S12X_EBI
Block Guide
V02.02

Original Release Date: 13 Februar 2004
Revised: 05 Oct 2004

Motorola, Inc.

Revision History

Release Number	Date	Author	Summary of Changes
V01.28	13-Feb-04		Initial Customer Release
V02.00	14-Jun-04		- Major document rework. - Added EBISIZ register for scalable external address bus width (ASIZ[4:0]) and 8-bit data bus option (HDBE) - Added EXSTR[2:0] bits to MODE register - Added stretch functionality in Special Test Mode - Made ECLKX2 available in all modes. - Moved addresses $000A (EIFCTL->MEMCTL0) and $000B (MODE) to S12X_MMC Block Guide - Added EBICTL register at $000E - Moved EIFCTL bits NECLK, EDIVx, EWAITE to PIM - Moved EIFCTL register bit EWAITE to EBICTL - Moved MODE register bits ITHRS, IVIS to EBICTL - Moved ‘Modes of Operation’ description to S12X_MMC - Moved features ‘chip selects’ and ‘Chip operating mode control’ to S12X_MMC - Moved feature ‘Free-running clock outputs’ to PIM
V02.01	18-Jun-04		- Removed "Controlled Copy" banners.

Release Number	Date	Author	Summary of Changes
V02.02	05-Oct-04		- Changed external accesses in Special Test Mode to 1 cycle only - Removed internal visibility feature in Special Test Mode (along with IVIS register bit) - Removed external wait feature in Special Test Mode - Changed programmable stretch cycle range from 1 to 8 cycles - Major changes in control register definitions - Enhanced internal visibility access diagrams - Added full decoding of all 4 access control signal combinations in Normal Expanded Mode - Removed address and databus size config feature in Special Test Mode - Removed all references to security - Made reduced threshold depend on function activated on pin (also added tag pin references) - Added ‘Unimplemented Area Access’ in Summary of Funtions table and changed to 1 cycle in ES mode. - Added IVD and ‘No Access’ descriptions.

Table of Contents

List of Figures

List of Tables

Preface

Terminology

Table 0-1 Acronyms and Abbreviations

Logic level “1”	Voltage that corresponds to Boolean true state
Logic level “0”	Voltage that corresponds to Boolean false state
$	Represents hexadecimal number
x	Represents logic level ’don’t care’
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
NVM	Non-volatile Memory; Flash EEPROM or ROM
byte	8-bit data
word	16-bit data

Section 1 Introduction

This document describes the functionality of the S12X_EBI block controlling the External Bus Interface.

Figure 1-1 is a block diagram of the S12X_EBI with all related I/O signals.

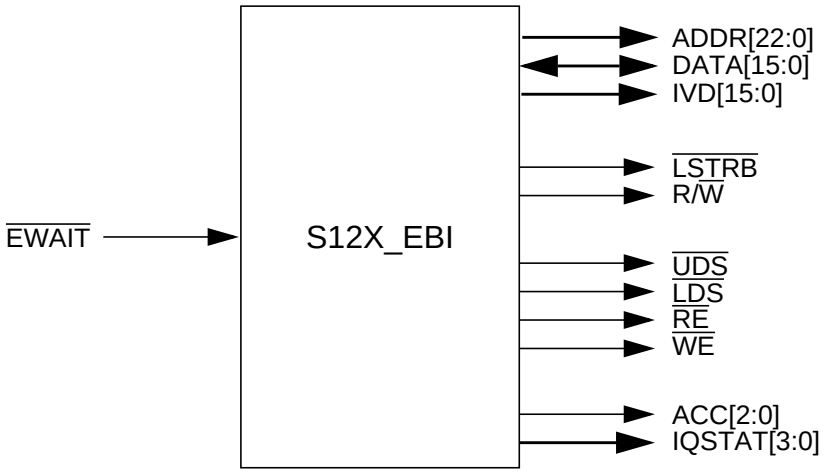

Figure 1-1 S12X_EBI Block Diagram

1.1 Overview

The S12X_EBI 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.

1.2 Features

The S12X_EBI includes the following features:

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

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 Block Guide for a detailed description of the MCU operating modes.

Section 2 External Signal Description

2.1 Overview

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

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

Table 2-1 outlines the pin names and gives a brief description of their function. Refer to the SoC Guide and PIM Block Guide for reset states of these pins and associated pull-ups or pull-downs.

