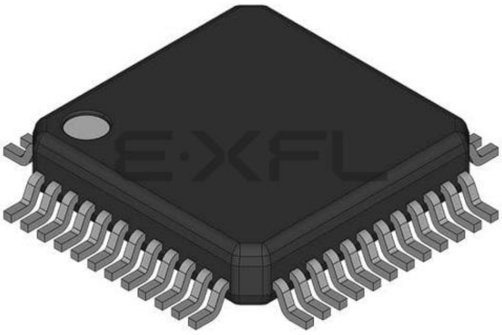
Freescale Semiconductor - MM912F634DV2AE Datasheet



Welcome to E-XFL.COM

Embedded - Microcontrollers - Application Specific: Tailored Solutions for Precision and Performance

Embedded - Microcontrollers - Application Specific represents a category of microcontrollers designed with unique features and capabilities tailored to specific application needs. Unlike general-purpose microcontrollers, application-specific microcontrollers are optimized for particular tasks, offering enhanced performance, efficiency, and functionality to meet the demands of specialized applications.

What Are Embedded - Microcontrollers - Application Specific?

Application specific microcontrollers are engineered to

Details	
Product Status	Active
Applications	Automotive
Core Processor	S12
Program Memory Type	FLASH (32KB)
Controller Series	HCS12
RAM Size	2K x 8
Interface	LIN, SCI
Number of I/O	9
Voltage - Supply	2.25V ~ 5.5V
Operating Temperature	-40°C ~ 105°C
Mounting Type	Surface Mount
Package / Case	48-LQFP Exposed Pad
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mm912f634dv2ae

4.2.3.1 Current Sense Module

For device options with the current sense module not available, the following considerations are to be made.

4.2.3.1.1 Pinout considerations

Table 69. ISENSE - Pin Considerations

PIN	PIN name for option 1	New PIN name	Comment
40	ISENSEL	NC	ISENSE feature not bonded and/or not tested. Connect PINs 40 and 41 (NC) to GND.
41	ISENSEH	NC	

4.2.3.1.2 Register Considerations

The Current Sense Register must remain in default (0x00) state.

Offset	Name		7	6	5	4	3	2	1	0
0x3C	CSR	R	CSE	0	0	0	CCD	CSGS		
	Current Sense Register	W								

The Conversion Control Register - Bit 9 must always be written 0.

0x82	ACCR (hi)	R	CH15	CH14	0	CH12	CH11	CH10	CH9	CH8
	ADC Conversion Ctrl Reg	W								

The Conversion Complete Register - Bit 9 must be ignored.

0x84	ACCSR (hi)	R	CC15	CC14	0	CC12	CC11	CC10	CC9	CC8
	ADC Conv Complete Reg	W								

The ADC Data Result Reg 9 must be ignored.

0x98	ADR9 (hi)	R	adr9 9	adr9 8	adr9 7	adr9 6	adr9 5	adr9 4	adr9 3	adr9 2
	ADC Data Result Register 9	W								
0x99	ADR9 (lo)	R	adr9 1	adr9 0	0	0	0	0	0	0
	ADC Data Result Register 9	W								

4.3.5 Sleep Mode

The Sleep mode will allow very low current consumption. In this mode, both voltage regulators (VDD and VDDX) are inactive.

The device can enter into Sleep mode by configuring the Mode Control Register (MCR) via the D2D- Interface. During Sleep mode, all unused internal blocks are deactivated to allow the lowest possible consumption. Power consumption will decrease further if the Cyclic Sense or Forced Wake-up feature are disabled. While in Sleep mode, the MM912F634 analog die will wake up on the following sources:

- Lx - Wake-up (maskable with selectable cyclic sense)
- Forced Wake-up (configurable timeout)
- LIN Wake-up

After Wake-up from the sources listed above or a reset condition, the device will transit to Reset mode.

See Section 4.8, “Wake-up / Cyclic Sense" for details.

4.3.6 Analog Die Functionality by Operation Mode

Table 71. Operation Mode Overview

Function	Reset	Normal	Stop	Sleep
VDD/VDDX	full	full	stop	OFF
HSUP	OFF	full	OFF	OFF
LSx		full	OFF	OFF
HSx		full	Cyclic Sense[62]	Cyclic Sense[62]
ADC		full	OFF	OFF
D2D		full	functional	OFF
Lx		full	Wake-up[62]	Wake-up[62]
PTBx		full	OFF	OFF
LIN		full	Wake-up[62]	Wake-up[62]
Watchdog		full[63]	OFF	OFF
VSENSE		full	OFF	OFF
CSENSE		full	OFF	OFF
Cyclic Sense		not active	Cyclic Sense[62]	Cyclic Sense[62]

Note:

[62] If configured.

[63] Special init through non window watchdog.

4.13 PWM Control Module (PWM8B2C)

4.13.1 Introduction

To control the High Side (HS1, HS2) and the Low Side (LS1, LS2) duty cycle as well as the PTB2 output, the PWM module is implemented. Refer to the individual driver section for details on the use of the internal PWM1 and PWM0 signal (Section 4.11, “High Side Drivers - HS", Section 4.12, “Low Side Drivers - LSx" and Section 4.17, “General Purpose I/O - PTB[0…2]")

The PWM definition is based on the HC12 PWM definitions with some of the simplifications incorporated. The PWM module has two channels with independent controls of left and center aligned outputs on each channel.

Each of the two channels has a programmable period and duty cycle as well as a dedicated counter. A flexible clock select scheme allows a total of four different clock sources to be used with the counters. Each of the modulators can create independent continuous waveforms with software-selectable duty rates from 0% to 100%.

