

Chapter 5

External Bus Interface (S12XEBIV4)

Table 5-1. Revision History

Rev. No. (Item No.)	Date (Submitted By)	Sections Affected	Substantial Change(s)
V04.01	12-Sep-05		- Added CSx stretch description.
V04.02	23-May-06		- Internal updates
V04.03	24-Jul-06		- Removed term IVIS

5.1 Introduction

This document describes the functionality of the XEBI block controlling the external bus interface.

The XEBI controls the functionality of a non-multiplexed external bus (a.k.a. 'expansion bus') in relationship with the chip operation modes. Dependent on the mode, the external bus can be used for data exchange with external memory, peripherals or PRU, and provide visibility to the internal bus externally in combination with an emulator.

5.1.1 Glossary or Terms

bus clock	System Clock. Refer to CRG Block Guide.
expanded modes	Normal Expanded Mode Emulation Single-Chip Mode Emulation Expanded Mode Special Test Mode
single-chip modes	Normal Single-Chip Mode Special Single-Chip Mode
emulation modes	Emulation Single-Chip Mode Emulation Expanded Mode
normal modes	Normal Single-Chip Mode Normal Expanded Mode
special modes	Special Single-Chip Mode Special Test Mode
NS	Normal Single-Chip Mode
SS	Special Single-Chip Mode
NX	Normal Expanded Mode
ES	Emulation Single-Chip Mode
EX	Emulation Expanded Mode
ST	Special Test Mode
external resource	Addresses outside MCU
PRR	Port Replacement Registers
PRU	Port Replacement Unit
EMULMEM	External emulation memory
access source	CPU or BDM or XGATE

5.1.2 Features

The XEBI includes the following features:

- Output of up to 23-bit address bus and control signals to be used with a non-muxed external bus
- Bidirectional 16-bit external data bus with option to disable upper half
- Visibility of internal bus activity

5.1.3 Modes of Operation

- Single-chip modes

The external bus interface is not available in these modes.

- Expanded modes

Address, data, and control signals are activated on the external bus in normal expanded mode and special test mode.

- Emulation modes

The external bus is activated to interface to an external tool for emulation of normal expanded mode or normal single-chip mode applications.

Refer to the S12X_MMC section for a detailed description of the MCU operating modes.

5.1.4 Block Diagram

Figure 5-1 is a block diagram of the XEBI with all related I/O signals.

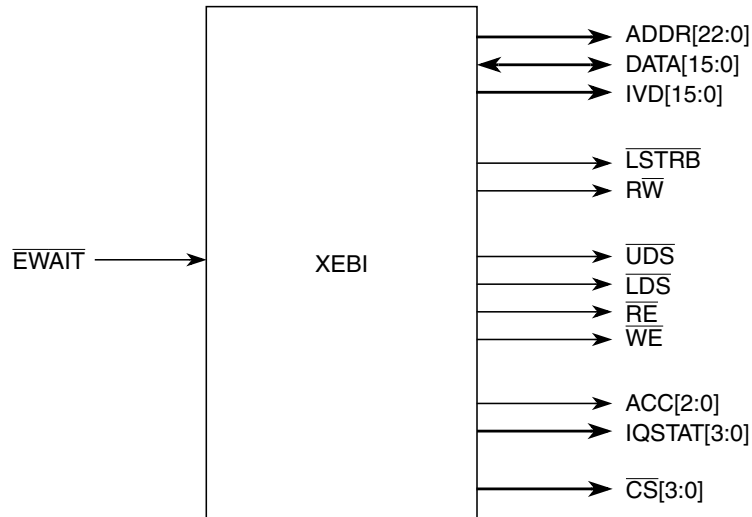


Figure 5-1. XEBI Block Diagram

5.2 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:

ECLK , ECLKX2 (free-running clocks) — PIM section

$\overline{\text{TAGHI}}$, $\overline{\text{TAGLO}}$ (tag inputs) — PIM section, S12X_DBG section

Table 5-2 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.

Table 5-2. External System Signals Associated with XEBI

Signal	I ¹ /O	EBI Signal Multiplex (T)ime ² (F)unction ³		Description	Available in Modes					
					NS	SS	NX	ES	EX	ST
\overline{RE}	O	—	—	Read Enable, indicates external read access	No	No	Yes	No	No	No
ADDR[22:20]	O	T	—	External address	No	No	Yes	Yes	Yes	Yes
ACC[2:0]	O		—	Access source	No	No	No	Yes	Yes	Yes
ADDR[19:16]	O	T	—	External address	No	No	Yes	Yes	Yes	Yes
IQSTAT[3:0]	O		—	Instruction Queue Status	No	No	No	Yes	Yes	Yes
ADDR[15:1]	O	T	—	External address	No	No	Yes	Yes	Yes	Yes
IVD[15:1]	O		—	Internal visibility read data	No	No	No	Yes	Yes	Yes
ADDR0	O	T	F	External address	No	No	No	Yes	Yes	Yes
IVD0	O			Internal visibility read data	No	No	No	Yes	Yes	Yes
\overline{UDS}	O	—	—	Upper Data Select, indicates external access to the high byte DATA[15:8]	No	No	Yes	No	No	No
\overline{LSTRB}	O	—	F	Low Strobe, indicates valid data on DATA[7:0]	No	No	No	Yes	Yes	Yes
\overline{LDS}	O	—		Lower Data Select, indicates external access to the low byte DATA[7:0]	No	No	Yes	No	No	No
\overline{RW}	O	—	F	Read/Write, indicates the direction of internal data transfers	No	No	No	Yes	Yes	Yes
\overline{WE}	O	—		Write Enable, indicates external write access	No	No	Yes	No	No	No
\overline{CS} [3:0]	O	—	—	Chip select	No	No	Yes	No	Yes	No
DATA[15:8]	I/O	—	—	Bidirectional data (even address)	No	No	Yes	Yes	Yes	Yes
DATA[7:0]	I/O	—	—	Bidirectional data (odd address)	No	No	Yes	Yes	Yes	Yes
\overline{EWAIT}	I	—	—	External control for external bus access stretches (adding wait states)	No	No	Yes	No	Yes	No

