NXP USA Inc. - MC9S12XA512VAG Datasheet

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	119
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 24x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	144-LQFP
Supplier Device Package	144-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/mc9s12xa512vag

Chapter 13
Periodic Interrupt Timer (S12PIT24B4CV1)

Chapter 14
Voltage Regulator (S12VREG3V3V5)

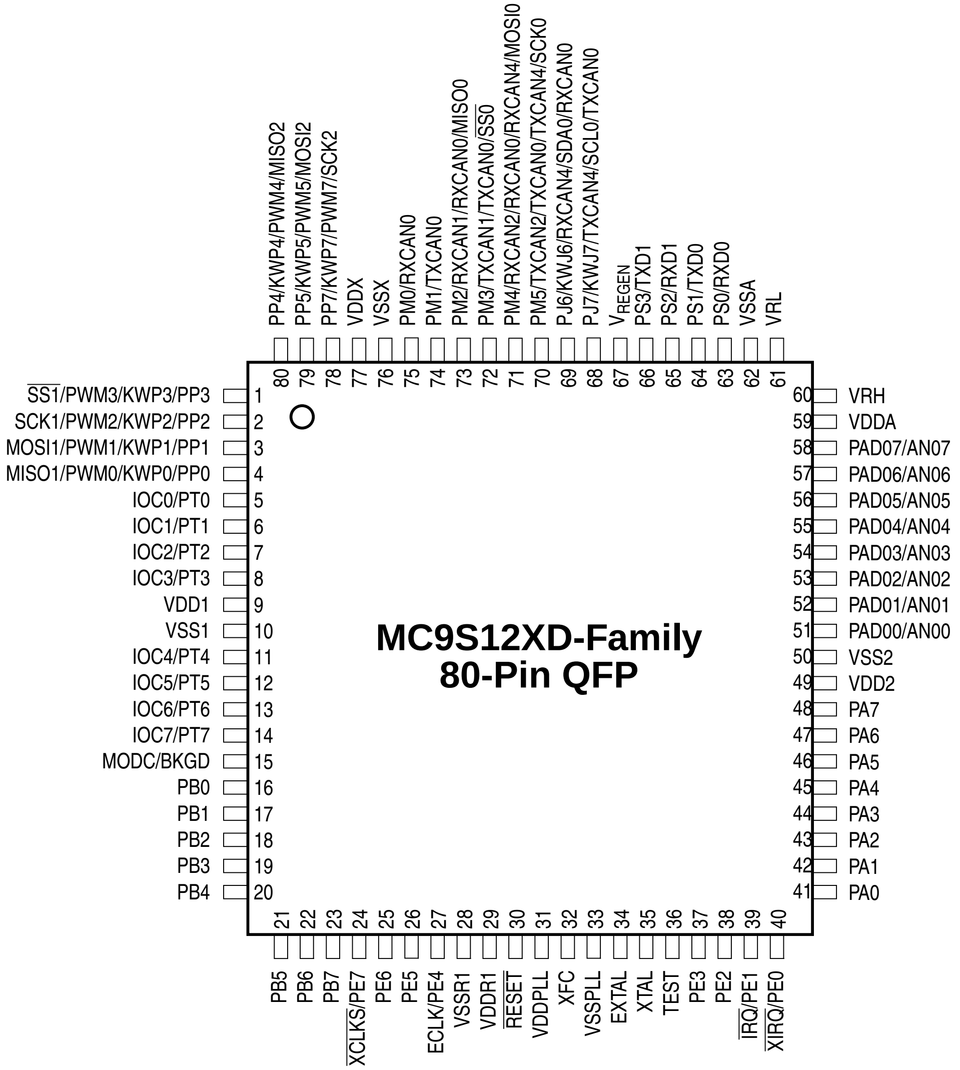


Figure 1-9. MC9S12XD Family Pin Assignments 80-Pin QFP Package

1.2.3.47 PM4 / RXCAN0 / RXCAN2 / RXCAN4 / MOSI0 — Port M I/O Pin 4

PM4 is a general-purpose input or output pin. It can be configured as the receive pin RXCAN of the scalable controller area network controllers 0, 2, or 4 (CAN0, CAN2, or CAN4). It can be configured as the master output (during master mode) or slave input pin (during slave mode) MOSI for the serial peripheral interface 0 (SPI0).

