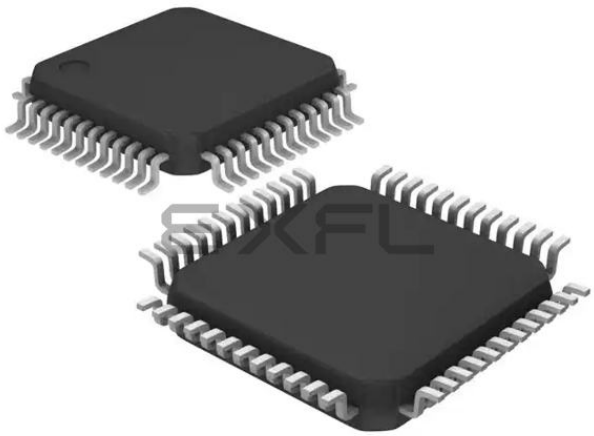


Welcome to E-XFL.COM

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 (48kB)
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
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mm912g634dc2apr2

For N < 100, the following equation is a good fit for the maximum jitter:

$$J(N) = \frac{j_1}{\sqrt{N}}$$

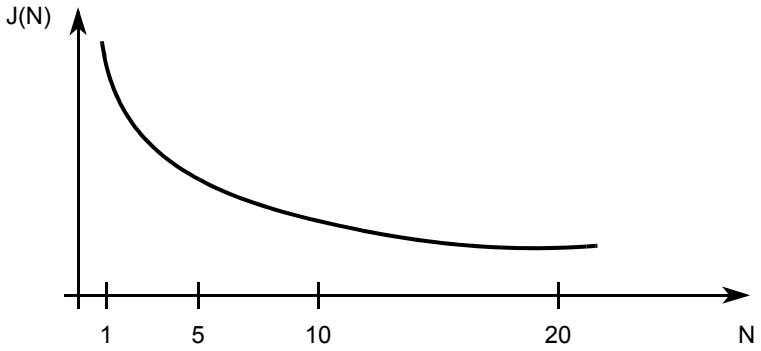


Figure 11. Maximum Bus Clock Jitter Approximation

NOTE

On timers and serial modules a prescaler will eliminate the effect of the jitter to a large extent.

4.6.2.2.2 Electrical Characteristics for the PLL(51)

Table 41. PLL Characteristics

Rating	Symbol	Min	Typ	Max	Unit
VCO Frequency During System Reset	f_{VCORST}	8.0	—	32	MHz
VCO Locking Range	f_{VCO}	32	—	64	MHz
Reference Clock	f_{REF}	1.0	—	—	MHz
Lock Detection	$\|\Delta_{Lock}\|$	0	—	1.5	%(52)
Un-Lock Detection	$\|\Delta_{unl}\|$	0.5	—	2.5	%(52)
Time to Lock	t_{lock}	—	—	150 + 256/f_{REF}	µs
Jitter Fit Parameter 1(53)	j_1	—	—	1.2	%

Note:

51. the maximum device bus clock is specified as fBUS.
52. % deviation from target frequency.
53. f_{REF} = 1.0 MHz, f_{BUS} = 32 MHz equivalent f_{PLL} = 64 MHz, REFRQ=00, SYNDIV=$1F, VCOFRQ=01, POSTDIV=$00.

4.6.2.3 Electrical Characteristics for the IRC1M

Table 42. IRC1M Characteristics

Rating	Symbol	Min	Typ	Max	Unit
Internal Reference Frequency, Factory Trimmed -40 °C ≤ T_J ≤ 150 °C	f_{IRC1M_TRIM}	0.987	1.0	1.013	MHz

Table 55. 0x0010–0x001B Memory Map Control (MMC) Map 2 of 2

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0010	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0011	DIRECT	R	DP15	DP14	DP13	DP12	DP11	DP10	DP9	DP8
		W								
0x0012	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0013	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0014	Reserved	R	0	0	0	0	0	0	0	0
		W								
0x0015	PPAGE	R	0	0	0	0	PIX3	PIX2	PIX1	PIX0
		W								

Table 56. 0x0016–0x0019 Reserved

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0016- 0x0019	Reserved	R	0	0	0	0	0	0	0	0
		W								

Table 57. 0x001A–0x001B Device ID Register (PARTIDH/PARTIDL)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x001A	PARTIDH	R	PARTIDH							
		W								
0x001B	PARTIDL	R	PARTIDL							
		W								

Table 58. 0x001C–0x001E Reserved

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x001C- 0x001E	Reserved	R	0	0	0	0	0	0	0	0
		W								

Table 59. 0x001F Interrupt Module (INT)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x001F	IVBR	R	IVB_ADDR[7:0]							
		W								

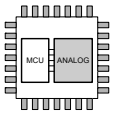
Table 60. 0x0020–0x002F Debug Module (DBG)

Address	Name		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0020	DBGC1	R	ARM	0	0	BDM	DBGBRK	0	COMRV	
		W		TRIG						
0x0021	DBGSR	R	TBF	0	0	0	0	SSF2	SSF1	SSF0
		W								

Table 74. Analog die Registers[63] - 0x0200–0x02FF D2D Blocking Access (D2DI) 2 of 3/ 0x0300–0x03FF D2D Non Blocking Access (D2DI) 3 of 3