¹ All inputs are capable of reducing input threshold level

² Time-multiplex means that the respective signals share the same pin on chip level and are active alternating in a dedicated time slot (in modes where applicable).

³ Function-multiplex means that one of the respective signals sharing the same pin on chip level continuously uses the pin depending on configuration and reset state.

5.3 Memory Map and Register Definition

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

5.3.1 Module Memory Map

The registers associated with the XEBI block are shown in Figure 5-2.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0E EBICTL0	R	ITHRS	0	HDBE	ASIZ4	ASIZ3	ASIZ2	ASIZ1	ASIZ0
	W								
0x0F EBICTL1	R	0	EXSTR12	EXSTR11	EXSTR10	0	EXSTR02	EXSTR01	EXSTR00
	W								

= Unimplemented or Reserved

Figure 5-2. XEBI Register Summary

5.3.2 Register Descriptions

The following sub-sections provide a detailed description of each register and the individual register bits.

All control bits can be written anytime, but this may have no effect on the related function in certain operating modes. This allows specific configurations to be set up before changing into the target operating mode.

NOTE

Depending on the operating mode an available function may be enabled, disabled or depend on the control register bit. Reading the register bits will reflect the status of related function only if the current operating mode allows user control. Please refer the individual bit descriptions.

5.3.2.1 External Bus Interface Control Register 0 (EBICTL0)

Module Base +0x000E (PRR)

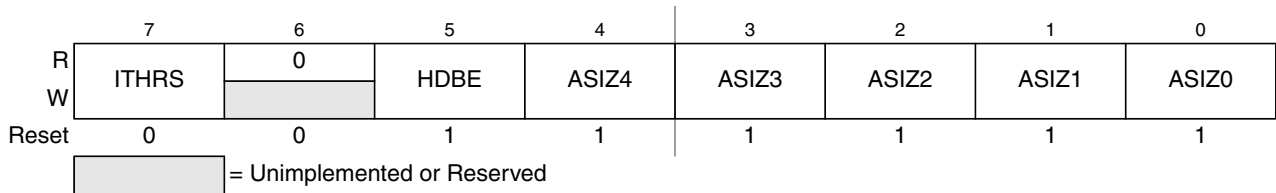


Figure 5-3. External Bus Interface Control Register 0 (EBICTL0)

Read: Anytime. In emulation modes, read operations will return the data from the external bus, in all other modes, the data is read from this register.

Write: Anytime. In emulation modes, write operations will also be directed to the external bus.

This register controls input pin threshold level and determines the external address and data bus sizes in normal expanded mode. If not in use with the external bus interface, the related pins can be used for alternative functions.

External bus is available as programmed in normal expanded mode and always full-sized in emulation modes and special test mode; function not available in single-chip modes.

Table 5-3. EBICTL0 Field Descriptions

Field	Description
7 ITHRS	<p>Reduced Input Threshold — This bit selects reduced input threshold on external data bus pins and specific control input signals which are in use with the external bus interface in order to adapt to external devices with a 3.3 V, 5 V tolerant I/O.</p> <p>The reduced input threshold level takes effect depending on ITHRS, the operating mode and the related enable signals of the EBI pin function as summarized in Table 5-4.</p> <p>0 Input threshold is at standard level on all pins 1 Reduced input threshold level enabled on pins in use with the external bus interface</p>
5 HDBE	<p>High Data Byte Enable — This bit enables the higher half of the 16-bit data bus. If disabled, only the lower 8-bit data bus can be used with the external bus interface. In this case the unused data pins and the data select signals (\overline{UDS} and \overline{LDS}) are free to be used for alternative functions.</p> <p>0 DATA[15:8], \overline{UDS}, and \overline{LDS} disabled 1 DATA[15:8], \overline{UDS}, and \overline{LDS} enabled</p>
4–0 ASIZ[4:0]	<p>External Address Bus Size — These bits allow scalability of the external address bus. The programmed value corresponds to the number of available low-aligned address lines (refer to Table 5-5). All address lines ADDR[22:0] start up as outputs after reset in expanded modes. This needs to be taken into consideration when using alternative functions on relevant pins in applications which utilize a reduced external address bus.</p>