1.2.3.48 PM3 / TXCAN1 / TXCAN0 / $\overline{\text{SS0}}$ — Port M I/O Pin 3

PM3 is a general-purpose input or output pin. It can be configured as the transmit pin TXCAN of the scalable controller area network controllers 1 or 0 (CAN1 or CAN0). It can be configured as the slave select pin $\overline{\text{SS}}$ of the serial peripheral interface 0 (SPI0).

1.2.3.49 PM2 / RXCAN1 / RXCAN0 / MISO0 — Port M I/O Pin 2

PM2 is a general-purpose input or output pin. It can be configured as the receive pin RXCAN of the scalable controller area network controllers 1 or 0 (CAN1 or CAN0). It can be configured as the master input (during master mode) or slave output pin (during slave mode) MISO for the serial peripheral interface 0 (SPI0).

1.2.3.50 PM1 / TXCAN0 — Port M I/O Pin 1

PM1 is a general-purpose input or output pin. It can be configured as the transmit pin TXCAN of the scalable controller area network controller 0 (CAN0).

1.2.3.51 PM0 / RXCAN0 — Port M I/O Pin 0

PM0 is a general-purpose input or output pin. It can be configured as the receive pin RXCAN of the scalable controller area network controller 0 (CAN0).

1.2.3.52 PP7 / KWP7 / PWM7 / SCK2 — Port P I/O Pin 7

PP7 is a general-purpose input or output pin. It can be configured to generate an interrupt causing the MCU to exit stop or wait mode. It can be configured as pulse width modulator (PWM) channel 7 output. It can be configured as serial clock pin SCK of the serial peripheral interface 2 (SPI2).

1.2.3.53 PP6 / KWP6 / PWM6 / $\overline{\text{SS2}}$ — Port P I/O Pin 6

PP6 is a general-purpose input or output pin. It can be configured to generate an interrupt causing the MCU to exit stop or wait mode. It can be configured as pulse width modulator (PWM) channel 6 output. It can be configured as slave select pin $\overline{\text{SS}}$ of the serial peripheral interface 2 (SPI2).

1.2.3.54 PP5 / KWP5 / PWM5 / MOSI2 — Port P I/O Pin 5

PP5 is a general-purpose input or output pin. It can be configured to generate an interrupt causing the MCU to exit stop or wait mode. It can be configured as pulse width modulator (PWM) channel 5 output. It can

Table 4-1. ATD10B16CV4 Memory Map

Address Offset	Use	Access
0x0000	ATD Control Register 0 (ATDCTL0)	R/W
0x0001	ATD Control Register 1 (ATDCTL1)	R/W
0x0002	ATD Control Register 2 (ATDCTL2)	R/W
0x0003	ATD Control Register 3 (ATDCTL3)	R/W
0x0004	ATD Control Register 4 (ATDCTL4)	R/W
0x0005	ATD Control Register 5 (ATDCTL5)	R/W
0x0006	ATD Status Register 0 (ATDSTAT0)	R/W
0x0007	Unimplemented	
0x0008	ATD Test Register 0 (ATDTEST0)[1]	R
0x0009	ATD Test Register 1 (ATDTEST1)	R/W
0x000A	ATD Status Register 2 (ATDSTAT2)	R
0x000B	ATD Status Register 1 (ATDSTAT1)	R
0x000C	ATD Input Enable Register 0 (ATDDIEN0)	R/W
0x000D	ATD Input Enable Register 1 (ATDDIEN1)	R/W
0x000E	Port Data Register 0 (PORTAD0)	R
0x000F	Port Data Register 1 (PORTAD1)	R
0x0010, 0x0011	ATD Result Register 0 (ATDDR0H, ATDDR0L)	R/W
0x0012, 0x0013	ATD Result Register 1 (ATDDR1H, ATDDR1L)	R/W
0x0014, 0x0015	ATD Result Register 2 (ATDDR2H, ATDDR2L)	R/W
0x0016, 0x0017	ATD Result Register 3 (ATDDR3H, ATDDR3L)	R/W
0x0018, 0x0019	ATD Result Register 4 (ATDDR4H, ATDDR4L)	R/W
0x001A, 0x001B	ATD Result Register 5 (ATDDR5H, ATDDR5L)	R/W
0x001C, 0x001D	ATD Result Register 6 (ATDDR6H, ATDDR6L)	R/W
0x001E, 0x001F	ATD Result Register 7 (ATDDR7H, ATDDR7L)	R/W
0x0020, 0x0021	ATD Result Register 8 (ATDDR8H, ATDDR8L)	R/W
0x0022, 0x0023	ATD Result Register 9 (ATDDR9H, ATDDR9L)	R/W
0x0024, 0x0025	ATD Result Register 10 (ATDDR10H, ATDDR10L)	R/W
0x0026, 0x0027	ATD Result Register 11 (ATDDR11H, ATDDR11L)	R/W
0x0028, 0x0029	ATD Result Register 12 (ATDDR12H, ATDDR12L)	R/W
0x002A, 0x002B	ATD Result Register 13 (ATDDR13H, ATDDR13L)	R/W
0x002C, 0x002D	ATD Result Register 14 (ATDDR14H, ATDDR14L)	R/W
0x002E, 0x002F	ATD Result Register 15 (ATDDR15H, ATDDR15L)	R/W