Offset	Name		7	6	5	4	3	2	1	0
0xA5	ADR15 (lo)	R	adr15 1	adr15 0	0	0	0	0	0	0
	ADC Data Result Reg 15	W								
0xC0	TIOS	R	0	0	0	0	IOS3	IOS2	IOS1	IOS0
	TIM InCap/OutComp Select	W								
0xC1	CFORC	R	0	0	0	0	0	0	0	0
	Timer Compare Force Reg	W					FOC3	FOC2	FOC1	FOC0
0xC2	OC3M	R	0	0	0	0	OC3M3	OC3M2	OC3M1	OC3M0
	Output Comp 3 Mask Reg	W								
0xC3	OC3D	R	0	0	0	0	OC3D3	OC3D2	OC3D1	OC3D0
	Output Comp 3 Data Reg	W								
0xC4	TCNT (hi)	R	tcnt 15	tcnt 14	tcnt 13	tcnt 12	tcnt 11	tcnt 10	tcnt 9	tcnt 8
	Timer Count Register	W								
0xC5	TCNT (lo)	R	tcnt 7	tcnt 6	tcnt 5	tcnt 4	tcnt 3	tcnt 2	tcnt 1	tcnt 0
	Timer Count Register	W								
0xC6	TSCR1	R	TEN	0	0	TFFCA	0	0	0	0
	Timer System Control Reg 1	W								
0xC7	TTOV	R	0	0	0	0	TOV3	TOV2	TOV1	TOV0
	Timer Toggle Overflow Reg	W								
0xC8	TCTL1	R	OM3	OL3	OM2	OL2	OM1	OL1	OM0	OL0
	Timer Control Register 1	W								
0xC9	TCTL2	R	EDG3B	EDG3A	EDG2B	EDG2A	EDG1B	EDG1A	EDG0B	EDG0A
	Timer Control Register 2	W								
0xCA	TIE	R	0	0	0	0	C3I	C2I	C1I	C0I
	Timer Interrupt Enable Reg	W								
0xCB	TSCR2	R	TOI	0	0	0	TCRE	PR2	PR1	PR0
	Timer System Control Reg 2	W								
0xCC	TFLG1	R	0	0	0	0	C3F	C2F	C1F	C0F
	Main Timer Interrupt Flag 1	W								
0xCD	TFLG2	R	TOF	0	0	0	0	0	0	0
	Main Timer Interrupt Flag 2	W								
0xCE	TC0 (hi)	R	tc0 15	tc0 14	tc0 13	tc0 12	tc0 11	tc0 10	tc0 9	tc0 8
	TIM InCap/OutComp Reg 0	W								
0xCF	TC0 (lo)	R	tc0 7	tc0 6	tc0 5	tc0 4	tc0 3	tc0 2	tc0 1	tc0 0
	TIM InCap/OutComp Reg 0	W								
0xD0	TC1 (hi)	R	tc1 15	tc1 14	tc1 13	tc1 12	tc1 11	tc1 10	tc1 9	tc1 8
	TIM InCap/OutComp Reg 1	W								
0xD1	TC1 (lo)	R	tc1 7	tc1 6	tc1 5	tc1 4	tc1 3	tc1 2	tc1 1	tc1 0
	TIM InCap/OutComp Reg 1	W								
0xD2	TC2 (hi)	R	tc2 15	tc2 14	tc2 13	tc2 12	tc2 11	tc2 10	tc2 9	tc2 8
	TIM InCap/OutComp Reg 2	W								
0xD3	TC2 (lo)	R	tc2 7	tc2 6	tc2 5	tc2 4	tc2 3	tc2 2	tc2 1	tc2 0
	TIM InCap/OutComp Reg 2	W								

5.5 Power Supply



The MM912_634 analog die supplies VDD (2.5 V), VDDX (5.0 V), and HSUP, based on the supply voltage applied to the VS1 pin. VDD is cascaded of the VDDX regulator. To separate the High Side outputs from the main power supply, the VS2 pin does only power the High Side drivers. Both supply pins have to be externally protected against reverse battery conditions. To supply external Hall Effect Sensors, the HSUP pin will supply a switchable regulated supply. See Section 5.11, “Hall Sensor Supply Output - HSUP".

A reverse battery protected input (VSENSE) is implemented to measure the Battery Voltage directly. A serial resistor (R_{VSENSE}) is required on this pin. See Section 5.23, “Supply Voltage Sense - VSENSE". In addition, the VS1 supply can be routed to the ADC (VS1SENSE) to measure the VS1 pin voltage directly. See Section 5.24, “Internal Supply Voltage Sense - VS1SENSE".

To have an independent ADC verification, the internal sleep mode bandgap voltage can be routed to the ADC (BANDGAP). As this node is independent from the ADC reference, any out of range result would indicate malfunctioning ADC or Bandgap reference. See Section 5.25, “Internal Bandgap Reference Voltage Sense - BANDGAP".

To stabilize the internal ADC reference voltage for higher precision measurements, the current limited ADC2p5 pin needs to be connected to an external filter capacitor (C_{ADC2p5}). It is not recommended to connect additional loads to this pin. See Section 5.20, “Analog Digital Converter - ADC".

The following safety features are implemented:

- LBI - Low Battery Interrupt, internally measured at VSENSE
- LVI - Low Voltage Interrupt, internally measured at VS1
- HVI - High Voltage Interrupt, internally measured at VS2
- VROVI - Voltage Regulator Over-voltage Interrupt internally measured at VDD and VDDX
- LVR - Low Voltage Reset, internally measured at VDD
- LVRX - Low Voltage Reset, internally measured at VDDX
- HTI - High Temperature Interrupt measured between the VDD and VDDX regulators
- Over-temperature Shutdown measured between the VDD and VDDX regulators

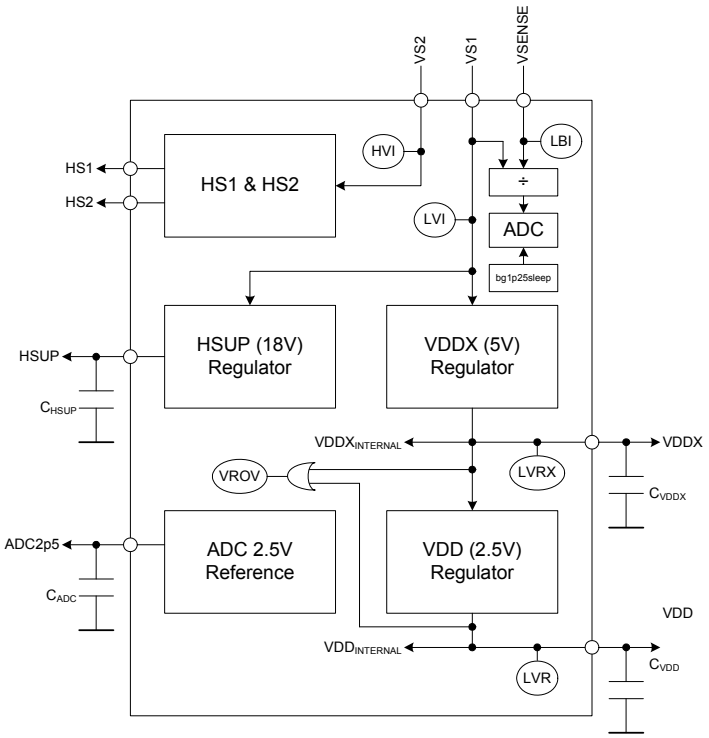


Figure 17. MM912_634 Power Supply

5.9.1.3 Forced Wake-up

Configuring the Forced Wake-up Multiplier (FWM) in the Timing Control Register (TCR) will enable the forced wake-up based on the selected Cyclic Sense Timing (CST). Forced Wake-up can be combined with all other wake-up sources considering the timing dependencies.

5.9.1.4 LIN - Wake-up

While in Low-Power mode the MM912_634 analog die monitors the activity on the LIN bus. A dominant pulse longer than t_{PROPWL} followed by a dominant to recessive transition will cause a LIN Wake-up. This behavior protects the system from a short-to-ground bus condition.

5.9.1.5 D2D - Wake-up (Stop Mode only)

Receiving a Normal mode request via the D2D interface (MODE=0, Mode Control Register (MCR)) will result in a wake-up from stop mode. As this condition is controlled by the MCU, no wake-up status bit does indicate this wake-up source.