4.13.1.1 Features

The PWM block includes these distinctive features:

- Two independent PWM channels with programmable periods and duty cycles
- Dedicated counter for each PWM channel
- Programmable PWM enable/disable for each channel
- Software selection of PWM duty pulse polarity for each channel
- Period and duty cycle are double buffered. Change takes effect when the end of the effective period is reached (PWM counter reaches zero), or when the channel is disabled
- Programmable center or left aligned outputs on individual channels
- Four clock sources (A, B, SA, and SB) provide for a wide range of frequencies
- Programmable clock select logic

4.13.1.2 Modes of Operation

The PWM8B2C module does operate in Normal mode only.

4.13.3 Register Descriptions

This section describes in detail all the registers and register bits in the PWM module. Reserved bits within a register will always read as 0 and the write will be unimplemented. Unimplemented functions are indicated by shading the bit.

Table 109. PWM Register Summary

Name / Offset[86]		7	6	5	4	3	2	1	0
0x60 PWMCTL	R	CAE1	CAE0	PCLK1	PCLK0	PPOL1	PPOL0	PWME1	PWME0
	W								
0x61 PWMPRCLK	R	0	PCKB2	PCKB1	PCKB0	0	PCKA2	PCKA1	PCKA0
	W								
0x62 PWMSCLA	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
0x63 PWMSCLB	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
0x64 PWMCNT0	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0
0x65 PWMCNT1	R	Bit 7	6	5	4	3	2	1	Bit 0
	W	0	0	0	0	0	0	0	0
0x66 PWMPER0	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
0x67 PWMPER1	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
0x68 PWMDTY0	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
0x69 PWMDTY1	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								

Note:

[86] Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

4.13.3.1 PWM Control Register (PWMCTL)

Table 110. PWM Control Register (PWMCTL)

Offset[87] 0x60 Access: User read/write

	7	6	5	4	3	2	1	0
R	CAE1	CAE0	PCLK1	PCLK0	PPOL1	PPOL0	PWME1	PWME0
W								
Reset	0	0	0	0	0	0	0	0

Note:

[87] Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

Table 111. PWMCTL - Register Field Descriptions

Field	Description
7–6 CAE[1:0]	Center Aligned Output Modes on Channels 1–0 0 Channels 1–0 operate in left aligned output mode. 1 Channels 1–0 operate in center aligned output mode.
5 PCLK1	Pulse Width Channel 1 Clock Select 0 Clock B is the clock source for PWM channel 1. 1 Clock SB is the clock source for PWM channel 1.
4 PCLK0	Pulse Width Channel 0 Clock Select 0 Clock A is the clock source for PWM channel 0. 1 Clock SA is the clock source for PWM channel 0.
3–2 PPOL[1:0]	Pulse Width Channel 1–0 Polarity Bits 0 PWM channel 1–0 outputs are low at the beginning of the period, then go high when the duty count is reached. 1 PWM channel 1–0 outputs are high at the beginning of the period, then go low when the duty count is reached.
1-0 PWME[1:0]	Pulse Width Channel 1–0 Enable 0 Pulse width channel 1–0 is disabled. 1 Pulse width channel 1–0 is enabled. The pulse modulated signal becomes available at PWM, output bit 1 when its clock source begins its next cycle.

4.13.3.1.1 PWM Enable (PWMEx)

NOTE

The first PWM cycle after enabling the channel can be irregular. If both PWM channels are disabled (PWME1–0 = 0), the prescaler counter shuts off for power savings.

Each PWM channel has an enable bit (PWMEx) to start its waveform output. When any of the PWMEx bits are set (PWMEx = 1), the associated PWM output is enabled immediately. However, the actual PWM waveform is not available on the associated PWM output until its clock source begins its next cycle, due to the synchronization of PWMEx and the clock source.

4.13.3.1.2 PWM Polarity (PPOLx)

NOTE

PPOLx register bits can be written anytime. If the polarity changes while a PWM signal is being generated, a truncated or stretched pulse can occur during the transition

The starting polarity of each PWM channel waveform is determined by the associated PPOLx bit. If the polarity bit is one, the PWM channel output is high at the beginning of the cycle and then goes low when the duty count is reached. Conversely, if the polarity bit is zero, the output starts low and then goes high when the duty count is reached.

4.13.3.1.3 PWM Clock Select (PCLKx)

NOTE

Register bits PCLK0 and PCLK1 can be written anytime. If a clock select changes while a PWM signal is being generated, a truncated or stretched pulse can occur during the transition.

Each PWM channel has a choice of two clocks to use as the clock source for that channel as described by the following.

4.15.2.2 SCI Control Register 1 (SCIC1)

This read/write register is used to control various optional features of the SCI system.

