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	HCS12X
Core Size	16-Bit
Speed	40MHz
Connectivity	CANbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	91
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	1.72V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	112-LQFP
Supplier Device Package	112-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s12xs128j1calr

1.3 System Clock Description

The clock and reset generator module (CRG) provides the internal clock signals for the core and all peripheral modules. Figure 1-6 shows the clock connections from the CRG to all modules.

Consult the S12XECRG section for details on clock generation.

NOTE

The XS family uses the XE family clock and reset generator module. Therefore all CRG references are related to S12XECRG.

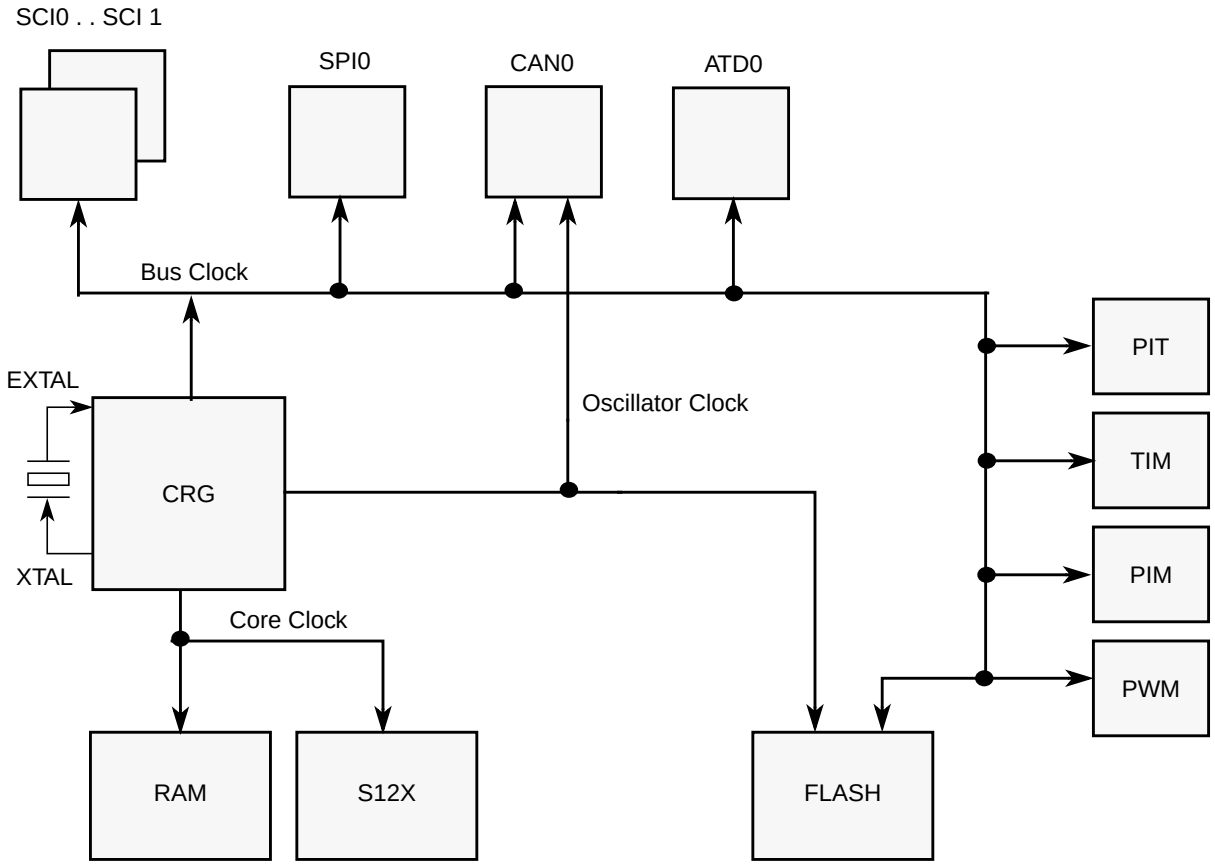


Figure 1-6. Clock Connections

The system clock can be supplied in several ways enabling a range of system operating frequencies to be supported:

- The on-chip phase locked loop (PLL)
- the PLL self clocking
- the oscillator

2.3.33 PIM Reserved Register

Address 0x024F — Access: User read[1]

	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 u = Unaffected by reset

Figure 2-31. PIM Reserved Register

1 Read: Always reads 0x00
Write: Unimplemented

2.3.34 Port M Data Register (PTM)

Address 0x0250 — Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	PTM7	PTM6	PTM5	PTM4	PTM3	PTM2	PTM1	PTM0
Altern. Function	—	—	(SCK0)	(MOSI0)	($\overline{SS0}$)	(MISO0)	TXCAN0	RXCAN0
	—	—	—	—	—	—	(TXD1)	(RXD1)
Reset	0	0	0	0	0	0	0	0

Figure 2-32. Port M Data Register (PTM)

1 Read: Anytime, the data source depends on the data direction value
Write: Anytime

Table 2-30. PTM Register Field Descriptions

Field	Description
7-6 PTM	**Port M general purpose input/output data**—Data Register When not used with the alternative function, the associated pin can be used as general purpose I/O. In general purpose output mode the register bit value is driven to the pin. If the associated data direction bit is set to 1, a read returns the value of the port register bit, otherwise the buffered pin input state is read.
5 PTM	**Port M general purpose input/output data**—Data Register, routed SPI0 SCK input/output When not used with the alternative function, the associated pin can be used as general purpose I/O. In general purpose output mode the register bit value is driven to the pin. If the associated data direction bit is set to 1, a read returns the value of the port register bit, otherwise the buffered pin input state is read. • The SPI0 function takes precedence over the general purpose I/O function if enabled.

2.3.67 Port AD0 Data Register 1 (PT1AD0)

Address 0x0271 Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	PT1AD07	PT1AD06	PT1AD05	PT1AD04	PT1AD03	PT1AD02	PT1AD01	PT1AD00
Altern. Function	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0
Reset	0	0	0	0	0	0	0	0

Figure 2-65. Port AD0 Data Register 1 (PT1AD0)

1 Read: Anytime, the data source depends on the data direction value
Write: Anytime

Table 2-64. PT1AD0 Register Field Descriptions

Field	Description
7-0 PT1AD0	**Port AD0 general purpose input/output data**—Data Register, ATD AN analog input When not used with the alternative function, the associated pin can be used as general purpose I/O. In general purpose output mode the register bit value is driven to the pin. If the associated data direction bit is set to 1, a read returns the value of the port register bit, otherwise the buffered pin input state is read.

