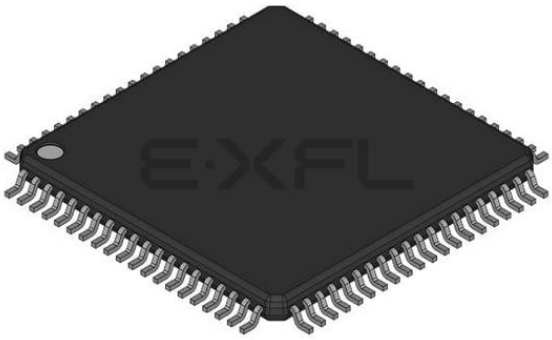


Welcome to E-XFL.COM

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	Obsolete
Core Processor	HCS12X
Core Size	16-Bit
Speed	80MHz
Connectivity	EBI/EMI, I²C, IrDA, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	59
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-QFP
Supplier Device Package	80-QFP (14x14)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s12xa256caar

Table 1-5. XGATE Resources (see Figure 1-6)

Device	XGRAMSIZE	XGRAM_LOW
9S12XDG128	12K	0x0F_D000
3S12XDG128	12K	0x0F_D000
9S12XD128	8K	0x0F_E000
9S12XD64	4K	0x0F_F000
9S12XB128	6K	0x0F_E800
9S12XA128	12K	0x0F_D000

Table 1-9. Chip Modes and Data Sources

Chip Modes	BKGD = MODC	PE6 = MODB	PE5 = MODA	PK7 = ROMCTL	PE3 = EROMCTL	Data Source[1]
Normal single chip	1	0	0	X	X	Internal
Special single chip	0	0	0			
Emulation single chip	0	0	1	X	0	Emulation memory
				X	1	Internal Flash
Normal expanded	1	0	1	0	X	External application
				1	X	Internal Flash
Emulation expanded	0	1	1	0	X	External application
				1	0	Emulation memory
				1	1	Internal Flash
Special test	0	1	0	0	X	External application
				1	X	Internal Flash

1 Internal means resources inside the MCU are read/written.
Internal Flash means Flash resources inside the MCU are read/written.
Emulation memory means resources inside the emulator are read/written (PRU registers, Flash replacement, RAM, EEPROM, and register space are always considered internal).
External application means resources residing outside the MCU are read/written.

The configuration of the oscillator can be selected using the $\overline{\text{XCLKS}}$ signal (see Table 1-10). For a detailed description please refer to the S12CRG section.

Table 1-10. Clock Selection Based on PE7

PE7 = $\overline{\text{XCLKS}}$	Description
0	Full swing Pierce oscillator or external clock source selected
1	Loop controlled Pierce oscillator selected

The logic level on the voltage regulator enable pin V_{REGEN} determines whether the on-chip voltage regulator is enabled or disabled (see Table 1-11).

Table 1-11. Voltage Regulator VREGEN

V_{REGEN}	Description
1	Internal voltage regulator enabled
0	Internal voltage regulator disabled, $V_{DD1,2}$ and V_{DDPLL} must be supplied externally

3.1.3 Block Diagram

Figure 3-1 shows a block diagram of the XOSC.

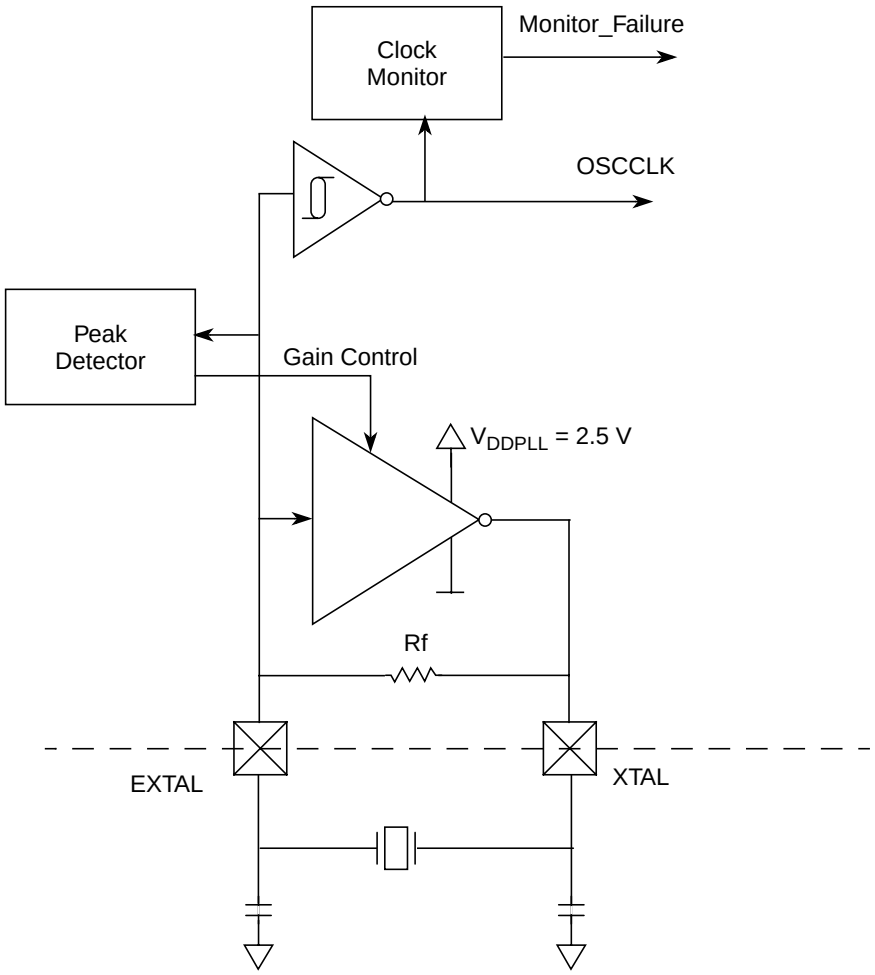


Figure 3-1. XOSC Block Diagram

3.2 External Signal Description

This section lists and describes the signals that connect off chip

3.2.1 V_{DDPLL} and V_{SSPLL} — Operating and Ground Voltage Pins

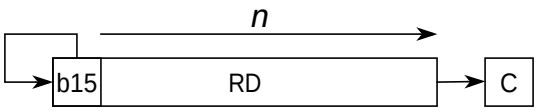
Theses pins provides operating voltage (V_{DDPLL}) and ground (V_{SSPLL}) for the XOSC circuitry. This allows the supply voltage to the XOSC to be independently bypassed.