Table 128. SCI Control Register 1 (SCIC1)

Offset[98] 0x42 Access: User read/write

	7	6	5	4	3	2	1	0
R	LOOPS	0	RSRC	M	0	ILT	PE	PT
W								
Reset	0	0	0	0	0	0	0	0

Note:

98. Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

Table 129. SCIC1 Field Descriptions

Field	Description
7 LOOPS	**Loop Mode Select** — Selects between loop back modes and normal 2-pin full-duplex modes. When LOOPS = 1, the transmitter output is internally connected to the receiver input. 0 Normal operation — RxD and TxD use separate pins. 1 Loop mode or single-wire mode where transmitter outputs are internally connected to receiver input. (See RSRC bit.) RxD pin is not used by SCI.
5 RSRC	**Receiver Source Select** — This bit has no meaning or effect unless the LOOPS bit is set to 1. When LOOPS = 1, the receiver input is internally connected to the TxD pin and RSRC determines whether this connection is also connected to the transmitter output. 0 Provided LOOPS = 1, RSRC = 0 selects internal loop back mode and the SCI does not use the RxD pins. 1 Single-wire SCI mode where the TxD pin is connected to the transmitter output and receiver input.
4 M	**9-Bit or 8-Bit Mode Select** 0 Normal — start + 8 data bits (LSB first) + stop. 1 Receiver and transmitter use 9-bit data characters start + 8 data bits (LSB first) + 9th data bit + stop.
2 ILT	**Idle Line Type Select** — Setting this bit to 1 ensures that the stop bit and logic 1 bits at the end of a character do not count toward the 10 or 11 bit times of logic high level needed by the idle line detection logic. Refer to Section 4.15.3.3.2.1, “Idle-line Wake-up"” for more information. 0 Idle character bit count starts after start bit. 1 Idle character bit count starts after stop bit.
1 PE	**Parity Enable** — Enables hardware parity generation and checking. When parity is enabled, the most significant bit (MSB) of the data character (eighth or ninth data bit) is treated as the parity bit. 0 No hardware parity generation or checking. 1 Parity enabled.
0 PT	**Parity Type** — Provided parity is enabled (PE = 1), this bit selects even or odd parity. Odd parity means the total number of 1s in the data character, including the parity bit, is odd. Even parity means the total number of 1s in the data character, including the parity bit, is even. 0 Even parity. 1 Odd parity.

NOTE

A successful channel 3 output compare overrides any channel 2:0 compares. If forced output compare on any channel occurs at the same time as the successful output compare then forced output compare action will take precedence and interrupt flag will not get set.

A write to this register with the corresponding (FOC 3:0) data bit(s) set causes the action programmed for output compare on channel “n” to occur immediately.The action taken is the same as if a successful comparison had just taken place with the TCn register except the interrupt flag does not get set.

4.18.3.3.3 Output Compare 3 Mask Register (OC3M)

NOTE

A successful channel 3 output compare overrides any channel 2:0 compares. For each OC3M bit that is set, the output compare action reflects the corresponding OC3D bit

Table 156. Output Compare 3 Mask Register (OC3M)

Offset[117] 0xC2 Access: User read/write

	7	6	5	4	3	2	1	0
R	0	0	0	0	OC3M3	OC3M2	OC3M1	OC3M0
W								
Reset	0	0	0	0	0	0	0	0

Note:

[117] Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

Table 157. OC3M - Register Field Descriptions

Field	Description
3-0 OC3M[3-0]	Output Compare 3 Mask “n” Channel bit 0 - Does not set the corresponding port to be an output port 1 - Sets the corresponding port to be an output port when this corresponding TIOS bit is set to be an output compare

Setting the OC3Mn (n ranges from 0 to 2) will set the corresponding port to be an output port when the corresponding TIOSn (n ranges from 0 to 2) bit is set to be an output compare.

4.18.3.3.4 Output Compare 3 Data Register (OC3D)

NOTE

A channel 3 output compare will cause bits in the output compare 3 data register to transfer to the timer port data register if the corresponding output compare 3 mask register bits are set.

Table 158. Output Compare 3 Data Register (OC3D)

Offset[118] 0xC3 Access: User read/write

	7	6	5	4	3	2	1	0
R	0	0	0	0	OC3D3	OC3D2	OC3D1	OC3D0
W								
Reset	0	0	0	0	0	0	0	0

Note:

[118] Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

4.18.3.3.13 Main Timer Interrupt Flag 2 (TFLG2)

NOTE

The TFLG2 register indicates when an interrupt has occurred. Writing a one to the TOF bit will clear it. Any access to TCNT will clear TOF bit of TFLG2 register if the TFFCA bit in TSCR register is set.

Table 179. Main Timer Interrupt Flag 2 (TFLG2)

Offset[127] 0xCD Access: User read/write

	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

Note:

[127] Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

Table 180. TFLG2 - Register Field Descriptions

Field	Description
7 TOF	Timer Overflow Flag 1 = Indicates that an Interrupt has occurred (Set when 16-bit free-running timer counter overflows from $FFFF to $0000) 0 = Flag indicates an Interrupt has not occurred.

4.18.3.3.14 Timer Input Capture/Output Compare Registers (TC3 - TC0)

NOTE

TRead anytime. Write anytime for output compare function. Writes to these registers have no effect during input capture.

Depending on the TIOS bit for the corresponding channel, these registers are used to latch the value of the free-running counter when a defined transition is sensed by the corresponding input capture edge detector or to trigger an output action for output compare.

Read/Write access in byte mode for high byte should takes place before low byte otherwise it will give a different result.

Table 181. Timer Input Capture/Output Compare Register 0 (TC0)

Offset[128] 0xCE, 0xCF Access: User read(anytime)/write (special mode)

	15	14	13	12	11	10	9	8
R	tc0_15	tc0_14	tc0_13	tc0_12	tc0_11	tc0_10	tc0_9	tc0_8
W								
Reset	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
R	tc0_7	tc0_6	tc0_5	tc0_4	tc0_3	tc0_2	tc0_1	tc0_0
W								
Reset	0	0	0	0	0	0	0	0

Note:

[128] Offset related to 0x0200 for blocking access and 0x300 for non blocking access within the global address space.