2.3.68 Port AD0 Data Direction Register 0 (DDR0AD0)

Address 0x0272 Access: User read/write[1]

	7	6	5	4	3	2	1	0
R W	DDR0AD07	DDR0AD06	DDR0AD05	DDR0AD04	DDR0AD03	DDR0AD02	DDR0AD01	DDR0AD00
Reset	0	0	0	0	0	0	0	0

Figure 2-66. Port AD0 Data Direction Register 0 (DDR0AD0)

1 Read: Anytime
Write: Anytime

Table 2-65. DDR0AD0 Register Field Descriptions

Field	Description
7-0 DDR0AD0	**Port AD0 data direction**— This bit determines whether the associated pin is an input or output. To use the digital input function the ATD Digital Input Enable Register (ATD0DIEN) has to be set to logic level “1”. 1 Associated pin configured as output 0 Associated pin configured as input

2.4.2.7 Wired-or mode register (WOMx)

If the pin is used as an output this register turns off the active high drive. This allows wired-or type connections of outputs.

2.4.2.8 Interrupt enable register (PIEx)

If the pin is used as an interrupt input this register serves as a mask to the interrupt flag to enable/disable the interrupt.

2.4.2.9 Interrupt flag register (PIFx)

If the pin is used as an interrupt input this register holds the interrupt flag after a valid pin event.

2.4.2.10 Module routing registers (MODRR, PTTRR)

These registers allow software re-configuration of the pinouts of the different package options for specific peripherals:

- MODRR supports the re-routing of the SCI1 and SPI0 pins to alternative ports
- PTTRR supports the re-routing of the PWM and TIM channels to alternative ports

2.4.3 Pins and Ports

NOTE

Please refer to the device pinout section to determine the pin availability in the different package options.

2.4.3.1 BKGD pin

The BKGD pin is associated with the BDM module.

During reset, the BKGD pin is used as MODC input.

2.4.3.2 Port A, B

Port A pins PA[7:0] and Port B pins PB[7:0] can be used for general-purpose I/O.

2.4.3.3 Port E

Port E is associated with the free-running clock outputs ECLK, ECLKX2 and interrupt inputs $\overline{\text{IRQ}}$ and $\overline{\text{XIRQ}}$.

Port E pins PE[7:2] can be used for either general-purpose I/O or with the alternative functions.

Port E pin PE[7] an be used for either general-purpose I/O or as the free-running clock ECLKX2 output running at the core clock rate.

Port E pin PE[4] an be used for either general-purpose I/O or as the free-running clock ECLK output running at the bus clock rate or at the programmed divided clock rate.

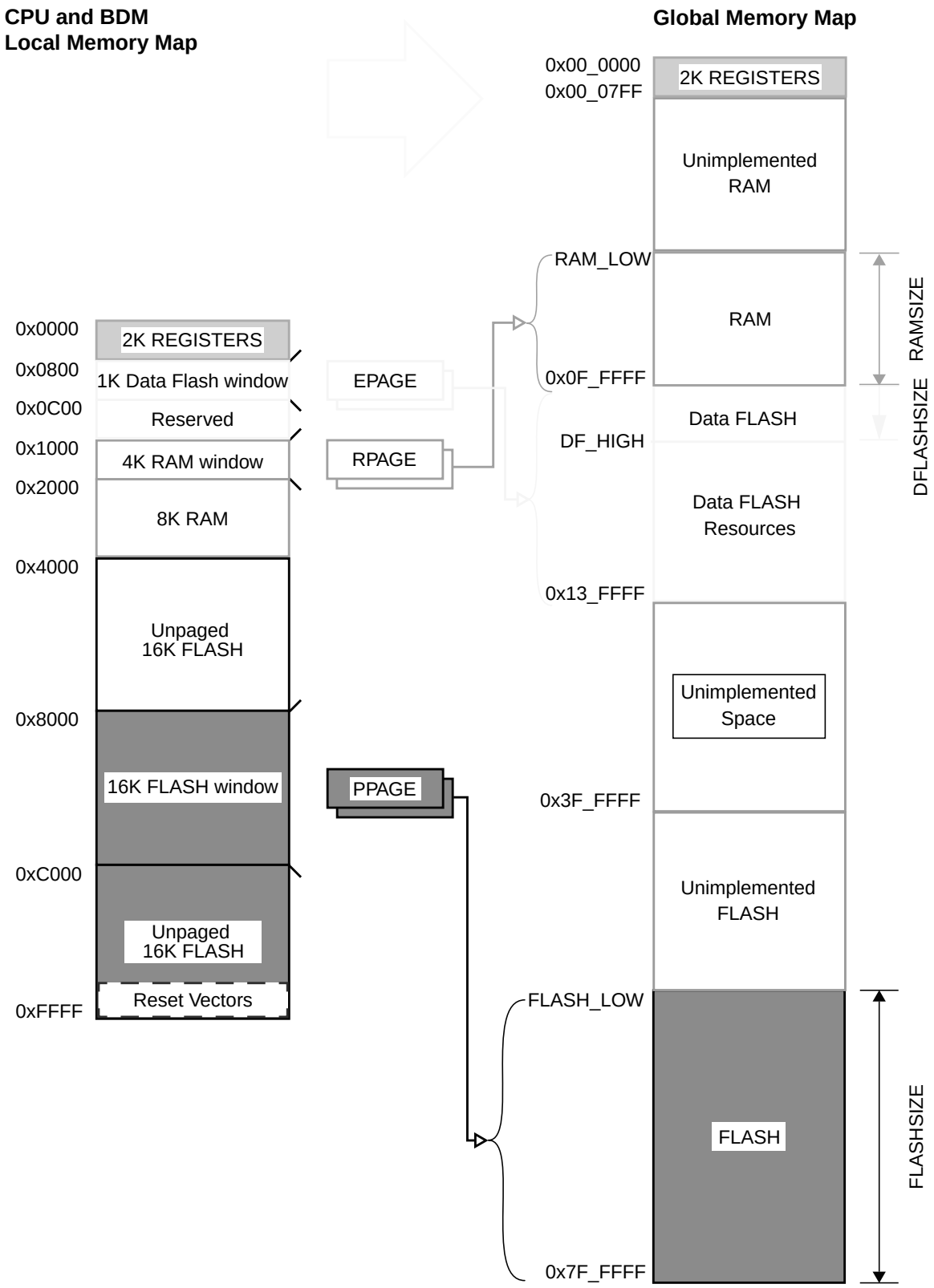


Figure 3-19. S12X CPU & BDM Global Address Mapping

GO_UNTIL command can not be aborted. Only the corresponding ACK pulse can be aborted by the SYNC command.

