Reference Design for MIPS32® M14K™ Cores Application Note

Document Number: MD00716
Revision 01.00
March 25, 2010

MIPS Technologies, Inc.
955 East Arques Avenue
Sunnyvale, CA 94085-4521

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Table of Contents

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

1 Introduction

The MIPS32® M14K™ Processor Core is cache-less and instead uses a SRAM-style bus that may be configured with either a Dual Memory Interface or a Unified Memory Interface. The Dual Memory Interface has separate instruction (ISRAM) and data (DSRAM) memory interfaces, and provides a redirection mechanism that permits D-side references to be handled by the I-side. The Unified Memory Interface has a single memory interface that shares both data and instructions.

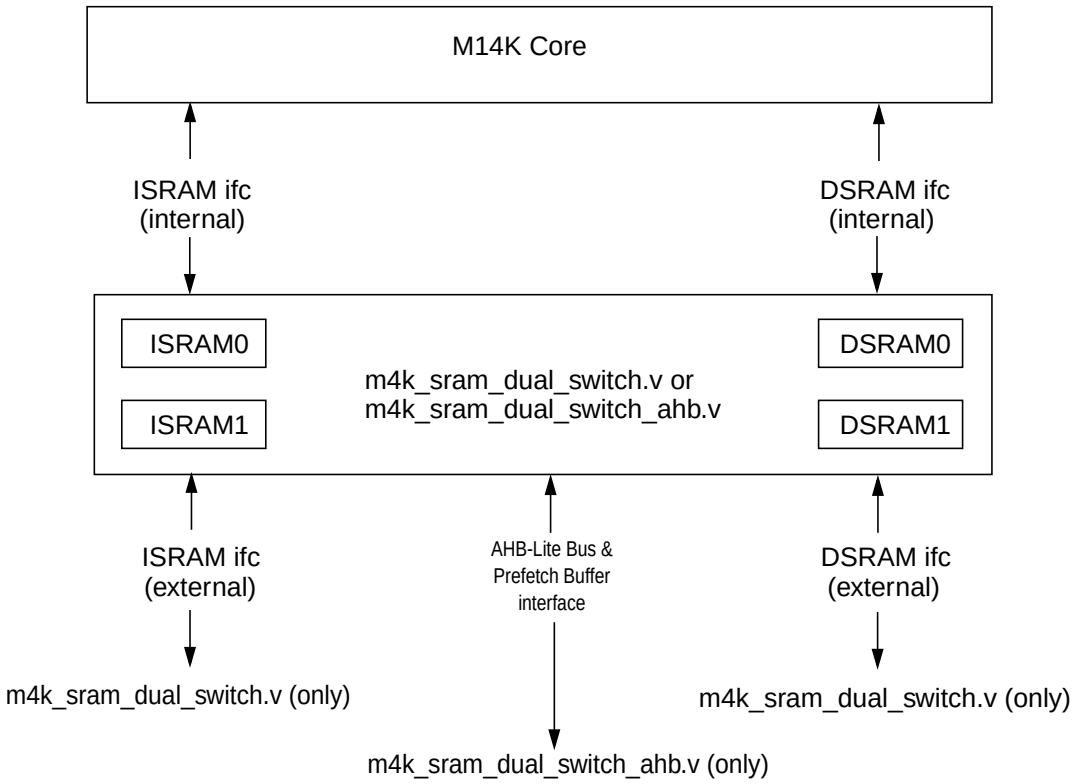
This application note will introduce you to working with the higher performance Dual Memory Interface (ISRAM and DSRAM) with an optimized interface for flash memory and native AMBA®-3 AHB-Lite Bus Interface Unit. Source code and scripts accompany this application note that demonstrates how to build and debug an application in ISRAM/DSRAM. Target platforms include the CASim™ Cycle Accurate Simulator and the SEAD™-3/YAMON™ hardware platform.