5.9.1.6 Wake-up Due to Internal / External Reset (STOP Mode Only)

While in Stop mode, a Reset due to a VDD low voltage condition or an external Reset applied on the $\overline{\text{RESET\_A}}$ pin will result in a Wake-up with immediate transition to Reset mode. In this case, the LVR or EXR bits in the Reset Status Register will indicate the source of the event.

5.9.1.7 Wake-up Due to Loss of Supply Voltage (SLEEP Mode Only)

While in Sleep mode, a supply voltage VS1 < VPOR will result in a transition to Power On mode.

5.9.2 Register Definition

5.9.2.1 Wake-up Control Register (WCR)

Table 94. Wake-up Control Register (WCR)

Offset[75] 0x12 — Access: User read/write

	7	6	5	4	3	2	1	0
R W	CSSEL		L5WE	L4WE	L3WE	L2WE	L1WE	L0WE
Reset	0	0	1	1	1	1	1	1

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

Table 95. WCR - Register Field Descriptions

Field	Description
7-6 CSSEL	Cyclic Sense Select - Configures the HSx output for the cyclic sense event. Note, with no LxWE selected - only the selected HSx output will be switched periodically, no Lx state change would be detected. For all configurations, the Forced Wake-up can be activated in parallel in Section 5.9.2.2, “Timing Control Register (TCR)" 00 - Cyclic Sense Off 01 - Cyclic Sense with periodic HS1on 10 - Cyclic Sense with periodic HS2 on 11 - Cyclic Sense with periodic HS1 and HS2 on.
5 - L5WE	Wake-up Input 5 Enabled - L5 Wake-up Select Bit. 0 - L5 Wake-up Disabled 1 - L5 Wake-up Enabled
4 - L4WE	Wake-up Input 4 Enabled - L4 Wake-up Select Bit. 0 - L4 Wake-up Disabled 1 - L4 Wake-up Enabled

5.14.3.1.3 PWM Clock Select (PCLKx)

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

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.

5.14.3.1.4 PWM Center Align Enable (CAEx)

The CAEx bits select either center aligned outputs or left aligned output for both PWM channels. If the CAEx bit is set to a one, the corresponding PWM output will be center aligned. If the CAEx bit is cleared, the corresponding PWM output will be left aligned. See Section 5.14.4.2.5, “Left Aligned Outputs" and Section 5.14.4.2.6, “Center Aligned Outputs" for a more detailed description of the PWM output modes.

NOTE

Write these bits only when the corresponding channel is disabled.

5.14.3.2 PWM Prescale Clock Select Register (PWMPRCLK)

This register selects the prescale clock source for clocks A and B independently.

Table 121. PWM Prescale Clock Select Register (PWMPRCLK)

Offset(91) 0x61 — Access: User read/write

	7	6	5	4	3	2	1	0
R	0	PCKB2	PCKB1	PCKB0	0	PCKA2	PCKA1	PCKA0
W								
Reset	0	0	0	0	0	0	0	0

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

Table 122. PWMPRCLK - Register Field Descriptions

Field	Description
6–4 PCKB[2:0]	Prescaler Select for Clock B — Clock B is one of two clock sources which can be used for channel 1. These three bits determine the rate of clock B, as shown in Table 123.
2–0 PCKA[2:0]	Prescaler Select for Clock A — Clock A is one of two clock sources which can be used for channel 0. These three bits determine the rate of clock A, as shown in Table 124.

Table 123. Clock B Prescaler Selects

PCKB2	PCKB1	PCKB0	Value of Clock B
0	0	0	D2D clock
0	0	1	D2D clock / 2
0	1	0	D2D clock / 4
0	1	1	D2D clock / 8
1	0	0	D2D clock / 16
1	0	1	D2D clock / 32
1	1	0	D2D clock / 64
1	1	1	D2D clock / 128

Table 124. Clock A Prescaler Selects

PCKA2	PCKA1	PCKA0	Value of Clock A
0	0	0	D2D clock
0	0	1	D2D clock / 2
0	1	0	D2D clock / 4
0	1	1	D2D clock / 8
1	0	0	D2D clock / 16
1	0	1	D2D clock / 32
1	1	0	D2D clock / 64
1	1	1	D2D clock / 128

NOTE

PCKB2–0 and PCKA2–0 register bits can be written anytime. If the clock pre-scale is changed while a PWM signal is being generated, a truncated or stretched pulse can occur during the transition.

5.14.3.3 PWM Scale A Register (PWMSCLA)

PWMSCLA is the programmable scale value used in scaling clock A to generate clock SA. Clock SA is generated by taking clock A, dividing it by the value in the PWMSCLA register and dividing that by two.

Clock SA = Clock A / (2 * PWMSCLA)

NOTE

When PWMSCLA = $00, PWMSCLA value is considered a full scale value of 256. Clock A is thus divided by 512.

Any value written to this register will cause the scale counter to load the new scale value (PWMSCLA)

.

Table 125. PWM Scale A Register (PWMSCLA)

Offset[92] 0x62 Access: User read/write

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

Note:

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

5.14.3.4 PWM Scale B Register (PWMSCLB)

PWMSCLB is the programmable scale value used in scaling clock B to generate clock SB. Clock SB is generated by taking clock B, dividing it by the value in the PWMSCLB register and dividing that by two.

Clock SB = Clock B / (2 * PWMSCLB)

NOTE

When PWMSCLB = $00, PWMSCLB value is considered a full scale value of 256. Clock B is thus divided by 512.

Any value written to this register will cause the scale counter to load the new scale value (PWMSCLB).

NOTE

Clock SA = Clock A / (2 * PWMSCLA)

When PWMSCLA = $00, PWMSCLA value is considered a full scale value of 256. Clock A is thus divided by 512.

Similarly, clock B is used as an input to an 8-bit down counter followed by a divide by two producing clock SB. Thus, clock SB equals clock B divided by two times the value in the PWMSCLB register.

NOTE

Clock SB = Clock B / (2 * PWMSCLB)

When PWMSCLB = $00, PWMSCLB value is considered a full scale value of 256. Clock B is thus divided by 512.

As an example, consider the case in which the user writes $FF into the PWMSCLA register. Clock A for this case will be E divided by 4. A pulse will occur at a rate of once every 255x4 E cycles. Passing this through the divide by two circuit produces a clock signal at an E divided by 2040 rate. Similarly, a value of $01 in the PWMSCLA register when clock A is E divided by 4 will produce a clock at an E divided by 8 rate.

Writing to PWMSCLA or PWMSCLB causes the associated 8-bit down counter to be re-loaded. Otherwise, when changing rates the counter would have to count down to $01 before counting at the proper rate. Forcing the associated counter to re-load the scale register value every time PWMSCLA or PWMSCLB is written prevents this.

NOTE

Writing to the scale registers while channels are operating can cause irregularities in the PWM outputs.

5.14.4.1.3 Clock Select

Each PWM channel has the capability of selecting one of two clocks. For channels 0 the clock choice is clock A or clock SA. For channels 1 the choice is clock B or clock SB. The clock selection is done with the PCLKx control bits in the PWMCTL register.