Table 5-4. Input Threshold Levels on External Signals

ITHRS	External Signal	NS	SS	NX	ES	EX	ST
0	DATA[15:8] TAGHI, TAGLO	Standard	Standard	Standard	Reduced	Reduced	Standard
	DATA[7:0]				Standard	Standard	
	$\overline{\text{EWAIT}}$						
1	DATA[15:8] TAGHI, TAGLO	Standard	Standard	Reduced if HDBE = 1	Reduced	Reduced	Reduced
	DATA[7:0]			Reduced			
	$\overline{\text{EWAIT}}$			Reduced if EWAIT enabled ¹	Standard	Reduced if EWAIT enabled ¹	Standard

¹ EWAIT function is enabled if at least one $\overline{\text{CSx}}$ line is configured respectively in MMCCTL0. Refer to S12X_MMC section and Table 5-6.

Table 5-5. External Address Bus Size

ASIZ[4:0]	Available External Address Lines
00000	None
00001	$\overline{\text{UDS}}$
00010	ADDR1, $\overline{\text{UDS}}$
00011	ADDR[2:1], $\overline{\text{UDS}}$
:	:
10110	ADDR[21:1], $\overline{\text{UDS}}$
10111	ADDR[22:1], $\overline{\text{UDS}}$
:	:
11111	

5.3.2.2 External Bus Interface Control Register 1 (EBICTL1)

Module Base +0x000F (PRR)

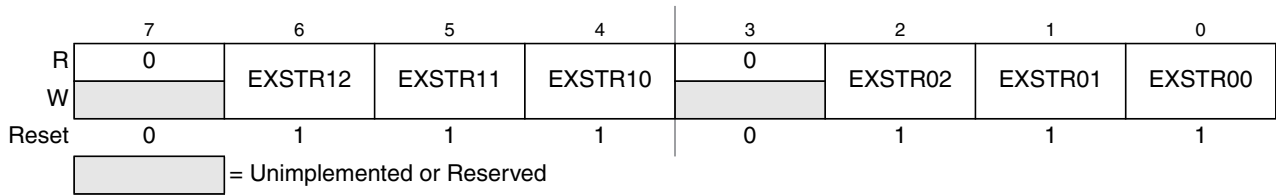


Figure 5-4. External Bus Interface Control Register 1 (EBICTL1)

Read: Anytime. In emulation modes, read operations will return the data from the external bus, in all other modes the data is read from this register.

Write: Anytime. In emulation modes, write operations will also be directed to the external bus.

This register allows programming of two independent values determining the amount of additional stretch cycles for external accesses (wait states).

With two bits in S12X_MMC register MMCCTL0 for every individual \overline{CSx} line one of the two counter options or the \overline{EWAIT} input is selected as stretch source. The chip select outputs can also be disabled to free up the pins for alternative functions (Table 5-6). Refer also to S12X_MMC section for register bit descriptions.

Table 5-6. Chip select function

CSxE1	CSxE0	Function
0	0	\overline{CSx} disabled
0	1	\overline{CSx} stretched with EXSTR0
1	0	\overline{CSx} stretched with EXSTR1
1	1	\overline{CSx} stretched with \overline{EWAIT}

If \overline{EWAIT} input usage is selected in MMCCTL0 the minimum number of stretch cycles is 2 for accesses to the related address range.

If configured respectively, stretch cycles are added as programmed or dependent on \overline{EWAIT} in normal expanded mode and emulation expanded mode; function not available in all other operating modes.

Table 5-7. EBICTL1 Field Descriptions

Field	Description
6–4 EXSTR1[2:0]	External Access Stretch Option 1 Bits 2, 1, 0 — This three bit field determines the amount of additional clock stretch cycles on every access to the external address space as shown in Table 5-8.
2–0 EXSTR0[2:0]	External Access Stretch Option 0 Bits 2, 1, 0 — This three bit field determines the amount of additional clock stretch cycles on every access to the external address space as shown in Table 5-8.

Table 5-8. External Access Stretch Bit Definition

EXSTRx[2:0]	Number of Stretch Cycles
000	1
001	2
010	3
011	4
100	5
101	6
110	7
111	8

5.4 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.

5.4.1 Operating Modes and External Bus Properties

A summary of the external bus interface functions for each operating mode is shown in [Table 5-9](#).

Table 5-9. 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]

Table 5-9. 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
Data select signals (if 16-bit data bus)	—	—	\overline{UDS} \overline{LDS}	$\overline{ADDR0}$ \overline{LSTRB}	$\overline{ADDR0}$ \overline{LSTRB}	$\overline{ADDR0}$ \overline{LSTRB}
Data direction signals	—	—	\overline{RE} \overline{WE}	RW	RW	RW
Chip Selects	—	—	$\overline{CS0}$ $\overline{CS1}$ $\overline{CS2}$ $\overline{CS3}$	—	$\overline{CS0}$ $\overline{CS1}$ $\overline{CS2}$ $\overline{CS3}$	—
External wait feature	—	—	EWAIT	—	EWAIT	—
Reduced input threshold enabled on	—	—	Refer to Table 5-4	DATA[15:0] EWAIT	DATA[15:0] EWAIT	Refer to Table 5-4

¹ Incl. S12X_EBI registers

² Refer to S12X_MMC section.