Although it is not recommended, the host could abort a pending BDM command by issuing a low pulse in the BKGD pin shorter than 128 serial clock cycles, which will not be interpreted as the SYNC command. The ACK is actually aborted when a negative edge is perceived by the target in the BKGD pin. The short abort pulse should have at least 4 clock cycles keeping the BKGD pin low, in order to allow the negative edge to be detected by the target. In this case, the target will not execute the SYNC protocol but the pending command will be aborted along with the ACK pulse. The potential problem with this abort procedure is when there is a conflict between the ACK pulse and the short abort pulse. In this case, the target may not perceive the abort pulse. The worst case is when the pending command is a read command (i.e., READ_BYTE). If the abort pulse is not perceived by the target the host will attempt to send a new command after the abort pulse was issued, while the target expects the host to retrieve the accessed memory byte. In this case, host and target will run out of synchronism. However, if the command to be aborted is not a read command the short abort pulse could be used. After a command is aborted the target assumes the next negative edge, after the abort pulse, is the first bit of a new BDM command.

NOTE

The details about the short abort pulse are being provided only as a reference for the reader to better understand the BDM internal behavior. It is not recommended that this procedure be used in a real application.

Since the host knows the target serial clock frequency, the SYNC command (used to abort a command) does not need to consider the lower possible target frequency. In this case, the host could issue a SYNC very close to the 128 serial clock cycles length. Providing a small overhead on the pulse length in order to assure the SYNC pulse will not be misinterpreted by the target. See Section 5.4.9, “SYNC — Request Timed Reference Pulse”.

Figure 5-13 shows a SYNC command being issued after a READ_BYTE, which aborts the READ_BYTE command. Note that, after the command is aborted a new command could be issued by the host computer.

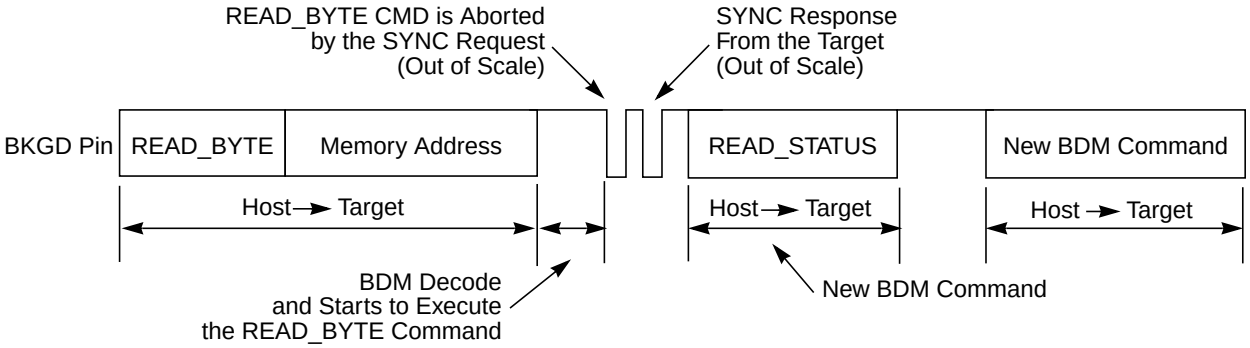


Figure 5-13. ACK Abort Procedure at the Command Level

NOTE

Figure 5-13 does not represent the signals in a true timing scale

Figure 5-14 shows a conflict between the ACK pulse and the SYNC request pulse. This conflict could occur if a POD device is connected to the target BKGD pin and the target is already in debug active mode.

Table 6-9. TRANGE Trace Range Encoding

TRANGE	Tracing Range
00	Trace from all addresses (No filter)
01	Trace only in address range from $00000 to Comparator D
10	Trace only in address range from Comparator C to $7FFFFF
11	Trace only in range from Comparator C to Comparator D

Table 6-10. TRCMOD Trace Mode Bit Encoding

TRCMOD	Description
00	Normal
01	Loop1
10	Detail
11	Pure PC

Table 6-11. TALIGN Trace Alignment Encoding

TALIGN	Description
00	Trigger at end of stored data
01	Trigger before storing data
10	Trace buffer entries before and after trigger
11	Reserved

6.3.2.4 Debug Control Register2 (DBGC2)

Address: 0x0023

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

= Unimplemented or Reserved

Figure 6-6. Debug Control Register2 (DBGC2)

Read: Anytime

Write: Anytime the module is disarmed.

This register configures the comparators for range matching.

Table 6-12. DBGC2 Field Descriptions

Field	Description
3–2 CDCM[1:0]	**C and D Comparator Match Control** — These bits determine the C and D comparator match mapping as described in Table 6-13.
1–0 ABCM[1:0]	**A and B Comparator Match Control** — These bits determine the A and B comparator match mapping as described in Table 6-14.

6.3.2.8.5 Debug Comparator Data High Register (DBGXDH)

Address: 0x002C

	7	6	5	4	3	2	1	0
R W	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
Reset	0	0	0	0	0	0	0	0

Figure 6-18. Debug Comparator Data High Register (DBGXDH)

Read: Anytime. See Table 6-26 for visible register encoding.

Write: If DBG not armed. See Table 6-26 for visible register encoding.

Table 6-32. DBGXAH Field Descriptions

Field	Description
7–0 Bits[15:8]	**Comparator Data High Compare Bits** — The Comparator data high compare bits control whether the selected comparator compares the data bus bits [15:8] to a logic one or logic zero. The comparator data compare bits are only used in comparison if the corresponding data mask bit is logic 1. This register is available only for comparators A and C. 0 Compare corresponding data bit to a logic zero 1 Compare corresponding data bit to a logic one

6.3.2.8.6 Debug Comparator Data Low Register (DBGXDL)

Address: 0x002D

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

Figure 6-19. Debug Comparator Data Low Register (DBGXDL)

Read: Anytime. See Table 6-26 for visible register encoding.

Write: If DBG not armed. See Table 6-26 for visible register encoding.

