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	Not For New Designs
Core Processor	HCS12X
Core Size	16-Bit
Speed	80MHz
Connectivity	CANbus, I²C, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	59
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	12K x 8
Voltage - Supply (Vcc/Vdd)	2.35V ~ 5.5V
Data Converters	A/D 8x10b, 16x10b
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/product-detail/nxp-semiconductors/s912xdg128f2caar

Revision History

Date	Revision Level	Description
April, 2005	02.07	New Book
May, 2005	02.08	Minor corrections
May, 2005	02.09	removed ESD Machine Model from electrical characteristics added thermal characteristics added more details to run current measurement configurations VDDA supply voltage range 3.15V - 3.6V fot ATD Operating Characteristics I/O Chararcteristics for alll pins except EXTAL, XTAL .... corrected VREG electrical spec IDD wait max 95mA
May 2005	02.10	Improvements to NVM reliabity spec, added part numbers
July 2005	02.11	Added ROM parts to App.
October 2005	02.12	Single Souce S12XD Fam. Document, New Memory Map Figures,
May 2006	2.13	SPI electricals updated Voltage Regulator electricals updated Added Partnumbers and 1L15Y maskset Updated App. E 6SCI's on 112 pin DT/P512 and 3 SPI's on all D256 parts
June 2006	2.14	Data Sheet covers S12XD/B & A Family Included differnt pull device specification for differnt masksets
July 2006	2.15	Minor Corrections and Improvments
June 2007	2.16	Added 2M42E and 1M84E masksets
July 2007	2.17	Modified Appendix
April 2008	2.18	Better explanation of ATD0/1 for S12XD-Family see page 1305 S12XB256 ATD specification changed see Appendix E.6 added M23S maskset
August 2008	2.19	Corrected XGRAMSIZE of S12XD256 on page 44 Corrected 17.4.2.4 XGATE Memory Map Scheme Corrected 18.4.2.4 XGATE Memory Map Scheme
September 2009	2.20	Corrected Table E-6 , 30K flash memory available for XGATE on B256
October 2009	2.21	Corrected Footnote in Appendix E3 regarding Shared XGATE/CPU area

Chapter 5Analog-to-Digital Converter (S12ATD10B8CV3)

Chapter 6
XGATE (S12XGATEV2)

Table 1-12. Interrupt Vector Locations (Sheet 1 of 3)

Vector Address[1]	XGATE Channel ID[2]	Interrupt Source	CCR Mask	Local Enable
$FFFE	—	System reset or illegal access reset	None	None
$FFFC	—	Clock monitor reset	None	PLLCTL (CME, SCME)
$FFFA	—	COP watchdog reset	None	COP rate select
Vector base + $F8	—	Unimplemented instruction trap	None	None
Vector base+ $F6	—	SWI	None	None
Vector base+ $F4	—	$\overline{\text{XIRQ}}$	X Bit	None
Vector base+ $F2	—	$\overline{\text{IRQ}}$	I bit	IRQCR (IRQEN)
Vector base+ $F0	$78	Real time interrupt	I bit	CRGINT (RTIE)
Vector base+ $EE	$77	Enhanced capture timer channel 0	I bit	TIE (C0I)
Vector base + $EC	$76	Enhanced capture timer channel 1	I bit	TIE (C1I)
Vector base+ $EA	$75	Enhanced capture timer channel 2	I bit	TIE (C2I)
Vector base+ $E8	$74	Enhanced capture timer channel 3	I bit	TIE (C3I)
Vector base+ $E6	$73	Enhanced capture timer channel 4	I bit	TIE (C4I)
Vector base+ $E4	$72	Enhanced capture timer channel 5	I bit	TIE (C5I)
Vector base + $E2	$71	Enhanced capture timer channel 6	I bit	TIE (C6I)
Vector base+ $E0	$70	Enhanced capture timer channel 7	I bit	TIE (C7I)
Vector base+ $DE	$6F	Enhanced capture timer overflow	I bit	TSRC2 (TOF)
Vector base+ $DC	$6E	Pulse accumulator A overflow	I bit	PACTL (PAOVI)
Vector base + $DA	$6D	Pulse accumulator input edge	I bit	PACTL (PAI)
Vector base + $D8	$6C	SPI0	I bit	SPI0CR1 (SPIE, SPTIE)
Vector base+ $D6	$6B	SCI0	I bit	SCI0CR2 (TIE, TCIE, RIE, ILIE)
Vector base + $D4	$6A	SCI1	I bit	SCI1CR2 (TIE, TCIE, RIE, ILIE)
Vector base + $D2	$69	ATD0	I bit	ATD0CTL2 (ASCIE)
Vector base + $D2	Reserved			
Vector base + $D0	$68	ATD1	I bit	ATD1CTL2 (ASCIE)
Vector base + $CE	$67	Port J	I bit	PIEJ (PIEJ7-PIEJ0)
Vector base + $CC	$66	Port H	I bit	PIEH (PIEH7-PIEH0)
Vector base + $CA	$65	Modulus down counter underflow	I bit	MCCTL(MCZI)
Vector base + $C8	$64	Pulse accumulator B overflow	I bit	PBCTL(PBOVI)
Vector base + $C6	$63	CRG PLL lock	I bit	CRGINT(LOCKIE)
Vector base + $C4	$62	CRG self-clock mode	I bit	CRGINT (SCMIE)
Vector base + $C2	Reserved			
Vector base + $C0	$60	IIC0 bus	I bit	IBCR0 (IBIE)

