

# **CoreMemCtrl v2.0**

*Handbook*

---

## **Actel Corporation, Mountain View, CA 94043**

© 2009 Actel Corporation. All rights reserved.

Printed in the United States of America

Part Number: 50200115-0

Release: March 2009

No part of this document may be copied or reproduced in any form or by any means without prior written consent of Actel.

Actel makes no warranties with respect to this documentation and disclaims any implied warranties of merchantability or fitness for a particular purpose. Information in this document is subject to change without notice. Actel assumes no responsibility for any errors that may appear in this document.

This document contains confidential proprietary information that is not to be disclosed to any unauthorized person without prior written consent of Actel Corporation.

### **Trademarks**

Actel and the Actel logo are registered trademarks of Actel Corporation.

Adobe and Acrobat Reader are registered trademarks of Adobe Systems, Inc.

All other products or brand names mentioned are trademarks or registered trademarks of their respective holders.

# Table of Contents

	Introduction . . . . .	5
	Core Overview . . . . .	5
	Key Features . . . . .	6
	Supported Device Families . . . . .	6
	Core Version . . . . .	6
	Supported Interfaces . . . . .	6
	Device Utilization and Performance . . . . .	7
<b>1</b>	<b>Tool Flows . . . . .</b>	<b>9</b>
	SmartDesign . . . . .	9
	Example System . . . . .	11
	Simulation . . . . .	11
	Synthesis in Libero IDE . . . . .	12
	Place-and-Route in Libero IDE . . . . .	12
<b>2</b>	<b>Connecting to External Memories . . . . .</b>	<b>13</b>
<b>3</b>	<b>Registers . . . . .</b>	<b>21</b>
<b>4</b>	<b>Memory Map . . . . .</b>	<b>23</b>
<b>5</b>	<b>Interface Descriptions . . . . .</b>	<b>25</b>
	Parameters/Generics . . . . .	25
	Ports . . . . .	27
<b>6</b>	<b>Waveforms . . . . .</b>	<b>29</b>
<b>7</b>	<b>Ordering Information . . . . .</b>	<b>31</b>
	Ordering Codes . . . . .	31
<b>A</b>	<b>Product Support . . . . .</b>	<b>33</b>
	Customer Service . . . . .	33
	Actel Customer Technical Support Center . . . . .	33
	Actel Technical Support . . . . .	33
	Website . . . . .	33
	Contacting the Customer Technical Support Center . . . . .	33
<b>B</b>	<b>Index . . . . .</b>	<b>35</b>



# Introduction

## Core Overview

CoreMemCtrl is an advanced high-performance bus (AHB) slave component that interfaces to external flash and SRAM memory devices. Both synchronous and asynchronous SRAM are supported. The core is suitable for connection to either the CoreAHB or CoreAHBLite bus.

CoreMemCtrl uses 256 MB of address space on the AHB or AHB-Lite bus to which it is connected. This address space is evenly split between flash and SRAM memory, so that up to 128 MB of each type of memory can be accessed. The core has a REMAP input which can be used to swap the positions of flash and SRAM in the memory map.

Various configuration options exist on the core to allow a variety of different memory devices to be supported. [Figure 1](#) shows a block diagram of CoreMemCtrl.

**Note:** CoreMemCtrl v2.0 is not pin compatible with previous versions of the core. CoreMemCtrl v2.0 has only a single AHB interface, whereas earlier versions had two AHB interfaces. A new REMAP input is also present on CoreMemCtrl v2.0.

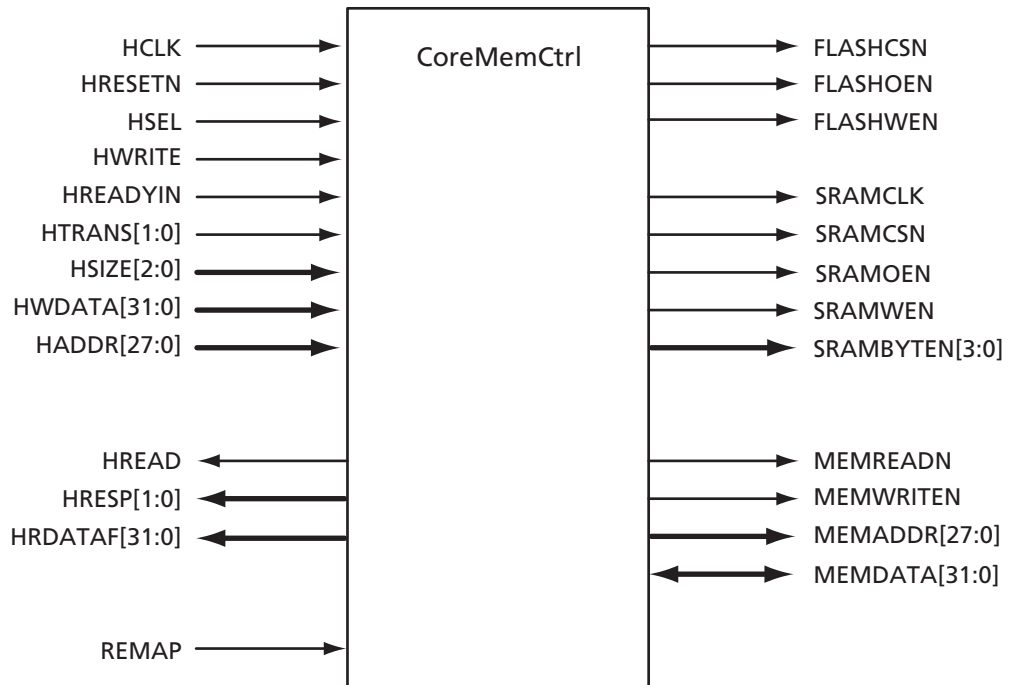


Figure 1 · CoreMemCtrl Block Diagram

## Key Features

CoreMemCtrl has the following features:

- Provides an AHB interface to external memory devices
- Configurable external memory interface
- Interfaces to external flash and either synchronous or asynchronous external SRAM
- Supports 32-bit word, 16-bit halfword, and 8-bit byte accesses to SRAM
- Supports 32-bit word accesses to flash; halfword and byte accesses to flash are not supported.
- The locations of flash and SRAM in the address space can be swapped by asserting the REMAP input.

## Supported Device Families

- IGLOO®
- IGLOOe
- IGLOO PLUS
- ProASIC®3
- ProASIC3E
- ProASIC3L
- Fusion
- Accelerator®
- RTAX-S
- RTSX-S

## Core Version

This handbook applies to CoreMemCtrl v2.0.

## Supported Interfaces

CoreMemCtrl has an advanced microcontroller bus architecture (AMBA) AHB slave interface through which an AHB master can initiate read and write accesses to external memory.

**Note:** CoreMemCtrl v2.0 is not pin compatible with previous versions of the core. CoreMemCtrl v2.0 has only a single AHB interface, whereas earlier versions had two AHB interfaces. A new REMAP input is also present on CoreMemCtrl v2.0

The core also has a group of ports for interfacing to external flash and SRAM memory devices. The design of the core assumes that the address bus and bidirectional data bus are common to both flash and SRAM devices. Aside from these busses, chip select, output enable, and write enable signals are provided for connection to the memory devices.

It is possible, via the configuration window for the core, to choose to use common read and write enable signals for flash and SRAM. When this option is selected, the MEMREADN and MEMWRITEN signals can be connected to both flash and SRAM. This option can be useful when the number of pins available for interfacing to external memories is limited.

## Device Utilization and Performance

Table 1 gives resource usage and performance figures for various configurations of CoreMemCtrl. Table 1 does not cover every possible configuration but instead lists a range of configurations which should give a good indication of the expected resource usage and performance of the core.