NOTE

Changing clock control bits while channels are operating can cause irregularities in the PWM outputs.

5.14.4.2 PWM Channel Timers

The main part of the PWM module are the actual timers. Each of the timer channels has a counter, a period register, and a duty register (each are 8-bit). The waveform output period is controlled by a match between the period register and the value in the counter. The duty is controlled by a match between the duty register and the counter value, and causes the state of the output to change during the period. The starting polarity of the output is also selectable on a per channel basis. Shown in Figure 24 is the block diagram for the PWM timer.

Table 198. Analog Digital Converter Module - Memory Map

Register / Offset[138]		Bit 7	6	5	4	3	2	1	Bit 0
0x8F	R	ADR4[1:0]							
ADR4 (lo)	W								
0x90	R	ADR5[9:2]							
ADR5 (hi)	W								
0x91	R	ADR5[1:0]							
ADR5 (lo)	W								
0x92	R	ADR6[9:2]							
ADR6 (hi)	W								
0x93	R	ADR6[1:0]							
ADR6 (lo)	W								
0x94	R	ADR7[9:2]							
ADR7 (hi)	W								
0x95	R	ADR7[1:0]							
ADR7 (lo)	W								
0x96	R	ADR8[9:2]							
ADR8 (hi)	W								
0x97	R	ADR8[1:0]							
ADR8 (lo)	W								
0x98	R	ADR9[9:2]							
ADR9 (hi)	W								
0x99	R	ADR9[1:0]							
ADR9 (lo)	W								
0x9A	R	ADR10[9:2]							
ADR10 (hi)	W								
0x9B	R	ADR10[1:0]							
ADR10 (lo)	W								
0x9C	R	ADR11[9:2]							
ADR11 (hi)	W								
0x9D	R	ADR11[1:0]							
ADR11 (lo)	W								
0x9E	R	ADR12[9:2]							
ADR12 (hi)	W								
0x9F	R	ADR12[1:0]							
ADR12 (lo)	W								
0xA0	R								
Reserved	W								
0xA1	R								
Reserved	W								
0xA2	R	ADR14[9:2]							
ADR14 (hi)	W								
0xA3	R	ADR14[1:0]							
ADR14 (lo)	W								

5.23 Supply Voltage Sense - VSENSE

The reverse battery protected VSENSE pin has been implemented to allow a direct measurement of the Battery level voltage. Bypassing the device VSUP capacitor and external reverse battery diode will detect under-voltage conditions without delay. A series resistor is required to protect the MM912_634 analog die from fast transients.

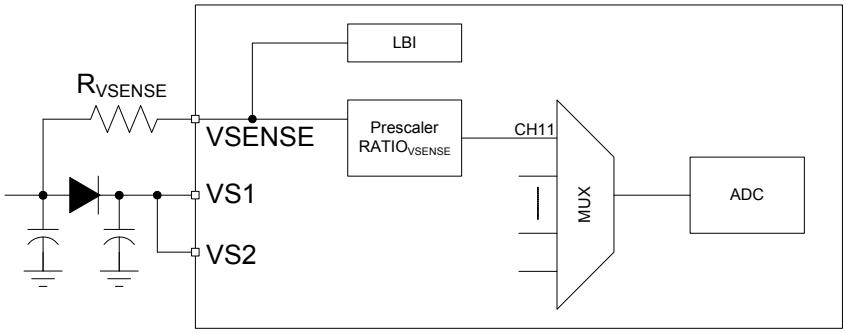


Figure 39. VSENSE Module

The voltage present on the VSENSE pin can be routed via an internal divider to the internal Analog Digital Converter or issue an interrupt (LBI) to alert the MCU.

For the interrupt based alert, see Section 5.5, “Power Supply". For VSENSE measurement using the internal ADC see Section 5.20, “Analog Digital Converter - ADC".

5.24 Internal Supply Voltage Sense - VS1SENSE

In addition to the VSENSE module, the internal VS1 supply can be routed to the analog digital converter as well. See Section 5.20, “Analog Digital Converter - ADC" for details on the acquisition.

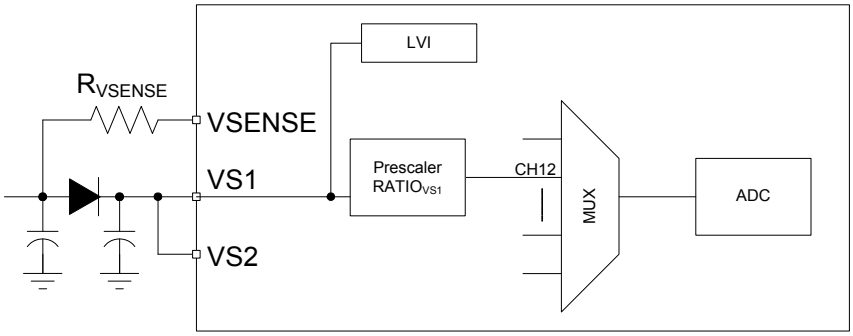


Figure 40. VS1Sense Module

5.25 Internal Bandgap Reference Voltage Sense - BANDGAP

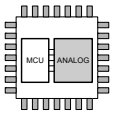
The internal reference bandgap voltage “bg1p25sleep” is generated fully independent from the Analog Digital Converter reference voltages.

Measuring(147) the “bg1p25sleep” reference through the ADC-CH14 allows should return a conversion result within ADCH14 under normal conditions. Any result outside the range would indicate faulty behavior of either the ADC chain or the 2p5sleep Bandgap circuity.

Note:

147. The maximum allowed sample frequency for Channel 14 is limited to fCH14. Increasing the sample frequency above can result in unwanted turn off of the LS drivers due to a false VREG over-voltage.

5.26 MM912_634 - Analog Die Trimming



A trimming option is implemented to increase some device parameter accuracy. As the MM912_634 analog die is exclusively combined with a FLASH- MCU, the required trimming values can be calculated during the final test of the device, and stored to a fixed position in the FLASH memory. During start-up of the system, the trimming values have to be copied into the MM912_634 analog die trimming registers.

The trimming registers will maintain their content during Low Power mode, Reset will set the default value.

5.26.1 Memory Map and Register Definition

5.26.1.1 Module Memory Map

There are four trimming registers implemented (CTR0…CTR3), with CTR2 being reserved for future use. The following table shows the registers used.