3.2.2 EXTAL and XTAL — Input and Output Pins

These pins provide the interface for either a crystal or a CMOS compatible clock to control the internal clock generator circuitry. EXTAL is the external clock input or the input to the crystal oscillator amplifier. XTAL is the output of the crystal oscillator amplifier. The MCU internal system clock is derived from the

ASR — Arithmetic Shift Right — ASR

Operation



n = RS or IMM4

Shifts the bits in register RD *n* positions to the right. The higher *n* bits of the register RD become filled with the sign bit (RD[15]). The carry flag will be updated to the bit contained in RD[n-1] before the shift for *n* > 0.

n can range from 0 to 16.

In immediate address mode, *n* is determined by the operand IMM4. *n* is considered to be 16 in IMM4 is equal to 0.

In dyadic address mode, *n* is determined by the content of RS. *n* is considered to be 16 if the content of RS is greater than 15.

CCR Effects

N	Z	V	C
Δ	Δ	0	Δ

N: Set if bit 15 of the result is set; cleared otherwise.

Z: Set if the result is $0000; cleared otherwise.

V: Set if a two´s complement overflow resulted from the operation; cleared otherwise.
$RD[15]_{old}$ ^ $RD[15]_{new}$

C: Set if *n* > 0 and RD[n-1] = 1; if n = 0 unaffected.

Code and CPU Cycles

Source Form	Address Mode	Machine Code												Cycles
ASR RD, #IMM4	IMM4	0	0	0	0	1	RD	IMM4		1	0	0	1	P
ASR RD, RS	DYA	0	0	0	0	1	RD	RS	1	0	0	0	1	P

7.3.2.24 Input Control System Control Register (ICSYS)

	7	6	5	4	3	2	1	0
R W	SH37	SH26	SH15	SH04	TFMOD	PACMX	BUFEN	LATQ
Reset	0	0	0	0	0	0	0	0

Figure 7-46. Input Control System Register (ICSYS)

Read: Anytime

Write: Once in normal modes

All bits reset to zero.

Table 7-30. ICSYS Field Descriptions

Field	Description
7:4 SHxy	**Share Input action of Input Capture Channels x and y** 0 Normal operation 1 The channel input ‘x’ causes the same action on the channel ‘y’. The port pin ‘x’ and the corresponding edge detector is used to be active on the channel ‘y’.
3 TFMOD	**Timer Flag Setting Mode** — Use of the TFMOD bit in conjunction with the use of the ICOVW register allows a timer interrupt to be generated after capturing two values in the capture and holding registers instead of generating an interrupt for every capture. By setting TFMOD in queue mode, when NOVWx bit is set and the corresponding capture and holding registers are emptied, an input capture event will first update the related input capture register with the main timer contents. At the next event, the TCx data is transferred to the TCxH register, the TCx is updated and the CxF interrupt flag is set. In all other input capture cases the interrupt flag is set by a valid external event on PTx. 0 The timer flags C3F–C0F in TFLG1 are set when a valid input capture transition on the corresponding port pin occurs. 1 If in queue mode (BUFEN = 1 and LATQ = 0), the timer flags C3F–C0F in TFLG1 are set only when a latch on the corresponding holding register occurs. If the queue mode is not engaged, the timer flags C3F–C0F are set the same way as for TFMOD = 0.
2 PACMX	**8-Bit Pulse Accumulators Maximum Count** 0 Normal operation. When the 8-bit pulse accumulator has reached the value 0x00FF, with the next active edge, it will be incremented to 0x0000. 1 When the 8-bit pulse accumulator has reached the value 0x00FF, it will not be incremented further. The value 0x00FF indicates a count of 255 or more.
1 BUFFEN	**IC Buffer Enable** 0 Input capture and pulse accumulator holding registers are disabled. 1 Input capture and pulse accumulator holding registers are enabled. The latching mode is defined by LATQ control bit.

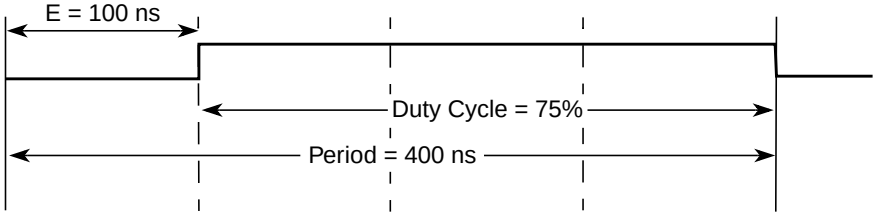


Figure 8-21. PWM Left Aligned Output Example Waveform

8.4.2.6 Center Aligned Outputs

For center aligned output mode selection, set the CAEx bit (CAEx = 1) in the PWMCAE register and the corresponding PWM output will be center aligned.

The 8-bit counter operates as an up/down counter in this mode and is set to up whenever the counter is equal to $00. The counter compares to two registers, a duty register and a period register as shown in the block diagram in Figure 8-19. When the PWM counter matches the duty register, the output flip-flop changes state, causing the PWM waveform to also change state. A match between the PWM counter and the period register changes the counter direction from an up-count to a down-count. When the PWM counter decrements and matches the duty register again, the output flip-flop changes state causing the PWM output to also change state. When the PWM counter decrements and reaches zero, the counter direction changes from a down-count back to an up-count and a load from the double buffer period and duty registers to the associated registers is performed, as described in Section 8.4.2.3, “PWM Period and Duty”. The counter counts from 0 up to the value in the period register and then back down to 0. Thus the effective period is PWMPERx*2.

NOTE

Changing the PWM output mode from left aligned to center aligned output (or vice versa) while channels are operating can cause irregularities in the PWM output. It is recommended to program the output mode before enabling the PWM channel.