Table 1 · CoreMemCtrl Device Utilization and Performance

Parameters										Tiles	Frequency (MHz)
SYNC_SRAM	FLOW_THROUGH	FLASH_16BIT	NUM_WS_FLASH_READ	NUM_WS_FLASH_WRITE	NUM_WS_SRAM_READ	NUM_WS_SRAM_WRITE	SHARED_RW	FLASH_ADDR_SEL	SRAM_ADDR_SEL		
1	0	0	1	1	1	1	0	2	2	391	137
1	0	0	3	3	1	1	0	2	2	393	137
1	0	0	1	1	1	1	1	2	2	384	136
1	0	0	1	1	1	1	0	1	1	400	137
1	0	0	1	1	1	1	0	0	0	391	137
1	0	0	1	1	1	1	0	1	2	419	139
1	0	0	1	1	1	1	0	0	2	391	137
1	1	0	1	1	1	1	0	2	2	448	71
1	0	1	1	1	1	1	0	2	2	476	124
1	1	1	1	1	1	1	0	2	2	553	62
0	0	0	1	1	1	1	0	2	2	352	154
0	0	0	2	2	2	2	0	2	2	364	163
0	0	0	3	3	3	3	0	2	2	349	161
0	0	1	1	1	1	1	0	2	2	443	154
0	0	0	1	1	1	1	0	1	2	376	163

*Note:* Data in this table were obtained for an A3P1000 device, speed grade -2.



## Tool Flows

### SmartDesign

CoreMemCtrl is available for download to the SmartDesign IP Catalog via the Libero® Integrated Design Environment (IDE) web repository. For information on using SmartDesign to instantiate, configure, connect, and generate cores, refer to the Libero IDE online help.

In a typical system, the AHB slave interface of CoreMemCtrl is connected to one of the slave slots on CoreAHB or CoreAHBLite. CoreMemCtrl is often connected to slot 0 on the bus, so that the external memory appears at address 0x00000000 in the address map of a processor connected as a master in the system. The ports for connection to external memory devices should be routed to the top level of the design and assigned to appropriate pins for connection to the external memories.

CoreMemCtrl is suitable for interfacing to a variety of flash and SRAM memories, but the core must be configured to suit the particular devices being used. Figure 1-1 shows the CoreMemCtrl configuration window, along with cross references to the corresponding top-level parameters. The parameters/generics of the core are fully described in “Parameters/Generics” on page 25.

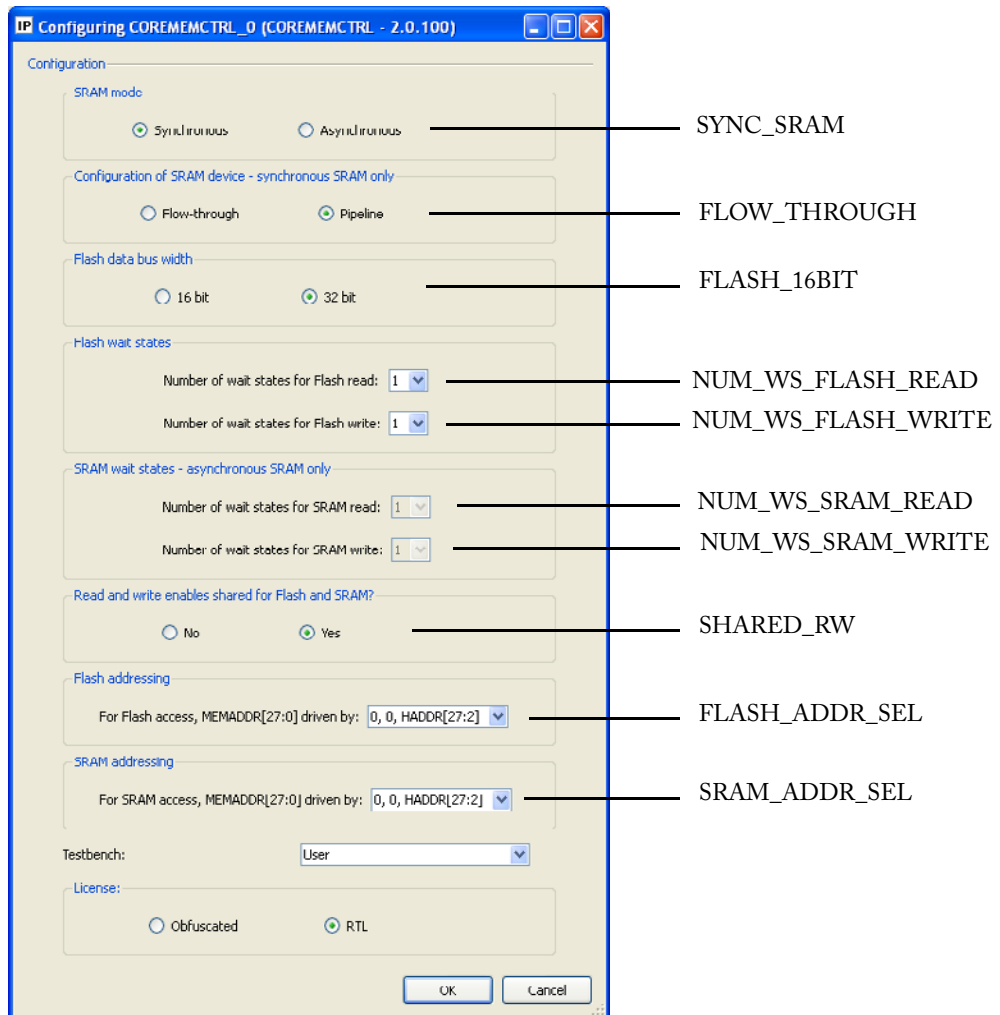


Figure 1-1 · CoreMemCtrl Configuration Window

The configuration options for CoreMemCtrl are described in the following paragraphs. The CoreMemCtrl configuration window is used to adjust the values of the underlying parameters/generics in the RTL code for the core. Each configuration option presented in the configuration window corresponds directly to an actual parameter/generic in the RTL code for CoreMemCtrl.

## SRAM Type

You can use the SYNC\_SRAM parameter to set the interfaced SRAM to either synchronous or asynchronous. If this parameter is set to synchronous SRAM, the FLOW\_THROUGH parameter must be set to match the mode of operation of the synchronous SRAM device. Some devices only operate in a flow-through manner (where the data appears in the clock cycle after the address), whereas others can be configured to operate in pipeline mode (with the data appearing two cycles after the address).

## Flash Data Bus Width

The core has a FLASH\_16BIT parameter, which allows for the use of only a single 16-bit flash device, as opposed to using two flash devices to suit a 32-bit data bus. When FLASH\_16BIT is set, the lower 16-bits of the memory data bus (MEMDATA[15:0]) must be connected to the 16-bit flash device. Note that, regardless of the setting of FLASH\_16BIT, only 32-bit word accesses to flash are supported from the AHB (or AHB-Lite) master. Halfword and byte accesses to flash are not supported. When FLASH\_16BIT is set, each 32-bit word access is transparently broken into two separate 16-bit accesses to the external flash device.

The setting for the flash data bus width is ignored when the word address is used for flash accesses. It is assumed that there is a 32-bit data bus to external flash when the word address is used to address the flash. See [“Address Bus Handling”](#) for more information.

## Number of Wait States

It is possible to adjust the number of wait states inserted during read and write accesses to flash and to asynchronous SRAM. Up to three wait states can be configured for each type of access. Increasing the number of wait states enables more clock cycles for signals to and from the memory device to stabilize. In systems where a relatively slow flash or asynchronous SRAM device is used, increasing the number of wait states enables the system clock speed to be increased while still respecting the timing requirements of the memory devices. However, there is a performance cost to increasing the number of wait states because several cycles will be required for each access to the external memory. The optimum balance between system clock frequency and the number of cycles required to access memory will depend on the system design and the performance required.