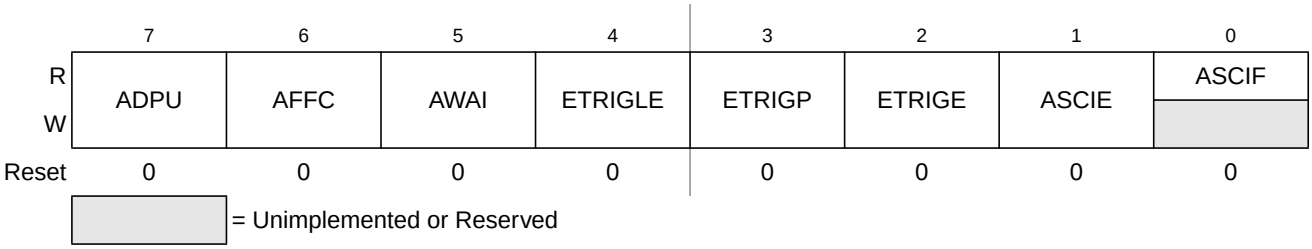


Figure 4-5. ATD Control Register 2 (ATDCTL2)

Read: Anytime

Write: Anytime

Table 4-6. ATDCTL2 Field Descriptions

Field	Description
7 ADPU	**ATD Power Down** — This bit provides on/off control over the ATD10B16C block allowing reduced MCU power consumption. Because analog electronic is turned off when powered down, the ATD requires a recovery time period after ADPU bit is enabled. 0 Power down ATD 1 Normal ATD functionality
6 AFFC	**ATD Fast Flag Clear All** 0 ATD flag clearing operates normally (read the status register ATDSTAT1 before reading the result register to clear the associate CCF flag). 1 Changes all ATD conversion complete flags to a fast clear sequence. Any access to a result register will cause the associate CCF flag to clear automatically.
5 AWAI	**ATD Power Down in Wait Mode** — When entering Wait Mode this bit provides on/off control over the ATD10B16C block allowing reduced MCU power. Because analog electronic is turned off when powered down, the ATD requires a recovery time period after exit from Wait mode. 0 ATD continues to run in Wait mode 1 Halt conversion and power down ATD during Wait mode After exiting Wait mode with an interrupt conversion will resume. But due to the recovery time the result of this conversion should be ignored.
4 ETRIGLE	**External Trigger Level/Edge Control** — This bit controls the sensitivity of the external trigger signal. See Table 4-7 for details.
3 ETRIGP	**External Trigger Polarity** — This bit controls the polarity of the external trigger signal. See Table 4-7 for details.
2 ETRIGE	**External Trigger Mode Enable** — This bit enables the external trigger on one of the AD channels or one of the ETRIG[3:0] inputs as described in Table 4-5. If external trigger source is one of the AD channels, the digital input buffer of this channel is enabled. The external trigger allows to synchronize the start of conversion with external events. 0 Disable external trigger 1 Enable external trigger
1 ASCIE	**ATD Sequence Complete Interrupt Enable** 0 ATD Sequence Complete interrupt requests are disabled. 1 ATD Interrupt will be requested whenever ASCIF = 1 is set.
0 ASCIF	**ATD Sequence Complete Interrupt Flag** — If ASCIE = 1 the ASCIF flag equals the SCF flag (see Section 4.3.2.7, “ATD Status Register 0 (ATDSTAT0)”), else ASCIF reads zero. Writes have no effect. 0 No ATD interrupt occurred 1 ATD sequence complete interrupt pending

CMP — Compare — CMP

Operation

RS2 – RS1 ⇒ NONE (translates to SUB R0, RS1, RS2)
RD – IMM16 ⇒ NONE (translates to CMPL RD, #IMM16[7:0]; CPCH RD, #IMM16[15:8])

Subtracts two 16 bit values and discards the result.

CCR Effects

N	Z	V	C
Δ	Δ	Δ	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.
RS1[15] & $\overline{RS2[15]}$ & $\overline{result[15]}$ | $\overline{RS1[15]}$ & RS2[15] & result[15]
RD[15] & $\overline{IMM16[15]}$ & $\overline{result[15]}$ | $\overline{RD[15]}$ & IMM16[15] & result[15]

C: Set if there is a carry from the bit 15 of the result; cleared otherwise.
$\overline{RS1[15]}$ & RS2[15] | $\overline{RS1[15]}$ & result[15] | RS2[15] & result[15]
$\overline{RD[15]}$ & IMM16[15] | $\overline{RD[15]}$ & result[15] | IMM16[15] & result[15]

Code and CPU Cycles

Source Form	Address Mode	Machine Code											Cycles
CMP RS1, RS2	TRI	0	0	0	1	1	0	0	0	RS1	RS2	0	0	P
CMP RS, #IMM16	IMM8	1	1	0	1	0	RS			IMM16[7:0]				P
	IMM8	1	1	0	1	1	RS			IMM16[15:8]				P

6.9 Initialization and Application Information

6.9.1 Initialization

The recommended initialization of the XGATE is as follows:

1. Clear the XGE bit to suppress any incoming service requests.
2. Make sure that no thread is running on the XGATE. This can be done in several ways:
   a) Poll the XGCHID register until it reads $00. Also poll XGDBG and XGSWEIF to make sure that the XGATE has not been stopped.
   b) Enter Debug Mode by setting the XGDBG bit. Clear the XGCHID register. Clear the XGDBG bit.