Table 2-1 External system signals associated with S12X_EBI

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	-	-	✓	-	-	-
ADDR[22:20]	O	T	-	External address	-	-	✓	✓	✓	✓
ACC[2:0]	O		-	Access source	-	-	-	✓	✓	✓
ADDR[19:16]	O	T	-	External address	-	-	✓	✓	✓	✓
IQSTAT[3:0]	O		-	Instruction Queue Status	-	-	-	✓	✓	✓
ADDR[15:1]	O	T	-	External address	-	-	✓	✓	✓	✓
IVD[15:1]	O		-	Internal visibility read data (IVIS = 1)	-	-	-	✓	✓	✓
ADDR0	O	T	F	External address	-	-	-	✓	✓	✓
IVD0	O			Internal visibility read data (IVIS = 1)	-	-	-	✓	✓	✓
$\overline{UDS}$	O	-		Upper Data Select, indicates external access to the high byte DATA[15:8]	-	-	✓	-	-	-
$\overline{LSTRB}$	O	-	F	Low Strobe, indicates valid data on DATA[7:0]	-	-	-	✓	✓	✓
$\overline{LDS}$	O	-		Lower Data Select, indicates external access to the low byte DATA[7:0]	-	-	✓	-	-	-
R/$\overline{W}$	O	-	F	Read/Write, indicates the direction of internal data transfers	-	-	-	✓	✓	✓
$\overline{WE}$	O	-		Write Enable, indicates external write access	-	-	✓	-	-	-
DATA[15:8]	I/O	-	-	Bidirectional data (even address)	-	-	✓	✓	✓	✓
DATA[7:0]	I/O	-	-	Bidirectional data (odd address)	-	-	✓	✓	✓	✓
$\overline{EWAIT}$	I	-	-	External control for external bus access stretches (adding wait states)	-	-	✓	-	✓	-

NOTES:
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 alternatingly 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.

Section 3 Memory Map/Register Definition

The registers associated with the S12X_EBI block are shown in **Figure 3-1**.

Address	Name		Bit 7	6	5	4	3	2	1	Bit 0
$000E	EBICTL0	Read	ITHRS	0	HDBE	ASIZ4	ASIZ3	ASIZ2	ASIZ1	ASIZ0
		Write								
$000F	EBICTL1	Read	EWAITE	0	0	0	0	EXSTR2	EXSTR1	EXSTR0
		Write								

= Unimplemented

Figure 3-1 S12X_EBI Register Map Summary

3.1 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 although 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.*

3.1.1 External Bus Interface Control Register 0 (EBICTL0)

Address: $000E (PRR)

	BIT 7	6	5	4	3	2	1	BIT 0
Read:	ITHRS	0	HDBE	ASIZ4	ASIZ3	ASIZ2	ASIZ1	ASIZ0
Write:								
Reset:	0	0	1	1	1	1	1	1

= Unimplemented

Figure 3-2 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 are 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.

ITHRS — 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 3.3V, 5V tolerant I/O.

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 3-1**.

1 = Reduced input threshold level enabled on pins in use with the external bus interface

0 = Input threshold is at standard level on all pins

Table 3-1 Input Threshold Levels on External Signals

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

HDBE - 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.

1 = DATA[15:8], $\overline{UDS}$, and $\overline{LDS}$ enabled

0 = DATA[15:8], $\overline{UDS}$, and $\overline{LDS}$ disabled

ASIZ[4:0] - 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 3-2**). 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.

Table 3-2 External Address Bus Size

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

3.1.2 External Bus Interface Control Register 1 (EBICTL1)

Address:	$000F (PRR)							
	BIT 7	6	5	4	3	2	1	BIT 0
Read:	EWAITE	0	0	0	0	EXSTR2	EXSTR1	EXSTR0
Write:								
Reset:	0	0	0	0	0	1	1	1

= Unimplemented

Figure 3-3 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 are read from this register.

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

This register is used to configure the external access stretch (wait) function.

EWAITE — External Wait Enable

This bit enables the external access stretch function using the external $\overline{\text{EWAIT}}$ input pin. Enabling this feature may have effect on the minimum number of additional stretch cycles (refer to **Table 3-3**).

External wait feature is only active if enabled in Normal Expanded Mode and Emulation Expanded Mode; function not available in all other operating modes.

1 = External Wait is enabled
0 = External Wait is disabled

EXSTR2, EXSTR1, EXSTR0 - External Access Stretch 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 3-3**. The minimum number of stretch cycles depends on the EWAITE setting.

Stretch cycles are added as programmed in Normal Expanded Mode and Emulation Expanded Mode; function not available in all other operating modes.