³ If EWAIT enabled for at least one \overline{CSx} line (refer to S12X_MMC section), the minimum number of external bus cycles is 3.

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

5.4.2 Internal Visibility

Internal visibility allows the observation of the internal CPU address and data bus as well as the determination of the access source and the CPU pipe (queue) status through the external bus interface.

Internal visibility is always enabled in emulation single chip mode and emulation expanded mode. Internal CPU accesses are made visible on the external bus interface except CPU execution of BDM firmware instructions.

Internal reads are made visible on $\overline{ADDRx/IVDx}$ (address and read data multiplexed, see Table 5-12 to Table 5-14), internal writes on \overline{ADDRx} and \overline{DATAx} (see Table 5-15 to Table 5-17). \overline{RW} and \overline{LSTRB} show the type of access. External read data are also visible on \overline{IVDx} .

During ‘no access’ cycles \overline{RW} is held in read position while \overline{LSTRB} is undetermined.

All accesses which make use of the external bus interface are considered external accesses.

5.4.2.1 Access Source Signals (ACC)

The access source can be determined from the external bus control signals ACC[2:0] as shown in Table 5-10.

Table 5-10. Determining Access Source from Control Signals

ACC[2:0]	Access Description
000	Repetition of previous access cycle
001	CPU access
010	BDM external access

Table 5-10. Determining Access Source from Control Signals

ACC[2:0]	Access Description
011	XGATE PRR access
100	No access ¹
101	CPU access error
110, 111	Reserved

¹ Denotes also CPU accesses to BDM firmware and BDM registers (IQSTATx are 'XXXX' and $R\bar{W} = 1$ in these cases)

5.4.2.2 Instruction Queue Status Signals (IQSTAT)

The CPU instruction queue status (execution-start and data-movement information) is brought out as IQSTAT[3:0] signals. For decoding of the IQSTAT values, refer to the S12X_CPU section.

5.4.2.3 Internal Visibility Data (IVD)

Depending on the access size and alignment, either a word of read data is made visible on the address lines or only the related data byte will be presented in the ECLK low phase. For details refer to Table 5-11.

Invalid IVD are brought out in case of non-CPU read accesses.

Table 5-11. IVD Read Data Output

Access	IVD[15:8]	IVD[7:0]
Word read of data at an even and even+1 address	ivd(even)	ivd(even+1)
Word read of data at an odd and odd+1 internal RAM address (misaligned)	ivd(odd+1)	ivd(odd)
Byte read of data at an even address	ivd(even)	addr[7:0] (rep.)
Byte read of data at an odd address	addr[15:8] (rep.)	ivd(odd)

5.4.2.4 Emulation Modes Timing

A bus access lasts 1 ECLK cycle. In case of a stretched external access (emulation expanded mode), up to an infinite amount of ECLK cycles may be added. ADDR_x values will only be shown in ECLK high phases, while ACC_x, IQSTAT_x, and IVD_x values will only be presented in ECLK low phases.

Based on this multiplex timing, ACC_x are only shown in the current (first) access cycle. IQSTAT_x and (for read accesses) IVD_x follow in the next cycle. If the access takes more than one bus cycle, ACC_x display NULL (0x000) in the second and all following cycles of the access. IQSTAT_x display NULL (0x0000) from the third until one cycle after the access to indicate continuation.

The resulting timing pattern of the external bus signals is outlined in the following tables for read, write and interleaved read/write accesses. Three examples represent different access lengths of 1, 2, and n–1 bus cycles. Non-shaded bold entries denote all values related to Access #0.

The following terminology is used:

- ‘addr’ — value(ADDRx); small letters denote the logic values at the respective pins
- ‘x’ — Undefined output pin values
- ‘z’ — Tristate pins
- ‘?’ — Dependent on previous access (read or write); IVDx: ‘ivd’ or ‘x’; DATAx: ‘data’ or ‘z’

5.4.2.4.1 Read Access Timing

Table 5-12. Read Access (1 Cycle)

Bus cycle ->	...	Access #0				Access #1		...
		1		2		3		
ECLK phase	...	high	low	high	low	high	low	...
ADDR[22:20] / ACC[2:0]	...	addr 0	acc 0	addr 1	acc 1	addr 2	acc 2	...
ADDR[19:16] / IQSTAT[3:0]	...		iqstat -1		iqstat 0		iqstat 1	...
ADDR[15:0] / IVD[15:0]	...		?		ivd 0		ivd 1	...
DATA[15:0] (internal read)	...	?	z	z	z	z	z	...
DATA[15:0] (external read)	...	?	z	data 0	z	data 1	z	...
RW	...	1	1	1	1	1	1	...