Table 6-33. DBGXDL Field Descriptions

Field	Description
7–0 Bits[7:0]	**Comparator Data Low Compare Bits** — The Comparator data low compare bits control whether the selected comparator compares the data bus bits [7:0] to a logic one or logic zero. The comparator data compare bits are only used in comparison if the corresponding data mask bit is logic 1. This register is available only for comparators A and C. 0 Compare corresponding data bit to a logic zero 1 Compare corresponding data bit to a logic one

8.4.3.3 Stop Mode

All clocks are stopped in STOP mode, dependent of the setting of the PCE, PRE and PSTP bit. The oscillator is disabled in STOP mode unless the PSTP bit is set. If the PRE or PCE bits are set, the RTI or COP continues to run in Pseudo Stop Mode. In addition to disabling system and core clocks the S12XECRG requests other functional units of the MCU (e.g. voltage-regulator) to enter their individual power saving modes (if available).

If the PLLSEL bit is still set when entering Stop Mode, the S12XECRG will switch the system and core clocks to OSCCLK by clearing the PLLSEL bit. Then the S12XECRG disables the IPLL, disables the core clock and finally disables the remaining system clocks.

If Pseudo Stop Mode is entered from Self-Clock Mode the S12XECRG will continue to check the clock quality until clock check is successful. In this case the IPLL and the voltage regulator (VREG) will remain enabled. If Full Stop Mode (PSTP = 0) is entered from Self-Clock Mode the ongoing clock quality check will be stopped. A complete timeout window check will be started when Stop Mode is left again.

There are two ways to restart the MCU from Stop Mode:

1. Any reset
2. Any interrupt

If the MCU is woken-up from Full Stop Mode by an interrupt and the fast wake-up feature is enabled (FSTWKP=1 and SCME=1), the system will immediately (no clock quality check) resume operation in Self-Clock Mode (see Section 8.4.1.4, “Clock Quality Checker”). The SCMIF flag will not be set for this special case. The system will remain in Self-Clock Mode with oscillator disabled until FSTWKP bit is cleared. The clearing of FSTWKP will start the oscillator and the clock quality check. If the clock quality check is successful, the S12XECRG will switch all system clocks to oscillator clock. The SCMIF flag will be set. See application examples in Figure 8-19 and Figure 8-20.

Because the IPLL has been powered-down during Stop Mode the PLLSEL bit is cleared and the MCU runs on OSCCLK after leaving Stop-Mode. The software must manually set the PLLSEL bit again, in order to switch system and core clocks to the PLLCLK.

NOTE

In Full Stop Mode or Self-Clock Mode caused by the fast wake-up feature the clock monitor and the oscillator are disabled.

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000	ATDCTL0	R	Reserved	0	0	0	WRAP3	WRAP2	WRAP1	WRAP0
		W								
0x0001	ATDCTL1	R	ETRIGSEL	SRES1	SRES0	SMP_DIS	ETRIGCH3	ETRIGCH2	ETRIGCH1	ETRIGCH0
		W								
0x0002	ATDCTL2	R	0	AFFC	ICLKSTP	ETRIGLE	ETRIGP	ETRIGE	ASCIE	ACMPIE
		W								
0x0003	ATDCTL3	R	DJM	S8C	S4C	S2C	S1C	FIFO	FRZ1	FRZ0
		W								
0x0004	ATDCTL4	R	SMP2	SMP1	SMP0	PRS[4:0]				
		W								
0x0005	ATDCTL5	R	0	SC	SCAN	MULT	CD	CC	CB	CA
		W								
0x0006	ATDSTAT0	R	SCF	0	ETORF	FIFOR	CC3	CC2	CC1	CC0
		W								
0x0007	Unimple-mented	R	0	0	0	0	0	0	0	0
		W								
0x0008	ATDCMPEH	R	CMPE[15:8]							
		W								
0x0009	ATDCMPEL	R	CMPE[7:0]							
		W								
0x000A	ATDSTAT2H	R	CCF[15:8]							
		W								
0x000B	ATDSTAT2L	R	CCF[7:0]							
		W								
0x000C	ATDDIENH	R	IEN[15:8]							
		W								
0x000D	ATDDIENL	R	IEN[7:0]							
		W								
0x000E	ATDCMPHTH	R	CMPHT[15:8]							
		W								
0x000F	ATDCMPHTL	R	CMPHT[7:0]							
		W								
0x0010	ATDDR0	R	See Section 10.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 10.3.2.12.2, “Right Justified Result Data (DJM=1)”							
		W								
0x0012	ATDDR1	R	See Section 10.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 10.3.2.12.2, “Right Justified Result Data (DJM=1)”							
		W								
0x0014	ATDDR2	R	See Section 10.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 10.3.2.12.2, “Right Justified Result Data (DJM=1)”							
		W								
0x0016	ATDDR3	R	See Section 10.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 10.3.2.12.2, “Right Justified Result Data (DJM=1)”							
		W								
0x0018	ATDDR4	R	See Section 10.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 10.3.2.12.2, “Right Justified Result Data (DJM=1)”							
		W								
0x001A	ATDDR5	R	See Section 10.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 10.3.2.12.2, “Right Justified Result Data (DJM=1)”							
		W								
0x001C	ATDDR6	R	See Section 10.3.2.12.1, “Left Justified Result Data (DJM=0)” and Section 10.3.2.12.2, “Right Justified Result Data (DJM=1)”							
		W								

= Unimplemented or Reserved

Figure 10-3. ADC12B16C Register Summary (Sheet 1 of 2)

Table 11-7. Baud Rate Prescaler

BRP5	BRP4	BRP3	BRP2	BRP1	BRP0	Prescaler value (P)
0	0	0	0	0	0	1
0	0	0	0	0	1	2
0	0	0	0	1	0	3
0	0	0	0	1	1	4
:	:	:	:	:	:	:
1	1	1	1	1	1	64

11.3.2.4 MSCAN Bus Timing Register 1 (CANBTR1)

The CANBTR1 register configures various CAN bus timing parameters of the MSCAN module.