4.19 Analog Digital Converter - ADC

4.19.1 Introduction

4.19.1.1 Overview

In order to sample the MM912F634 analog die analog sources, a 10-bit resolution successive approximation Analog to Digital Converter has been implemented. Controlled by the A/D Control Logic (ADC Wrapper), the Analog Digital Converter allows fast and high precision conversions.

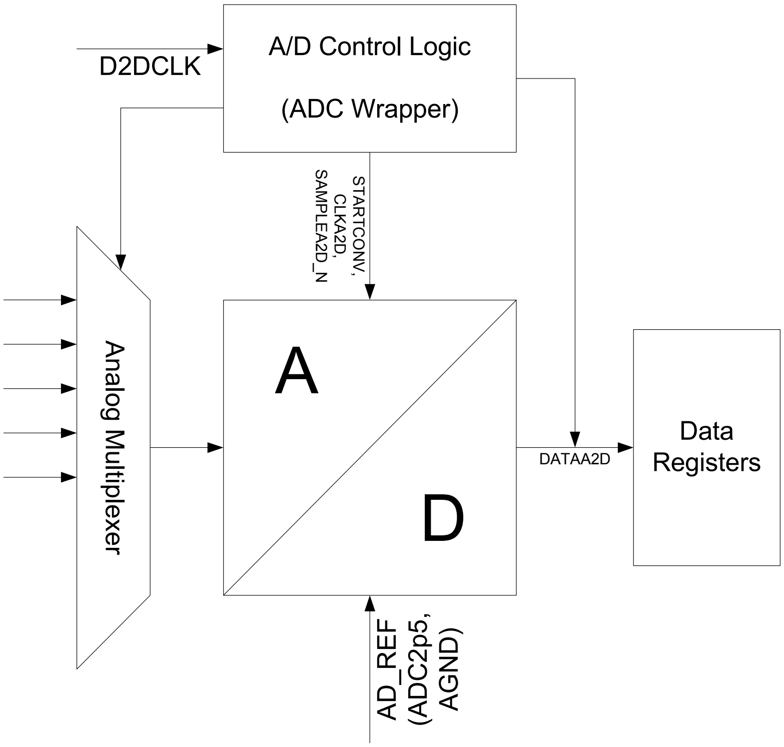


Figure 37. Analog Digital Converter Block Diagram

4.19.1.2 Features

- 10-bit resolution
- 13 µs (typ.), 10-bit Single Sample + Conversion Time
- External ADC2p5 pin with over-current protection to filter the analog reference voltage
- Total Error (TE) of ± 5 LSB without offset calibration active
- Integrated selectable offset compensation
- 14 + 1 analog channels (AD0…8; ISENSE, TSENSE and VSENSE, VS1SENSE, BANDGAP, plus calibration channel)
- Sequence- and Continuous Conversion Mode with IRQ for Sequence Complete indication
- Dedicated Result register for each channel

4.19.2 Modes of Operation

The Analog Digital Converter Module is active only in normal mode; it is disabled in Sleep and Stop mode.

4.27.3.9 PIM Reserved Registers

Table 231. PIM Reserved Registers

Address 0x0008-0x0019 Access: User read[162]

	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

Note:

[162] Read: Anytime.
Write: Unimplemented. Writing to these registers has no effect.

4.27.3.10 Port A Input Register (PTIA)

Table 232. Port A Input Register (PTIA)

Address 0x0120 Access: User read[163]

	7	6	5	4	3	2	1	0
R	0	0	PTIA5	PTIA4	PTIA3	PTIA2	PTIA1	PTIA0
W								
Reset[164]	u	u	u	u	u	u	u	u

Note:

[163] Read: Anytime.
Write: Unimplemented. Writing to this register has no effect.

[164] u = Unaffected by reset

Table 233. PTIA Register Field Descriptions

Field	Description
5-0 PTIA	**Port A input data—** This register always reads back the buffered and synchronized state of the associated pins. This can also be used to detect overload or short circuit conditions on output pins.

4.27.3.11 PIM Reserved Register

Table 234. PIM Reserved Register

Address 0x0121 Access: User read[165]

	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

Note:

[165] Read: Anytime.
Write: Unimplemented. Writing to this register has no effect.

4.30.4.4 Standard BDM Firmware Commands

Firmware commands are used to access and manipulate CPU resources. The system must be in active BDM to execute standard BDM firmware commands, see Section 4.30.4.2, “Enabling and Activating BDM"”. Normal instruction execution is suspended while the CPU executes the firmware located in the standard BDM firmware lookup table. The hardware command BACKGROUND is the usual way to activate BDM.

As the system enters active BDM, the standard BDM firmware lookup table and BDM registers become visible in the on-chip memory map at 0x3_FF00–0x3_FFFF, and the CPU begins executing the standard BDM firmware. The standard BDM firmware watches for serial commands and executes them as they are received.

The firmware commands are shown in Table 257.