Table 5-13. Read Access (2 Cycles)

Bus cycle ->	...	Access #0				Access #1		...	
		1		2		3			
ECLK phase	...	high	low	high	low	high	low	...	
ADDR[22:20] / ACC[2:0]	...	addr 0	acc 0	addr 0	000	addr 1	acc 1	...	
ADDR[19:16] / IQSTAT[3:0]	...		iqstat-1		iqstat 0		0000	0000	...
ADDR[15:0] / IVD[15:0]	...		?		x		x	ivd 0	...
DATA[15:0] (internal read)	...	?	z	z	z	z	z	...	
DATA[15:0] (external read)	...	?	z	z	z	data 0	z	...	
RW	...	1	1	1	1	1	1	...	

Table 5-14. Read Access (n-1 Cycles)

Bus cycle ->	...	Access #0						Access #1		...	
		1		2		3		n			
ECLK phase	...	high	low	high	low	high	low	...	high	low	...
ADDR[22:20] / ACC[2:0]	...	addr 0	acc 0	addr 0	000	addr 0	000	addr 1	acc 1	...	
ADDR[19:16] / IQSTAT[3:0]	...		iqstat-1		iqstat 0		0000		0000	0000	...
ADDR[15:0] / IVD[15:0]	...		?		x		x		x	ivd 0	...
DATA[15:0] (internal read)	...	?	z	z	z	z	z	z	z	...	
DATA[15:0] (external read)	...	?	z	z	z	z	z	data 0	z	...	
RW	...	1	1	1	1	1	1	...	1	1	...

5.4.2.4.2 Write Access Timing

Table 5-15. Write Access (1 Cycle)

		Access #0		Access #1		Access #2		
Bus cycle ->	...	1		2		3		...
ECLK phase	...	high	low	high	low	high	low	...
ADDR[22:20] / ACC[2:0]	...	addr 0	acc 0	addr 1	acc 1	addr 2	acc 2	...
ADDR[19:16] / IQSTAT[3:0]	...		iqstat -1		iqstat 0		iqstat 1	...
ADDR[15:0] / IVD[15:0]	...		?		x		x	...
DATA[15:0] (write)	...	?	data 0		data 1		data 2	...
R \bar{W}	...	0	0	1	1	1	1	...

Table 5-16. Write Access (2 Cycles)

		Access #0				Access #1		
Bus cycle ->	...	1		2		3		...
ECLK phase	...	high	low	high	low	high	low	...
ADDR[22:20] / ACC[2:0]	...	addr 0	acc 0	addr 0	000	addr 1	acc 1	...
ADDR[19:16] / IQSTAT[3:0]	...		iqstat-1		iqstat 0		0000	...
ADDR[15:0] / IVD[15:0]	...		?		x		x	...
DATA[15:0] (write)	...	?	data 0				x	...
R \bar{W}	...	0	0	0	0	1	1	...

Table 5-17. Write Access (n-1 Cycles)

		Access #0						Access #1			
Bus cycle ->	...	1		2		3		...	n		
ECLK phase	...	high	low	high	low	high	low	...	high	low	
ADDR[22:20] / ACC[2:0]	...	addr 0	acc 0	addr 0	000	addr 0	000	...	addr 1	acc 1	
ADDR[19:16] / IQSTAT[3:0]	...		iqstat-1		iqstat 0		0000	...		0000	...
ADDR[15:0] / IVD[15:0]	...		?		x		x	...		x	...
DATA[15:0] (write)	...	?	data 0						x	...	
R \bar{W}	...	0	0	0	0	0	0	...	1	1	

5.4.2.4.3 Read-Write-Read Access Timing

Table 5-18. Interleaved Read-Write-Read Accesses (1 Cycle)

		Access #0		Access #1		Access #2		
Bus cycle ->	...	1		2		3		...
ECLK phase	...	high	low	high	low	high	low	...
ADDR[22:20] / ACC[2:0]	...	addr 0	acc 0	addr 1	acc 1	addr 2	acc 2	...
ADDR[19:16] / IQSTAT[3:0]	...		iqstat -1		iqstat 0		iqstat 1	...
ADDR[15:0] / IVD[15:0]	...		?		ivd 0		x	...

Table 5-18. Interleaved Read-Write-Read Accesses (1 Cycle)

DATA[15:0] (internal read)	...	?	z	z	(write) data 1	z	...
DATA[15:0] (external read)	...	?	z	data 0	(write) data 1	z	...
R \bar{W}	...	1	1	0	0	1	1

5.4.3 Accesses to Port Replacement Registers

All read and write accesses to PRR addresses take two bus clock cycles independent of the operating mode. If writing to these addresses in emulation modes, the access is directed to both, the internal register and the external resource while reads will be treated external.

The XEBI control registers also belong to this category.

5.4.4 Stretched External Bus Accesses

In order to allow fast internal bus cycles to coexist in a system with slower external resources, the XEBI supports stretched external bus accesses (wait states) for each external address range related to one of the 4 chip select lines individually.

This feature is available in normal expanded mode and emulation expanded mode for accesses to all external addresses except emulation memory and PRR. In these cases the fixed access times are 1 or 2 cycles, respectively.