   The recommended method is a).
3. Set the XGVBR register to the lowest address of the XGATE vector space.
4. Clear all Channel ID flags.
5. Copy XGATE vectors and code into the RAM.
6. Initialize the S12X_INT module.
7. Enable the XGATE by setting the XGE bit.

The following code example implements the XGATE initialization sequence.

6.9.2 Code Example (Transmit "Hello World!" on SCI)

```
                        CPU   S12X
                        ;###########################################
                        ;#                   SYMBOLS                #
                        ;###########################################
SCI_REGS                EQU   $00C8           ;SCI register space
SCIBDH                  EQU   SCI_REGS+$00    ;SCI Baud Rate Register
SCIBDL                  EQU   SCI_REGS+$00    ;SCI Baud Rate Register
SCICR2                  EQU   SCI_REGS+$03    ;SCI Control Register 2
SCISR1                  EQU   SCI_REGS+$04    ;SCI Status  Register 1
SCIDRL                  EQU   SCI_REGS+$07    ;SCI Control Register 2
TIE                     EQU   $80             ;TIE bit mask
TE                      EQU   $08             ;TE bit mask
RE                      EQU   $04             ;RE bit mask
SCI_VEC                 EQU   $D6             ;SCI vector number

INT_REGS                      EQU             $0120          ;S12X_INT register space
INT_CFADDR              EQU   INT_REGS+$07    ;Interrupt Configuration Address Register
INT_CFDATA              EQU   INT_REGS+$08    ;Interrupt Configuration Data Registers
RQST                    EQU   $80             ;RQST bit mask

XGATE_REGS              EQU   $0380           ;XGATE register space
XGMCTL                  EQU   XGATE_REGS+$00  ;XGATE Module Control Register
XGMCTL_CLEAR            EQU   $FA02           ;Clear all XGMCTL bits
XGMCTL_ENABLE           EQU   $8282           ;Enable XGATE
XGCHID                  EQU   XGATE_REGS+$02  ;XGATE Channel ID Register
XGVBR                   EQU   XGATE_REGS+$06  ;XGATE ISP Select Register
XGIF                    EQU   XGATE_REGS+$08  ;XGATE Interrupt Flag Vector
```