## Shared Read and Write Enables

CoreMemCtrl generates read and write enable signals for both flash and SRAM. These signals are named FLASHOEN, FLASHWEN, SRAMOEN, and SRAMWEN. The core also generates two common read and write enable signals, MEMREADN and MEMWRITEN, which can be connected to both flash and SRAM devices when the SHARED\_RW parameter is set. This option is intended for use in situations where the number of FPGA pins available for interfacing to external memory is limited.

## Address Bus Handling

The final two configuration options for CoreMemCtrl are used to control how the address bus connecting to the external memory devices relates to the AHB or AHB-Lite address bus, HADDR. For both flash and SRAM accesses, it is possible to drive the memory address bus (MEMADDR) with the byte, halfword, or word address. This essentially translates into driving the least significant bit of the memory address bus with HADDR[0], HADDR[1], or HADDR[2]. These address-related configuration options are provided to allow flexibility in how memory systems can be constructed and to accommodate devices which use different approaches to addressing. For example, some flash devices only use their A0 pin when operating in 8-bit mode, whereas other flash devices always make use of their A0 pin irrespective of the operating mode.

When the memory address bus is driven by the word address for flash accesses (by selecting the "0, 0, HADDR[27:2]" option for flash addressing), the setting for the flash data bus width, which can be 16-bit or 32-bit, is ignored. It is assumed that there is a 32-bit data bus connection to external flash when the word address is used for flash accesses.

## Example System

A typical system that includes CoreMemCtrl is shown in the [Figure 1-2](#). AHB/AHB-Lite bus interfaces can be auto-connected in SmartDesign using the Auto Connect menu option.

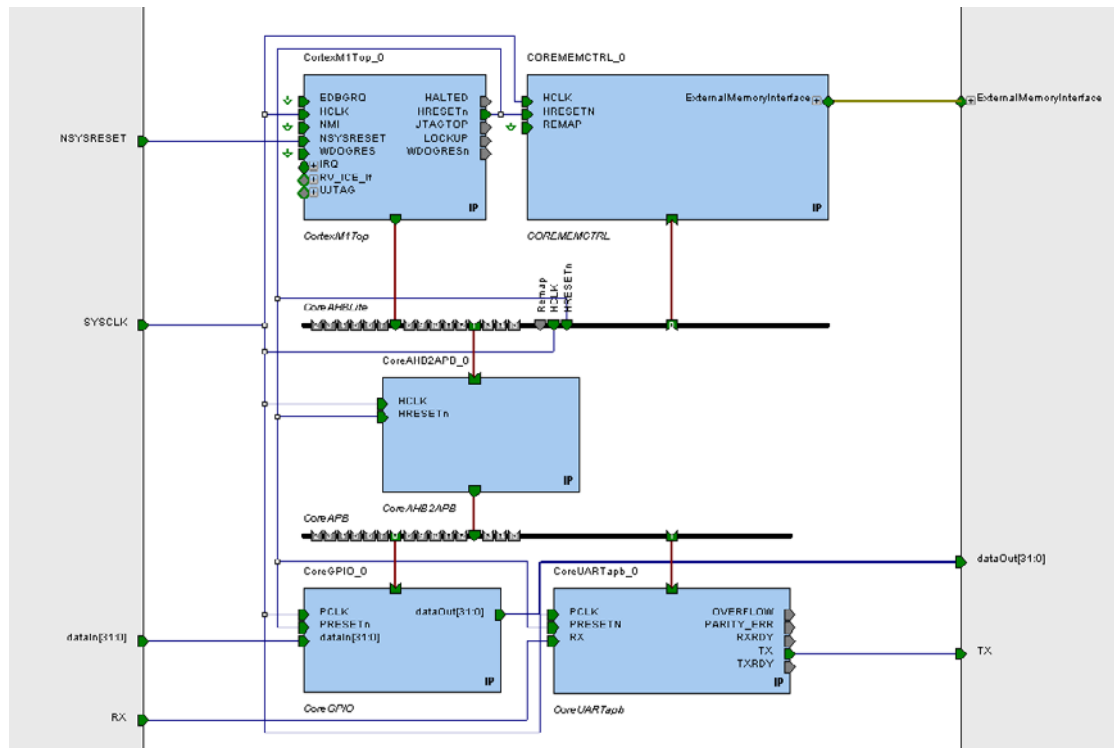


Figure 1-2 · Example System Including CoreMemCtrl

## Simulation

CoreMemCtrl comes with a simple user testbench, which can be invoked from the Libero IDE Project Manager. To run the testbench, use the configuration window to set the testbench configurable option for CoreMemCtrl to **User** before generating a CoreMemCtrl design. After the design has been generated, click the **Simulation** button in the Libero IDE GUI to run the testbench automatically.

**Note:** If CoreMemCtrl is included as a component within a larger design, then the CoreMemCtrl component must be set as the design root (from the right-click menu, select **Set As Root**), before running the user testbench.

An overview of the CoreMemCtrl user testbench is shown in Figure 1-3.

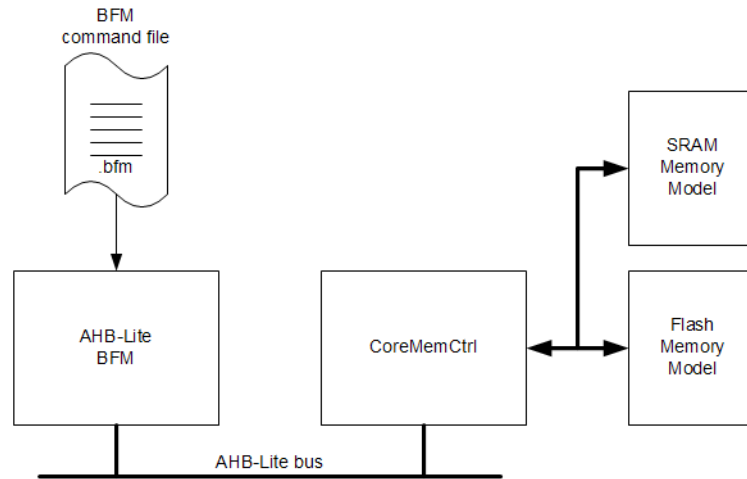


Figure 1-3 · Overview of CoreMemCtrl Testbench

The testbench is based around a bus functional model (BFM) which acts as an AHB-Lite master and is controlled by a command file, named `corememctrl_usertb.bfm`. Simple memory models are used to model external flash and SRAM devices. The BFM writes to and reads from the memory models via CoreMemCtrl.

Each time the testbench is run, the `corememctrl_usertb.bfm` command file is processed by an executable to create `corememctrl_usertb.vec`. This instruction vectors file is read in by the BFM module in the testbench. The `corememctrl_usertb.bfm` file is a text file and can be edited to suit any specific needs. After generating a CoreMemCtrl design from SmartDesign, the file can be found in the `<project>/simulation` directory.

## Synthesis in Libero IDE

To run synthesis on the core with the parameter settings selected in SmartDesign, set the design root appropriately, and click the **Synthesis** button in the Project Manager. The Synthesis window appears, displaying the Synplicity® project. To perform synthesis, click the **Run** button.

## Place-and-Route in Libero IDE

After setting the design root appropriately and running synthesis, click the **Layout** button in the Project Manager to invoke Designer. CoreMemCtrl requires no special place-and-route settings.

## Connecting to External Memories

This section includes a number of diagrams which illustrate how to connect the external memory interface ports of CoreMemCtrl to SRAM and flash devices. In the diagrams, generic representations of SRAM and flash devices are used; these representations are intended to cover devices from a range of manufacturers. Some manufacturers label the pins of their devices differently from others. In the following diagrams, the pin names on the SRAM and flash memories should be interpreted as described in [Table 1 on page 7](#). [Table 2-1](#) Lists the pin descriptions as used in the diagrams.