Module Base + 0x0003 Access: User read/write(1)

	7	6	5	4	3	2	1	0
R W	SAMP	TSEG22	TSEG21	TSEG20	TSEG13	TSEG12	TSEG11	TSEG10
Reset:	0	0	0	0	0	0	0	0

Figure 11-7. MSCAN Bus Timing Register 1 (CANBTR1)

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

Table 11-8. CANBTR1 Register Field Descriptions

Field	Description
7 SAMP	**Sampling** — This bit determines the number of CAN bus samples taken per bit time. 0 One sample per bit. 1 Three samples per bit(1). If SAMP = 0, the resulting bit value is equal to the value of the single bit positioned at the sample point. If SAMP = 1, the resulting bit value is determined by using majority rule on the three total samples. For higher bit rates, it is recommended that only one sample is taken per bit time (SAMP = 0).
6-4 TSEG2[2:0]	**Time Segment 2** — Time segments within the bit time fix the number of clock cycles per bit time and the location of the sample point (see Figure 11-44). Time segment 2 (TSEG2) values are programmable as shown in Table 11-9.
3-0 TSEG1[3:0]	**Time Segment 1** — Time segments within the bit time fix the number of clock cycles per bit time and the location of the sample point (see Figure 11-44). Time segment 1 (TSEG1) values are programmable as shown in Table 11-10.

1. In this case, PHASE_SEG1 must be at least 2 time quanta (Tq).

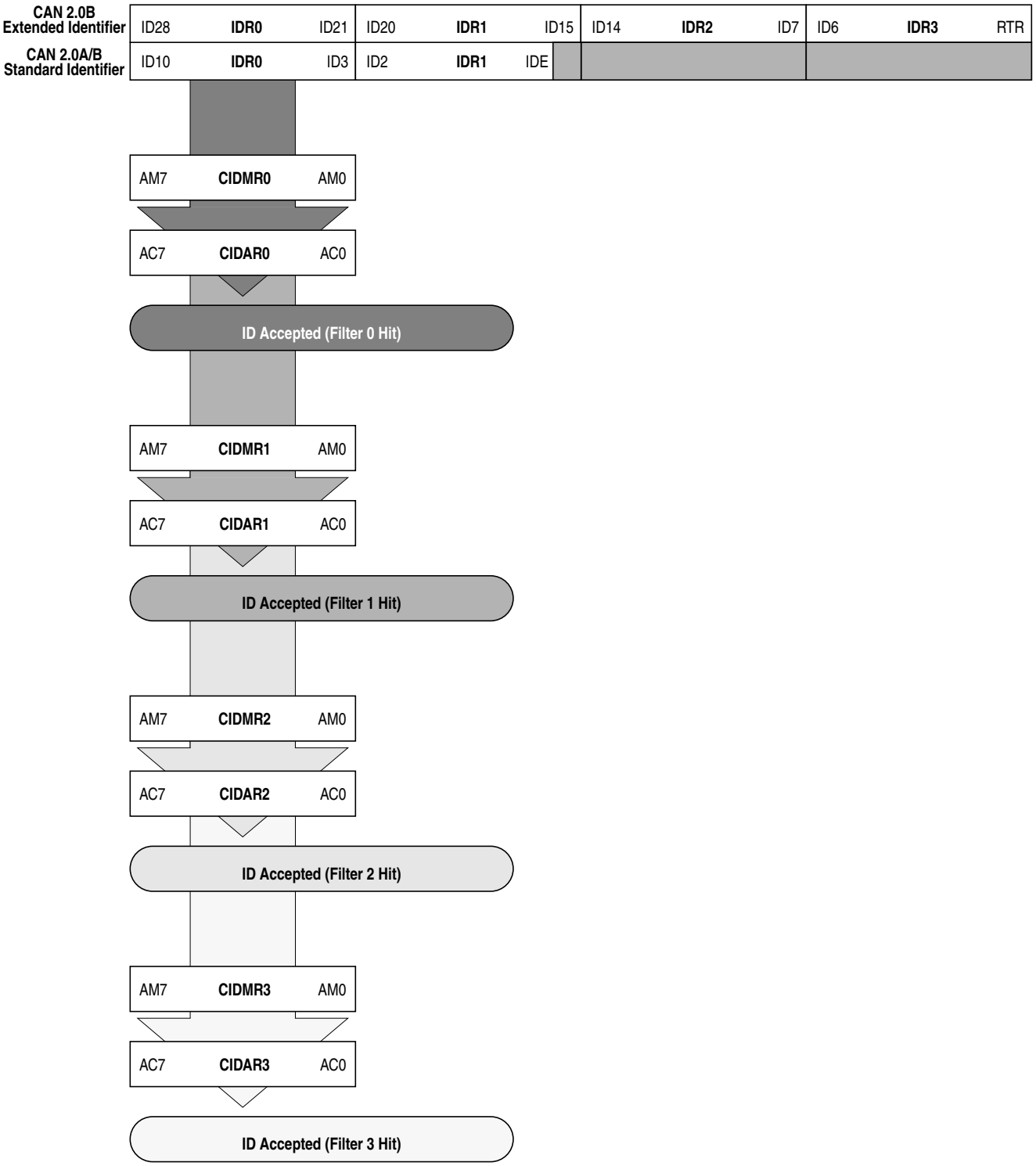
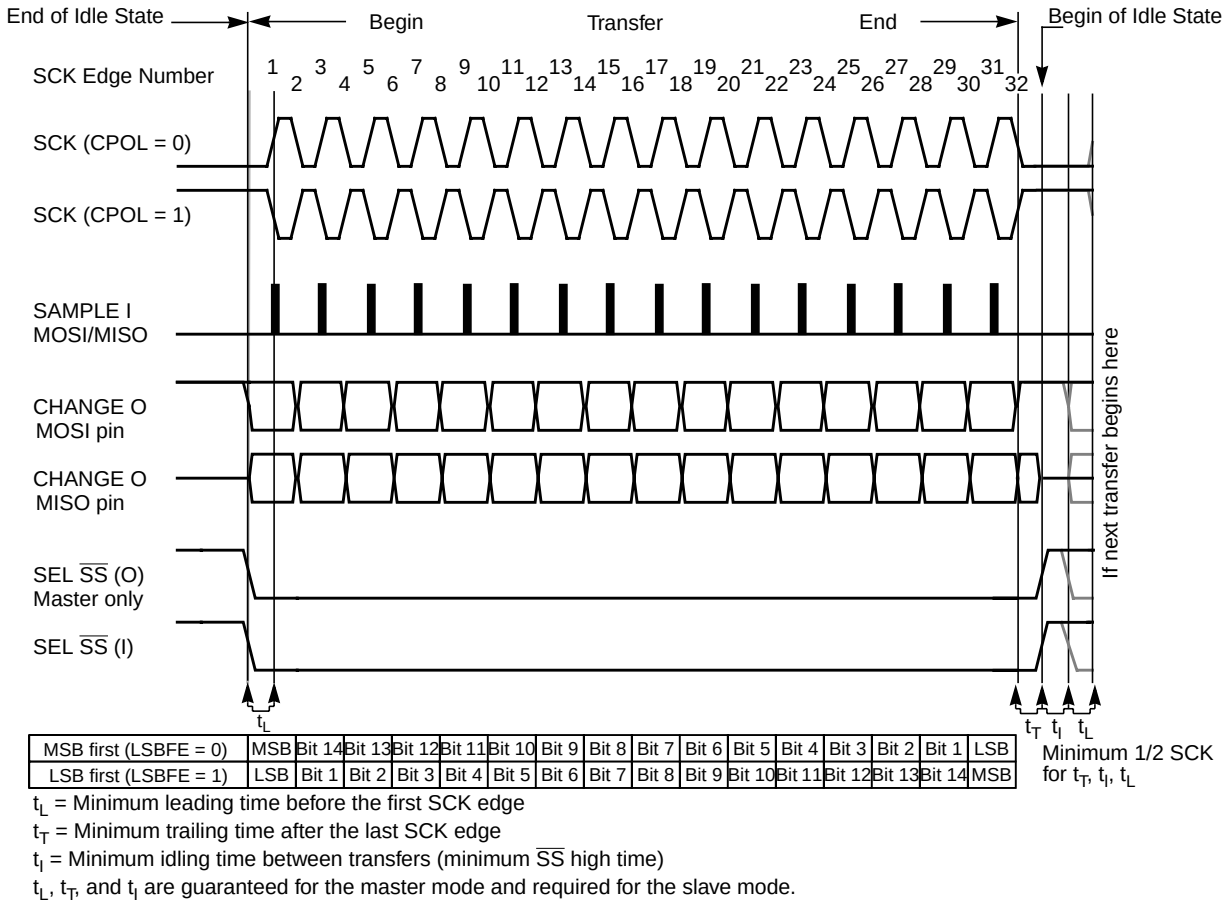