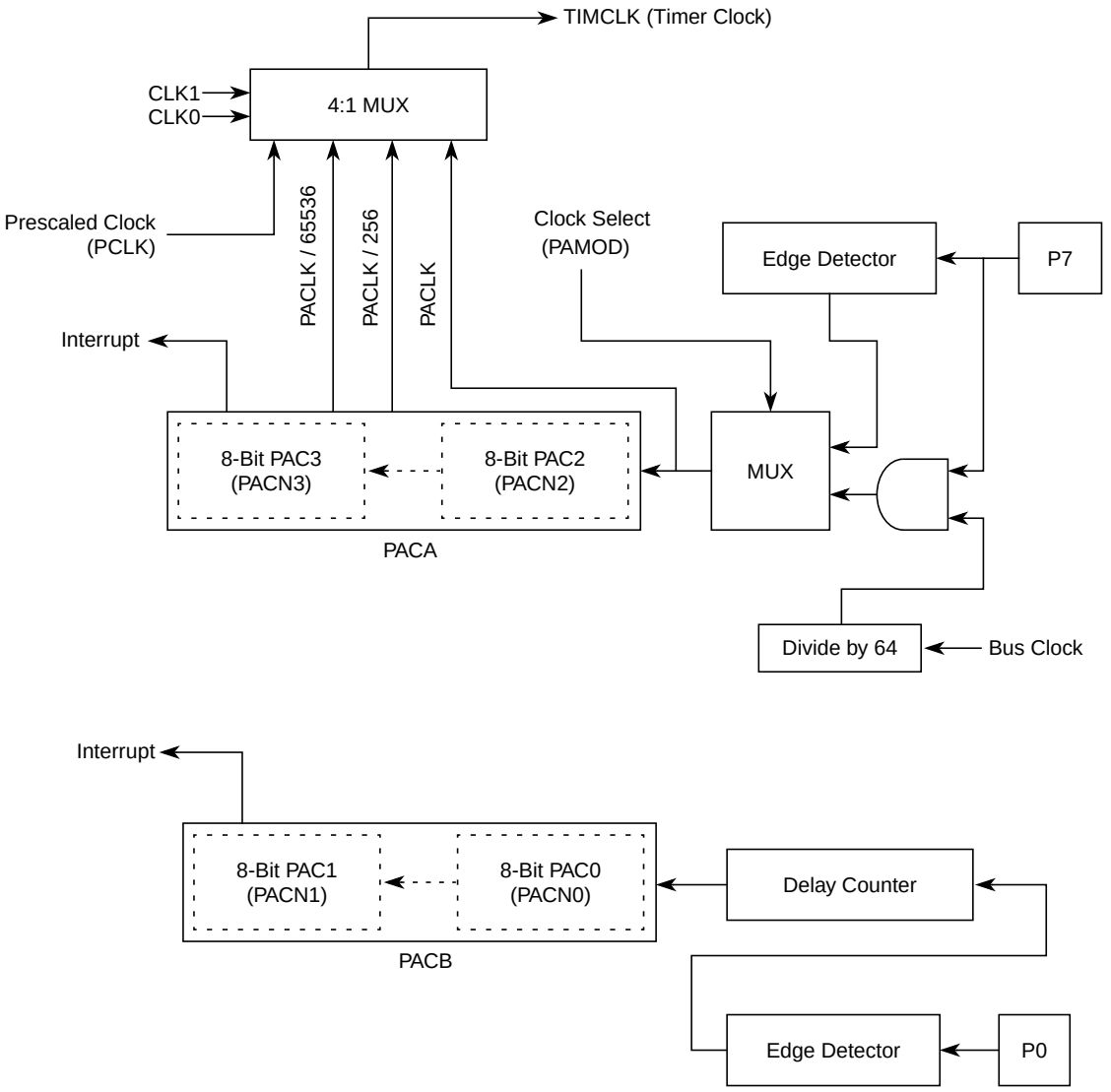


Figure 7-70. 16-Bit Pulse Accumulators Block Diagram

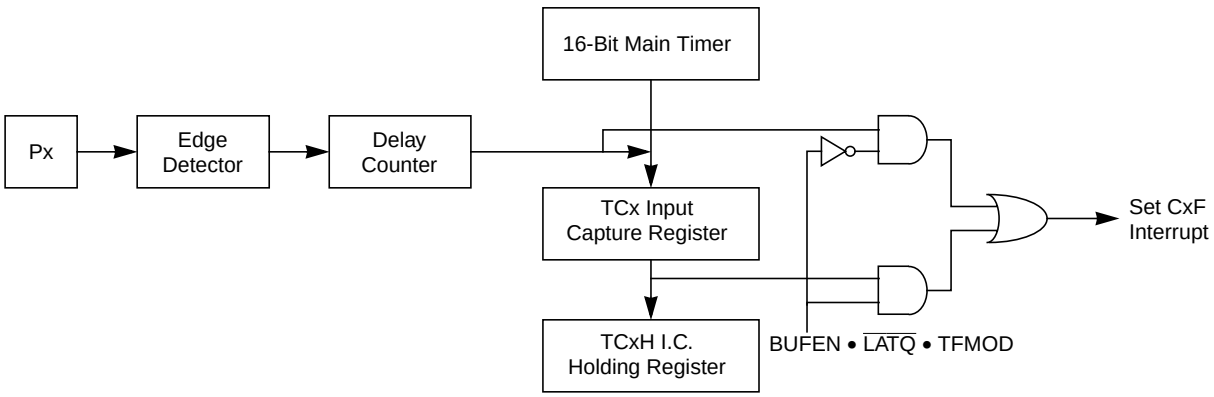


Figure 7-71. Block Diagram for Port 7 with Output Compare/Pulse Accumulator A

The MMCCTL0 register is used to control external bus functions, i.e., availability of chip selects.

CAUTION

XGATE write access to this register during an CPU access which makes use of this register could lead to unexpected results.

Table 17-4. MMCCTL0 Field Descriptions

Field	Description
3–0 CS[3:0]E	**Chip Select Enables** — Each of these bits enables one of the external chip selects $\overline{CS3}$, $\overline{CS2}$, $\overline{CS1}$, and $\overline{CS0}$ outputs which are asserted during accesses to specific external addresses. The associated global address ranges are shown in Table 1-6 and Table 1-21 and Figure 1-23. Chip selects are only active if enabled in normal expanded mode, Emulation expanded mode and special test mode. The function disabled in all other operating modes. 0 Chip select is disabled 1 Chip select is enabled

Table 17-5. Chip Select Signals

Global Address Range	Asserted Signal
0x00_0800–0x0F_FFFF	$\overline{CS3}$
0x10_0000–0x1F_FFFF	$\overline{CS2}$
0x20_0000–0x3F_FFFF	$\overline{CS1}$
0x40_0000–0x7F_FFFF	$\overline{CS0}$[1]

1 When the internal NVM is enabled (see ROMON in Section 1.3.2.5, “MMC Control Register (MMCCTL1)”) the $\overline{CS0}$ is not asserted in the space occupied by this on-chip memory block.

Table 18-14. PPAGE Field Descriptions

Field	Description
7–0 PIX[7:0]	**Program Page Index Bits 7–0** — These page index bits are used to select which of the 256 FLASH or ROM array pages is to be accessed in the Program Page Window.

The fixed 16K page from 0x4000–0x7FFF (when ROMHM = 0) is the page number 0xFD.

The reset value of 0xFE ensures that there is linear Flash space available between addresses 0x4000 and 0xFFFF out of reset.

The fixed 16K page from 0xC000-0xFFFF is the page number 0xFF.