[1] ATDTEST0 is intended for factory test purposes only.

NOTE

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

Chapter 5 Analog-to-Digital Converter (S12ATD10B8CV2)

5.1 Introduction

The ATD10B8C is an 8-channel, 10-bit, multiplexed input successive approximation analog-to-digital converter. Refer to device electrical specifications for ATD accuracy.

5.1.1 Features

- 8/10-bit resolution
- 7 µsec, 10-bit single conversion time
- Sample buffer amplifier
- Programmable sample time
- Left/right justified, signed/unsigned result data
- External trigger control
- Conversion completion interrupt generation
- Analog input multiplexer for 8 analog input channels
- Analog/digital input pin multiplexing
- 1-to-8 conversion sequence lengths
- Continuous conversion mode
- Multiple channel scans
- Configurable external trigger functionality on any AD channel or any of four additional external trigger inputs. The four additional trigger inputs can be chip external or internal. Refer to the device overview chapter for availability and connectivity.
- Configurable location for channel wrap around (when converting multiple channels in a sequence).

5.1.2 Modes of Operation

5.1.2.1 Conversion Modes

There is software programmable selection between performing single or continuous conversion on a single channel or multiple channels.

In freeze mode all clocks of the XGATE module may be stopped, depending on the module configuration (see Section 6.3.1.1, “XGATE Control Register (XGMCTL)”).

6.1.4 Block Diagram

Figure Figure 6-1 shows a block diagram of the XGATE.

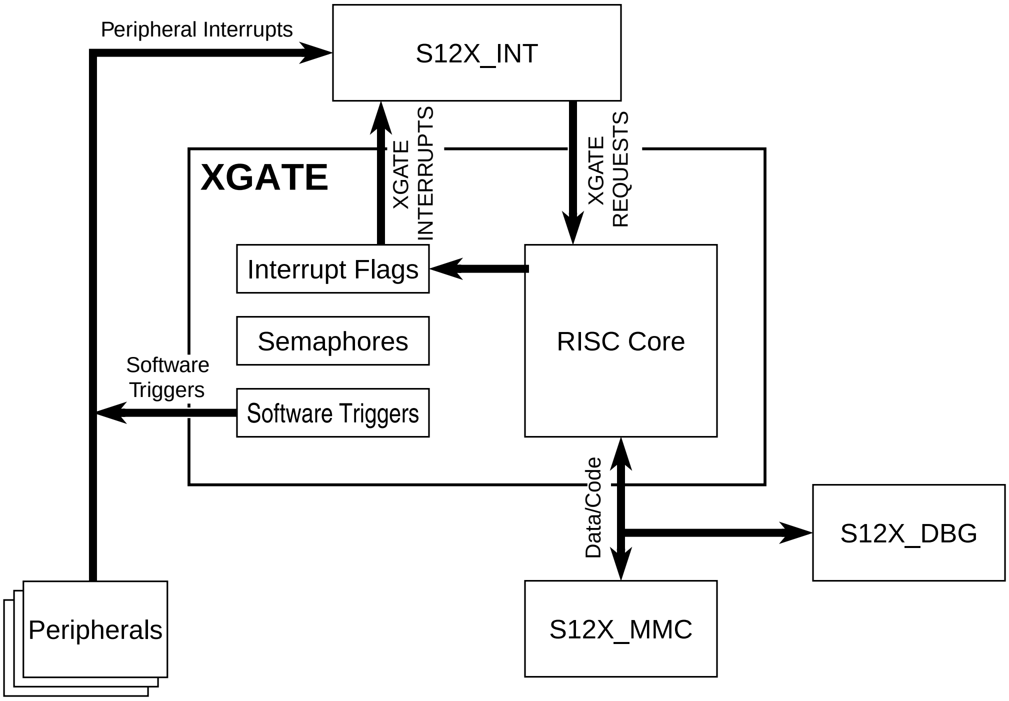


Figure 6-1. XGATE Block Diagram

6.2 External Signal Description

The XGATE module has no external pins.