Table 2-1 · Pin descriptions for Connection Diagrams

Pin Name in Diagrams	Description
E#	Chip enable/chip select
G#	Read or output enable
W#	Write enable
BW#	Byte write enable on SRAM
LB#	Lower byte enable on SRAM
UB#	Upper byte enable on SRAM
BYTE#	Byte (8 bit) mode operation on flash
A0, A1, ..., An	Address bus pins. An is the most significant address bit.
DQ[n:m]	Bidirectional data bus
CLK	Clock input on synchronous SRAM
ADV#	Burst address advance on synchronous SRAM
ADSP#	Address status processor on synchronous SRAM
ADSC#	Address status controller on synchronous SRAM
FT#	Flow-through/pipeline mode operation on synchronous SRAM

16-bit flash devices are available from a number of manufacturers and normally have a BYTE# input which, when asserted Low, causes the device to operate in 8-bit mode. Only the lower half of the data bus (that is, DQ[7:0]) is used to carry data in 8-bit mode. Manufacturers of flash devices generally use one of two approaches to addressing:

- The A0 pin is only used (for the least significant address bit) when in 8-bit mode. When operating in 16-bit mode, the A0 pin is not used; typically the A0 input buffer is turned off in this type of flash device when the BYTE# pin is High. The remaining address pins (A[n:1]) are always used for addressing.
- All of the address pins, including A0, are used for addressing in both 8-bit and 16-bit mode. On this type of flash device, the upper data pin, DQ[15], is typically reused as the least significant address bit input when operating in 8-bit mode.

Refer to the datasheet of the external flash device being used with CoreMemCtrl to check how the device handles addressing. The flash addressing configurable option should be set to suit the flash device(s) in use. Flash addressing must also be correctly set when interfacing to a single 16-bit flash device, that is, when the flash data bus width is set to 16 bit.

**Note:** From the AHB or AHB-Lite master's viewpoint, only 32-bit word accesses to flash are supported. Byte and halfword accesses to flash are not supported.

Snippets from the CoreMemCtrl configuration window are included in each of the following connection diagrams to show the correspondence between the configuration options and the associated memory connections.

Figure Figure 2-1 to Figure 2-4 on page 17 show four different external memory systems. Figure 2-1 and Figure 2-3 on page 16 illustrate flash devices which use addressing as described in the first addressing approach. The flash devices in Figure 2-2 on page 15 and in Figure 2-4 use the approach to addressing described in the second addressing approach. Figure 2-5 and Figure 2-6 on page 19 illustrate additional pin connections relevant to synchronous SRAM devices which are not shown in previous diagrams.

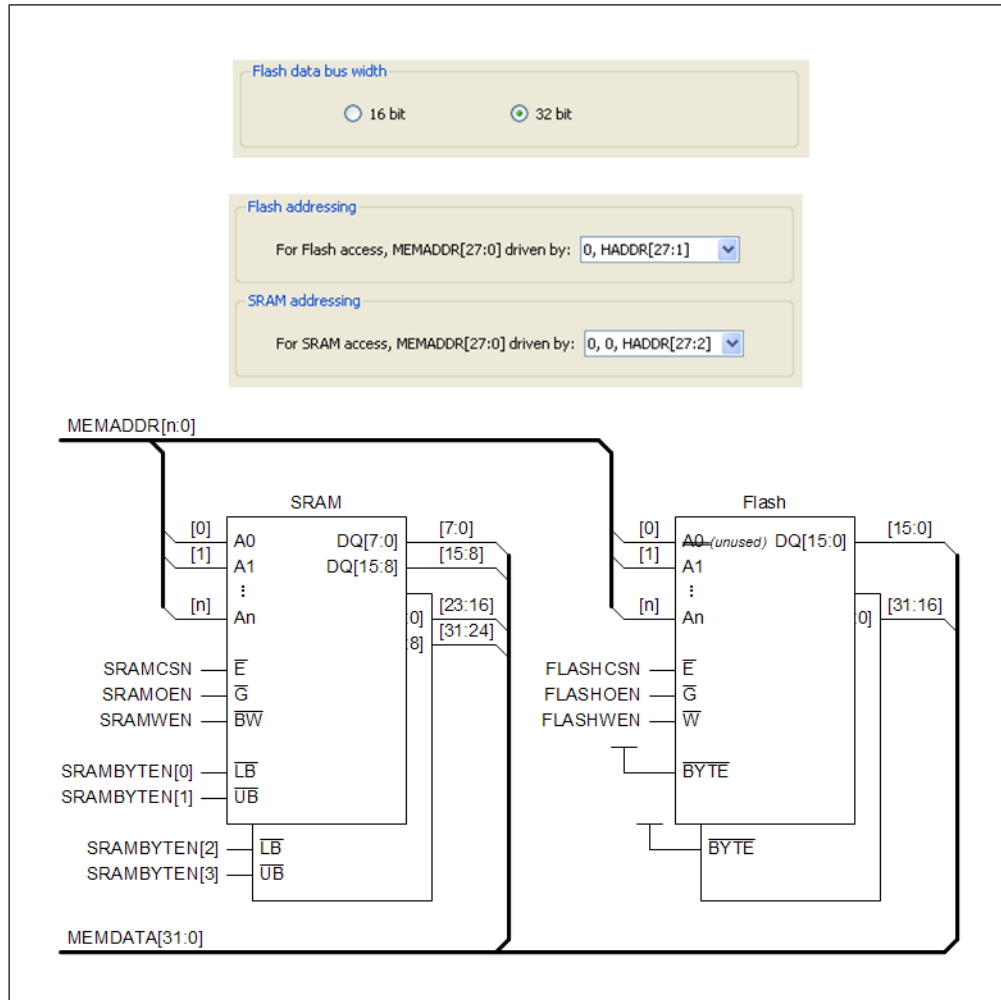


Figure 2-1 · Connecting to SRAM and Flash with 32-bit Flash Data Bus and Flash of Type 1