18.3.2.9 RAM Write Protection Control Register (RAMWPC)

Address: 0x011C

	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 18-17. RAM Write Protection Control Register (RAMWPC)

Read: Anytime

Write: Anytime

Table 18-15. RAMWPC Field Descriptions

Field	Description
0 RWPE	**RAM Write Protection Enable** — This bit enables the RAM write protection mechanism. When the RWPE bit is cleared, there is no write protection and any memory location is writable by the CPU module and the XGATE module. When the RWPE bit is set the write protection mechanism is enabled and write access of the CPU or to the XGATE RAM region. Write access performed by the XGATE module to outside of the XGATE RAM region or the shared region is suppressed as well in this case. 0 RAM write protection check is disabled, region boundary registers can be written. 1 RAM write protection check is enabled, region boundary registers cannot be written.
1 AVIE	**CPU Access Violation Interrupt Enable** — This bit enables the Access Violation Interrupt. If AVIE is set and AVIF is set, an interrupt is generated. 0 CPU Access Violation Interrupt Disabled. 1 CPU Access Violation Interrupt Enabled.
0 AVIF	**CPU Access Violation Interrupt Flag** — When set, this bit indicates that the CPU has tried to write a memory location inside the XGATE RAM region. This flag can be reset by writing ’1’ to the AVIF bit location. 0 No access violation by the CPU was detected. 1 Access violation by the CPU was detected.

Table 18-21 shows the address boundaries of each chip select and the relationship with the implemented resources (internal) parameters.

Table 18-21. Global Chip Selects Memory Space

Chip Selects	Bottom Address	Top Address
$\overline{CS3}$	0x00_0800	0x0F_FFFF minus RAMSIZE[1]
$\overline{CS2}$	0x10_0000	0x13_FFFF minus EEPROMSIZE[2]
$\overline{CS2}$[3]	0x14_0000	0x1F_FFFF
$\overline{CS1}$	0x20_0000	0x3F_FFFF
$\overline{CS0}$[4]	0x40_0000	0x7F_FFFF minus FLASHSIZE[5]

1 External RPAGE accesses in (NX, EX and ST)

2 External EPAGE accesses in (NX, EX and ST)

3 When ROMHM is set (see ROMHM in Table 18-19) the $\overline{CS2}$ is asserted in the space occupied by this on-chip memory block.

4 When the internal NVM is enabled (see ROMON in Section 18.3.2.5, “MMC Control Register (MMCCTL1)) the $\overline{CS0}$ is not asserted in the space occupied by this on-chip memory block.

5 External PPAGE accesses in (NX, EX and ST)

Figure 18-23. Local to Implemented Global Address Mapping (Without GPAGE)

18.4.2.4 XGATE Memory Map Scheme

18.4.2.4.1 Expansion of the XGATE Local Address Map

The XGATE 64 Kbyte memory space allows access to internal resources only (Registers, RAM, and FLASH). The 2 Kilobyte register address range is the same register address range as for the CPU and the BDM module . XGATE can access the FLASH in single chip modes, even when the MCU is secured. In expanded modes, XGATE can not access the FLASH when MCU is secured.

The local address of the XGATE RAM access is translated to the global RAM address range. The XGATE shares the RAM resource with the CPU and the BDM module . The local address of the XGATE FLASH access is translated to the global address as shown in Figure 18-24. For the implemented memory spaces and addresses please refer to Table 1-4 and Table 1-5.