Figure 11-42. 8-bit Maskable Identifier Acceptance Filters



t_L = Minimum leading time before the first SCK edge
t_T = Minimum trailing time after the last SCK edge
t_I = Minimum idling time between transfers (minimum $\overline{SS}$ high time)
t_L, t_T, and t_I are guaranteed for the master mode and required for the slave mode.

Figure 15-13. SPI Clock Format 0 (CPHA = 0), with 16-Bit Transfer Width selected (XFRW = 1)

In slave mode, if the $\overline{SS}$ line is not deasserted between the successive transmissions then the content of the SPI data register is not transmitted; instead the last received data is transmitted. If the $\overline{SS}$ line is deasserted for at least minimum idle time (half SCK cycle) between successive transmissions, then the content of the SPI data register is transmitted.

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

15.4.3.3 CPHA = 1 Transfer Format

Some peripherals require the first SCK edge before the first data bit becomes available at the data out pin, the second edge clocks data into the system. In this format, the first SCK edge is issued by setting the CPHA bit at the beginning of the n[1]-cycle transfer operation.

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

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

1. n depends on the selected transfer width, please refer to Section 15.3.2.2, “SPI Control Register 2 (SPICR2)

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.

15.4.7 Low Power Mode Options

15.4.7.1 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.

15.4.7.2 SPI in Wait Mode

SPI operation in wait mode depends upon the state of the SPISWAI bit in SPI control register 2.

- If SPISWAI is clear, the SPI operates normally when the CPU is in wait mode
- If SPISWAI is set, SPI clock generation ceases and the SPI module enters a power conservation state when the CPU is in wait mode.
  - If SPISWAI is set and the SPI is configured for master, any transmission and reception in progress stops at wait mode entry. The transmission and reception resumes when the SPI exits wait mode.
  - If SPISWAI is set and the SPI is configured as a slave, any transmission and reception in progress continues if the SCK continues to be driven from the master. This keeps the slave synchronized to the master and the SCK.

    If the master transmits several bytes while the slave is in wait mode, the slave will continue to send out bytes consistent with the operation mode at the start of wait mode (i.e., if the slave is currently sending its SPIDR to the master, it will continue to send the same byte. Else if the slave is currently sending the last received byte from the master, it will continue to send each previous master byte).