Table 257. Firmware Commands

Command[174]	Opcode (hex)	Data	Description
READ_NEXT[175]	62	16-bit data out	Increment X index register by 2 (X = X + 2), then read word X points to.
READ_PC	63	16-bit data out	Read program counter.
READ_D	64	16-bit data out	Read D accumulator.
READ_X	65	16-bit data out	Read X index register.
READ_Y	66	16-bit data out	Read Y index register.
READ_SP	67	16-bit data out	Read stack pointer.
WRITE_NEXT	42	16-bit data in	Increment X index register by 2 (X = X + 2), then write word to location pointed to by X.
WRITE_PC	43	16-bit data in	Write program counter.
WRITE_D	44	16-bit data in	Write D accumulator.
WRITE_X	45	16-bit data in	Write X index register.
WRITE_Y	46	16-bit data in	Write Y index register.
WRITE_SP	47	16-bit data in	Write stack pointer.
GO	08	none	Go to user program. If enabled, ACK will occur when leaving active background mode.
GO_UNTIL[176]	0C	none	Go to user program. If enabled, ACK will occur upon returning to active background mode.
TRACE1	10	none	Execute one user instruction then return to active BDM. If enabled, ACK will occur upon returning to active background mode.
TAGGO -> GO	18	none	(Previous enable tagging and go to user program.) This command will be deprecated and should not be used anymore. Opcode will be executed as a GO command.

Note:

[174] If enabled, ACK will occur when data is ready for transmission for all BDM READ commands and will occur after the write is complete for all BDM WRITE commands.

[175] When the firmware command READ_NEXT or WRITE_NEXT is used to access the BDM address space the BDM resources are accessed rather than user code. Writing BDM firmware is not possible.

[176] System stop disables the ACK function and ignored commands will not have an ACK-pulse (e.g., CPU in stop or wait mode). The GO_UNTIL command will not get an Acknowledge if CPU executes the wait or stop instruction before the “UNTIL” condition (BDM active again) is reached (see Section 4.30.4.7, “Serial Interface Hardware Handshake Protocol"” last Note).

Table 259. Quick Reference to DBG Registers (continued)

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x002D	DBGADL	R W	Bit 7	6	5	4	3	2	1	Bit 0
0x002E	DBGADHM	R W	Bit 15	14	13	12	11	10	9	Bit 8
0x002F	DBGADLM	R W	Bit 7	6	5	4	3	2	1	Bit 0

Note:

177. This bit is visible at DBGCNT[7] and DBGSR[7]
178. This represents the contents if the Comparator A control register is blended into this address.
179. This represents the contents if the Comparator B control register is blended into this address.
180. This represents the contents if the Comparator C control register is blended into this address.

4.31.3.2 Register Descriptions

This section consists of the DBG control and trace buffer register descriptions in address order. Each comparator has a bank of registers that are visible through an 8-byte window between 0x0028 and 0x002F in the DBG module register address map. When ARM is set in DBGC1, the only bits in the DBG module registers that can be written are ARM, TRIG, and COMRV[1:0]

4.31.3.2.1 Debug Control Register 1 (DBGC1)

NOTE

When disarming the DBG by clearing ARM with software, the contents of bits[4:3] are not affected by the write, since up until the write operation, ARM = 1 preventing these bits from being written. These bits must be cleared using a second write if required.

Table 260. Debug Control Register (DBGC1)

Address: 0x0020

	7	6	5	4	3	2	1	0
R	ARM	0	0	BDM	DBGBRK	0	COMRV	
W		TRIG						
Reset	0	0	0	0	0	0	0	0

Read: Anytime

Write: Bits 7, 1, 0 anytime
Bit 6 can be written anytime but always reads back as 0.
Bits 4:3 anytime DBG is not armed.

Table 261. DBGC1 Field Descriptions

Field	Description
7 ARM	**Arm Bit** — The ARM bit controls whether the DBG module is armed. This bit can be set and cleared by user software and is automatically cleared on completion of a tracing session, or if a breakpoint is generated with tracing not enabled. On setting this bit the state sequencer enters State1. 0 Debugger disarmed 1 Debugger armed

4.32.4.2 Stop Mode Using DCO Clock as a Bus Clock

An example of what happens going into stop mode and exiting stop mode after an interrupt is shown in Figure 74.

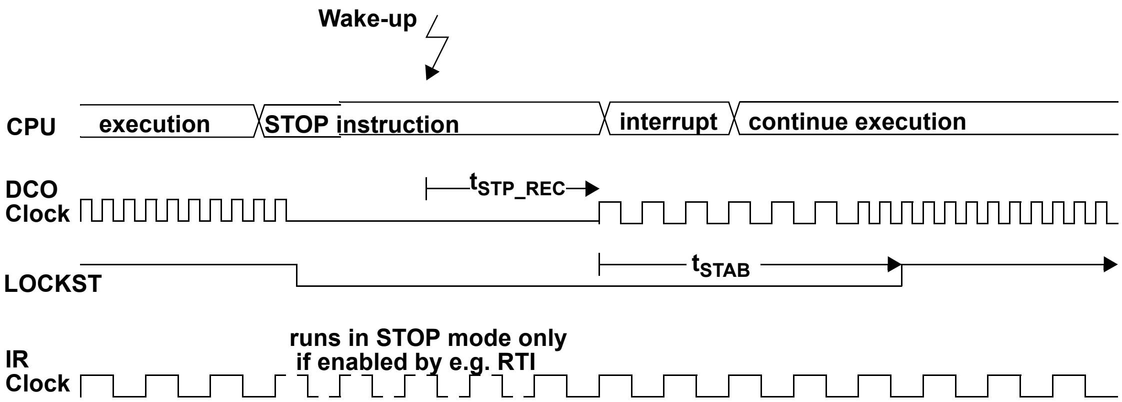


Figure 74. Example of STOP Mode Using DCO Clock as Bus Clock

4.32.4.3 Stop Mode Using Oscillator Clock as Bus Clock

An example of what happens going into stop mode and exiting stop mode after an interrupt is shown in Figure 75.