BPL Branch if Plus BPL

Operation

If N = 0, then PC + $0002 + (REL9 << 1) ⇒ PC

Tests the Sign flag and branches if N = 0.

CCR Effects

N	Z	V	C
—	—	—	—

N: Not affected.
Z: Not affected.
V: Not affected.
C: Not affected.

Code and CPU Cycles

Source Form	Address Mode	Machine Code								Cycles
BPL REL9	REL9	0	0	1	0	1	0	0	REL9	PP/P

SEX Sign Extend Byte to Word SEX

Operation

The result in RD is the 16 bit sign extended representation of the original two’s complement number in the low byte of RD.L.

CCR Effects

N	Z	V	C
Δ	Δ	0	—

N: Set if bit 15 of the result is set; cleared otherwise.
Z: Set if the result is $0000; cleared otherwise.
V: 0; cleared.
C: Not affected.

Code and CPU Cycles

Source Form	Address Mode	Machine Code														Cycles
SEX RD	MON	0	0	0	0	0	RD	1	1	1	1	0	1	0	0	P

7.3.2.29 8-Bit Pulse Accumulators Holding Registers (PA3H–PA0H)

	7	6	5	4	3	2	1	0
R	PA3H7	PA3H6	PA3H5	PA3H4	PA3H3	PA3H2	PA3H1	PA3H0
W	▨	▨	▨	▨	▨	▨	▨	▨
Reset	0	0	0	0	0	0	0	0

▨ = Unimplemented or Reserved

Figure 7-51. 8-Bit Pulse Accumulators Holding Register 3 (PA3H)

	7	6	5	4	3	2	1	0
R	PA2H7	PA2H6	PA2H5	PA2H4	PA2H3	PA2H2	PA2H1	PA2H0
W	▨	▨	▨	▨	▨	▨	▨	▨
Reset	0	0	0	0	0	0	0	0

▨ = Unimplemented or Reserved

Figure 7-52. 8-Bit Pulse Accumulators Holding Register 2 (PA2H)

	7	6	5	4	3	2	1	0
R	PA1H7	PA1H6	PA1H5	PA1H4	PA1H3	PA1H2	PA1H1	PA1H0
W	▨	▨	▨	▨	▨	▨	▨	▨
Reset	0	0	0	0	0	0	0	0

▨ = Unimplemented or Reserved

Figure 7-53. 8-Bit Pulse Accumulators Holding Register 1 (PA1H)

	7	6	5	4	3	2	1	0
R	PA0H7	PA0H6	PA0H5	PA0H4	PA0H3	PA0H2	PA0H1	PA0H0
W	▨	▨	▨	▨	▨	▨	▨	▨
Reset	0	0	0	0	0	0	0	0

▨ = Unimplemented or Reserved

Figure 7-54. 8-Bit Pulse Accumulators Holding Register 0 (PA0H)

Read: Anytime.

Write: Has no effect.

All bits reset to zero.

These registers are used to latch the value of the corresponding pulse accumulator when the related bits in register ICPAR are enabled (see Section 7.4.1.3, “Pulse Accumulators”).

Figure 11-17 shows two cases of break detect. In trace RXD_1 the break symbol starts with the start bit, while in RXD_2 the break starts in the middle of a transmission. If BRKDFE = 1, in RXD_1 case there will be no byte transferred to the receive buffer and the RDRF flag will not be modified. Also no framing error or parity error will be flagged from this transfer. In RXD_2 case, however the break signal starts later during the transmission. At the expected stop bit position the byte received so far will be transferred to the receive buffer, the receive data register full flag will be set, a framing error and if enabled and appropriate a parity error will be set. Once the break is detected the BRKDIF flag will be set.