Stretched accesses are controlled by:

1. EXSTR1[2:0] and EXSTR0[2:0] bits in the EBICTL1 register configuring a fixed amount of stretch cycles individually for each \overline{CSx} line in MMCCTL0
2. Activation of the external wait feature for each \overline{CSx} line MMCCTL0 register
3. Assertion of the external \overline{EWAIT} signal when at least one \overline{CSx} line is configured for EWAIT

The EXSTRx[2:0] control bits can be programmed for generation of a fixed number of 1 to 8 stretch cycles. If the external wait feature is enabled, the minimum number of additional stretch cycles is 2. An arbitrary amount of stretch cycles can be added using the \overline{EWAIT} input.

\overline{EWAIT} needs to be asserted at least for a minimal specified time window within an external access cycle for the internal logic to detect it and add a cycle (refer to electrical characteristics). Holding it for additional cycles will cause the external bus access to be stretched accordingly.

Write accesses are stretched by holding the initiator in its current state for additional cycles as programmed and controlled by external wait after the data have been driven out on the external bus. This results in an extension of time the bus signals and the related control signals are valid externally.

Read data are not captured by the system in normal expanded mode until the specified setup time before the \overline{RE} rising edge.

Read data are not captured in emulation expanded mode until the specified setup time before the falling edge of ECLK.

In emulation expanded mode, accesses to the internal flash or the emulation memory (determined by EROMON and ROMON bits; see S12X_MMC section for details) always take 1 cycle and stretching is not supported. In case the internal flash is taken out of the map in user applications, accesses are stretched as programmed and controlled by external wait.

5.4.5 Data Select and Data Direction Signals

The S12X_EBI supports byte and word accesses at any valid external address. The big endian system of the MCU is extended to the external bus; however, word accesses are restricted to even aligned addresses. The only exception is the visibility of misaligned word accesses to addresses in the internal RAM as this module exclusively supports these kind of accesses in a single cycle.

With the above restriction, a fixed relationship is implied between the address parity and the dedicated bus halves where the data are accessed: DATA[15:8] is related to even addresses and DATA[7:0] is related to odd addresses.

In expanded modes the data access type is externally determined by a set of control signals, i.e., data select and data direction signals, as described below. The data select signals are not available if using the external bus interface with an 8-bit data bus.

5.4.5.1 Normal Expanded Mode

In normal expanded mode, the external signals \overline{RE} , \overline{WE} , \overline{UDS} , \overline{LDS} indicate the access type (read/write), data size and alignment of an external bus access (Table 5-19).

Table 5-19. Access in Normal Expanded Mode

Access	\overline{RE}	\overline{WE}	\overline{UDS}	\overline{LDS}	DATA[15:8]		DATA[7:0]	
					I/O	data(addr)	I/O	data(addr)
Word write of data on DATA[15:0] at an even and even+1 address	1	0	0	0	Out	data(even)	Out	data(odd)
Byte write of data on DATA[7:0] at an odd address	1	0	1	0	In	x	Out	data(odd)
Byte write of data on DATA[15:8] at an even address	1	0	0	1	Out	data(even)	In	x
Word read of data on DATA[15:0] at an even and even+1 address	0	1	0	0	In	data(even)	In	data(odd)
Byte read of data on DATA[7:0] at an odd address	0	1	1	0	In	x	In	data(odd)
Byte read of data on DATA[15:8] at an even address	0	1	0	1	In	data(even)	In	x
Indicates No Access	1	1	1	1	In	x	In	x
Unimplemented	1	1	1	0	In	x	In	x
	1	1	0	1	In	x	In	x

5.4.5.2 Emulation Modes and Special Test Mode

In emulation modes and special test mode, the external signals \overline{LSTRB} , \overline{RW} , and $ADDR0$ indicate the access type (read/write), data size and alignment of an external bus access. Misaligned accesses to the internal RAM and misaligned XGATE PRR accesses in emulation modes are the only type of access that are able to produce $\overline{LSTRB} = ADDR0 = 1$. This is summarized in Table 5-20.

Table 5-20. Access in Emulation Modes and Special Test Mode

Access	\overline{RW}	\overline{LSTRB}	$ADDR0$	DATA[15:8]		DATA[7:0]	
				I/O	data(addr)	I/O	data(addr)
Word write of data on DATA[15:0] at an even and even+1 address	0	0	0	Out	data(even)	Out	data(odd)
Byte write of data on DATA[7:0] at an odd address	0	0	1	In	x	Out	data(odd)
Byte write of data on DATA[15:8] at an even address	0	1	0	Out	data(odd)	In	x
Word write at an odd and odd+1 internal RAM address (misaligned — only in emulation modes)	0	1	1	Out	data(odd+1)	Out	data(odd)
Word read of data on DATA[15:0] at an even and even+1 address	1	0	0	In	data(even)	In	data(even+1)
Byte read of data on DATA[7:0] at an odd address	1	0	1	In	x	In	data(odd)
Byte read of data on DATA[15:8] at an even address	1	1	0	In	data(even)	In	x
Word read at an odd and odd+1 internal RAM address (misaligned - only in emulation modes)	1	1	1	In	data(odd+1)	In	data(odd)

5.4.6 Low-Power Options

The XEBI does not support any user-controlled options for reducing power consumption.

5.4.6.1 Run Mode

The XEBI does not support any options for reducing power in run mode.