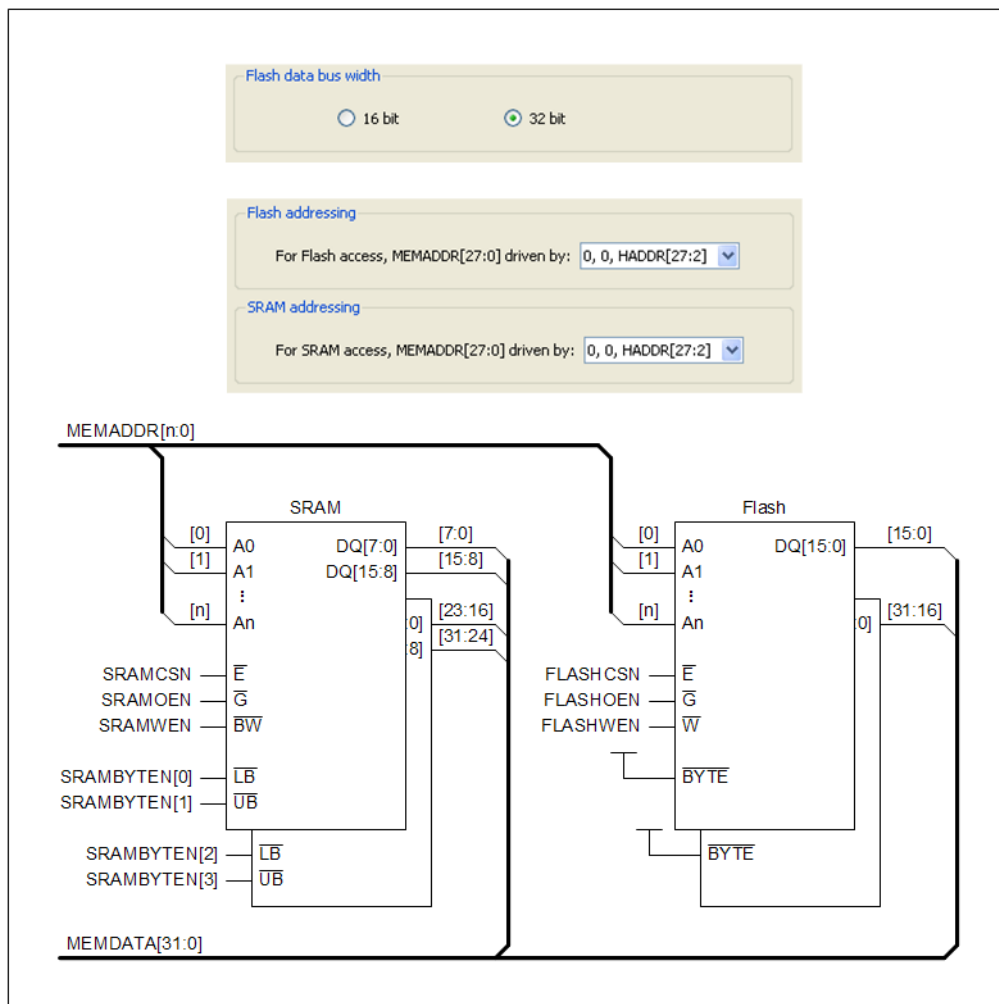


Figure 2-2 · Connecting to SRAM and Flash with 32-bit Flash Data Bus and Flash of Type 2

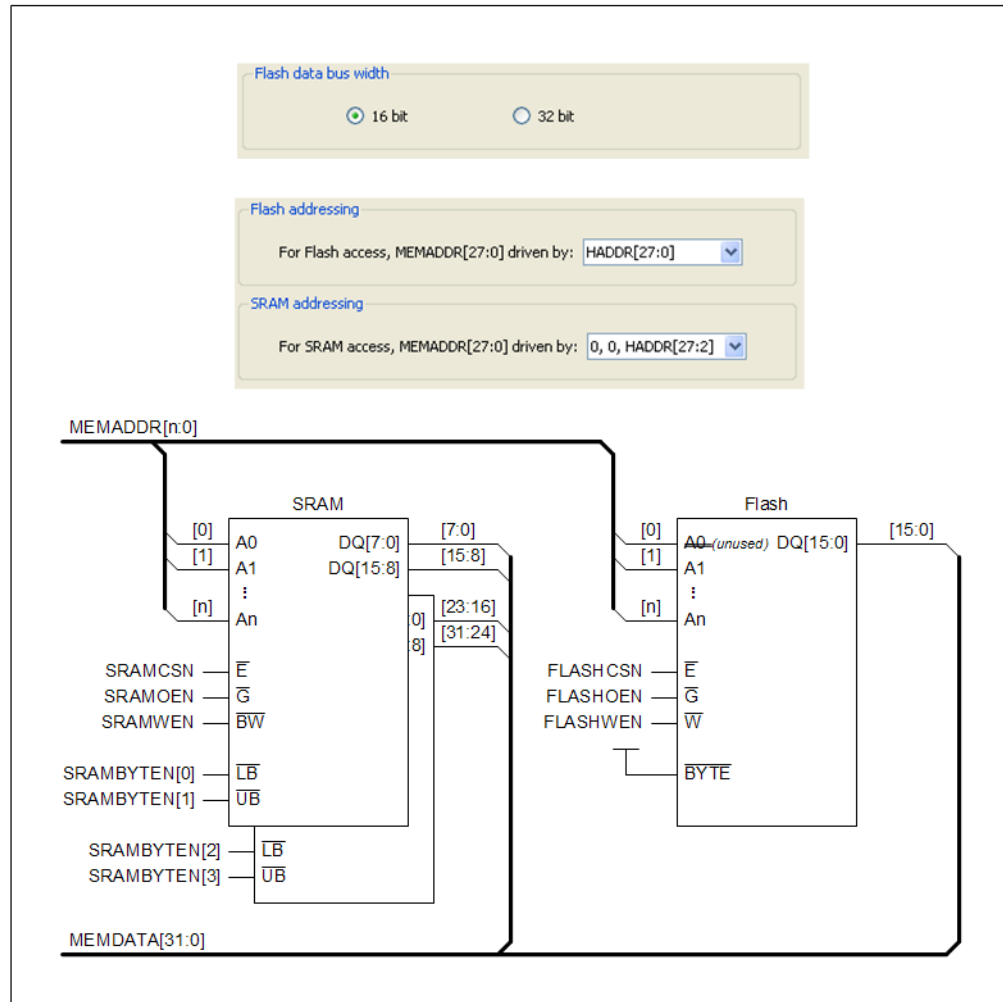


Figure 2-3 · Connecting to SRAM and Single Flash Device with 16-bit Flash Data Bus and Flash of Type 1

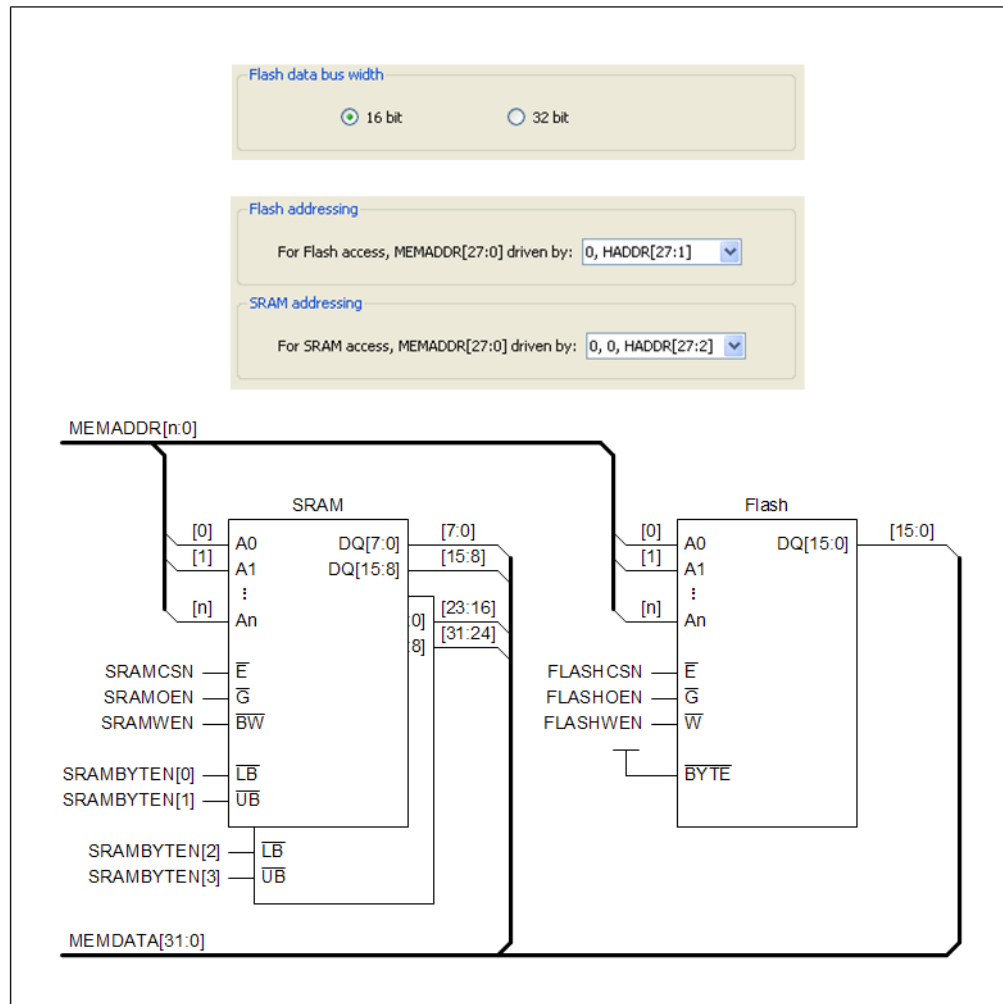


Figure 2-4 · Connecting to SRAM and Single Flash Device with 16-bit Flash Data Bus and Flash of Type 2