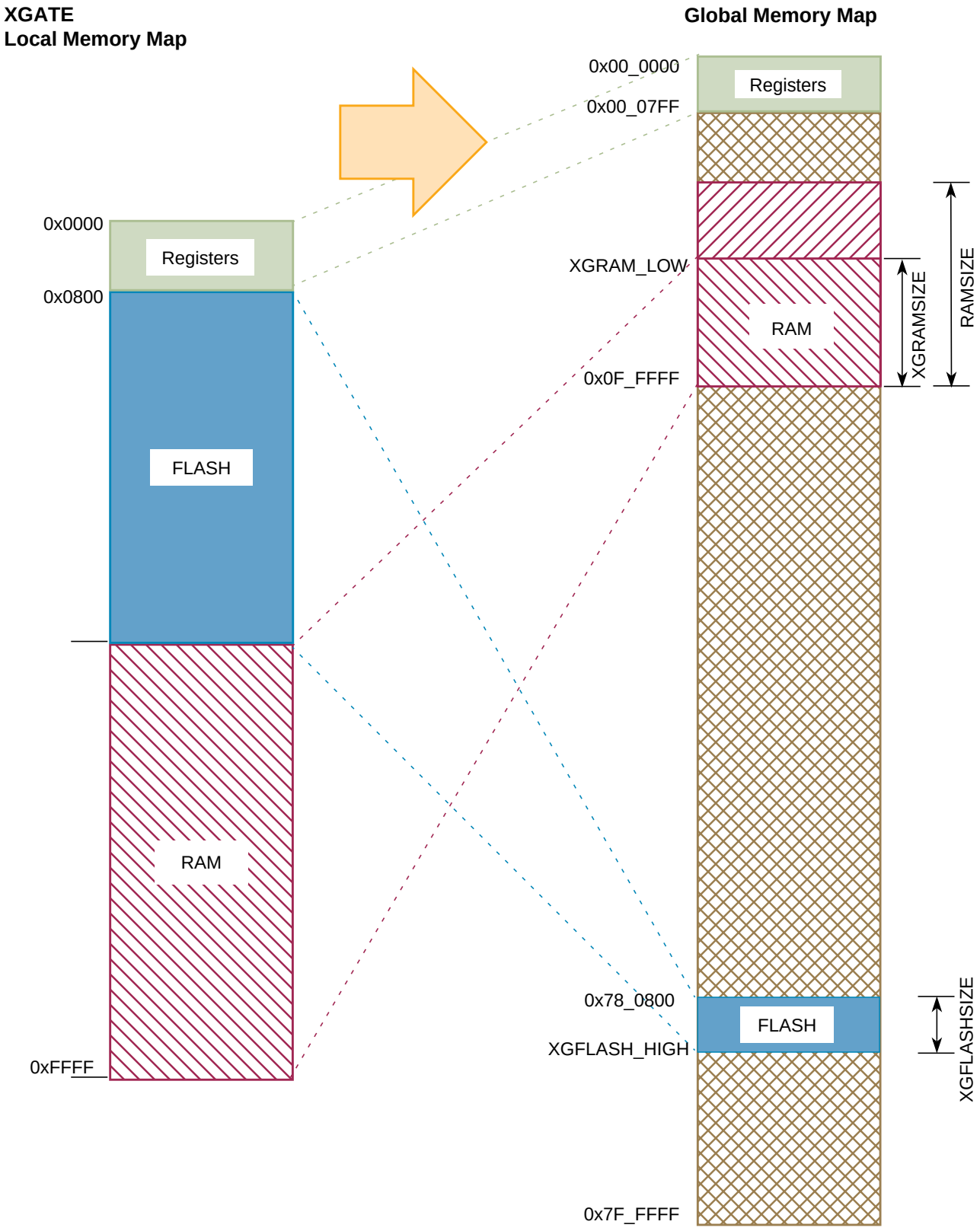


Figure 18-24. XGATE Global Address Mapping

21.4 Functional Description

This section describes the functions of the external bus interface. The availability of external signals and functions in relation to the operating mode is initially summarized and described in more detail in separate sub-sections.

21.4.1 Operating Modes and External Bus Properties

A summary of the external bus interface functions for each operating mode is shown in Table 21-7.

Table 21-7. Summary of Functions

Properties (if Enabled)	Single-Chip Modes		Expanded Modes			
	Normal Single-Chip	Special Single-Chip	Normal Expanded	Emulation Single-Chip	Emulation Expanded	Special Test
Timing Properties						
PRR access[1]	2 cycles read internal write internal	2 cycles read internal write internal	2 cycles read internal write internal	2 cycles read external write int & ext	2 cycles read external write int & ext	2 cycles read internal write internal
Internal access visible externally	—	—	—	1 cycle	1 cycle	1 cycle
External address access and unimplemented area access[2]	—	—	Max. of 2 to 9 programmed cycles or n cycles of ext. wait[3]	1 cycle	Max. of 2 to 9 programmed cycles or n cycles of ext. wait[3]	1 cycle
Flash area address access[4]	—	—	—	1 cycle	1 cycle	1 cycle
Signal Properties						
Bus signals	—	—	ADDR[22:1] DATA[15:0]	ADDR[22:20]/ACC[2:0] ADDR[19:16]/IQSTAT[3:0] ADDR[15:0]/IVD[15:0] DATA[15:0]	ADDR[22:20]/ACC[2:0] ADDR[19:16]/IQSTAT[3:0] ADDR[15:0]/IVD[15:0] DATA[15:0]	ADDR[22:0] DATA[15:0]
Data select signals (if 16-bit data bus)	—	—	$\overline{UDS}$ $\overline{LDS}$	ADDR0 $\overline{LSTRB}$	ADDR0 $\overline{LSTRB}$	ADDR0 $\overline{LSTRB}$
Data direction signals	—	—	$\overline{RE}$ $\overline{WE}$	R/$\overline{W}$	R/$\overline{W}$	R/$\overline{W}$
External wait feature	—	—	$\overline{EWAIT}$	—	$\overline{EWAIT}$	—
Reduced input threshold enabled on	—	—	Refer to Table 21-3	DATA[15:0] $\overline{EWAIT}$	DATA[15:0] $\overline{EWAIT}$	Refer to Table 21-3

1 Incl. S12X_EBI registers

2 Refer to S12X_MMC section.

3 If EWAITE = 1, the minimum number of external bus cycles is 3.

4 Available only if configured appropriately by ROMON and EROMON (refer to S12X_MMC section).

22.3.2.22 Port T Polarity Select Register (PPST)

	7	6	5	4	3	2	1	0
R W	PPST7	PPST6	PPST5	PPST4	PPST3	PPST2	PPST1	PPST0
Reset	0	0	0	0	0	0	0	0

Figure 22-24. Port T Polarity Select Register (PPST)

Read: Anytime.

Write: Anytime.

This register selects whether a pull-down or a pull-up device is connected to the pin.

Table 22-26. PPST Field Descriptions

Field	Description
7–0 PPST[7:0]	**Pull Select Port T** 0 A pull-up device is connected to the associated port T pin, if enabled by the associated bit in register PERT and if the port is used as input. 1 A pull-down device is connected to the associated port T pin, if enabled by the associated bit in register PERT and if the port is used as input.

22.3.2.23 Port S Data Register (PTS)

	7	6	5	4	3	2	1	0
R W	PTS7	PTS6	PTS5	PTS4	PTS3	PTS2	PTS1	PTS0
SCI/SPI	$\overline{SS0}$	SCK0	MOSI0	MISO0	TXD1	RXD1	TXD0	RXD0
Reset	0	0	0	0	0	0	0	0

Figure 22-25. Port S Data Register (PTS)

Read: Anytime.

Write: Anytime.

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.