Table 3-3 External Access Stretch Bit Definition

EXSTR[2:0]	Number of Stretch Cycles	
	EWAITE = 0	EWAITE = 1
000	1 cycle	>= 2 cycles
001	2 cycles	>= 2 cycles
010	3 cycles	>= 3 cycles
011	4 cycles	>= 4 cycles
100	5 cycles	>= 5 cycles
101	6 cycles	>= 6 cycles
110	7 cycles	>= 7 cycles
111	8 cycles	>= 8 cycles

Section 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.

4.1 Operating Modes and External Bus Properties

A summary of the external bus interface functions for each operating mode is shown in **Table 4-1**.

Table 4-1 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 & 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 3-1**	DATA[15:0] $\overline{EWAIT}$	DATA[15:0] $\overline{EWAIT}$	refer to **Table 3-1**

NOTES:
1. Incl. S12X_EBI registers
2. Refer to S12X_MMC Block Guide.
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 Block Guide).

4.1.1 Internal Visibility

Internal visibility allows the observation of the internal MCU 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 and BDM accesses are made visible on the external bus interface, except those to BDM firmware and BDM registers.

Internal reads are made visible on ADDRx/IVDx (address and read data multiplexed, see **Table 4-3** to **Table 4-5**), internal writes on ADDRx and DATAx (see **Table 4-6** to **Table 4-8**). R/$\overline{W}$ and $\overline{LSTRB}$ show the type of access. External read data are also visible on IVDx.

4.1.1.1 Access Source and Instruction Queue Status Signals

The access source (bus master) can be determined from the external bus control signals ACC[2:0] as shown in **Table 4-2**.

Table 4-2 Determining Access Source from Control Signals

ACC[2:0]	Access description
000	Repetition of previous access cycle
001	CPU access
010	BDM access
011	XGATE PRR access [1]
100	No access [2]
101, 110, 111	Reserved

NOTES:
1. Note: Invalid IVD brought out in read cycles
2. Denotes also accesses to BDM firmware and BDM registers (IQSTATx are ‘XXXX’ and R/$\overline{W}$=1 in these cases)

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 Block Guide.

4.1.1.2 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. ADDRx values will only be shown in ECLK high phases while ACCx, IQSTATx and IVDx values will only be presented in ECLK low phases.

Based on this multiplex timing, ACCx are only shown in the current (first) access cycle. IQSTATx and (for read accesses) IVDx follow in the next cycle. If the access takes more than one bus cycle, ACCx display NULL ($000) in the second and all following cycles of the access. IQSTATx display NULL ($0000) 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’

1. Read Access timing

Table 4-3 Read 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]	...		**acc 0**		acc 1		acc 2	...
ADDR[19:16] / IQSTAT[3:0]	...	**addr 0**	iqstat -1	addr 1	**iqstat 0**	addr 2	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	...
R/$\overline{W}$	...	1	1	1	1	1	1	...

Table 4-4 Read 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]	...		**acc 0**		**000**		acc 1	...
ADDR[19:16] / IQSTAT[3:0]	...	**addr 0**	iqstat-1	**addr 0**	**iqstat 0**	addr 1	**0000**	...
ADDR[15:0] / IVD[15:0]	...		?		**x**		**ivd 0**	...
DATA[15:0] (internal read)	...	?	**z**	**z**	**z**	**z**	z	...
DATA[15:0] (external read)	...	?	**z**	**z**	**z**	**data 0**	z	...
R/$\overline{W}$	...	1	1	1	1	1	1	...

Table 4-5 Read 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]	...		**acc 0**		**000**		**000**	...		acc 1	...
ADDR[19:16] / IQSTAT[3:0]	...	**addr 0**	iqstat-1	**addr 0**	**iqstat 0**	**addr 0**	**0000**	...	addr 1	**0000**	...
ADDR[15:0] / IVD[15:0]	...		?		**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	...
R/$\overline{W}$	...	1	1	1	1	1	1	...	1	1	...

2. Write Access timing

Table 4-6 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]	...		**acc 0**		acc 1		acc 2	...
ADDR[19:16] / IQSTAT[3:0]	...	**addr 0**	iqstat -1	addr 1	**iqstat 0**	addr 2	iqstat 1	...
ADDR[15:0] / IVD[15:0]	...		?		**x**		x	...
DATA[15:0] (write)	...	?	**data 0**		data 1		data 2	...
R/$\overline{W}$	...	0	0	1	1	1	1	...