Power consumption is reduced in single-chip modes due to the absence of the external bus interface. Operation in expanded modes results in a higher power consumption, however any unnecessary toggling of external bus signals is reduced to the lowest indispensable activity by holding the previous states between external accesses.

5.4.6.2 Wait Mode

The XEBI does not support any options for reducing power in wait mode.

5.4.6.3 Stop Mode

The XEBI will cease to function in stop mode.

5.5 Initialization/Application Information

This section describes the external bus interface usage and timing. Typical customer operating modes are normal expanded mode and emulation modes, specifically to be used in emulator applications. Taking the availability of the external wait feature into account the use cases are divided into four scenarios:

- Normal expanded mode
 - External wait feature disabled
 - External wait feature enabled
- Emulation modes
 - Emulation single-chip mode (without wait states)
 - Emulation expanded mode (with optional access stretching)

Normal single-chip mode and special single-chip mode do not have an external bus. Special test mode is used for factory test only. Therefore, these modes are omitted here.

All timing diagrams referred to throughout this section are available in the Electrical Characteristics appendix of the SoC section.

5.5.1 Normal Expanded Mode

This mode allows interfacing to external memories or peripherals which are available in the commercial market. In these applications the normal bus operation requires a minimum of 1 cycle stretch for each external access.

5.5.1.1 Example 1a: External Wait Feature Disabled

The first example of bus timing of an external read and write access with the external wait feature disabled is shown in

- Figure ‘Example 1a: Normal Expanded Mode — Read Followed by Write’

The associated supply voltage dependent timing are numbers given in

- Table ‘Example 1a: Normal Expanded Mode Timing $V_{DD5} = 5.0\text{ V}$ (EWAIT disabled)’
- Table ‘Example 1a: Normal Expanded Mode Timing $V_{DD5} = 3.0\text{ V}$ (EWAIT disabled)’

Systems designed this way rely on the internal programmable access stretching. These systems have predictable external memory access times. The additional stretch time can be programmed up to 8 cycles to provide longer access times.

5.5.1.2 Example 1b: External Wait Feature Enabled

The external wait operation is shown in this example. It can be used to exceed the amount of stretch cycles over the programmed number in EXSTR[2:0]. The feature must be enabled by configuring at least one \overline{CSx} line for EWAIT.

If the \overline{EWAIT} signal is not asserted, the number of stretch cycles is forced to a minimum of 2 cycles. If \overline{EWAIT} is asserted within the predefined time window during the access it will be strobed active and another stretch cycle is added. If strobed inactive, the next cycle will be the last cycle before the access is finished. \overline{EWAIT} can be held asserted as long as desired to stretch the access.

An access with 1 cycle stretch by \overline{EWAIT} assertion is shown in

- Figure ‘Example 1b: Normal Expanded Mode — Stretched Read Access’
- Figure ‘Example 1b: Normal Expanded Mode — Stretched Write Access’

The associated timing numbers for both operations are given in

- Table ‘Example 1b: Normal Expanded Mode Timing $V_{DD5} = 5.0\text{ V}$ (EWAIT enabled)’
- Table ‘Example 1b: Normal Expanded Mode Timing $V_{DD5} = 3.0\text{ V}$ (EWAIT enabled)’

It is recommended to use the free-running clock (ECLK) at the fastest rate (bus clock rate) to synchronize the \overline{EWAIT} input signal.

5.5.2 Emulation Modes

In emulation mode applications, the development systems use a custom PRU device to rebuild the single-chip or expanded bus functions which are lost due to the use of the external bus with an emulator.

Accesses to a set of registers controlling the related ports in normal modes (refer to SoC section) are directed to the external bus in emulation modes which are substituted by PRR as part of the PRU. Accesses to these registers take a constant time of 2 cycles.

Depending on the setting of ROMON and EROMON (refer to S12X_MMC section), the program code can be executed from internal memory or an optional external emulation memory (EMULMEM). No wait state operation (stretching) of the external bus access is done in emulation modes when accessing internal memory or emulation memory addresses.

In both modes observation of the internal operation is supported through the external bus (internal visibility).

5.5.2.1 Example 2a: Emulation Single-Chip Mode

This mode is used for emulation systems in which the target application is operating in normal single-chip mode.

Figure 5-5 shows the PRU connection with the available external bus signals in an emulator application.