23.0.5.35 Port M Polarity Select Register (PPSM)

	7	6	5	4	3	2	1	0
R W	PPSM7	PPSM6	PPSM5	PPSM4	PPSM3	PPSM2	PPSM1	PPSM0
Reset	0	0	0	0	0	0	0	0

Figure 23-37. Port M Polarity Select Register (PPSM)

Read: Anytime.

Write: Anytime.

This register selects whether a pull-down or a pull-up device is connected to the pin. If CAN is active, a pull-up device can be activated on the related RXCAN inputs, but not a pull-down.

Table 23-36. PPSM Field Descriptions

Field	Description
7–0 PPSM[7:0]	**Pull Select Port M** 0 A pull-up device is connected to the associated port M pin, if enabled by the associated bit in register PERM and if the port is used as general purpose or RXCAN input. 1 A pull-down device is connected to the associated port M pin, if enabled by the associated bit in register PERM and if the port is used as a general purpose but not as RXCAN.

23.0.5.36 Port M Wired-OR Mode Register (WOMM)

	7	6	5	4	3	2	1	0
R W	WOMM7	WOMM6	WOMM5	WOMM4	WOMM3	WOMM2	WOMM1	WOMM0
Reset	0	0	0	0	0	0	0	0

Figure 23-38. Port M Wired-OR Mode Register (WOMM)

Read: Anytime.

Write: Anytime.

This register configures the output pins as wired-OR. If enabled the output is driven active low only (open-drain). A logic level of “1” is not driven. It applies also to the CAN outputs and allows a multipoint connection of several serial modules. This bit has no influence on pins used as inputs.

Table 23-37. WOMM Field Descriptions

Field	Description
7–0 WOMM[7:0]	**Wired-OR Mode Port M** 0 Output buffers operate as push-pull outputs. 1 Output buffers operate as open-drain outputs.

Table 24-28. PTM Field Descriptions (continued)

Field	Description
3–2 PTM[3:2]	The routed CAN0 function (TXCAN0 and RXCAN0) takes precedence over the routed SPI0 and the general purpose I/O function if the routed CAN0 module is enabled. *Refer to MSCAN section for details.* The routed SPI0 function ($\overline{SS0}$ and MISO0) takes precedence of the general purpose I/O function if the routed SPI0 is enabled and not in bidirectional mode. *Refer to SPI section for details.*
1–0 PTM[1:0]	The CAN0 function (TXCAN0 and RXCAN0) takes precedence over the general purpose I/O function if the CAN0 module is enabled. *Refer to MSCAN section for details.*

24.0.5.27 Port M Input Register (PTIM)

	7	6	5	4	3	2	1	0
R	PTIM7	PTIM6	PTIM5	PTIM4	PTIM3	PTIM2	PTIM1	PTIM0
W								
Reset[1]	—	—	—	—	—	—	—	—

= Unimplemented or Reserved

Figure 24-29. Port M Input Register (PTIM)

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 can also be used to detect overload or short circuit conditions on output pins.

24.0.5.28 Port M Data Direction Register (DDRM)

	7	6	5	4	3	2	1	0
R W	DDRM7	DDRM6	DDRM5	DDRM4	DDRM3	DDRM2	DDRM1	DDRM0
Reset	0	0	0	0	0	0	0	0

Figure 24-30. Port M Data Direction Register (DDRM)

Read: Anytime.

Write: Anytime.

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

The CAN forces the I/O state to be an output for each port line associated with an enabled output (TXCAN). It also forces the I/O state to be an input for each port line associated with an enabled input (RXCAN). In those cases the data direction bits will not change.

	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

Figure 28-22. RESERVED3

All bits read 0 and are not writable.

28.3.2.14 RESERVED4

This register is reserved for factory testing and is not accessible.

	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

Figure 28-23. RESERVED4

All bits read 0 and are not writable.

28.4 Functional Description

28.4.1 Flash Command Operations

Write operations are used to execute program, erase, erase verify, erase abort, and data compress algorithms described in this section. The program and erase algorithms are controlled by a state machine whose timebase, FCLK, is derived from the oscillator clock via a programmable divider. The command register, as well as the associated address and data registers, operate as a buffer and a register (2-stage FIFO) so that a second command along with the necessary data and address can be stored to the buffer while the first command is still in progress. This pipelined operation allows a time optimization when programming more than one word on a specific row in the Flash block as the high voltage generation can be kept active in between two programming commands. The pipelined operation also allows a simplification of command launching. Buffer empty as well as command completion are signalled by flags in the Flash status register with corresponding interrupts generated, if enabled.

The next sections describe:

1. How to write the FCLKDIV register
2. Command write sequences to program, erase, erase verify, erase abort, and data compress operations on the Flash memory
3. Valid Flash commands
4. Effects resulting from illegal Flash command write sequences or aborting Flash operations

between the written data and the backdoor key data stored in the Flash memory. If all four words of data are written to the correct addresses in the correct order and the data matches the backdoor keys stored in the Flash memory, the MCU will be unsecured. The data must be written to the backdoor keys sequentially starting with 0x7F_FF00–1 and ending with 0x7F_FF06–7. 0x0000 and 0xFFFF are not permitted as backdoor keys. While the KEYACC bit is set, reads of the Flash memory will return invalid data.

The user code stored in the Flash memory must have a method of receiving the backdoor keys from an external stimulus. This external stimulus would typically be through one of the on-chip serial ports.