Table 4-7 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]	...		**acc 0**		**000**		acc 1	...
ADDR[19:16] / IQSTAT[3:0]	...	**addr 0**	iqstat-1	**addr 0**	**iqstat 0**	addr 1	**0000**	...
ADDR[15:0] / IVD[15:0]	...		?		**x**		**x**	...
DATA[15:0] (write)	...	?	**data 0**				x	...
R/$\overline{W}$	...	0	0	0	0	1	1	...

Table 4-8 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]	...		**acc 0**		**000**		**000**	**...**		acc 1	...
ADDR[19:16] / IQSTAT[3:0]	...	**addr 0**	iqstat-1	**addr 0**	**iqstat 0**	**addr 0**	**0000**	**...**	addr 1	**0000**	...
ADDR[15:0] / IVD[15:0]	...		?		**x**		**x**	**...**		**x**	...
DATA[15:0] (write)	...	?	**data 0**							x	...
R/$\overline{W}$	...	0	0	0	0	0	0	...	1	1	...

3. Read-Write-Read Access timing

Table 4-9 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]	...		**acc 0**		acc 1		acc 2	...
ADDR[19:16] / IQSTAT[3:0]	...	**addr 0**	iqstat -1	addr 1	**iqstat 0**	addr 2	iqstat 1	...
ADDR[15:0] / IVD[15:0]	...		?		**ivd 0**		x	...
DATA[15:0] (internal read)	...	?	**z**	**z**	(write) data 1		z	...
DATA[15:0] (external read)	...	?	**z**	**data 0**	(write) data 1		z	...
R/$\overline{W}$	...	1	1	0	0	1	1	...

4.1.1.3 Internal Visibility Data

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 4-10**.

Table 4-10 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)

4.1.2 Accesses to Port Replacement Registers

All read and write accesses to PRR addresses take 2 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 S12X_EBI control registers also belong to this category.

4.1.3 Stretched External Bus Accesses

In order to allow fast internal bus cycles to coexist in a system with slower external resources, the S12X_EBI supports stretched external bus accesses (wait states).

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. EXSTR[2:0] bits in the EBICTL1 register configuring fixed amount of stretch cycles
2. Activation of the external wait feature by EWAITE in EBICTL1 register

3. Assertion of the external $\overline{\text{EWAIT}}$ signal when EWAITE = 1

The EXSTR[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 $\overline{\text{EWAIT}}$ input.

$\overline{\text{EWAIT}}$ 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 $\overline{\text{RE}}$ 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 Block Guide 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.

4.1.4 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.

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 4-11).

Table 4-11 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

2. Emulation modes and Special Test Mode

In emulation modes and Special Test Mode the external signals $\overline{LSTRB}$, R/$\overline{W}$, 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 4-12.

Table 4-12 Access in emulation modes and Special Test Mode

Access	R/$\overline{W}$	$\overline{LSTRB}$	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)

4.2 Low-Power Options

The S12X_EBI does not support any user-controlled options for reducing power consumption.

4.2.1 Run Mode

The S12X_EBI 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.

4.2.2 Wait Mode

The S12X_EBI does not support any options for reducing power in wait mode.

4.2.3 Stop Mode

The S12X_EBI will cease to function in stop mode.

Section 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 and 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 Guide.

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.

5.1.1 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}=5.0V (EWAITE = 0)’
- Table ‘Example 1a: Normal Expanded Mode Timing V_{DD5}=3.0V (EWAITE = 0)’

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.

5.1.2 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 writing EWAITE = 1.

If the $\overline{\text{EWAIT}}$ signal is not asserted, the number of stretch cycles is forced to a minimum of 2 cycles. If $\overline{\text{EWAIT}}$ 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. $\overline{\text{EWAIT}}$ can be held asserted as long as desired to stretch the access.

An access with 1 cycle stretch by $\overline{\text{EWAIT}}$ 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}=5.0V (EWAITE = 1)’
- Table ‘Example 1b: Normal Expanded Mode Timing VDD5=3.0V (EWAITE = 1)’

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

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 Guide) 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 Block Guide) 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).

5.2.1 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 5-1 shows the PRU connection with the available external bus signals in an emulator application.