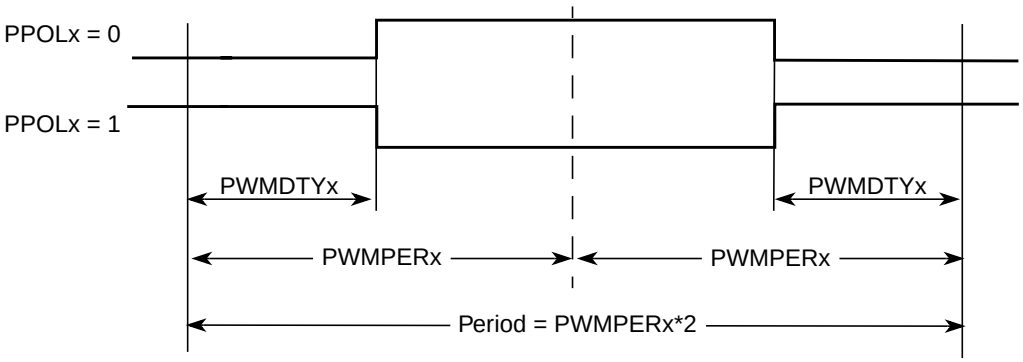


Figure 8-22. PWM Center Aligned Output Waveform

Table 10-33. Data Length Codes

Data Length Code				Data Byte Count
DLC3	DLC2	DLC1	DLC0	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8

10.3.3.4 Transmit Buffer Priority Register (TBPR)

This register defines the local priority of the associated message buffer. The local priority is used for the internal prioritization process of the MSCAN and is defined to be highest for the smallest binary number. The MSCAN implements the following internal prioritization mechanisms:

- All transmission buffers with a cleared TXEx flag participate in the prioritization immediately before the SOF (start of frame) is sent.
- The transmission buffer with the lowest local priority field wins the prioritization.

In cases of more than one buffer having the same lowest priority, the message buffer with the lower index number wins.

	7	6	5	4	3	2	1	0
R W	PRIO7	PRIO6	PRIO5	PRIO4	PRIO3	PRIO2	PRIO1	PRIO0
Reset:	0	0	0	0	0	0	0	0

Figure 10-36. Transmit Buffer Priority Register (TBPR)

Read: Anytime when TXEx flag is set (see Section 10.3.2.7, "MSCAN Transmitter Flag Register (CANTFLG)") and the corresponding transmit buffer is selected in CANTBSEL (see Section 10.3.2.11, "MSCAN Transmit Buffer Selection Register (CANTBSEL)").

Write: Anytime when TXEx flag is set (see Section 10.3.2.7, "MSCAN Transmitter Flag Register (CANTFLG)") and the corresponding transmit buffer is selected in CANTBSEL (see Section 10.3.2.11, "MSCAN Transmit Buffer Selection Register (CANTBSEL)").

10.3.3.5 Time Stamp Register (TSRH–TSRL)

If the TIME bit is enabled, the MSCAN will write a time stamp to the respective registers in the active transmit or receive buffer right after the EOF of a valid message on the CAN bus (see Section 10.3.2.1,

The SCI also sets a flag, the transmit data register empty flag (TDRE), every time it transfers data from the buffer (SCIDRH/L) to the transmitter shift register.The transmit driver routine may respond to this flag by writing another byte to the Transmitter buffer (SCIDRH/SCIDRL), while the shift register is still shifting out the first byte.

To initiate an SCI transmission:

1. Configure the SCI:
   a) Select a baud rate. Write this value to the SCI baud registers (SCIBDH/L) to begin the baud rate generator. Remember that the baud rate generator is disabled when the baud rate is zero. Writing to the SCIBDH has no effect without also writing to SCIBDL.
   b) Write to SCICR1 to configure word length, parity, and other configuration bits (LOOPS,RSRC,M,WAKE,ILT,PE,PT).
   c) Enable the transmitter, interrupts, receive, and wake up as required, by writing to the SCICR2 register bits (TIE,TCIE,RIE,ILIE,TE,RE,RWU,SBK). A preamble or idle character will now be shifted out of the transmitter shift register.
2. Transmit Procedure for each byte:
   a) Poll the TDRE flag by reading the SCISR1 or responding to the TDRE interrupt. Keep in mind that the TDRE bit resets to one.
   b) If the TDRE flag is set, write the data to be transmitted to SCIDRH/L, where the ninth bit is written to the T8 bit in SCIDRH if the SCI is in 9-bit data format. A new transmission will not result until the TDRE flag has been cleared.
3. Repeat step 2 for each subsequent transmission.

NOTE

The TDRE flag is set when the shift register is loaded with the next data to be transmitted from SCIDRH/L, which happens, generally speaking, a little over half-way through the stop bit of the previous frame. Specifically, this transfer occurs 9/16ths of a bit time AFTER the start of the stop bit of the previous frame.

Writing the TE bit from 0 to a 1 automatically loads the transmit shift register with a preamble of 10 logic 1s (if M = 0) or 11 logic 1s (if M = 1). After the preamble shifts out, control logic transfers the data from the SCI data register into the transmit shift register. A logic 0 start bit automatically goes into the least significant bit position of the transmit shift register. A logic 1 stop bit goes into the most significant bit position.

Hardware supports odd or even parity. When parity is enabled, the most significant bit (MSB) of the data character is the parity bit.

The transmit data register empty flag, TDRE, in SCI status register 1 (SCISR1) becomes set when the SCI data register transfers a byte to the transmit shift register. The TDRE flag indicates that the SCI data register can accept new data from the internal data bus. If the transmit interrupt enable bit, TIE, in SCI control register 2 (SCICR2) is also set, the TDRE flag generates a transmitter interrupt request.

Whenever a 16-bit timer counter and the connected 8-bit micro timer counter have counted to zero, the PITLD register is reloaded and the corresponding time-out flag PTF in the PIT time-out flag (PITTF) register is set, as shown in Figure 13-20. The time-out period is a function of the timer load (PITLD) and micro timer load (PITMTLD) registers and the bus clock f_{BUS}:

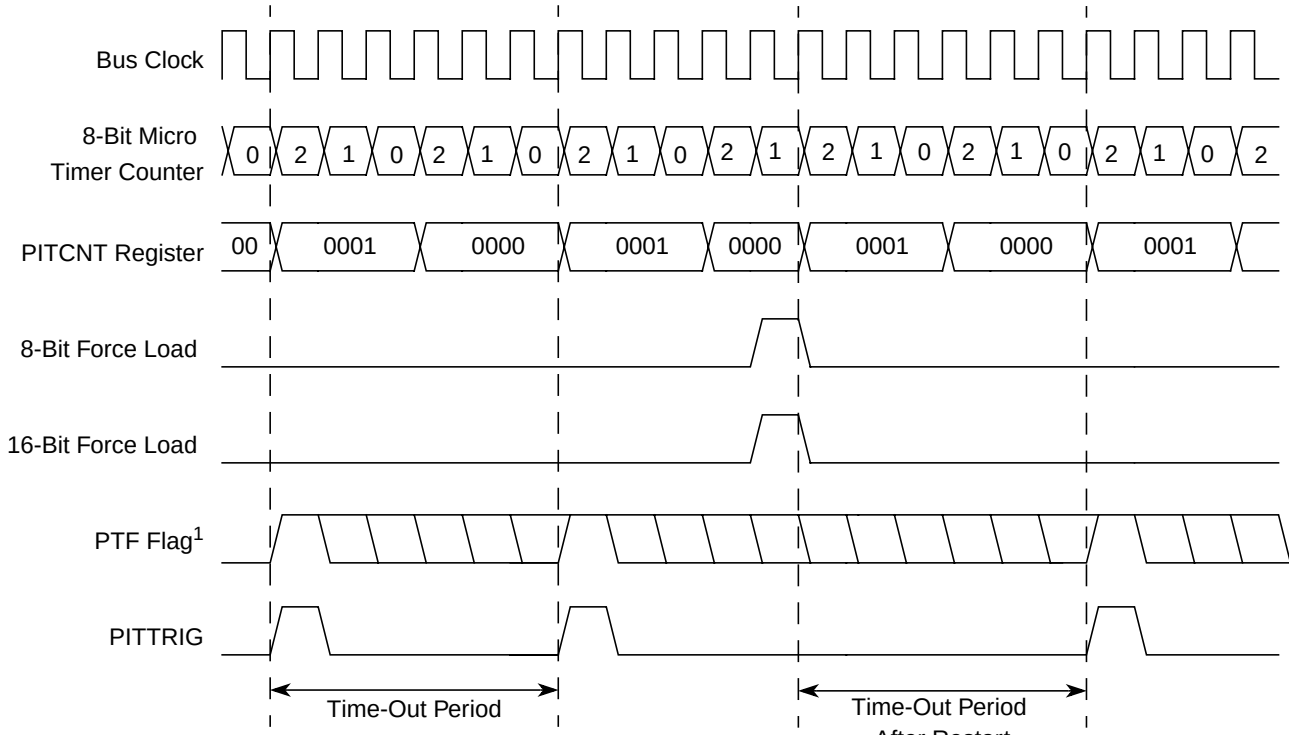
time-out period = (PITMTLD + 1) * (PITLD + 1) / f_{BUS}.

For example, for a 40 MHz bus clock, the maximum time-out period equals:

256 * 65536 * 25 ns = 419.43 ms.

The current 16-bit modulus down-counter value can be read via the PITCNT register. The micro timer down-counter values cannot be read.

The 8-bit micro timers can individually be restarted by writing a one to the corresponding force load micro timer PFLMT bits in the PIT control and force load micro timer (PITCFLMT) register. The 16-bit timers can individually be restarted by writing a one to the corresponding force load timer PFLT bits in the PIT forceload timer (PITFLT) register. If desired, any group of timers and micro timers can be restarted at the same time by using one 16-bit write to the adjacent PITCFLMT and PITFLT registers with the relevant bits set, as shown in Figure 13-20.



Note 1. The PTF flag clearing depends on the software

Figure 13-20. PIT Trigger and Flag Signal Timing

13.4.2 Interrupt Interface

Each time-out event can be used to trigger an interrupt service request. For each timer channel, an individual bit PINTE in the PIT interrupt enable (PITINTE) register exists to enable this feature. If PINTE

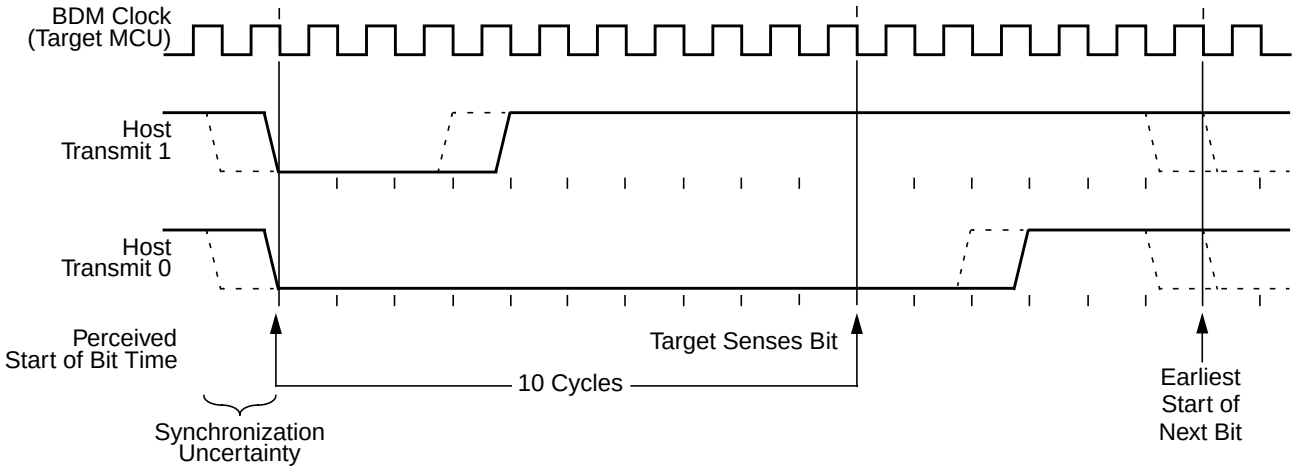


Figure 15-8. BDM Host-to-Target Serial Bit Timing

The receive cases are more complicated. Figure 15-9 shows the host receiving a logic 1 from the target system. Since the host is asynchronous to the target, there is up to one clock-cycle delay from the host-generated falling edge on BKGD to the perceived start of the bit time in the target. The host holds the BKGD pin low long enough for the target to recognize it (at least two target clock cycles). The host must release the low drive before the target drives a brief high speedup pulse seven target clock cycles after the perceived start of the bit time. The host should sample the bit level about 10 target clock cycles after it started the bit time.