Table 212. MM912_634 Analog Die Trimming Registers

Offset	Name		7	6	5	4	3	2	1	0
0xF0	CTR0	R	LINTRE	LINTR	WDCTRE	CTR0_4	CTR0_3	WDCTR2	WDCTR1	WDCTR0
	Trimming Reg 0	W								
0xF1	CTR1	R	BGTRE	CTR1_6	BGTRIMUP	BGTRIMDN	IREFTRE	IREFTR2	IREFTR1	IREFTR0
	Trimming Reg 1	W								
0xF2	CTR2	R	0	0	0	SLPBGTRE	SLPBG_LOCK	SLPBGTR2	SLPBGTR1	SLPBGTR0
	Trimming Reg 2	W								
0xF3	CTR3	R	OFFCTRE	OFFCTR2	OFFCTR1	OFFCTR0	CTR3_E	CTR3_2	CTR3_1	CTR3_0
	Trimming Reg 3	W								

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

At system startup, the trimming information have to be copied from the MCU IFR Flash location to the corresponding MM912_634 analog die trimming registers. The following table shows the register correlation.

Table 213. MM912_634 - MCU vs. Analog Die Trimming Register Correlation

Name	MCU IFR Address	Analog Offset[149]
CTR0	0x0_40C0	0xF0
CTR1	0x0_40C1	0xF1
CTR2	0x0_40C2	0xF2
CTR3	0x0_40C3	0xF3

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

NOTE

Two word (16-Bit) transfers including CTR2 are recommended at system startup. The IFR register has to be enabled for reading (Program Page Index Register (PPAGE))

NOTE

To trim the bg1p25sleep there is two steps:

Step 1: First choose the right trim step by adjusting SLPBGTR[2:0] with SLPBGTRE=1, SLPBG_LOCK bit has to stay at 0.

Step 2: Once the trim value is known, correct SLPBGTR[2:0], SLPBGTRE and SLPBG_LOCK bits have to be set at the same time to apply and lock the trim. Once the trim is locked, no other trim on the parameter is possible.

5.27.3.8 Serial Peripheral Interface Module (SPI)

- Configurable 8 or 16-bit data size
- Full-duplex or single-wire bidirectional
- Double-buffered transmit and receive
- Master or slave mode
- MSB-first or LSB-first shifting
- Serial clock phase and polarity options

5.27.3.9 On-Chip Voltage Regulator (VREG)

- Linear voltage regulator with bandgap reference
- Low-voltage detect (LVD) with low-voltage interrupt (LVI)
- Power-on reset (POR) circuit
- Low-voltage reset (LVR)

5.27.3.10 Background Debug (BDM)

- Non-intrusive memory access commands
- Supports in-circuit programming of on-chip nonvolatile memory

5.27.3.11 Debugger (DBG)

- Trace buffer with depth of 64 entries
- Three comparators (A, B and C)
  - Comparator A compares the full address bus and full 16-bit data bus
  - Exact address or address range comparisons
- Two types of comparator matches
  - Tagged: This matches just before a specific instruction begins execution
  - Force: This is valid on the first instruction boundary after a match occurs
- Four trace modes
- Four stage state sequencer

5.27.3.12 Die to Die Initiator (D2DI)

- Up to 2.0 Mbyte/s data rate
- Configurable 4-bit or 8-bit wide data path

Figure 41 shows MC9S12I64 CPU and BDM local address translation to the global memory map. It indicates also the location of the internal resources in the memory map. The whole 256 k global memory space is visible through the P-Flash window located in the 64 k local memory map located at 0x8000 - 0xBFFF using the PPAGE register.

5.29.1.4.2 Security

S12I derives can be secured to prohibit external access to the on-chip P-Flash. The S12PMMC module determines the access permissions to the on-chip memories in secured and unsecured state.

5.29.1.5 Block Diagram

Figure 44 shows a block diagram of the S12PMMC.

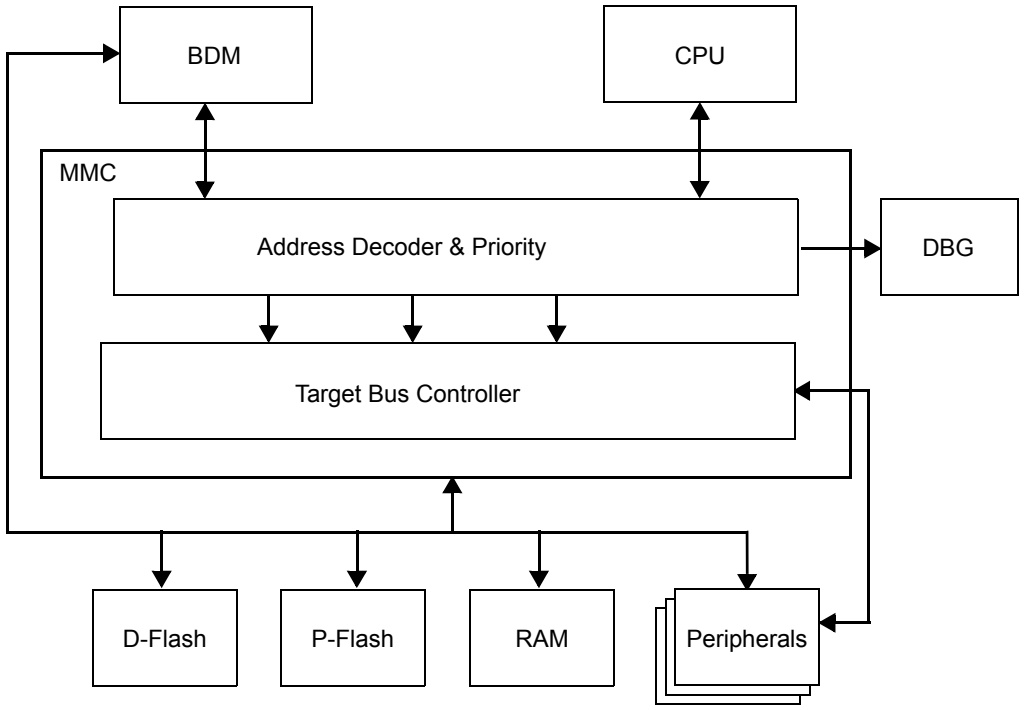


Figure 44. S12PMMC Block Diagram

5.29.2 External Signal Description

The S12PMMC uses two external pins to determine the devices operating mode: RESET and MODC (Table 247) See Device User Guide (DUG) for the mapping of these signals to device pins.

Table 247. External System Pins Associated With S12PMMC

Pin Name	Pin Functions	Description
$\overline{\text{RESET}}$ (See DUG)	$\overline{\text{RESET}}$	The $\overline{\text{RESET}}$ pin is used the select the MCU's operating mode.
$\overline{\text{MODC}}$ (See DUG)	MODC	The MODC pin is captured at the rising edge of the $\overline{\text{RESET}}$ pin. The captured value determines the MCU's operating mode.

5.29.3 Memory Map and Registers

5.29.3.1 Module Memory Map

A summary of the registers associated with the S12PMMC block is shown in Table 248. Detailed descriptions of the registers and bits are given in the subsections that follow.

5.30.4 Functional Description

The INT module processes all exception requests to be serviced by the CPU module. These exceptions include interrupt vector requests and reset vector requests. Each of these exception types and their overall priority level is discussed in the subsections below.