2 Programming the m4k_sram_dual_switch*.v

This section discusses programming the `m4k_sram_dual_switch.v` (for SRAM interface) and `m4k_sram_dual_switch_ahb.v` (for `AHB-Lite` bus interface) customer example modules. These files are located in the `$MIPS_PROJECT/proc/design/rtl` directory of the M14K deliverable, and they may be modified by the customer to add application-specific functionality. All SEAD-3 hardware implementations include the deliverable version of the `m4k_sram_dual_switch_ahb.v` module. A block diagram of the system interface for the `m4k_sram_dual_switch.v` or `m4k_sram_dual_switch_ahb.v` modules is shown in Figure 1.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Figure 1 SRAM/DSRAM Interface Block Diagram



2.1 Theory of Operation

Following reset, the ISRAM and DSRAM in the `m4k_sram_dual_switch*.v` module are disabled, and therefore all memory operations from the core bypass the ISRAM, DSRAM, and FLASH interfaces and go through the AHB-lite interface. The `m4k_sram_dual_switch_ahb.v` module merges the separate (external) ISRAM & DSRAM and Flash interfaces into the AHB-lite interface bus that is connected to the system memory controller. Software can enable or disable the ISRAM & DSRAM & FLASH interface and configure their physical base address decoding. The base addresses of the ISRAM & DSRAM & FLASH interface may be initialized at any physical address as long as it's aligned with the physical size of the RAM, i.e., 16KB, 32KB, etc. When the DSRAM is enabled, all CPU loads and stores within its configured address range go to the DSRAM. When the ISRAM or FLASH interface is enabled, all CPU instruction fetches within its configured address range come from the ISRAM or FLASH interface respectively. Also, CPU loads and stores can go to the ISRAM or FLASH interface if REDIRECTION is enabled. Memory references outside the configured address ranges of ISRAM and DSRAM and FLASH interface will be forwarded to the system memory controller via the AHB-lite interface bus.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

2.2 Control Register Summary

Table 1 lists the control registers in order of physical address.

Table 1 Control Registers

Register Address (Physical)	Register Name	Function
0x1e000000	*ISRAM0 Configuration*	ISRAM0 Control
0x1e000010	*ISRAM0 Mask*	ISRAM0 Size and Alignment
0x1e000004	*ISRAM1 Configuration*	ISRAM1 Control
0x1e000014	*ISRAM1 Mask*	ISRAM1 Size and Alignment
0x1e000008	*Prefetch-Buffer Configuration*	Prefetch-Buffer Control
0x1e000018	*Prefetch-Buffer Mask*	Prefetch-Buffer Size and Alignment
0x1e000020	*DSRAM0 Configuration*	DSRAM0 Control
0x1e000030	*DSRAM0 Mask*	DSRAM0 Size and Alignment
0x1e000024	*DSRAM1 Configuration*	DSRAM1 Control
0x1e000034	*DSRAM1 Mask*	DSRAM1 Size and Alignment

2.3 ISRAM0 Configuration Register (0x1e000000)

The ISRAM0 is controlled by the *ISRAM0 Configuration* register located at physical address 0x1e000000. It contains a 22-bit field that sets the I-side physical base address. This is the physical starting address at which the ISRAM0 will be decoded, and it must be loaded with an address that is aligned with the block size of the ISRAM0. The *Enable* (*E*) bit enables the decoding of CPU fetches from ISRAM0. The *Redirection* (*r*) bit enables CPU loads and stores to ISRAM0, so instructions may be copied to it. Figure 2 shows the format of the *ISRAM0 Configuration* register; Table 2 describes the *ISRAM0 Configuration* register fields.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Figure 2 ISRAM0 Configuration Register Format

Table 2 ISRAM0 Configuration Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
I-Side Base Address	31:10	I-side physical base address (must be aligned with block size).	R/W	0
0	9:2	Must be written with zero; returns zero on read.	0	0
R	1	Redirect D-side if hitting I-side address range (and not hitting D-side address range).	R/W	
E	0	Enable the ISRAM0.	R/W	0

2.4 ISRAM1 Configuration Register (0x1e000004)

The ISRAM1 is controlled by the *ISRAM1 Configuration* register located at physical address 0x1e000004. It contains a 22-bit field that sets the I-side physical base address. This is the physical starting address at which the ISRAM1 will be decoded, and it must be loaded with an address that is aligned with the block size of the ISRAM1. The *Enable* (*E*) bit enables the decoding of CPU fetches from ISRAM1. The *Redirection* (*r*) bit enables CPU loads and stores to ISRAM1, so instructions may be copied to it. Figure 2 shows the format of the *ISRAM1 Configuration* register; Table 2 describes the *ISRAM1 Configuration* register fields.