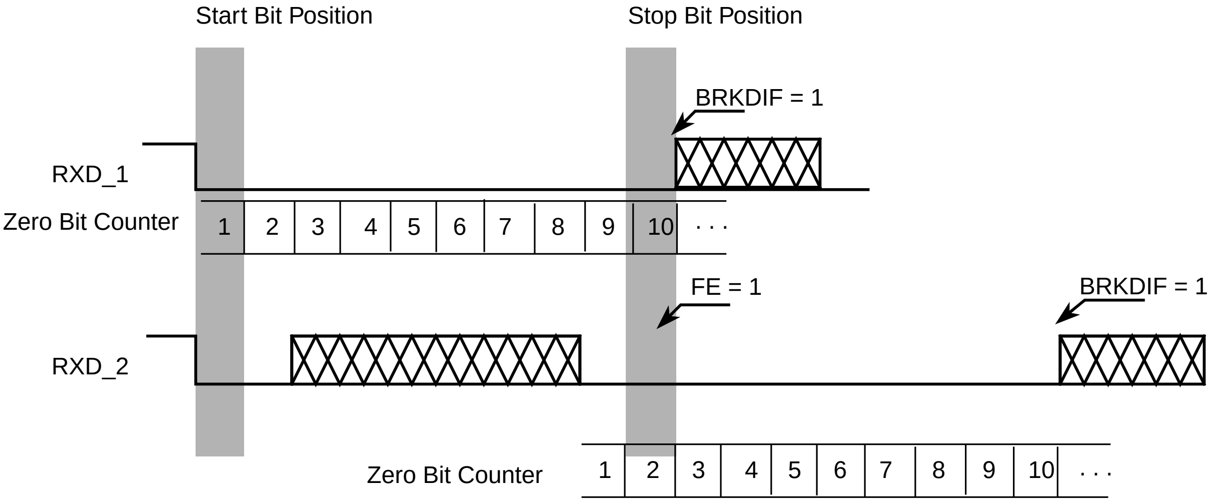


Figure 11-17. Break Detection if BRKDFE = 1 (M = 0)

11.4.5.4 Idle Characters

An idle character (or preamble) contains all logic 1s and has no start, stop, or parity bit. Idle character length depends on the M bit in SCI control register 1 (SCICR1). The preamble is a synchronizing idle character that begins the first transmission initiated after writing the TE bit from 0 to 1.

If the TE bit is cleared during a transmission, the TXD pin becomes idle after completion of the transmission in progress. Clearing and then setting the TE bit during a transmission queues an idle character to be sent after the frame currently being transmitted.

NOTE

When queueing an idle character, return the TE bit to logic 1 before the stop bit of the current frame shifts out through the TXD pin. Setting TE after the stop bit appears on TXD causes data previously written to the SCI data register to be lost. Toggle the TE bit for a queued idle character while the TDRE flag is set and immediately before writing the next byte to the SCI data register.

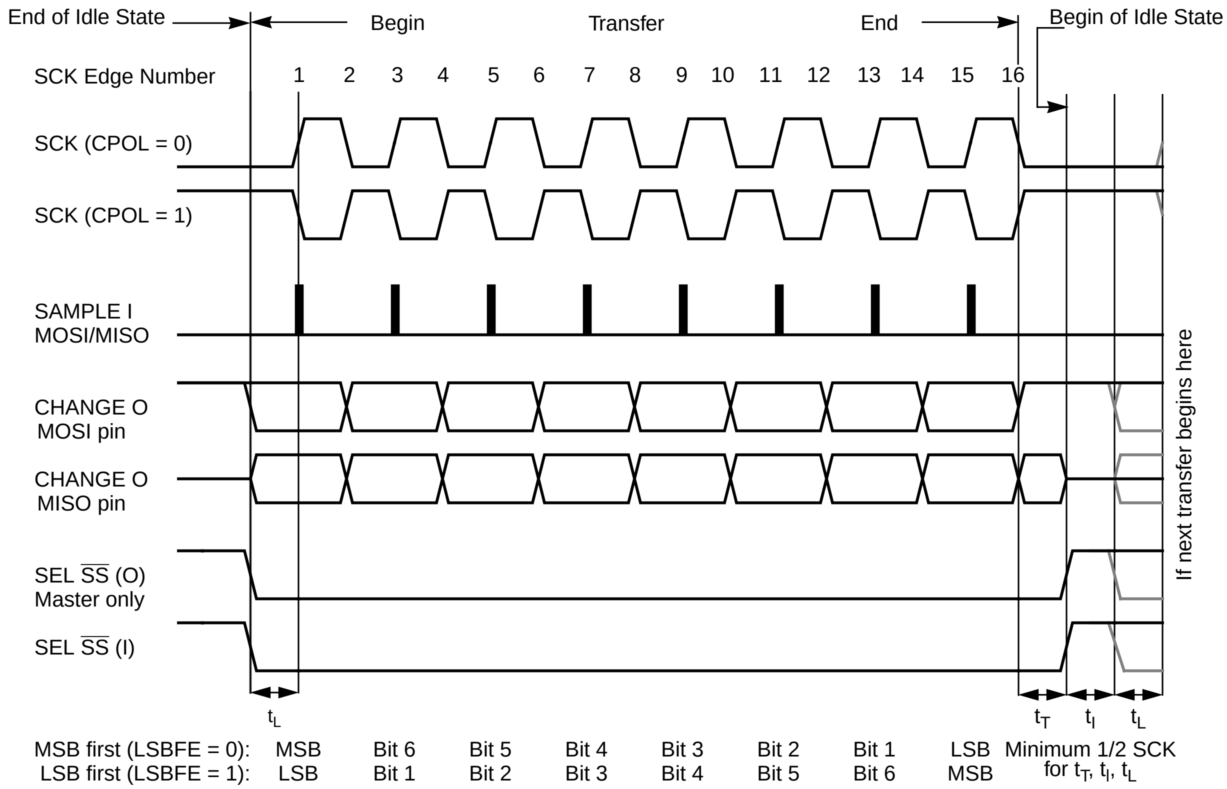
If the TE bit is clear and the transmission is complete, the SCI is not the master of the TXD pin