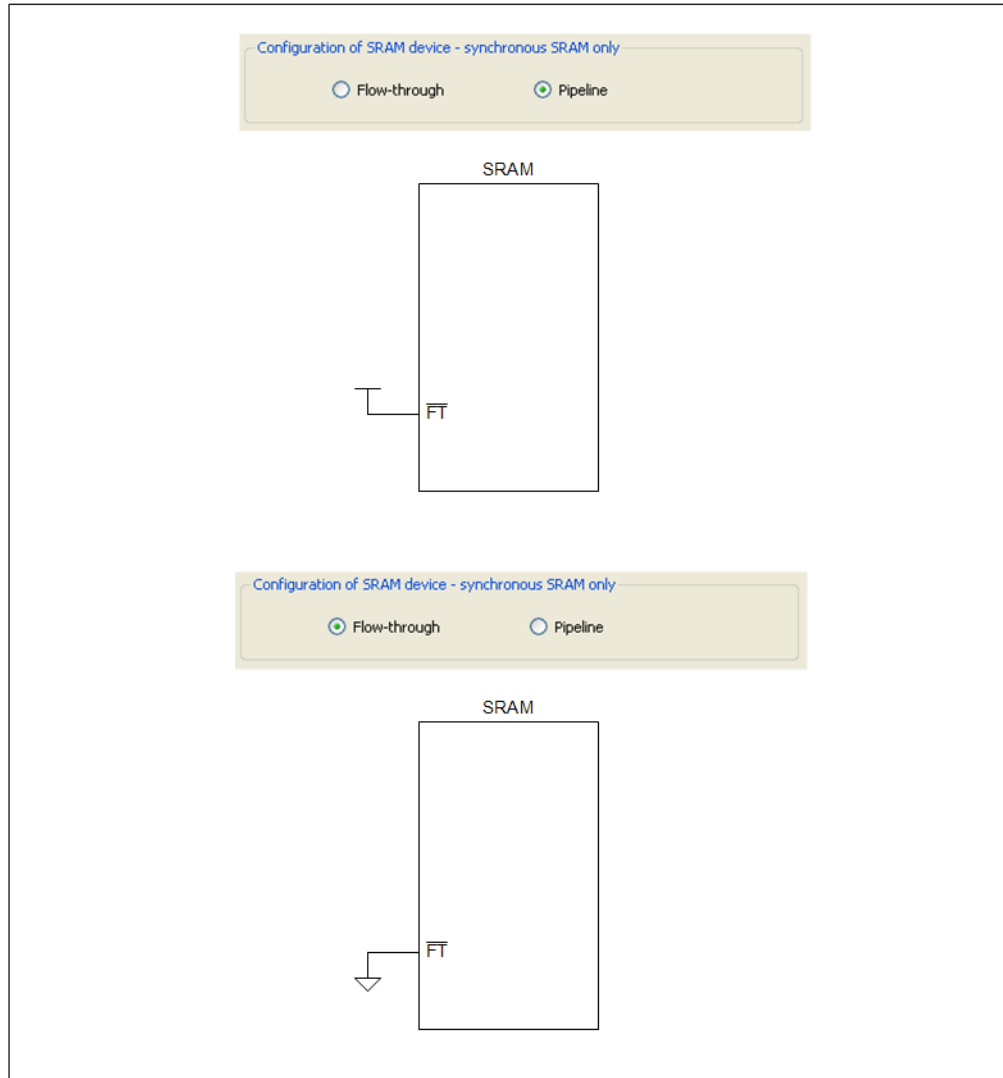


Figure 2-5 · Matching CoreMemCtrl Configuration to FT# Pin Level on Synchronous SRAM Devices with FT# Input

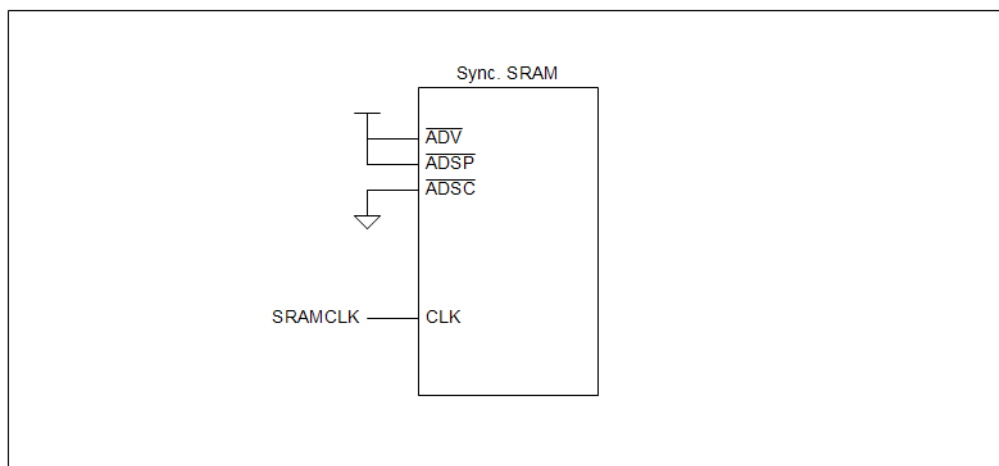


Figure 2-6 · Additional Connections for Synchronous SRAM Devices



---

## Registers

CoreMemCtrl does not contain any memory-mapped registers such as control or status registers.



## Memory Map

CoreMemCtrl uses 256 MB of address space on the AHB or AHB-Lite bus to which it is connected. This address space is evenly divided between flash memory and SRAM memory, so that each memory type can be up to 128 MB in size. The REMAP input is used to control which memory type appears in which half of the 256 MB slot. [Table 4-1](#) shows the relationship between the REMAP input and address offsets of flash and SRAM.

Table 4-1 · REMAP and Address Offset Relationship

REMAP	Address Offset	Memory Type
0	0x00000000 to 0x07FFFFFF	Flash
	0x08000000 to 0x0FFFFFFF	SRAM
1	0x00000000 to 0x07FFFFFF	SRAM
	0x08000000 to 0x0FFFFFFF	Flash



## Interface Descriptions

### Parameters/Generics

Table 5-1 · CoreMemCtrl Parameters/Generics

Parameter	Values	Default	Description
FAMILY	9, 10, 11, 12, 15, 16, 17, 20, 21, 22, 23	17	Must be set to match the supported FPGA family. 9 - RTSX-S 10 - EX 11 - AX 12 - RTAX-S 15 - ProASIC3 16 - ProASIC3E 17 - Fusion 20 - IGLOO 21 - IGLOOe 22 - ProASIC3L 23 - IGLOO PLUS
SYNC_SRAM	0, 1	1	Selects the type of external SRAM. 0 = Interfacing to asynchronous SRAM 1 = Interfacing to synchronous SRAM
FLOW_THROUGH	0, 1	0	This parameter is only relevant when SYNC_SRAM = 1 and should be set to match the operation of the external synchronous SRAM device. Some synchronous SRAM devices can be configured to operate in either pipeline or flow-through mode. 0 = External synchronous SRAM device is operating in pipeline mode. 1 = External synchronous SRAM device is operating in flow-through mode.
FLASH_16BIT	0, 1	0	0 = 32-bit data bus to external flash 1 = 16-bit data bus to external flash. This setting is intended to facilitate the use of a single 16-bit flash device. <i>Note: Only (32-bit) word accesses to external flash are supported from the AHB or AHB-Lite master. Halfword and byte accesses to flash are not supported. This is true even when FLASH_16BIT = 1, in which case each word access is transparently broken down into two separate halfword accesses to the actual flash device.</i>

Table 5-1 · CoreMemCtrl Parameters/Generics (continued)