5.30.4.1 S12S Exception Requests

The CPU handles both reset requests and interrupt requests. A priority decoder is used to evaluate the priority of pending interrupt requests.

5.30.4.2 Interrupt Prioritization

The INT module contains a priority decoder to determine the priority for all interrupt requests pending for the CPU. If more than one interrupt request is pending, the interrupt request with the higher vector address wins the prioritization.

The following conditions must be met for an I bit maskable interrupt request to be processed.

1. The local interrupt enabled bit in the peripheral module must be set.
2. The I bit in the condition code register (CCR) of the CPU must be cleared.
3. There is no SWI, TRAP, or X bit maskable request pending.

NOTE

All non I bit maskable interrupt requests always have higher priority than the I bit maskable interrupt requests. If the X bit in the CCR is cleared, it is possible to interrupt an I bit maskable interrupt by an X bit maskable interrupt. It is possible to nest non maskable interrupt requests, e.g., by nesting SWI or TRAP calls.

Since an interrupt vector is only supplied at the time when the CPU requests it, it is possible that a higher priority interrupt request could override the original interrupt request 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 interrupt request first, before the original interrupt request is processed.

If the interrupt source is unknown (for example, in the case where an interrupt request becomes inactive after the interrupt has been recognized, but prior to the CPU vector request), the vector address supplied to the CPU will default to that of the spurious interrupt vector.

NOTE

Care must be taken to ensure that all interrupt requests remain active until the system begins execution of the applicable service routine; otherwise, the exception request may not get processed at all or the result may be a spurious interrupt request (vector at address (vector base + 0x0080)).

5.30.4.3 Reset Exception Requests

The INT module supports three system reset exception request types (please refer to the Clock and Reset generator module for details):

1. Pin reset, power-on reset or illegal address reset, low voltage reset (if applicable)
2. Clock monitor reset request
3. COP watchdog reset request

5.30.4.4 Exception Priority

The priority (from highest to lowest) and address of all exception vectors issued by the INT module upon request by the CPU is shown in Table 259.

Table 259. Exception Vector Map and Priority

Vector Address[172]	Source
0xFFFE	Pin reset, power-on reset, illegal address reset, low voltage reset (if applicable)
0xFFFC	Clock monitor reset
0xFFFA	COP watchdog reset
(Vector base + 0x00F8)	Unimplemented opcode trap
(Vector base + 0x00F6)	Software interrupt instruction (SWI) or BDM vector request

Table 266. BDMPPR Field Descriptions

Field	Description
7 BPAE	**BDM Program Page Access Enable Bit** — BPAE enables program page access for BDM hardware and firmware read/write instructions. The BDM hardware commands used to access the BDM registers (READ_BD and WRITE_BD) can not be used for global accesses even if the BGAE bit is set. 0 BDM Program Paging disabled 1 BDM Program Paging enabled
3–0 BPP[3:0]	**BDM Program Page Index Bits 3–0** — These bits define the selected program page. For more detailed information regarding the program page window scheme, please refer to the S12S_MMC Block Guide.

5.31.3.3 Family ID Assignment

The family ID is an 8-bit value located in the BDM ROM in active BDM (at global address: 0x3_FF0F). The read-only value is a unique family ID which is 0xC2 for devices with an HCS12S core.

5.31.4 Functional Description

The BDM receives and executes commands from a host via a single wire serial interface. There are two types of BDM commands: hardware and firmware commands.

Hardware commands are used to read and write target system memory locations and to enter active background debug mode, see Section 5.31.4.3, “BDM Hardware Commands". Target system memory includes all memory that is accessible by the CPU.

Firmware commands are used to read and write CPU resources and to exit from active background debug mode, see Section 5.31.4.4, “Standard BDM Firmware Commands". The CPU resources referred to are the accumulator (D), X index register (X), Y index register (Y), stack pointer (SP), and program counter (PC).

Hardware commands can be executed at any time and in any mode excluding a few exceptions as highlighted (see Section 5.31.4.3, “BDM Hardware Commands") and in secure mode (see Section 5.31.4.1, “Security"). BDM firmware commands can only be executed when the system is not secure and is in active background debug mode (BDM).

5.31.4.1 Security

If the user resets into special single chip mode with the system secured, a secured mode BDM firmware lookup table is brought into the map overlapping a portion of the standard BDM firmware lookup table. The secure BDM firmware verifies that the on-chip Flash EEPROM are erased. This being the case, the UNSEC and ENBDM bit will get set. The BDM program jumps to the start of the standard BDM firmware and the secured mode BDM firmware is turned off and all BDM commands are allowed. If the Flash do not verify as erased, the BDM firmware sets the ENBDM bit, without asserting UNSEC, and the firmware enters a loop. This causes the BDM hardware commands to become enabled, but does not enable the firmware commands. This allows the BDM hardware to be used to erase the Flash.

BDM operation is not possible in any other mode than special single chip mode when the device is secured. The device can only be unsecured via BDM serial interface in special single chip mode. For more information regarding security, please see the S12S_9SEC Block Guide.

5.31.4.2 Enabling and Activating BDM

The system must be in active BDM to execute standard BDM firmware commands. BDM can be activated only after being enabled. BDM is enabled by setting the ENBDM bit in the BDM status (BDMSTS) register. The ENBDM bit is set by writing to the BDM status (BDMSTS) register, via the single-wire interface, using a hardware command such as WRITE_BD_BYTE.

After being enabled, BDM is activated by one of the following[177]:

- Hardware BACKGROUND command
- CPU BGND instruction
- Breakpoint force or tag mechanism[178]

When BDM is activated, the CPU finishes executing the current instruction and then begins executing the firmware in the standard BDM firmware lookup table. When BDM is activated by a breakpoint, the type of breakpoint used determines if BDM becomes active before or after execution of the next instruction.

Note:

177. BDM is enabled and active immediately out of special single-chip reset.
178. This method is provided by the S12S_DBG module.

5.31.4.7 Serial Interface Hardware Handshake Protocol

BDM commands that require CPU execution are ultimately treated at the MCU bus rate. Since the BDM clock source can be modified , it is very helpful to provide a handshake protocol in which the host could determine when an issued command is executed by the CPU. . The alternative is to always wait the amount of time equal to the appropriate number of cycles at the slowest possible rate the clock could be running. This sub-section will describe the hardware handshake protocol.