Data reception is double buffered. Data is shifted serially into the SPI shift register during the transfer and is transferred to the parallel SPI data register after the last bit is shifted in.

After the 16th (last) SCK edge:

- Data that was previously in the master SPI data register should now be in the slave data register and the data that was in the slave data register should be in the master.
- The SPIF flag in the SPI status register is set, indicating that the transfer is complete.

Figure 12-11 is a timing diagram of an SPI transfer where CPHA = 0. SCK waveforms are shown for CPOL = 0 and CPOL = 1. The diagram may be interpreted as a master or slave timing diagram because the SCK, MISO, and MOSI pins are connected directly between the master and the slave. The MISO signal is the output from the slave and the MOSI signal is the output from the master. The $\overline{SS}$ pin of the master must be either high or reconfigured as a general-purpose output not affecting the SPI.



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 12-11. SPI Clock Format 0 (CPHA = 0)

16.3.1.1 Interrupt Vector Base Register (IVBR)

Address: 0x0121

	7	6	5	4	3	2	1	0
R W	IVB_ADDR[7:0]							
Reset	1	1	1	1	1	1	1	1

Figure 16-3. Interrupt Vector Base Register (IVBR)

Read: Anytime

Write: Anytime

Table 16-2. IVBR Field Descriptions

Field	Description
7–0 IVB_ADDR[7:0]	**Interrupt Vector Base Address Bits** — These bits represent the upper byte of all vector addresses. Out of reset these bits are set to 0xFF (i.e., vectors are located at 0xFF10–0xFFFE) to ensure compatibility to HCS12. **Note:** A system reset will initialize the interrupt vector base register with “0xFF” before it is used to determine the reset vector address. Therefore, changing the IVBR has no effect on the location of the three reset vectors (0xFFFA–0xFFFE). **Note:** If the BDM is active (i.e., the CPU is in the process of executing BDM firmware code), the contents of IVBR are ignored and the upper byte of the vector address is fixed as “0xFF”.

20.3.2.6 Debug Count Register (DBGCNT)

Address: 0x0026

	7	6	5	4	3	2	1	0
R	0	CNT						
W	▨	▨	▨	▨	▨	▨	▨	▨
Reset	0	—	—	—	—	—	—	—
POR	0	0	0	0	0	0	0	0

▨ = Unimplemented or Reserved

Figure 20-8. Debug Count Register (DBGCNT)

Read: Anytime

Write: Never

Table 20-17. DBGCNT Field Descriptions

Field	Description
6–0 CNT[6:0]	**Count Value** — The CNT bits [6:0] indicate the number of valid data 64-bit data lines stored in the Trace Buffer. Table 20-18 shows the correlation between the CNT bits and the number of valid data lines in the Trace Buffer. When the CNT rolls over to zero, the TBF bit in DBGSR is set and incrementing of CNT will continue in end-trigger or mid-trigger mode. The DBGCNT register is cleared when ARM in DBGC1 is written to a one. The DBGCNT register is cleared by power-on-reset initialization but is not cleared by other system resets. Thus should a reset occur during a debug session, the DBGCNT register still indicates after the reset, the number of valid trace buffer entries stored before the reset occurred. The DBGCNT register is not decremented when reading from the trace buffer.