Parameter	Values	Default	Description
NUM_WS_FLASH_READ	0 to 3	1	Number of wait states inserted during a read from flash
NUM_WS_FLASH_WRITE	1 to 3	1	Number of wait states inserted during a write to flash
NUM_WS_SRAM_READ	0 to 3	1	Number of wait states inserted during a read from asynchronous SRAM. This parameter is only relevant when SYNC_SRAM = 0.
NUM_WS_SRAM_WRITE	1 to 3	1	Number of wait states inserted during a write to asynchronous SRAM. This parameter is only relevant when SYNC_SRAM = 0.
SHARED_RW	0, 1	0	0 = Separate read and write enable signals (that is, FLASHOEN, FLASHWEN, SRAMOEN and SRAMWEN) are used for flash and SRAM. 1 = Common read and write enable signals (that is, MEMREADN and MEMWRITEN) are used for both flash and SRAM. This setting is intended for use in cases where a limited number of pins are available for interfacing to external memory.
FLASH_ADDR_SEL	0 to 2	2	This parameter is used to control how the address bus to flash relates to the AHB or AHB-Lite address for the transfer. 0 = Byte address is used. That is, MEMADDR[27:0] is driven by HADDR[27:0] during flash accesses. 1 = Halfword address is used. That is, MEMADDR[27:0] is driven by {0, HADDR[27:1]} during flash accesses. 2 = Word address is used. That is MEMADDR[27:0] is driven by {0, 0, HADDR[27:2]} during flash accesses.
SRAM_ADDR_SEL	0 to 2	2	This parameter is used to control how the address bus to SRAM relates to the AHB or AHB-Lite address for the transfer. 0 = Byte address is used. That is, MEMADDR[27:0] is driven by HADDR[27:0] during SRAM accesses. 1 = Halfword address is used. That is, MEMADDR[27:0] is driven by {0, HADDR[27:1]} during SRAM accesses. 2 = Word address is used. That is MEMADDR[27:0] is driven by {0, 0, HADDR[27:2]} during SRAM accesses.

## Ports

Table 5-2 outlines the top-level signals for CoreMemCtrl.

Table 5-2 · CoreMemCtrl Ports

Name	Type	Description
<b>AHB/AHB-Lite Bus Signals</b>		
HCLK	Input	AHB system clock. Reference clock for all internal logic
HRESETN	Input	AHB active Low asynchronous reset
HSEL	Input	AHB slave select
HWRITE	Input	AHB write/read indication
HREADYIN	Input	AHB ready indication from bus
HTRANS[1:0]	Input	AHB transfer type
HSIZE[2:0]	Input	AHB size of transfer
HWDATA[31:0]	Input	AHB write data
HADDR[27:0]	Input	AHB address bus
HRDATA[31:0]	Output	AHB read data
HREADY	Output	AHB ready output to bus
HRESP[1:0]	Output	AHB response
HRDATA[31:0]	Output	AHB read data
<b>Remap Control</b>		
REMAP	Input	Remap control. When asserted, flash and SRAM locations are swapped in the address map.
<b>External Memory Interface</b>		
FLASHCSN	Output	Flash chip select, active Low
FLASHOEN	Output	Flash output enable, active Low
FLASHWEN	Output	Flash write enable, active Low
SRAMCLK	Output	SRAM clock. This clock signal is the inverse of the AHB clock input, HCLK.
SRAMCSN	Output	SRAM chip select, active Low
SRAMOEN	Output	SRAM output enable, active Low
SRAMWEN	Output	SRAM write enable, active Low
SRAMBYTEN[3:0]	Output	SRAM byte enables, active Low
MEMREADN	Output	Common read enable, active Low. Can be connected to both flash and SRAM when the SHARED_RW parameter is set.

*Note:* Unless otherwise noted, all of the signals above are active High

Table 5-2 · CoreMemCtrl Ports (continued)

Name	Type	Description
MEMWRITEN	Output	Common write enable, active Low. Can be connected to both flash and SRAM when the SHARED_RW parameter is set.
MEMADDR[27:0]	Output	Common flash/SRAM address bus
MEMDATA[31:0]	Inout	Common flash/SRAM data bus

*Note:* Unless otherwise noted, all of the signals above are active High

# Waveforms

The waveforms in this section show the timing of the CoreMemCtrl signals.

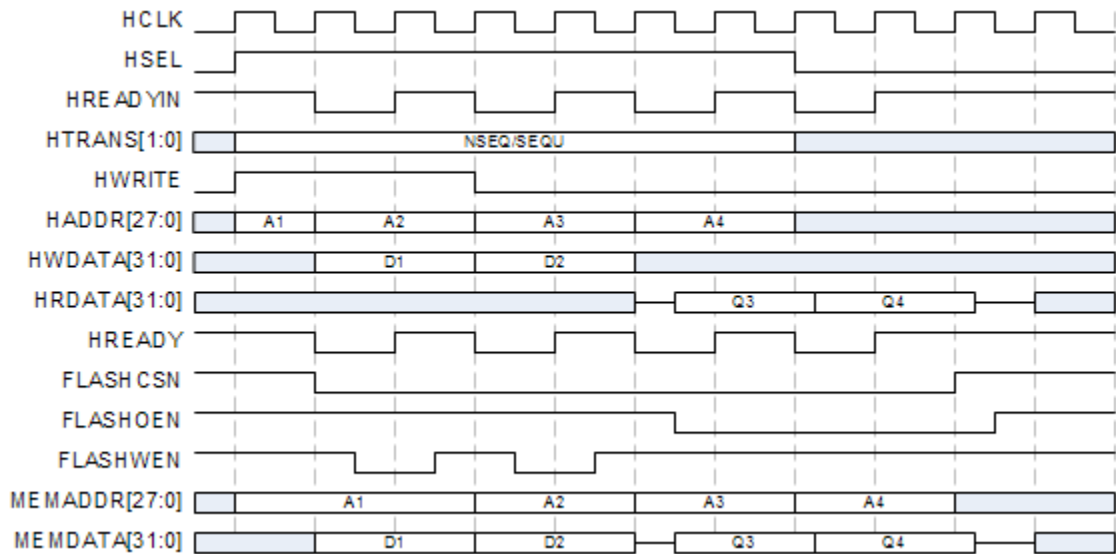


Figure 6-1 · Flash Access: Two Writes and Two Reads with One Wait State for Each Transfer

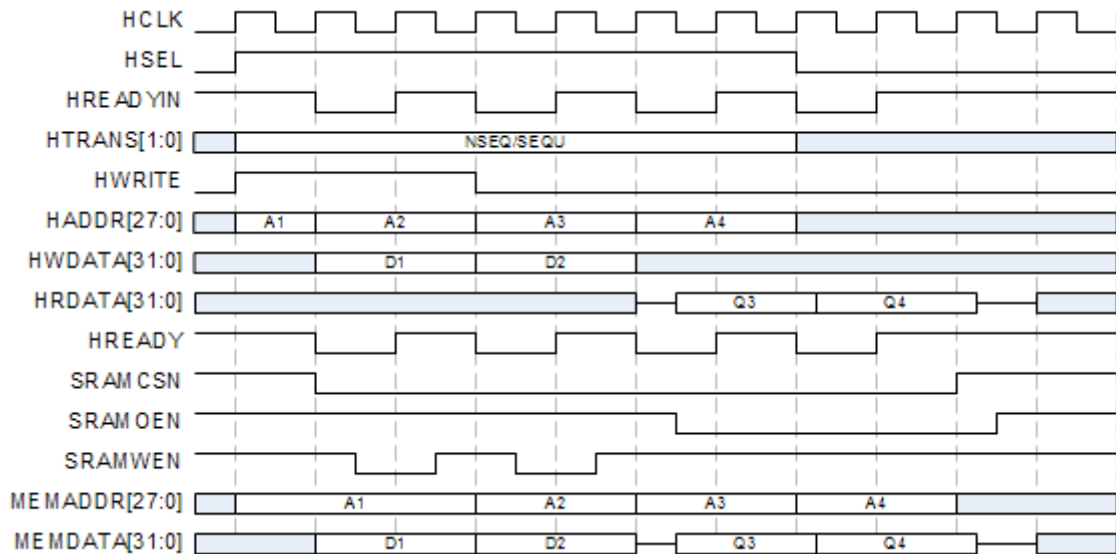


Figure 6-2 · Asynchronous SRAM Access: Two Writes and Two Reads with One Wait State for each Transfer