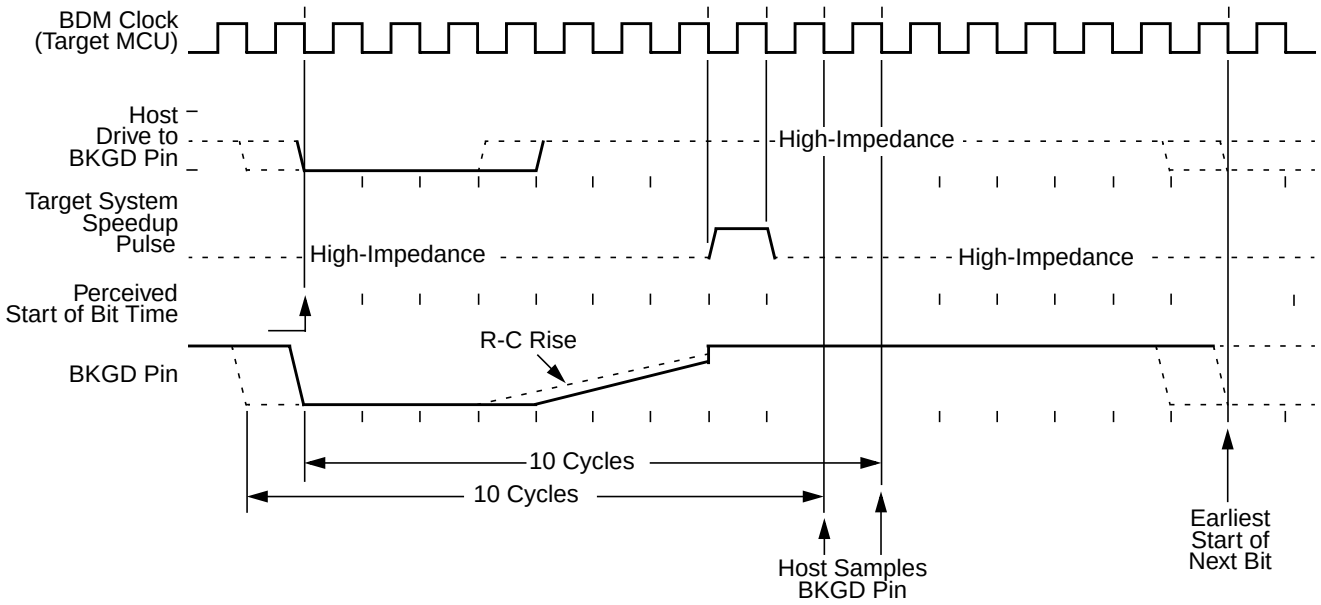


Figure 15-9. BDM Target-to-Host Serial Bit Timing (Logic 1)

XGATE activity can still be compared, traced and can be used to generate a breakpoint to the XGATE module. When the CPU enters active BDM mode through a BACKGROUND command, with the DBG module armed, the DBG remains armed.

The DBG module tracing is disabled if the MCU is secure. Breakpoints can however still be generated if the MCU is secure.

Table 19-1. Mode Dependent Restriction Summary

BDM Enable	BDM Active	MCU Secure	Comparator Matches Enabled	Breakpoints Possible	Tagging Possible	Tracing Possible
x	x	1	Yes	Yes	Yes	No
0	0	0	Yes	Only SWI	Yes	Yes
0	1	0	Active BDM not possible when not enabled			
1	0	0	Yes	Yes	Yes	Yes
1	1	0	XGATE only	XGATE only	XGATE only	XGATE only

19.1.4 Block Diagram

Figure 19-1 shows a block diagram of the debug module.

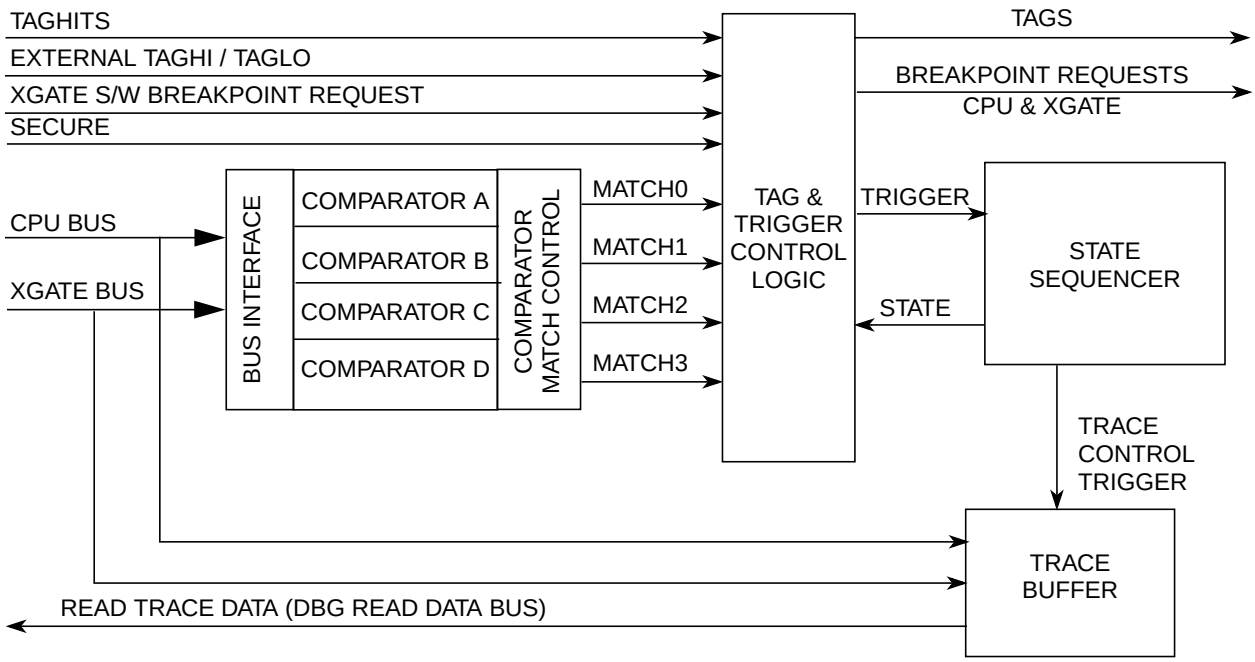


Figure 19-1. Debug Block Diagram

19.2 External Signal Description

The DBG sub-module features two external tag input signals (see Table 19-2). See Device User Guide (DUG) for the mapping of these signals to device pins. These tag pins may be used for the external tagging in emulation modes only