Table 20-18. CNT Decoding Table

TBF (DBGSR)	CNT[6:0]	Description
0	0000000	No data valid
0	0000001	32 bits of one line valid[1]
0	0000010 0000100 0000110 .. 1111100	1 line valid 2 lines valid 3 lines valid .. 62 lines valid
0	1111110	63 lines valid
1	0000000	64 lines valid; if using Begin trigger alignment, ARM bit will be cleared and the tracing session ends.
1	0000010 1111110	64 lines valid, oldest data has been overwritten by most recent data

[1] This applies to Normal/Loop1 Modes when tracing from either S12XCPU or XGATE only.

22.3 Memory Map and Register Definition

This section provides a detailed description of all PIM registers.

22.3.1 Module Memory Map

Table 22-2 shows the register map of the port integration module.

Table 22-2. PIM Memory Map (Sheet 1 of 3)

Address	Use	Access
0x0000	Port A Data Register (PORTA)	Read / Write
0x0001	Port B Data Register (PORTB)	Read / Write
0x0002	Port A Data Direction Register (DDRA)	Read / Write
0x0003	Port B Data Direction Register (DDRB)	Read / Write
0x0004	Port C Data Register (PORTC)	Read / Write
0x0005	Port D Data Register (PORTD)	Read / Write
0x0006	Port C Data Direction Register (DDRC)	Read / Write
0x0007	Port D Data Direction Register (DDRD)	Read / Write
0x0008	Port E Data Register (PORTE)	Read / Write[1]
0x0009	Port E Data Direction Register (DDRE)	Read / Write[1]
0x000A : 0x000B	Non-PIM Address Range	—
0x000C	Pull-up Up Control Register (PUCR)	Read / Write[1]
0x000D	Reduced Drive Register (RDRIV)	Read / Write[1]
0x000E : 0x001B	Non-PIM Address Range	—
0x001C	ECLK Control Register (ECLKCTL)	Read / Write[1]
0x001D	PIM Reserved	—
0x001E	IRQ Control Register (IRQCR)	Read / Write[1]
0x001F	PIM Reserved	—
0x0020 : 0x0031	Non-PIM Address Range	—
0x0032	Port K Data Register (PORTK)	Read / Write
0x0033	Port K Data Direction Register (DDRK)	Read / Write
0x0034 : 0x023F	Non-PIM Address Range	—
0x0240	Port T Data Register (PTT)	Read / Write

Chapter 23
DQ256 Port Integration Module (S12XDQ256PIMV2)

Introduction

The S12XD family port integration module (below referred to as PIM) establishes the interface between the peripheral modules including the non-multiplexed external bus interface module (S12X_EBI) and the I/O pins for all ports. It controls the electrical pin properties as well as the signal prioritization and multiplexing on shared pins.

This document covers the description of:

- Port A, B used as address output of the S12X_EBI and Port C, D used as data I/O of the S12X_EBI
- Port E associated with the S12X_EBI control signals and the IRQ, XIRQ interrupt inputs
- Port K associated with address output and control signals of the S12X_EBI
- Port T connected to the Enhanced Capture Timer (ECT) module
- Port S associated with 2 SCI and 1 SPI modules
- Port M associated with 3 MSCAN modules
- Port P connected to the PWM and 2 SPI modules — inputs can be used as an external interrupt source
- Port H associated with 1 SCI module — inputs can be used as an external interrupt source
- Port J associated with 1 MSCAN, 1 SCI, and 1 IIC module — inputs can be used as an external interrupt source
- Port AD0 and AD1 associated with one 8-channel andone 16-channel ATD module

Most I/O pins can be configured by register bits to select data direction and drive strength, to enable and select pull-up or pull-down devices. Interrupts can be enabled on specific pins resulting in status flags.

The I/O's of 2 MSCAN and 3 SPI modules can be routed from their default location to alternative port pins.

NOTE

The implementation of the PIM is device dependent. Therefore some functions are not available on certain derivatives or 112-pin and 80-pin package options.