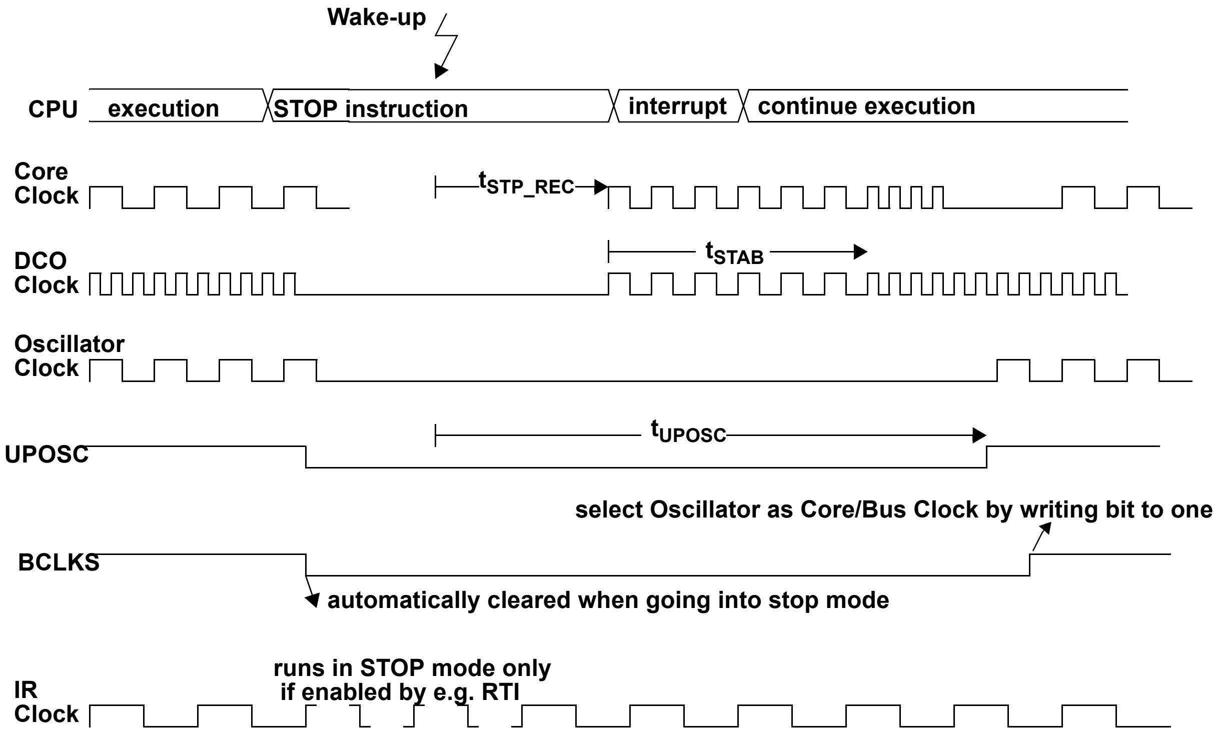


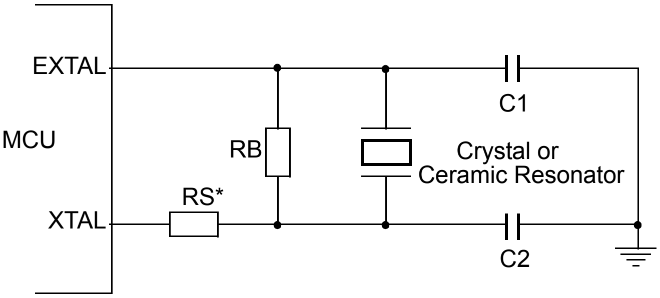
Figure 75. Example of STOP Mode Using Oscillator Clock as a Bus Clock

4.33.5 External Signals EXTAL and XTAL — Input and Output Pins

NOTE

> Freescale recommends an evaluation of the application board, and chosen resonator or crystal, by the resonator or crystal supplier.
>
> The oscillator circuit is not suited for overtone resonators and crystals.

EXTAL is the input to the crystal oscillator amplifier. XTAL is the output of the crystal oscillator amplifier.



* R_S can be zero (shorted) when use with higher frequency crystals.
Refer to manufacturer's data.

Figure 79. Full Swing Pierce Oscillator Connections

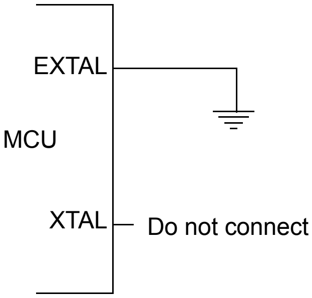


Figure 80. External Connections, if S12SCRG is Unused

The circuit shown in Figure 79 is recommended when using either a crystal or a ceramic resonator.

If S12SCRG is not used, it is recommended to pull the EXTAL input pin to GND, as shown in Figure 80. In Off mode the XTAL output will be forced to V_{DD} by the MCU.

Table 338. RTICTL Field Descriptions (continued)

Field	Description
4 RTISWAI	RTI Stops in Wait Mode Bit Normal modes: Write once Special modes: Write anytime. 0 RTI keeps running in Wait mode. 1 RTI stops and initializes the RTI counter whenever the part enters Wait mode.
3 RTIRSTP	**RTI Runs in Stop Mode Bit** Normal modes: Write once Special modes: Write anytime. 0 RTI stops in Stop mode 1 RTI continues in Stop mode **Note:** If the RTIRSTP bit is cleared the RTI counter will go static while in Stop mode. The RTI counter will not initialize like in Wait mode with RTISWAI bit set.
2 RTIE	Real Time Interrupt Enable Bit Write anytime. 0 Interrupt requests from RTI are disabled. 1 Interrupt will be requested whenever RTIF is set.
1–0 RTIRT[1:0]	**RTI Interrupt Prescaler Rate Select Bits** — These bits select the prescaler rate for the RTI. See Table 340., “RTI Frequency Divide Rates“ for selectable ratios in conjunction with RTICNT[7:0] counter select bits Write anytime in appliance of the write protection rules (see 4.34.7.1, “RTI register write protection rules“).