Table 19-21. State1 Sequencer Next Sate Selection

SC[3:0]	Description
0000	Any match triggers to state2
0001	Any match triggers to state3
0010	Any match triggers to final state
0011	Match2 triggers to State2..... Other matches have no effect
0100	Match2 triggers to State3..... Other matches have no effect
0101	Match2 triggers to final state..... Other matches have no effect
0110	Match0 triggers to State2..... Match1 triggers to State3..... Other matches have no effect
0111	Match1 triggers to State3..... Match0 triggers final state..... Other matches have no effect
1000	Match0 triggers to State2..... Match2 triggers to State3..... Other matches have no effect
1001	Match2 triggers to State3..... Match0 triggers final state..... Other matches have no effect
1010	Match1 triggers to State2..... Match3 triggers to State3..... Other matches have no effect
1011	Match3 triggers to State3..... Match1 triggers to final state..... Other matches have no effect
1100	Reserved
1101	Reserved
1110	Reserved
1111	Reserved

19.3.1.9 Debug State Control Register 2 (DBGSCR2)

0x0027

	7	6	5	4	3	2	1	0
R	0	0	0	0	SC3	SC2	SC1	SC0
W								
Reset	0	0	0	0	0	0	0	0

Unimplemented or Reserved

Figure 19-11. Debug State Control Register 2 (DBGSCR2)

Read: Anytime

Write: Anytime when DBG not armed.

This register is visible at 0x0027 only with COMRV[1:0] = 01. The state control register 2 selects the targeted next state while in State2. The matches refer to the match channels of the comparator match control logic as depicted in Figure 19-1 and described in Section 19.3.1.11.1, “Debug Comparator Control Register (DBGXCTL)”. Comparators must be enabled by setting the comparator enable bit in the associated DBGXCTL control register.

21.1.3 Block Diagram

Figure 21-1 is a block diagram of the XEBI with all related I/O signals.

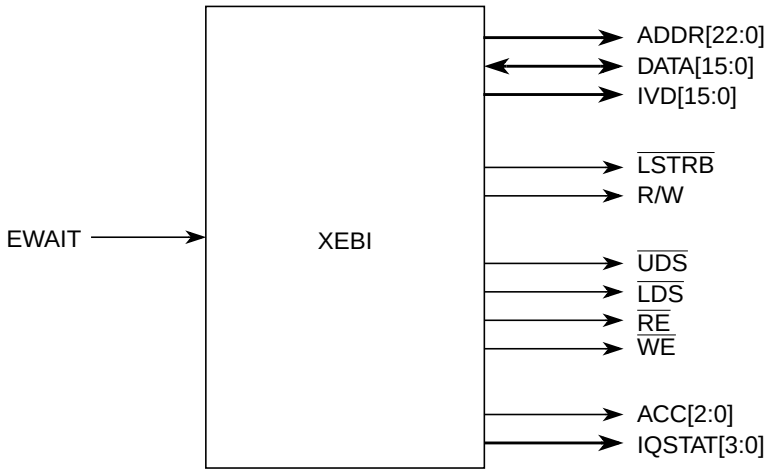


Figure 21-1. XEBI Block Diagram

21.2 External Signal Description

The user is advised to refer to the SoC section for port configuration and location of external bus signals.

NOTE

The following external bus related signals are described in other sections:
$\overline{CS2}$, $\overline{CS1}$, $\overline{CS0}$ (chip selects) — S12X_MMC section
ECLK, ECLKX2 (free-running clocks) — PIM section
$\overline{TAGHI}$, $\overline{TAGLO}$ (tag inputs) — PIM section, S12X_DBG section

Table 21-1 outlines the pin names and gives a brief description of their function. Refer to the SoC section and PIM section for reset states of these pins and associated pull-ups or pull-downs.