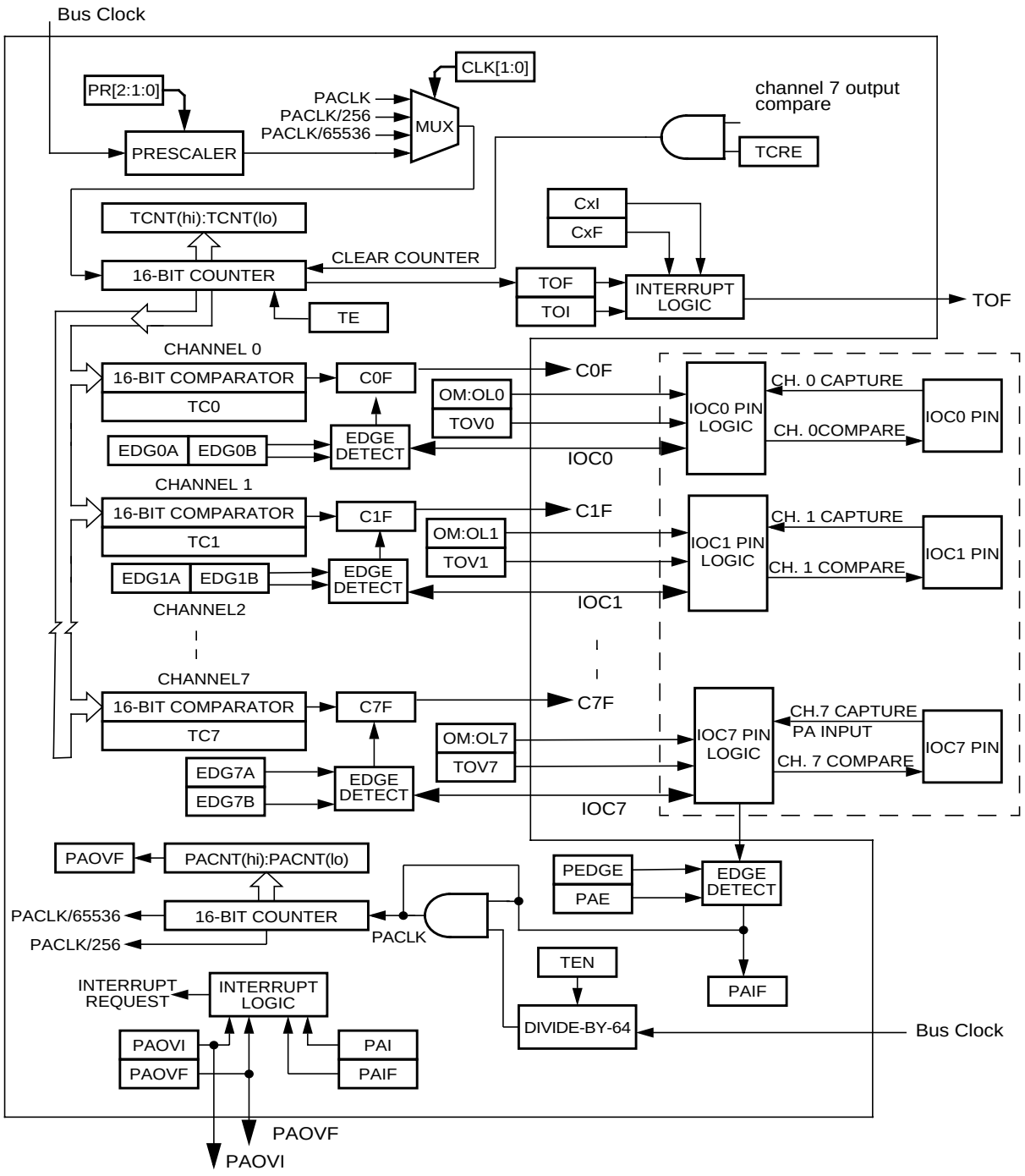


Figure 16-30. Detailed Timer Block Diagram

16.4.1 Prescaler

The prescaler divides the bus clock by 1,2,4,8,16,32,64 or 128. The prescaler select bits, PR[2:0], select the prescaler divisor. PR[2:0] are in timer system control register 2 (TSCR2).

Chapter 17
Voltage Regulator (S12VREGL3V3V1)

Table 17-1. Revision History Table

Rev. No. (Item No.)	Date (Submitted By)	Sections Affected	Substantial Change(s)
V01.02	09 Sep 2005		Updates for API external access and LVR flags.
V01.03	23 Sep 2005		VAE reset value is 1.
V01.04	08 Jun 2007		Added temperature sensor to customer information

17.1 Introduction

Module VREG_3V3 is a tri output voltage regulator that provides two separate 1.84V (typical) supplies differing in the amount of current that can be sourced and a 2.82V (typical) supply. The regulator input voltage range is from 3.3V up to 5V (typical).

17.1.1 Features

Module VREG_3V3 includes these distinctive features:

- Three parallel, linear voltage regulators with bandgap reference
- Low-voltage detect (LVD) with low-voltage interrupt (LVI)
- Power-on reset (POR)
- Low-voltage reset (LVR)
- High Temperature Detect (HTD) with High Temperature Interrupt (HTI)
- Autonomous periodical interrupt (API)

17.1.2 Modes of Operation

There are three modes VREG_3V3 can operate in:

1. Full performance mode (FPM) (MCU is not in stop mode)

   The regulator is active, providing the nominal supply voltages with full current sourcing capability. Features LVD (low-voltage detect), LVR (low-voltage reset), and POR (power-on reset) and HTD (High Temperature Detect) are available. The API is available.

2. Reduced power mode (RPM) (MCU is in stop mode)

   The purpose is to reduce power consumption of the device. The output voltage may degrade to a lower value than in full performance mode, additionally the current sourcing capability is substantially reduced. Only the POR is available in this mode, LVD, LVR and HTD are disabled. The API is available.

In Shutdown Mode an external supply driving VDD/VSS can replace the voltage regulator.

17.2.4 VDDF — Regulator Output2 (NVM Logic) Pins

Signals VDDF/VSS are the secondary outputs of VREG_3V3 that provide the power supply for the NVM logic. These signals are connected to device pins to allow external decoupling capacitors (220 nF, X7R ceramic).

In Shutdown Mode an external supply driving VDDF/VSS can replace the voltage regulator.

17.2.5 VDDPLL, VSSPLL — Regulator Output3 (PLL) Pins

Signals VDDPLL/VSSPLL are the secondary outputs of VREG_3V3 that provide the power supply for the PLL and oscillator. These signals are connected to device pins to allow external decoupling capacitors (100 nF...220 nF, X7R ceramic).

In Shutdown Mode, an external supply driving VDDPLL/VSSPLL can replace the voltage regulator.

17.2.6 VDDX — Power Input Pin

Signals VDDX/VSS are monitored by VREG_3V3 with the LVR feature.

17.2.7 VREGEN — Optional Regulator Enable Pin

This optional signal is used to shutdown VREG_3V3. In that case, VDD/VSS and VDDPLL/VSSPLL must be provided externally. Shutdown mode is entered with VREGEN being low. If VREGEN is high, the VREG_3V3 is either in Full Performance Mode or in Reduced Power Mode.

For the connectivity of VREGEN, see device specification.

NOTE

Switching from FPM or RPM to shutdown of VREG_3V3 and vice versa is not supported while MCU is powered.

17.2.8 VREG_API — Optional Autonomous Periodical Interrupt Output Pin

This pin provides the signal selected via APIEA if system is set accordingly. See 17.3.2.3, “Autonomous Periodical Interrupt Control Register (VREGAPICL) and 17.4.8, “Autonomous Periodical Interrupt (API) for details.

For the connectivity of VREG_API, see device specification.

17.3 Memory Map and Register Definition

This section provides a detailed description of all registers accessible in VREG_3V3.

If enabled in the system, the VREG_3V3 will abort all read and write accesses to reserved registers within it’s memory slice. See device level specification for details.

P-Flash phrase containing the D-Flash protection byte during the reset sequence, the DPOPEN bit will be cleared and DPS bits will be set to leave the D-Flash memory fully protected.

Trying to alter data in any protected area in the D-Flash memory will result in a protection violation error and the FPVIOL bit will be set in the FSTAT register. Block erase of the D-Flash memory is not possible if any of the D-Flash sectors are protected.