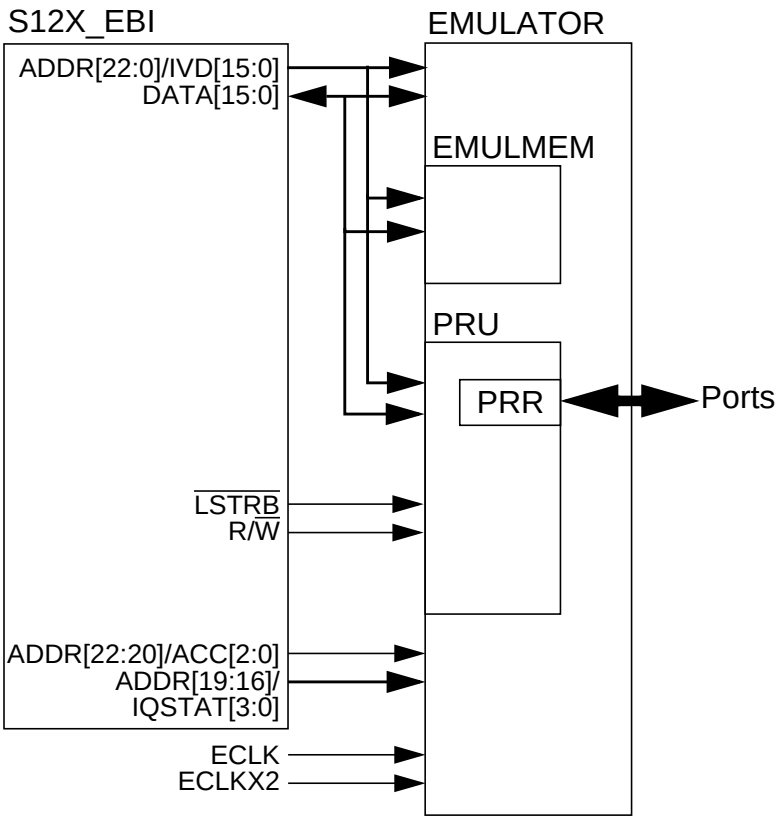

Figure 5-1 Application in Emulation Single-Chip Mode

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 (EWAITE = 0)’

Timing considerations:

- Signals muxed with address lines ADDRx, i.e. IVDx, IQSTATx and ACCx, have the same timing.
- $\overline{\text{LSTRB}}$ has the same timing as R/$\overline{\text{W}}$.
- 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.

5.2.2 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 ADDR0, $\overline{LSTRB}$, and R/$\overline{W}$ signals.

Figure 5-2 shows the PRU connection with the available external bus signals in an emulator application.

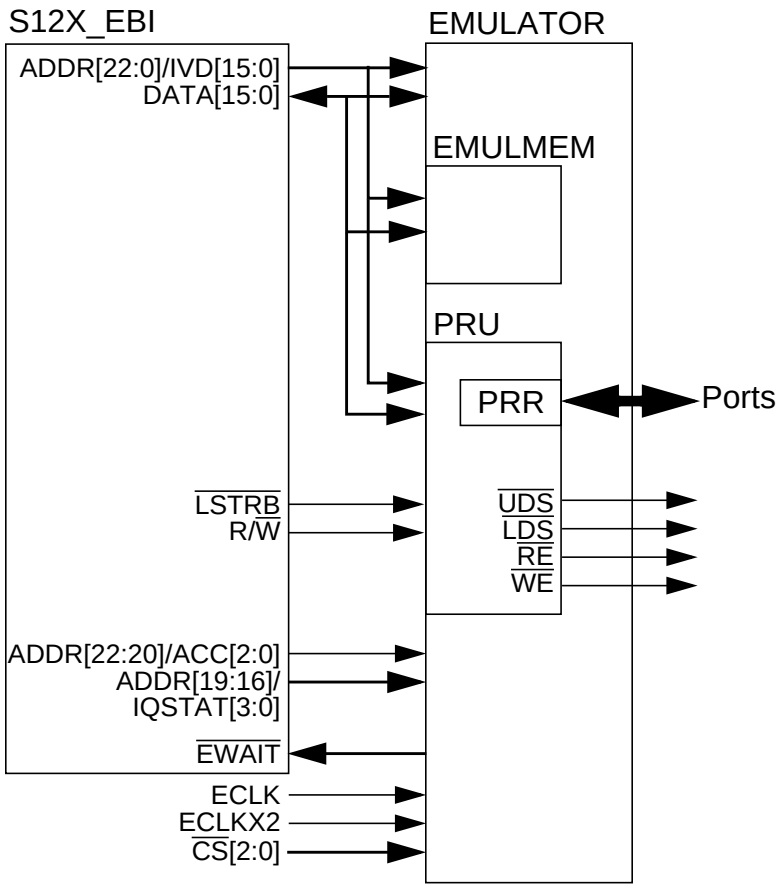

Figure 5-2 Application in Emulation Expanded Mode

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}=5.0V (EWAITE = 0)’ (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.

Index

Block Guide End Sheet

FINAL PAGE OF
36
PAGES