If the KEYEN[1:0] bits are in the enabled state (see Section 29.3.2.2, “Flash Security Register (FSEC)”), the MCU can be unsecured by the backdoor key access sequence described below:

1. Set the KEYACC bit in the Flash Configuration Register (FCNFG).
2. Write the correct four 16-bit words to Flash addresses 0xFF00–0xFF07 sequentially starting with 0x7F_FF00.
3. Clear the KEYACC bit. Depending on the user code used to write the backdoor keys, a wait cycle (NOP) may be required before clearing the KEYACC bit.
4. If all four 16-bit words match the backdoor keys stored in Flash addresses 0x7F_FF00–0x7F_FF07, the MCU is unsecured and the SEC[1:0] bits in the FSEC register are forced to the unsecure state of 1:0.

The backdoor key access sequence is monitored by an internal security state machine. An illegal operation during the backdoor key access sequence will cause the security state machine to lock, leaving the MCU in the secured state. A reset of the MCU will cause the security state machine to exit the lock state and allow a new backdoor key access sequence to be attempted. The following operations during the backdoor key access sequence will lock the security state machine:

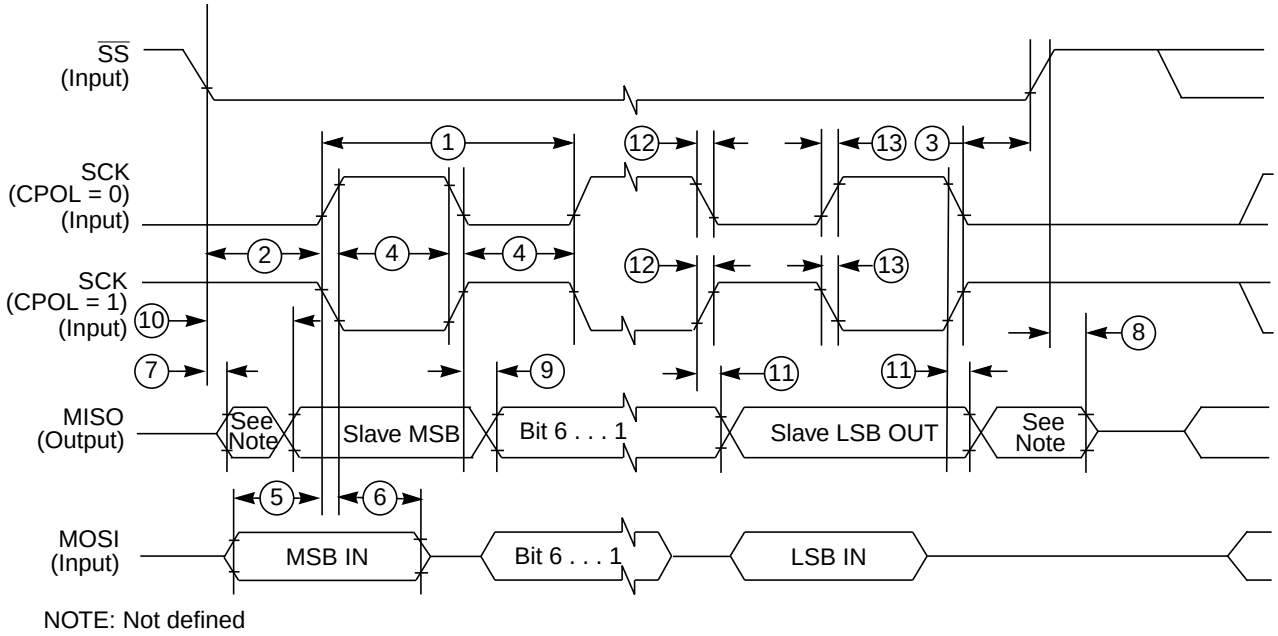
1. If any of the four 16-bit words does not match the backdoor keys programmed in the Flash array.
2. If the four 16-bit words are written in the wrong sequence.
3. If more than four 16-bit words are written.
4. If any of the four 16-bit words written are 0x0000 or 0xFFFF.
5. If the KEYACC bit does not remain set while the four 16-bit words are written.
6. If any two of the four 16-bit words are written on successive MCU clock cycles.

After the backdoor keys have been correctly matched, the MCU will be unsecured. Once the MCU is unsecured, the Flash security byte can be programmed to the unsecure state, if desired.

In the unsecure state, the user has full control of the contents of the backdoor keys by programming addresses 0x7F_FF00–0x7F_FF07 in the Flash Configuration Field.

The security as defined in the Flash security byte (0x7F_FF0F) is not changed by using the backdoor key access sequence to unsecure. The backdoor keys stored in addresses 0x7F_FF00–0x7F_FF07 are unaffected by the backdoor key access sequence. After the next reset of the MCU, the security state of the Flash module is determined by the Flash security byte (0x7F_FF0F). The backdoor key access sequence has no effect on the program and erase protections defined in the Flash protection register.

It is not possible to unsecure the MCU in special single chip mode by using the backdoor key access sequence in background debug mode (BDM).



NOTE: Not defined

Figure A-9. SPI Slave Timing (CPHA = 0)

Table E-1. Memory Sizes and Package Options S12XD-Family

Device	Package	Flash	RAM	EEPROM	ROM
9S12XD128	112 LQFP	128K	8K	2K	
	80 QFP				
9S12XD64	80 QFP	64K	4K	1K	