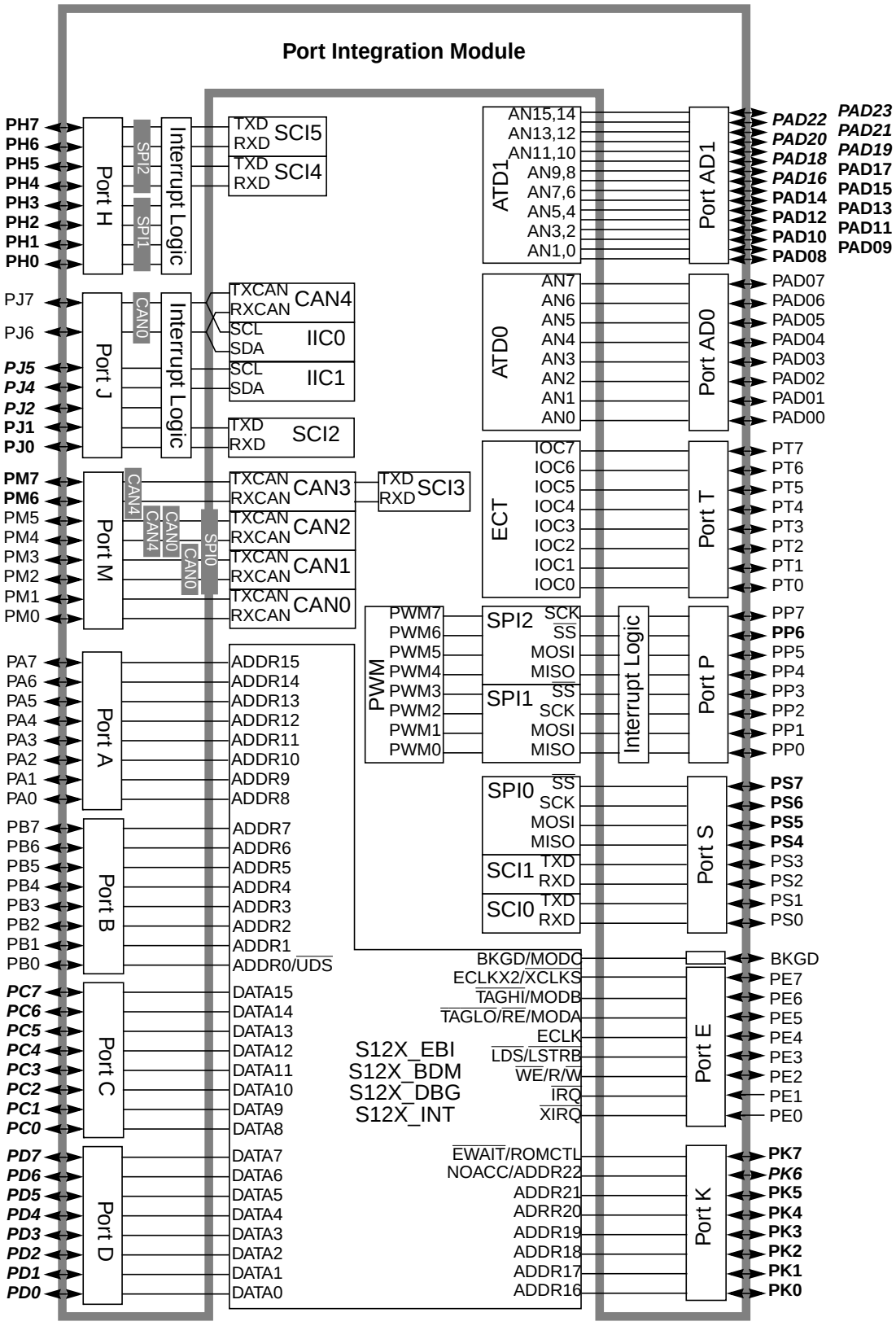


Figure 22-1. PIM Block Diagram

23.0.5.54 Port J Data Register (PTJ)

	7	6	5	4	3	2	1	0
R	PTJ7	PTJ6	PTJ5	PTJ4	0	PTJ2	PTJ1	PTJ0
W								
CAN4	TXCAN4	RXCAN4						
SCI2							TXD2	RXD2
IIC0	SCL0	SDA0						
Routed CAN0	TXCAN0	RXCAN0						
Alt. Function			$\overline{CS2}$	$\overline{CS0}$		$\overline{CS1}$		$\overline{CS3}$
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 23-56. Port J Data Register (PTJ)

Read: Anytime.

Write: Anytime.

Port J pins 7–4 and 2–0 are associated with the CAN4, SCI2, IIC0, the routed CAN0 modules and chip select signals ($\overline{CS0}$, $\overline{CS1}$, $\overline{CS2}$, $\overline{CS3}$). These pins can be used as general purpose I/O when not used with any of the peripherals.

If the data direction bits of the associated I/O pins are set to logic level “1”, a read returns the value of the port register, otherwise the buffered pin input state is read.

Table 23-51. PTJ Field Descriptions

Field	Description
7–6 PJ[7:6]	The CAN4 function (TXCAN4 and RXCAN4) takes precedence over the IIC0, the routed CAN0 and the general purpose I/O function if the CAN4 module is enabled. The IIC0 function (SCL0 and SDA0) takes precedence over the routed CAN0 and the general purpose I/O function if the IIC0 is enabled. If the IIC0 module takes precedence the SDA0 and SCL0 outputs are configured as open drain outputs. *Refer to IIC section for details.* The routed CAN0 function (TXCAN0 and RXCAN0) takes precedence over the general purpose I/O function if the routed CAN0 module is enabled. *Refer to MSCAN section for details.*
2 PJ2	The chip select function ($\overline{CS1}$) takes precedence over the general purpose I/O.
1 PJ1	The SCI2 function takes precedence over the general purpose I/O function if the SCI2 module is enabled. *Refer to SCI section for details.*
0 PJ0	The chip select ($\overline{CS3}$) takes precedence over the general purpose I/O function.

Port S pins 7–4 are associated with the SPI0. The SPI0 pin configuration is determined by several status bits in the SPI0 module. *Refer to SPI section for details.* When not used with the SPI0, these pins can be used as general purpose I/O.

Port S bits 3–0 are associated with the SCI1 and SCI0. The SCI ports associated with transmit pins 3 and 1 are configured as outputs if the transmitter is enabled. The SCI ports associated with receive pins 2 and 0 are configured as inputs if the receiver is enabled. *Refer to SCI section for details.* When not used with the SCI, these pins can be used as general purpose I/O.

If the data direction bits of the associated I/O pins are set to logic level “1”, a read returns the value of the port register, otherwise the buffered pin input state is read.

24.0.5.20 Port S Input Register (PTIS)

	7	6	5	4	3	2	1	0
R	PTIS7	PTIS6	PTIS5	PTIS4	PTIS3	PTIS2	PTIS1	PTIS0
W								
Reset[1]	—	—	—	—	—	—	—	—

= Unimplemented or Reserved

Figure 24-22. Port S Input Register (PTIS)

1. These registers are reset to zero. Two bus clock cycles after reset release the register values are updated with the associated pin values.

Read: Anytime.

Write: Never, writes to this register have no effect.

This register always reads back the buffered state of the associated pins. This also can be used to detect overload or short circuit conditions on output pins.

24.0.5.21 Port S Data Direction Register (DDRS)

	7	6	5	4	3	2	1	0
R W	DDRS7	DDRS6	DDRS5	DDRS4	DDRS3	DDRS2	DDRS1	DDRS0
Reset	0	0	0	0	0	0	0	0

Figure 24-23. Port S Data Direction Register (DDRS)

Read: Anytime.

Write: Anytime.

This register configures each port S pin as either input or output.

If SPI0 is enabled, the SPI0 determines the pin direction. *Refer to SPI section for details*.

This register always reads back the buffered state of the associated pins. This can also be used to detect overload or short circuit conditions on output pins.

24.0.5.44 Port H Data Direction Register (DDRH)

	7	6	5	4	3	2	1	0
R W	DDRH7	DDRH6	DDRH5	DDRH4	DDRH3	DDRH2	DDRH1	DDRH0
Reset	0	0	0	0	0	0	0	0

Figure 24-46. Port H Data Direction Register (DDRH)

Read: Anytime.

Write: Anytime.

This register configures each port H pin as either input or output.

If the associated routed SPI module is enabled this register has no effect on the pins.

If a SPI module is enabled, the SPI determines the pin direction. *Refer to SPI section for details.*

The DDRH bits revert to controlling the I/O direction of a pin when the associated peripheral modules are disabled.

Table 24-41. DDRH Field Descriptions

Field	Description
7–0 DDRH[7:0]	**Data Direction Port H** 0 Associated pin is configured as input. 1 Associated pin is configured as output. **Note:** Due to internal synchronization circuits, it can take up to 2 bus clock cycles until the correct value is read on PTH or PTIH registers, when changing the DDRH register.

24.0.5.45 Port H Reduced Drive Register (RDRH)

	7	6	5	4	3	2	1	0
R W	RDRH7	RDRH6	RDRH5	RDRH4	RDRH3	RDRH2	RDRH1	RDRH0
Reset	0	0	0	0	0	0	0	0

Figure 24-47. Port H Reduced Drive Register (RDRH)

Read: Anytime.

Write: Anytime.

This register configures the drive strength of each Port H output pin as either full or reduced. If the port is used as input this bit is ignored.