23.0.1 Features

A full-featured PIM module includes these distinctive registers:

- Data and data direction registers for Ports A, B, C, D, E, K, T, S, M, P, H, J, AD0, and AD1 when used as general-purpose I/O

Table 23-1. Pin Functions and Priorities (Sheet 4 of 7)

Port	Pin Name	Pin Function and Priority	I/O	Description	Pin Function after Reset
M	PM7	TXCAN4	O	MSCAN4 transmit pin	GPIO
		GPIO	I/O	General-purpose I/O	
	PM6	RXCAN4	I	MSCAN4 receive pin	
		GPIO	I/O	General-purpose I/O	
	PM5	TXCAN2	O	MSCAN2 transmit pin	
		TXCAN0	O	MSCAN0 transmit pin	
		TXCAN4	O	MSCAN4 transmit pin	
		SCK0	I/O	Serial Peripheral Interface 0 serial clock pin *If CAN0 is routed to PM[3:2] the SPI0 can still be used in bidirectional master mode.*	
		GPIO	I/O	General-purpose I/O	
	PM4	RXCAN2	I	MSCAN2 receive pin	
		RXCAN0	I	MSCAN0 receive pin	
		RXCAN4	I	MSCAN4 receive pin	
		MOSI0	I/O	Serial Peripheral Interface 0 master out/slave in pin *If CAN0 is routed to PM[3:2] the SPI0 can still be used in bidirectional master mode.*	
		GPIO	I/O	General-purpose I/O	
	PM3	TXCAN1	O	MSCAN1 transmit pin	
		TXCAN0	O	MSCAN0 transmit pin	
		$\overline{\text{SS0}}$	I/O	Serial Peripheral Interface 0 slave select output in master mode, input for slave mode or master mode.	
		GPIO	I/O	General-purpose I/O	
	PM2	RXCAN1	I	MSCAN1 receive pin	
		RXCAN0	I	MSCAN0 receive pin	
		MISO0	I/O	Serial Peripheral Interface 0 master in/slave out pin	
		GPIO	I/O	General-purpose I/O	
	PM1	TXCAN0	O	MSCAN0 transmit pin	
		GPIO	I/O	General-purpose I/O	
	PM0	RXCAN0	I	MSCAN0 receive pin	
		GPIO	I/O	General-purpose I/O	

region are shown in the EEPROM memory map. The default protection setting is stored in the EEPROM configuration field as described in Table 25-1.

Table 25-1. EEPROM Configuration Field

Global Address	Size (bytes)	Description
0x13_FFFC	1	Reserved
0x13_FFFD	1	EEPROM Protection byte Refer to Section 25.3.2.5, “EEPROM Protection Register (EPROT)”
0x13_FFFE – 0x13_FFFF	2	Reserved

27.4.2.6 Sector Erase Abort Command

The sector erase abort operation will terminate the active sector erase operation so that other sectors in a Flash block are available for read and program operations without waiting for the sector erase operation to complete.

An example flow to execute the sector erase abort operation is shown in Figure 27-31. The sector erase abort command write sequence is as follows:

1. Write to any Flash block address to start the command write sequence for the sector erase abort command. The address and data written are ignored.
2. Write the sector erase abort command, 0x47, to the FCMD register.
3. Clear the CBEIF flag in the FSTAT register by writing a 1 to CBEIF to launch the sector erase abort command.

If the sector erase abort command is launched resulting in the early termination of an active sector erase operation, the ACCERR flag will set once the operation completes as indicated by the CCIF flag being set. The ACCERR flag sets to inform the user that the Flash sector may not be fully erased and a new sector erase command must be launched before programming any location in that specific sector. If the sector erase abort command is launched but the active sector erase operation completes normally, the ACCERR flag will not set upon completion of the operation as indicated by the CCIF flag being set. Therefore, if the ACCERR flag is not set after the sector erase abort command has completed, a Flash sector being erased when the abort command was launched will be fully erased. The maximum number of cycles required to abort a sector erase operation is equal to four FCLK periods (see Section 27.4.1.1, "Writing the FCLKDIV Register") plus five bus cycles as measured from the time the CBEIF flag is cleared until the CCIF flag is set. If sectors in multiple Flash blocks are being simultaneously erased, the sector erase abort operation will be applied to all active Flash blocks without writing to each Flash block in the sector erase abort command write sequence.

NOTE

Since the ACCERR bit in the FSTAT register may be set at the completion of the sector erase abort operation, a command write sequence is not allowed to be buffered behind a sector erase abort command write sequence. The CBEIF flag will not set after launching the sector erase abort command to indicate that a command should not be buffered behind it. If an attempt is made to start a new command write sequence with a sector erase abort operation active, the ACCERR flag in the FSTAT register will be set. A new command write sequence may be started after clearing the ACCERR flag, if set.

NOTE

The sector erase abort command should be used sparingly since a sector erase operation that is aborted counts as a complete program/erase cycle.