	—	—	2.07	V
2	T	POR assert level	V_{PORA}	0.97	—	—	V
3	D	Reset input pulse width, minimum input time	PW_{RSTL}	2	—	—	t_{osc}
4	D	Startup from Reset	n_{RST}	192	—	196	n_{osc}
5	D	Interrupt pulse width, $\overline{IRQ}$ edge-sensitive mode	PW_{IRQ}	20	—	—	ns
6	D	Wait recovery startup time	t_{WRS}	—	—	14	t_{cyc}

A.4.1.1 POR

The release level V_{PORR} and the assert level V_{PORA} are derived from the V_{DD} supply. They are also valid if the device is powered externally. After releasing the POR reset the oscillator and the clock quality check are started. If after a time t_{CQOUT} no valid oscillation is detected, the MCU will start using the internal self clock. The fastest startup time possible is given by n_{uposc}.

A.4.1.2 LVR

The release level V_{LVRR} and the assert level V_{LVRA} are derived from the V_{DD} supply. They are also valid if the device is powered externally. After releasing the LVR reset the oscillator and the clock quality check are started. If after a time t_{CQOUT} no valid oscillation is detected, the MCU will start using the internal self clock. The fastest startup time possible is given by n_{uposc}.