The hardware handshake protocol signals to the host controller when an issued command was successfully executed by the target. This protocol is implemented by a 16 serial clock cycle low pulse followed by a brief speedup pulse in the BKGD pin. This pulse is generated by the target MCU when a command, issued by the host, has been successfully executed (see Figure 58). This pulse is referred to as the ACK pulse. After the ACK pulse has finished: the host can start the bit retrieval if the last issued command was a read command, or start a new command if the last command was a write command or a control command (BACKGROUND, GO, GO_UNTIL(182) or TRACE1). The ACK pulse is not issued earlier than 32 serial clock cycles after the BDM command was issued. The end of the BDM command is assumed to be the 16th tick of the last bit. This minimum delay assures enough time for the host to perceive the ACK pulse. Note also that, there is no upper limit for the delay between the command and the related ACK pulse, since the command execution depends upon the CPU bus, which in some cases could be very slow due to long accesses taking place.This protocol allows a great flexibility for the POD designers, since it does not rely on any accurate time measurement or short response time to any event in the serial communication.

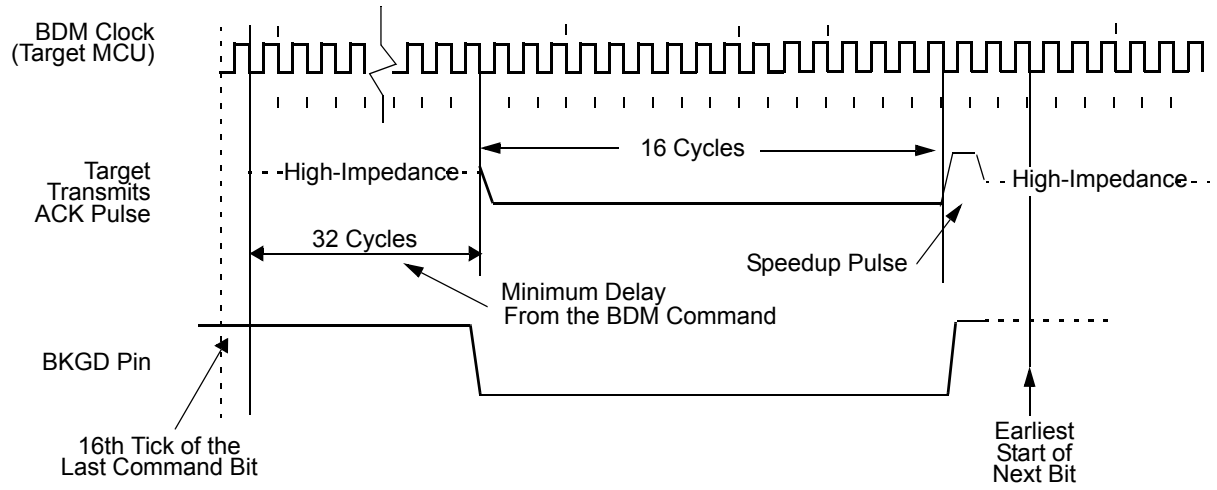


Figure 58. Target Acknowledge Pulse (ACK)

NOTE

If the ACK pulse was issued by the target, the host assumes the previous command was executed. If the CPU enters stop prior to executing a hardware command, the ACK pulse will not be issued meaning that the BDM command was not executed. After entering stop mode, the BDM command is no longer pending.

Figure 59 shows the ACK handshake protocol in a command level timing diagram. The READ_BYTE instruction is used as an example. First, the 8-bit instruction opcode is sent by the host, followed by the address of the memory location to be read. The target BDM decodes the instruction. A bus cycle is grabbed (free or stolen) by the BDM and it executes the READ_BYTE operation. Having retrieved the data, the BDM issues an ACK pulse to the host controller, indicating that the addressed byte is ready to be retrieved. After detecting the ACK pulse, the host initiates the byte retrieval process. Note that data is sent in the form of a word and the host needs to determine which is the appropriate byte based on whether the address was odd or even.

Table 299. Comparator Register Layout

0x0028	CONTROL	Read/Write	Comparators A,B and C
0x0029	ADDRESS HIGH	Read/Write	Comparators A,B and C
0x002A	ADDRESS MEDIUM	Read/Write	Comparators A,B and C
0x002B	ADDRESS LOW	Read/Write	Comparators A,B and C
0x002C	DATA HIGH COMPARATOR	Read/Write	Comparator A only
0x002D	DATA LOW COMPARATOR	Read/Write	Comparator A only
0x002E	DATA HIGH MASK	Read/Write	Comparator A only
0x002F	DATA LOW MASK	Read/Write	Comparator A only

5.32.3.2.8.1 Debug Comparator Control Register (DBGXCTL)

The contents of this register bits 7 and 6 differ depending upon which comparator registers are visible in the 8-byte window of the DBG module register address map.

Table 300. Debug Comparator Control Register DBGACTL (Comparator A)

Address: 0x0028

	7	6	5	4	3	2	1	0
R W	SZE	SZ	TAG	BRK	RW	RWE	NDB	COMPE
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Table 301. Debug Comparator Control Register DBGBCTL (Comparator B)

Address: 0x0028

	7	6	5	4	3	2	1	0
R W	SZE	SZ	TAG	BRK	RW	RWE	0	COMPE
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Table 302. Debug Comparator Control Register DBGCCTL (Comparator C)

Address: 0x0028

	7	6	5	4	3	2	1	0
R W	0	0	TAG	BRK	RW	RWE	0	COMPE
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Read: DBGACTL if COMRV[1:0] = 00
DBGBCTL if COMRV[1:0] = 01
DBGCCTL if COMRV[1:0] = 10

Write: DBGACTL if COMRV[1:0] = 00 and DBG not armed
DBGBCTL if COMRV[1:0] = 01 and DBG not armed
DBGCCTL if COMRV[1:0] = 10 and DBG not armed

5.38.3.2.11 Reserved Register CPMUTEST1

NOTE

This reserved register is designed for factory test purposes only, and is not intended for general user access. Writing to this register when in Special Mode can alter the S12CPMU's functionality.

Table 362. Reserved Register (CPMUTEST1)

0x003E

	7	6	5	4	3	2	1	0
R	pfd_force_en0	cpmu_test_clk_en0	cpmu_test_clk_sel[1]0	cpmu_test_clk_sel[0]0	osc_lcp_monitor_disable0	osc_lcp_extsqw_enable0	pfd_force_up0	pfd_force_down0
W								
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Read: Anytime