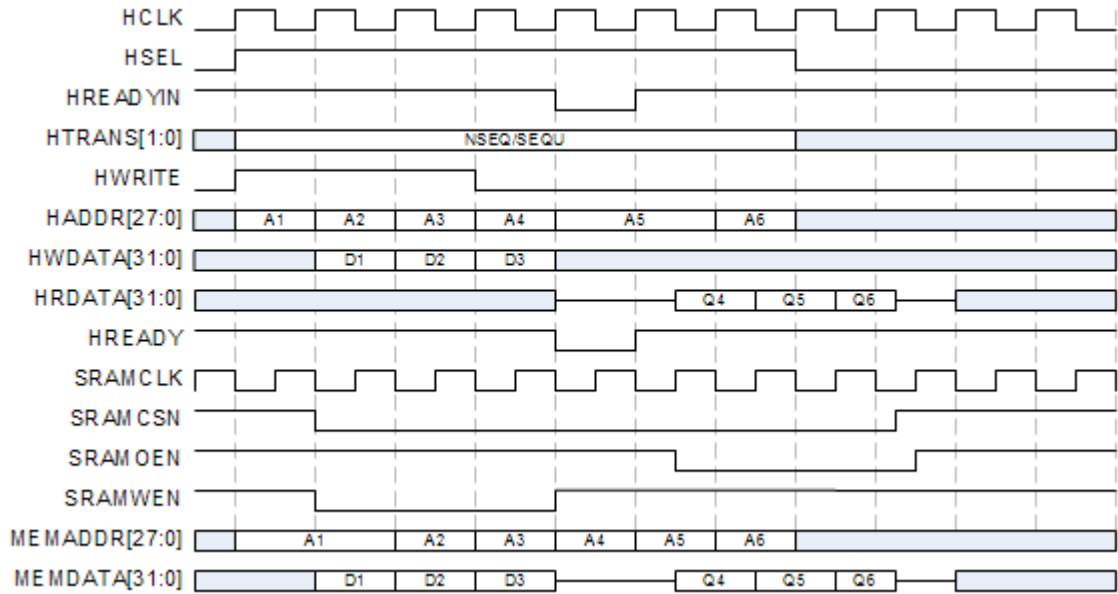


Figure 6-3 · Synchronous SRAM Access: Three Writes and Three Reads

---

# Ordering Information

## Ordering Codes

CoreMemCtrl can be ordered through your local Actel sales representative. It should be ordered using the following number scheme: CoreMemCtrl-XX, where XX is listed in [Table 7-1](#).

Table 7-1 · Ordering Codes

XX	Description
OM	RTL for Obfuscated RTL – multiple-use license
RM	RTL for RTL source – multiple-use license

*Note:* CoreMemCtrl-OM is included free with a Libero IDE license



---

## Product Support

Actel backs its products with various support services including Customer Service, a Customer Technical Support Center, a web site, an FTP site, electronic mail, and worldwide sales offices. This appendix contains information about contacting Actel and using these support services.

### Customer Service

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

From Northeast and North Central U.S.A., call 650.318.4480

From Southeast and Southwest U.S.A., call 650.318.4480

From South Central U.S.A., call 650.318.4434

From Northwest U.S.A., call 650.318.4434

From Canada, call 650.318.4480

From Europe, call 650.318.4252 or +44 (0) 1276 401 500

From Japan, call 650.318.4743

From the rest of the world, call 650.318.4743

Fax, from anywhere in the world 650.318.8044

### Actel Customer Technical Support Center

Actel staffs its Customer Technical Support Center with highly skilled engineers who can help answer your hardware, software, and design questions. The Customer Technical Support Center spends a great deal of time creating application notes and answers to FAQs. So, before you contact us, please visit our online resources. It is very likely we have already answered your questions.

### Actel Technical Support

Visit the [Actel Customer Support website \(www.actel.com/custsup/search.html\)](http://www.actel.com/custsup/search.html) for more information and support. Many answers available on the searchable web resource include diagrams, illustrations, and links to other resources on the Actel web site.

### Website

You can browse a variety of technical and non-technical information on Actel's [home page](http://www.actel.com), at [www.actel.com](http://www.actel.com).

### Contacting the Customer Technical Support Center

Highly skilled engineers staff the Technical Support Center from 7:00 A.M. to 6:00 P.M., Pacific Time, Monday through Friday. Several ways of contacting the Center follow:

#### Email

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. We constantly monitor the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and your contact information for efficient processing of your request.

The technical support email address is [tech@actel.com](mailto:tech@actel.com).

## Phone

Our Technical Support Center answers all calls. The center retrieves information, such as your name, company name, phone number and your question, and then issues a case number. The Center then forwards the information to a queue where the first available application engineer receives the data and returns your call. The phone hours are from 7:00 A.M. to 6:00 P.M., Pacific Time, Monday through Friday. The Technical Support numbers are:

**650.318.4460**

**800.262.1060**

Customers needing assistance outside the US time zones can either contact technical support via email ([tech@actel.com](mailto:tech@actel.com)) or contact a local sales office. [Sales office listings](#) can be found at [www.actel.com/contact/offices/index.html](http://www.actel.com/contact/offices/index.html).

# Index

## A

Actel  
  electronic mail 33  
  telephone 34  
  web-based technical support 33  
  website 33  
address bus handling 10  
AHB 5  
AHB/AHB-Lite bus signals 27

## C

contacting Actel  
  customer service 33  
  electronic mail 33  
  telephone 34  
  web-based technical support 33  
CoreMemCtrl  
  block diagram 5  
  configuration window 9  
  device utilization and performance 7  
  key features 6  
  overview 5  
  supported families 6  
  supported interfaces 6  
  version 6  
customer service 33

## E

external memories 13  
external memory interface 27

## F

flash data bus width 10

## G

generics 25

## M

memory-mapped registers 21

## O

ordering code 31

## P

parameters 25  
ports 27  
product support 33–34  
  customer service 33  
  electronic mail 33  
  technical support 33  
  telephone 34  
  website 33

## R

remap control 27

## S

shared read and write enables 10  
SRAM type 10

## T

technical support 33  
testbenches  
  operation 31  
toolflow  
  example system 11  
  Place-and-Route in Libero IDE 12  
  simulation 11  
  SmartDesign 9  
  synthesis in Libero IDE 12

## W

wait states 10  
web-based technical support 33



**Actel is the leader in low-power and mixed-signal FPGAs and offers the most comprehensive portfolio of system and power management solutions. Power Matters. Learn more at [www.actel.com](http://www.actel.com).**

**Actel Corporation** • 2061 Stierlin Court • Mountain View, CA 94043 • USA

Phone 650.318.4200 • Fax 650.318.4600 • Customer Service: 650.318.1010 • Customer Applications Center: 800.262.1060

**Actel Europe Ltd.** • River Court, Meadows Business Park • Station Approach, Blackwater • Camberley Surrey GU17 9AB • United Kingdom

Phone +44 (0) 1276 609 300 • Fax +44 (0) 1276 607 540

**Actel Japan** • EXOS Ebisu Building 4F • 1-24-14 Ebisu Shibuya-ku • Tokyo 150 • Japan

Phone +81.03.3445.7671 • Fax +81.03.3445.7668 • <http://jp.actel.com>

**Actel Hong Kong** • Room 2107, China Resources Building • 26 Harbour Road • Wanchai • Hong Kong

Phone +852 2185 6460 • Fax +852 2185 6488 • [www.actel.com.cn](http://www.actel.com.cn)