4.34.6.2.2 RTI Counter select bits (RTICNT)

This register is used to restart the RTI time-out period.

Table 339. RTICNT Register Diagram

0x003D

	7	6	5	4	3	2	1	0
R W	RTICNT7	RTICNT6	RTICNT5	RTICNT4	RTICNT3	RTICNT2	RTICNT1	RTICNT0
Reset	0	0	0	0	0	0	0	0

Read: Anytime

Write: Anytime in appliance of the write protection rules (see 4.34.7.1, “RTI register write protection rules“)

When the RTI is turned on the RTIF bit can be set with the following rates:

Table 340. RTI Frequency Divide Rates

RTICNT[7:0]	RTIRT[1:0] =			
	00 (OFF)	**01 (1)**	**10 (16)**	**11 (256)**
0000 0000 (÷1)	OFF[185]	OFF	16	256
0000 0001 (÷2)	OFF	2x1	2x16	2x256
0000 0010 (÷3)	OFF	3x1	3x16	3x256
0000 0011 (÷4)	OFF	4x1	4x16	4x256
......	....	....	....	....
1111 1110 (÷255)	OFF	255x1	255x16	255x256

Table 343. COPCTL Field Descriptions (continued)

Field	Description
3 COPRSTP	**COP Runs in Stop Mode Bit** Normal modes: Write once Special modes: Write anytime 0 COP stops in Stop mode 1 COP continues in Stop mode **Note:** If the COPRSTP bit is cleared the COP counter will go static while in Stop mode. The COP counter will not initialize like in Wait mode with COPSWAI bit set.
2–0 CR[2:0]	**COP Watchdog Timer Rate Select Bits** — These bits select the COP timeout rate (see Table 344). Writing a non-zero value to CR[2:0] enables the COP counter and starts the timeout period. A COP counter timeout causes a system reset. This can be avoided by periodically (before timeout) re-initialize the COP counter via the ARMCOP register. While all of the following four conditions are true the CR[2:0], WCOP bits are ignored and the COP operates at highest timeout period (2^{16} cycles) in normal COP mode (Window COP mode disabled): 1) COP is enabled (CR[2:0] is not 000) 2) BDM mode active 3) RSBCK = 0 4) Operation in special mode

Table 344. COP Watchdog Rates[187]

CR2	CR1	CR0	Input_CLK Cycles to Timeout
0	0	0	COP disabled
0	0	1	2^6
0	1	0	2^8
0	1	1	2^{10}
1	0	0	2^{12}
1	0	1	2^{14}
1	1	0	2^{15}
1	1	1	2^{16}

Note:

[187] Refer to Device User Guide (Section 4.35.4.1, “COP Configuration") for reset values of WCOP, CR2, CR1 and CR0.

Table 348. Flash Configuration Field (continued)

Address Relative to Flash Array Base	Size (bytes)	Description
0x7F0E	1	Flash Nonvolatile byte Refer to the SoC Guide
0x7F0F	1	Flash Security byte Refer to Section 4.36.3.3.2, “Flash Security Register (FSEC)"”

4.36.3.2 Flash IFR Map

The Flash IFR is a 128 byte nonvolatile information memory that is read accessible as defined in the SoC Guide. The MCU memory map places the Flash IFR addresses between Flash IFR base + 0x0000 and 0x007F as shown in Table 349.

Table 349. Flash IFR Description

Address Relative to Flash IFR Base[(188)]	Size (bytes)	Description
0x0000 - 0x000D	14	Wafer lot number, wafer number, X coordinate, Y coordinate
0x000E - 0x003F	50	Reserved for wafer test data
0x0040 - 0x004F	16	Flash memory controller parameters
0x0050 - 0x007B	44	Reserved
0x007C - 0x007F	4	MCU control parameters

Note:

188. Refer to the SoC Guide for details on how to enable the Flash IFR

4.36.3.3 Register Descriptions

The Flash module contains a set of 16 control and status registers located between Flash register base + 0x0000 and 0x000F. Flash registers are byte and word accessible. A summary of the Flash module registers is given in Table 350. Detailed descriptions of each register bit are provided in the following sections.

Table 350. S12SFTSR32K Register Summary (Normal/Special Mode)

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0100 FCLKDIV	R	FDIVLD	PRDIV8	FDIV5	FDIV4	FDIV3	FDIV2	FDIV1	FDIV0
	W								
0x0101 FSEC	R	KEYEN1	KEYEN0	0	0	0	0	SEC1	SEC0
	W								
0x0102 FRSV0	R	0	0	0	0	0	0	0	0
	W								
0x0103 FCNFG	R	CBEIE	CCIE	KEYACC	0	0	0	0	0
	W								
0x0104 FPROT	R	FPHS4	FPHS3	FPHS2	FPHS1	FPHS0	FPLS2	FPLS1	FPLS0
	W								
0x0105 FSTAT	R	CBEIF	CCIF	PVIOL	ACCERR	0	BLANK	0	0
	W								
0x0106 FCMD	R	0	CMDB6	CMDB5	CMDB4	CMDB3	CMDB2	CMDB1	CMDB0
	W								

4.36.4.2.6 Set Verify Margin Level Command

The set verify margin level operation, available only in special mode, will set the margin level in the Flash array sense-amps to allow content validation with margin to the normal level for subsequent Flash array reads. The set verify margin level command should only be used to validate initial programming of the Flash array.