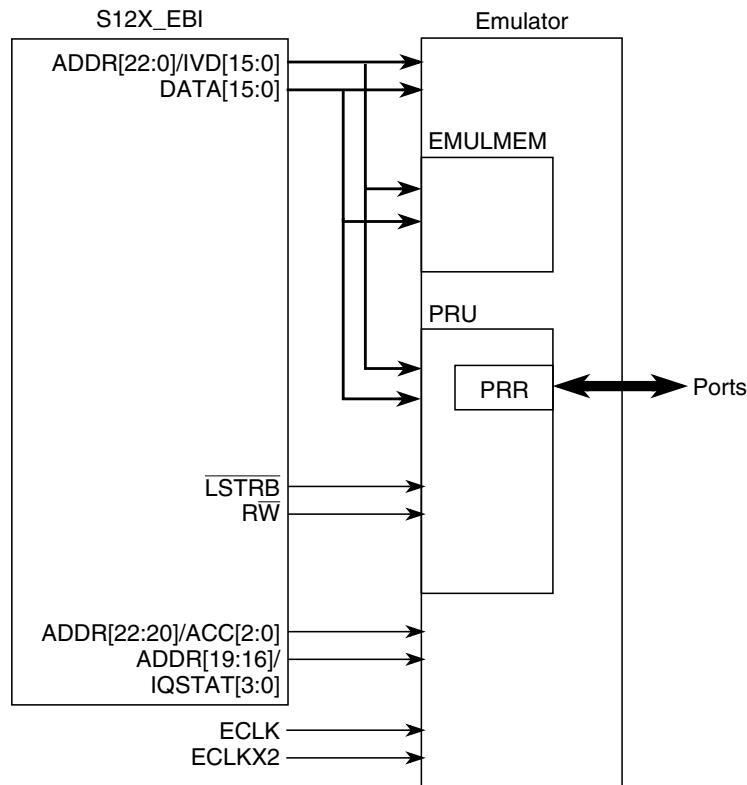


Figure 5-5. Application in Emulation Single-Chip Mode

The timing diagram for this operation is shown in:

- Figure ‘Example 2a: Emulation Single-Chip Mode — Read Followed by Write’

The associated timing numbers are given in:

- Table ‘Example 2a: Emulation Single-Chip Mode Timing (EWAIT disabled)’

Timing considerations:

- Signals muxed with address lines ADDR_x, i.e., IVD_x, IQSTAT_x and ACC_x, have the same timing.
- $\overline{\text{LSTRB}}$ has the same timing as $\overline{\text{RW}}$.
- ECLKX2 rising edges have the same timing as ECLK edges.
- The timing for accesses to PRU registers, which take 2 cycles to complete, is the same as the timing for an external non-PRR access with 1 cycle of stretch as shown in example 2b.

5.5.2.2 Example 2b: Emulation Expanded Mode

This mode is used for emulation systems in which the target application is operating in normal expanded mode.

If the external bus is used with a PRU, the external device rebuilds the data select and data direction signals \overline{UDS} , \overline{LDS} , \overline{RE} , and \overline{WE} from the $\overline{ADDR0}$, \overline{LSTRB} , and \overline{RW} signals.

Figure 5-6 shows the PRU connection with the available external bus signals in an emulator application.

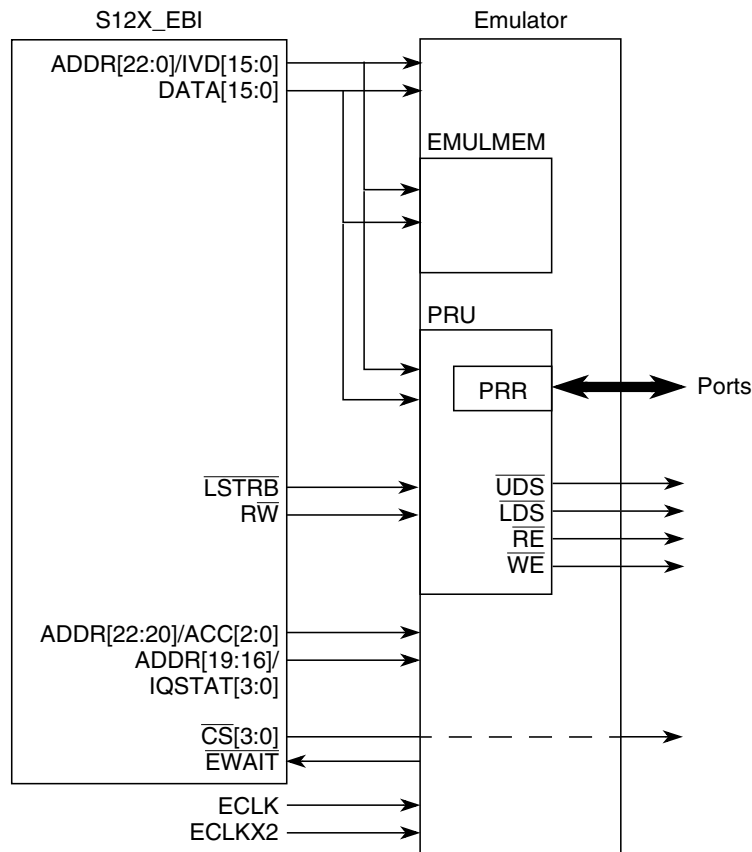


Figure 5-6. Application in Emulation Expanded Mode

The timings of accesses with 1 stretch cycle are shown in

- Figure ‘Example 2b: Emulation Expanded Mode — Read with 1 Stretch Cycle’
- Figure ‘Example 2b: Emulation Expanded Mode — Write with 1 Stretch Cycle’

The associated timing numbers are given in

- Table ‘Example 2b: Emulation Expanded Mode Timing $V_{DD5} = 5.0\text{ V}$ (EWAIT disabled)’ (this also includes examples for alternative settings of 2 and 3 additional stretch cycles)

Timing considerations:

- If no stretch cycle is added, the timing is the same as in Emulation Single-Chip Mode.