24.0.5.61 Port AD1 Data Direction Register 1 (DDR1AD1)

	7	6	5	4	3	2	1	0
R W	DDR1AD17	DDR1AD16	DDR1AD15	DDR1AD14	DDR1AD13	DDR1AD12	DDR1AD11	DDR1AD10
Reset	0	0	0	0	0	0	0	0

Figure 24-63. Port AD1 Data Direction Register 1 (DDR1AD1)

Read: Anytime.

Write: Anytime.

This register configures pins PAD[7:0] as either input or output.

Table 24-55. DDR1AD1 Field Descriptions

Field	Description
7–0 DDR1AD1[7:0]	**Data Direction Port AD1 Register 1** 0 Associated pin is configured as input. 1 Associated pin is configured as output. **Note:** Due to internal synchronization circuits, it can take up to 2 bus clock cycles until the correct value is read on PTAD11 register, when changing the DDR1AD1 register. **Note:** To use the digital input function on port AD1 the ATD1 digital input enable register (ATD1DIEN1) has to be set to logic level “1”.

24.0.5.62 Port AD1 Reduced Drive Register 0 (RDR0AD1)

	7	6	5	4	3	2	1	0
R W	RDR0AD115	RDR0AD114	RDR0AD113	RDR0AD112	RDR0AD111	RDR0AD110	RDR0AD19	RDR0AD18
Reset	0	0	0	0	0	0	0	0

Figure 24-64. Port AD1 Reduced Drive Register 0 (RDR0AD1)

Read: Anytime.

Write: Anytime.

This register configures the drive strength of each PAD[15:8] output pin as either full or reduced. If the port is used as input this bit is ignored.

Table 24-56. RDR0AD1 Field Descriptions

Field	Description
7–0 RDR0AD1[15:8]	**Reduced Drive Port AD1 Register 0** 0 Full drive strength at output. 1 Associated pin drives at about 1/6 of the full drive strength.

Table 29-8. FCNFG Field Descriptions

Field	Description
7 CBEIE	**Command Buffer Empty Interrupt Enable** — The CBEIE bit enables an interrupt in case of an empty command buffer in the Flash module. 0 Command buffer empty interrupt disabled. 1 An interrupt will be requested whenever the CBEIF flag (see Section 29.3.2.6, “Flash Status Register (FSTAT)”) is set.
6 CCIE	**Command Complete Interrupt Enable** — The CCIE bit enables an interrupt in case all commands have been completed in the Flash module. 0 Command complete interrupt disabled. 1 An interrupt will be requested whenever the CCIF flag (see Section 29.3.2.6, “Flash Status Register (FSTAT)”) is set.
5 KEYACC	**Enable Security Key Writing** 0 Flash writes are interpreted as the start of a command write sequence. 1 Writes to Flash array are interpreted as keys to open the backdoor. Reads of the Flash array return invalid data.

29.3.2.5 Flash Protection Register (FPROT)

The FPROT register defines which Flash sectors are protected against program or erase operations.

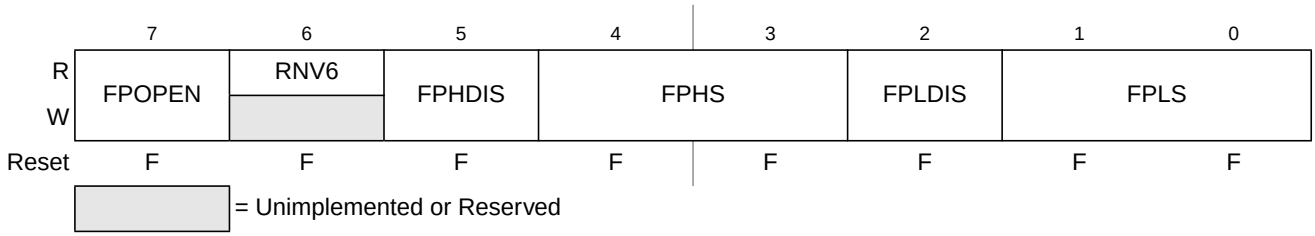


Figure 29-8. Flash Protection Register (FPROT)

All bits in the FPROT register are readable and writable with restrictions (see Section 29.3.2.5.1, “Flash Protection Restrictions”) except for RNV[6] which is only readable.

During the reset sequence, the FPROT register is loaded from the Flash Configuration Field at global address 0x7F_FF0D. To change the Flash protection that will be loaded during the reset sequence, the upper sector of the Flash memory must be unprotected, then the Flash Protect/Security byte located as described in Table 29-1 must be reprogrammed.

Trying to alter data in any protected area in the Flash memory will result in a protection violation error and the PVIOL flag will be set in the FSTAT register. The mass erase of a Flash block is not possible if any of the Flash sectors contained in the Flash block are protected.

29.4.3 Illegal Flash Operations

The ACCERR flag will be set during the command write sequence if any of the following illegal steps are performed, causing the command write sequence to immediately abort:

1. Writing to a Flash address before initializing the FCLKDIV register.
2. Writing a byte or misaligned word to a valid Flash address.
3. Starting a command write sequence while a data compress operation is active.
4. Starting a command write sequence while a sector erase abort operation is active.
5. Writing to any Flash register other than FCMD after writing to a Flash address.
6. Writing a second command to the FCMD register in the same command write sequence.
7. Writing an invalid command to the FCMD register.
8. When security is enabled, writing a command other than mass erase to the FCMD register when the write originates from a non-secure memory location or from the Background Debug Mode.
9. Writing to a Flash address after writing to the FCMD register.
10. Writing to any Flash register other than FSTAT (to clear CBEIF) after writing to the FCMD register.
11. Writing a 0 to the CBEIF flag in the FSTAT register to abort a command write sequence.

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

The ACCERR flag will also be set if any of the following events occur:

1. Launching the sector erase abort command while a sector erase operation is active which results in the early termination of the sector erase operation (see Section 29.4.2.6, “Sector Erase Abort Command”).
2. The MCU enters stop mode and a program or erase operation is in progress. The operation is aborted immediately and any pending command is purged (see Section 29.5.2, “Stop Mode”).

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 29.3.2.6, “Flash Status Register (FSTAT)”).

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 an address written in the command write sequence was in a protected area of the Flash memory
2. Writing the sector erase command if an address written in the command write sequence was in a protected area of the Flash memory
3. Writing the mass erase command to a Flash block while any Flash protection is enabled in the block

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 29.3.2.6, “Flash Status Register (FSTAT)”).