Table 18-22. DFPROT Field Descriptions

Field	Description
7 DPOPEN	**D-Flash Protection Control** 0 Enables D-Flash memory protection from program and erase with protected address range defined by DPS bits 1 Disables D-Flash memory protection from program and erase
4–0 DPS[4:0]	**D-Flash Protection Size** — The DPS[4:0] bits determine the size of the protected area in the D-Flash memory as shown in Table 18-23.

Table 18-23. D-Flash Protection Address Range

DPS[4:0]	Global Address Range	Protected Size
0_0000	0x10_0000 – 0x10_00FF	256 bytes
0_0001	0x10_0000 – 0x10_01FF	512 bytes
0_0010	0x10_0000 – 0x10_02FF	768 bytes
0_0011	0x10_0000 – 0x10_03FF	1024 bytes
0_0100	0x10_0000 – 0x10_04FF	1280 bytes
0_0101	0x10_0000 – 0x10_05FF	1536 bytes
0_0110	0x10_0000 – 0x10_06FF	1792 bytes
0_0111	0x10_0000 – 0x10_07FF	2048 bytes
0_1000	0x10_0000 – 0x10_08FF	2304 bytes
0_1001	0x10_0000 – 0x10_09FF	2560 bytes
0_1010	0x10_0000 – 0x10_0AFF	2816 bytes
0_1011	0x10_0000 – 0x10_0BFF	3072 bytes
0_1100	0x10_0000 – 0x10_0CFF	3328 bytes
0_1101	0x10_0000 – 0x10_0DFF	3584 bytes
0_1110	0x10_0000 – 0x10_0EFF	3840 bytes
0_1111	0x10_0000 – 0x10_0FFF	4096 bytes
1_0000	0x10_0000 – 0x10_10FF	4352 bytes
1_0001	0x10_0000 – 0x10_11FF	4608 bytes
1_0010	0x10_0000 – 0x10_12FF	4864 bytes
1_0011	0x10_0000 – 0x10_13FF	5120 bytes
1_0100	0x10_0000 – 0x10_14FF	5376 bytes
1_0101	0x10_0000 – 0x10_15FF	5632 bytes

20.4.1.1 Writing the FCLKDIV Register

Prior to issuing any Flash program or erase command after a reset, the user is required to write the FCLKDIV register to divide OSCCLK down to a target FCLK of 1 MHz. Table 20-7 shows recommended values for the FDIV field based on OSCCLK frequency.

NOTE

Programming or erasing the Flash memory cannot be performed if the bus clock runs at less than 1 MHz. Setting FDIV too high can destroy the Flash memory due to overstress. Setting FDIV too low can result in incomplete programming or erasure of the Flash memory cells.

When the FCLKDIV register is written, the FDIVLD bit is set automatically. If the FDIVLD bit is 0, the FCLKDIV register has not been written since the last reset. If the FCLKDIV register has not been written, any Flash program or erase command loaded during a command write sequence will not execute and the ACCERR bit in the FSTAT register will set.

20.4.1.2 Command Write Sequence

The Memory Controller will launch all valid Flash commands entered using a command write sequence.

Before launching a command, the ACCERR and FPVIOL bits in the FSTAT register must be clear (see Section 20.3.2.7) and the CCIF flag should be tested to determine the status of the current command write sequence. If CCIF is 0, the previous command write sequence is still active, a new command write sequence cannot be started, and all writes to the FCCOB register are ignored.

CAUTION

Writes to any Flash register must be avoided while a Flash command is active (CCIF=0) to prevent corruption of Flash register contents and Memory Controller behavior.

20.4.1.2.1 Define FCCOB Contents

The FCCOB parameter fields must be loaded with all required parameters for the Flash command being executed. Access to the FCCOB parameter fields is controlled via the CCOBIX bits in the FCCOBIX register (see Section 20.3.2.3).

The contents of the FCCOB parameter fields are transferred to the Memory Controller when the user clears the CCIF command completion flag in the FSTAT register (writing 1 clears the CCIF to 0). The CCIF flag will remain clear until the Flash command has completed. Upon completion, the Memory Controller will return CCIF to 1 and the FCCOB register will be used to communicate any results. The flow for a generic command write sequence is shown in Figure 20-26.

Table A-4. Operating Conditions

Voltage difference V_{DDR} to V_{DDX}	Δ_{VDDR}	-0.1	0	0.1	V
Voltage difference V_{SSX} to V_{SSA}	Δ_{VSSX}	refer to Table A-14			
Voltage difference V_{SS1} , V_{SS2} , V_{SS3} , V_{SSPLL} to V_{SSX}	Δ_{VSS}	-0.1	0	0.1	V
Digital logic supply voltage[1]	V_{DD}	1.72	1.8	1.98	V
PLL supply voltage	V_{DDPLL}	1.72	1.8	1.98	V
Oscillator[2] (Loop Controlled Pierce) (Full Swing Pierce)	f_{osc}	4 2	— —	16 40	MHz
Bus frequency[3]	f_{bus}	0.5	—	40	MHz
Temperature Option **C** Operating junction temperature range Operating ambient temperature range[4]	 T_J T_A	 –40 –40	 — 27	 110 85	°C
Temperature Option **V** Operating junction temperature range Operating ambient temperature range[4]	 T_J T_A	 –40 –40	 — 27	 130 105	°C
Temperature Option **M** Operating junction temperature range Operating ambient temperature range[4]	 T_J T_A	 –40 –40	 — 27	 150 125	°C

1 The device contains an internal voltage regulator to generate the logic and PLL supply out of the I/O supply.
2 This refers to the oscillator base frequency. Typical crystal & resonator tolerances are supported.
3 Please refer to Table A-24 for maximum bus frequency limits with frequency modulation enabled
4 Please refer to Section A.1.8, “Power Dissipation and Thermal Characteristics” for more details about the relation between ambient temperature T_A and device junction temperature T_J.

NOTE

Using the internal voltage regulator, operation is guaranteed in a power down until a low voltage reset assertion.

A.1.8 Power Dissipation and Thermal Characteristics

Power dissipation and thermal characteristics are closely related. The user must assure that the maximum operating junction temperature is not exceeded. The average chip-junction temperature (T_J) in °C can be obtained from:

$$T_J = T_A + (P_D \bullet \Theta_{JA})$$

T_J = Junction Temperature, [°C]

T_A = Ambient Temperature, [°C]

P_D = Total Chip Power Dissipation, [W]

Θ_{JA} = Package Thermal Resistance, [°C/W]