Write: Only in Special Mode

Table 363. CPMUTEST1 Field Descriptions

Field	Description
7 pfd_force_en	**Phase Detector Force Enable Bit**— This bit breaks the PLL feedback loop and allows force of phase detector via pfd_force_up or pfd_force_down bit or cpmu_test_xfc pin or fc_force_en. 0 Normal functionality of Phase Detector using REFCLK and FBCLK. 1 Phase detector de-connected from REFCLK and FBCLK (PLL loop open).
6 cpmu_test_clk_en	**CPMU test clock enable Bit**— This bits routes the clock selected by cpmu_test_clk_sel[1:0] to external pin cpmu_test_clk. 0 CPMU test clock not observable. 1 CPMU test clock observable at external pin.
5, 4 cpmu_test_clk_sel[1:0]	**CPMU test clock select Bits**— These bits select the CPMU test clock to be observed on external pin cpmu_test_clk for test or characterization purposes. 00 = IRCCLK, 01=OSCCLK, 10=VCOCLK, 11=VCOCLK_DIV4.
3 osc_lcp_monitor_disable	**Oscillator clock monitor disable Bit** — to disable the clock monitor in special single chip mode. 0 Clock monitor always enabled with OSCE=1. 1 Clock monitor disabled regardless of OSCE Bit.
2 osc_lcp_extsqw_enable	**Oscillator external square wave enable Bit** — Drives directly osc_lcp_extsqw_enable input of OSCLCP hardmacro.
1 pfd_force_up	**Phase Detector Force Up Bit** — If pfd_force_en=1, this bits force the PLL charge pump to drive the internal filter voltage down, that is VCOCLK frequency goes up. Using this test feature make sure that only one source is driving the internal filter node (FC). So for this case write xfc_en=fc_force_en=pfd_force_down=0. 0 No effect. 1 If pfd_force_en=1 then the charge pump continuously drives internal filter node down.
0 pfd_force_up	**Phase Detector Force Down Bit** — If pfd_force_en=1, this bits force the PLL charge pump to drive the internal filter voltage up, that is VCOCLK frequency goes down. Using this test feature make sure that only one source is driving the internal filter node (FC). So for this case write xfc_en=fc_force_en=pfd_force_up=0. 0 No effect. 1 If pfd_force_en=1 then the charge pump continuously drives internal filter node up.

5.39.4.6 Error Conditions

The SPI has one error condition:

Mode fault error

5.39.4.6.1 Mode Fault Error

If the $\overline{SS}$ input becomes low while the SPI is configured as a master, it indicates a system error where more than one master may be trying to drive the MOSI and SCK lines simultaneously. This condition is not permitted in normal operation, the MODF bit in the SPI status register is set automatically, provided the MODFEN bit is set.

In the special case where the SPI is in master mode and MODFEN bit is cleared, the $\overline{SS}$ pin is not used by the SPI. In this special case, the mode fault error function is inhibited and MODF remains cleared. In case the SPI system is configured as a slave, the $\overline{SS}$ pin is a dedicated input pin. Mode fault error doesn't occur in slave mode.

If a mode fault error occurs, the SPI is switched to slave mode, with the exception that the slave output buffer is disabled. So SCK, MISO, and MOSI pins are forced to be high impedance inputs to avoid any possibility of conflict with another output driver. A transmission in progress is aborted and the SPI is forced into idle state.

If the mode fault error occurs in the bidirectional mode for a SPI system configured in master mode, output enable of the MOMI (MOSI in bidirectional mode) is cleared if it was set. No mode fault error occurs in the bidirectional mode for SPI system configured in slave mode.

The mode fault flag is cleared automatically by a read of the SPI status register (with MODF set) followed by a write to SPI control register 1. If the mode fault flag is cleared, the SPI becomes a normal master or slave again.

NOTE

If a mode fault error occurs and a received data byte is pending in the receive shift register, this data byte will be lost.

5.39.4.7 Low Power Mode Options

5.39.4.7.1 SPI in Run Mode

In run mode with the SPI system enable (SPE) bit in the SPI control register clear, the SPI system is in a low-power, disabled state. SPI registers remain accessible, but clocks to the core of this module are disabled.

5.39.4.7.2 SPI in Stop Mode

Stop mode is dependent on the system. The SPI enters stop mode when the module clock is disabled (held high or low). If the SPI is in master mode and exchanging data when the CPU enters stop mode, the transmission is frozen until the CPU exits stop mode. After stop, data to and from the external SPI is exchanged correctly. In slave mode, the SPI will stay synchronized with the master.

The stop mode is not dependent on the SPISWAI bit.

5.39.4.7.3 Reset

The reset values of registers and signals are described in Section 5.39.3, “Memory Map and Register Definition", which details the registers and their bit fields.

- If a data transmission occurs in slave mode after reset without a write to SPIDR, it will transmit garbage, or the data last received from the master before the reset.
- Reading from the SPIDR after reset will always read zeros.

5.39.4.7.4 Interrupts

The SPI only originates interrupt requests when SPI is enabled (SPE bit in SPICR1 set). The following is a description of how the SPI makes a request and how the MCU should acknowledge that request. The interrupt vector offset and interrupt priority are chip dependent.

The interrupt flags MODF, SPIF, and SPTEF are logically ORed to generate an interrupt request.

Table 430. FSTAT Field Descriptions (continued)

Field	Description
2 RSVD	**Reserved Bit** — This bit is reserved and always reads 0.
1–0 MGSTAT[1:0]	**Memory Controller Command Completion Status Flag** — One or more MGSTAT flag bits are set if an error is detected during execution of a Flash command or during the Flash reset sequence. See Section 5.40.4.5, “Flash Command Description" and Section 5.40.6, “Initialization" for details.

5.40.3.2.8 Flash Error Status Register (FERSTAT)

The FERSTAT register reflects the error status of internal Flash operations.

Table 431. Flash Error Status Register (FERSTAT)

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

= Unimplemented or Reserved

All flags in the FERSTAT register are readable and only writable to clear the flag.

Table 432. FERSTAT Field Descriptions

Field	Description
1 DFDIF	**Double Bit Fault Detect Interrupt Flag** — The setting of the DFDIF flag indicates that a double bit fault was detected in the stored parity and data bits during a Flash array read operation, or that a Flash array read operation was attempted on a Flash block that was under a Flash command operation.(220) The DFDIF flag is cleared by writing a 1 to DFDIF. Writing a 0 to DFDIF has no effect on DFDIF. 0 No double bit fault detected 1 Double bit fault detected or an invalid Flash array read operation attempted
0 SFDIF	**Single Bit Fault Detect Interrupt Flag** — With the IGNSF bit in the FCNFG register clear, the SFDIF flag indicates that a single bit fault was detected in the stored parity and data bits during a Flash array read operation, or that a Flash array read operation was attempted on a Flash block that was under a Flash command operation. (220) The SFDIF flag is cleared by writing a 1 to SFDIF. Writing a 0 to SFDIF has no effect on SFDIF. 0 No single bit fault detected 1 Single bit fault detected and corrected or an invalid Flash array read operation attempted

Note:

220. The single bit fault and double bit fault flags are mutually exclusive for parity errors (an ECC fault occurrence can be either single fault or double fault, but never both). A simultaneous access collision (read attempted while command running) is indicated when both SFDIF and DFDIF flags are high.

5.40.3.2.9 P-Flash Protection Register (FPROT)

The FPROT register defines which P-Flash sectors are protected against program and erase operations.

Table 433. P-Flash Protection Register (FPROT)

	7	6	5	4	3	2	1	0
R	FPOPEN	RNV6	FPHDIS	FPHS[1:0]		FPLDIS	FPLS[1:0]	
W								
Reset	F	F	F	F	F	F	F	F

= Unimplemented or Reserved