An example flow to execute the set verify margin level operation is shown in Figure 91. The set verify margin level command write sequence is as follows:

1. Write to an aligned Flash block address to start the command write sequence for the set verify margin level command. The address will be ignored while the data written sets the margin level as shown in Table 384.
2. Write the set verify margin level command, 0x75, to the FCMD register.
3. Clear the CBEIF flag in the FSTAT register by writing a 1 to CBEIF to launch the set verify margin level command.

Once the set verify margin level command has successfully launched, the CCIF flag in the FSTAT register will set after the set verify margin level operation has completed.

Table 384. Flash Array Margin Level Settings

Command Data Field	Margin Level Setting	Description
0x0000	Normal	Sets normal level for Flash array reads
0x0005	Margin 0	Sets test level to validate margin to reading 0’s
0x0024	Margin 1	Sets test level to validate margin to reading 1’s

9. Writing a 0 to the CBEIF flag in the FSTAT register to abort a command write sequence.

The ACCERR flag will also be set if the MCU enters stop mode while any command is active (CCIF=0). The operation is aborted immediately and, if burst programming, any pending burst program command is purged (see Section 4.36.5.2, “Stop Mode"”).

The ACCERR flag will not be set if any Flash register is read during a valid command write sequence.

If the Flash memory is read during execution of an algorithm (CCIF = 0), the read operation will return invalid data and the ACCERR flag will not be set.

If the ACCERR flag is set in the FSTAT register, the user must clear the ACCERR flag before starting another command write sequence (see Section 4.36.3.4, “Flash Status Register (FSTAT)"”).

4.36.4.3.2 Flash Protection Violations

The PVIOL flag will be set after the command is written to the FCMD register during a command write sequence, if any of the following illegal operations are attempted, causing the command write sequence to immediately abort:

1. Writing the program command if the address written in the command write sequence was in a protected area of the Flash array.
2. Writing the sector erase command if the address written in the command write sequence was in a protected area of the Flash array.
3. Writing the mass erase command while any Flash protection is enabled.
4. Writing an invalid command if the address written in the command write sequence was in a protected area of the Flash array.

If the PVIOL flag is set in the FSTAT register, the user must clear the PVIOL flag before starting another command write sequence (see Section 4.36.3.4, “Flash Status Register (FSTAT)"”).

4.36.5 Operating Modes

4.36.5.1 Wait Mode

If a command is active (CCIF = 0) when the MCU enters wait mode, the active command and any buffered command will be completed.

The Flash module can recover the MCU from wait mode if the CBEIF and CCIF interrupts are enabled (see Section 4.36.8, “Interrupts"”).

4.36.5.2 Stop Mode

NOTE

As active commands are immediately aborted when the MCU enters stop mode, it is strongly recommended that the user does not use the STOP instruction during program or erase operations.

If a command is active (CCIF = 0) when the MCU enters stop mode, the operation will be aborted and, if the operation is program or erase, the Flash array data being programmed or erased may be corrupted and the CCIF and ACCERR flags will be set. If active, the high voltage circuitry to the Flash array will immediately be switched off when entering stop mode. Upon exit from stop mode, the CBEIF flag is set and any buffered command will not be launched. The ACCERR flag must be cleared before starting a command write sequence (see Section 4.36.4.1.2, “Command Write Sequence"”).

4.38 Serial Peripheral Interface (S12SPIV4)

4.38.1 Introduction

The SPI module allows a duplex, synchronous, serial communication between the MCU and peripheral devices. Software can poll the SPI status flags or the SPI operation can be interrupt driven.

4.38.1.1 Glossary of Terms

SPI — Serial Peripheral Interface

SS — Slave Select

SCK — Serial Clock

MOSI — Master Output, Slave Input

MISO — Master Input, Slave Output

MOMI — Master Output, Master Input

SISO — Slave Input, Slave Output

4.38.1.2 Features

The S12SPIV4 includes these distinctive features:

- Master mode and slave mode
- Bi-directional mode
- Slave select output
- Mode fault error flag with CPU interrupt capability
- Double-buffered data register
- Serial clock with programmable polarity and phase
- Control of SPI operation during wait mode

4.38.1.3 Modes of Operation

The SPI functions in three modes: run, wait, and stop.

- Run mode
  This is the basic mode of operation.
- Wait mode
  SPI operation in wait mode is a configurable low-power mode, controlled by the SPISWAI bit located in the SPICR2 register. In wait mode, if the SPISWAI bit is clear, the SPI operates like in run mode. If the SPISWAI bit is set, the SPI goes into a power conservative state, with the SPI clock generation turned off. If the SPI is configured as a master, any transmission in progress stops, but is resumed after CPU goes into run mode. If the SPI is configured as a slave, reception and transmission of a byte continues, so that the slave stays synchronized to the master.
- Stop mode
  The SPI is inactive in stop mode for reduced power consumption. If the SPI is configured as a master, any transmission in progress stops, but is resumed after CPU goes into run mode. If the SPI is configured as a slave, reception and transmission of a byte continues, so that the slave stays synchronized to the master.

This is a high level description only, detailed descriptions of operating modes are contained in Section 4.38.4.7, “Low Power Mode Options"”.