Figure 3 ISRAM1 Configuration Register Format

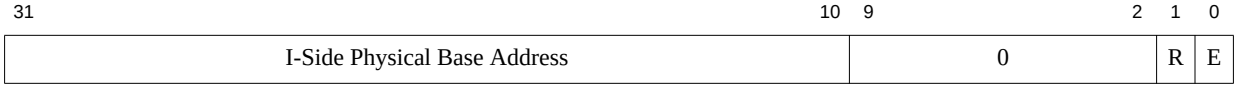


Table 3 ISRAM1 Configuration Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
I-Side Base Address	31:10	I-side physical base address (must be aligned with block size).	R/W	0
0	9:2	Must be written with zero; returns zero on read.	0	0
R	1	Redirect D-side if hitting I-side address range (and not hitting D-side address range).	R/W	
E	0	Enable the ISRAM1.	R/W	0

2.5 ISRAM0 Mask Register (0x1e000010)

The *ISRAM0 Mask* register is located at physical address 0x1e000010. It contains a 22-bit read-only mask used by software to mask and align the I-Side Address in the *ISRAM0 Configuration* register.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Figure 4 shows the format of the *ISRAM0 Mask* register; Table 4 describes the *ISRAM0 Mask* register fields.

Figure 4 ISRAM0 Mask Register Format

Table 4 ISRAM0 Mask Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
I-Side Address Mask	31:10	I-side address mask 32K=0xffff8000, 64K=0xffff0000.	R	0
0	9:0	Returns zero on read.	0	0

2.6 ISRAM1 Mask Register (0x1e000010)

The *ISRAM1 Mask* register is located at physical address 0x1e000010. It contains a 22-bit read-only mask used by software to mask and align the I-Side Address in the *ISRAM1 Configuration* register.

Figure 4 shows the format of the *ISRAM1 Mask* register; Table 4 describes the *ISRAM1 Mask* register fields.

Figure 5 ISRAM1 Mask Register Format

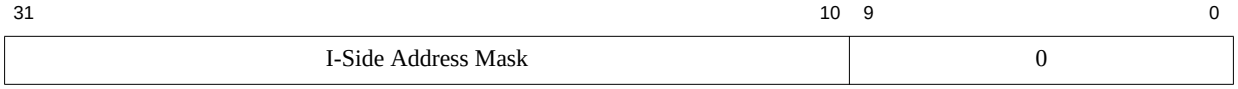


Table 5 ISRAM1 Mask Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
I-Side Address Mask	31:10	I-side address mask 32K=0xffff8000, 64K=0xffff0000.	R	0
0	9:0	Returns zero on read.	0	0

2.7 DSRAM0 Configuration Register (0x1e000020)

The DSRAM0 is controlled by the *DSRAM0 Configuration* register located at physical address 0x1e000020. It contains a 22-bit field for setting the D-side physical base address. This is the physical starting address at which the DSRAM0 will be decoded, and it must be loaded with an address that is aligned with the block size of the DSRAM0. The *Enable* (*E*) bit enables the decoding of CPU loads and stores to DSRAM0. The read-only *Redirection* (*R*) bit indicates a D-side access has been redirected to I-side.

Figure 6 shows the format of the *DSRAM0 Configuration* register; Table 6 describes the *DSRAM0 Configuration* register fields.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Figure 6 DSRAM0 Configuration Register Format

Table 6 DSRAM0 Configuration Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
I-Side Base Address	31:10	I-side physical base address (must be aligned with block size),	R/W	0
0	9:2	Must be written with zero; returns zero on read.	0	0
R	1	Redirect D-side if NOT hitting D-side address range (if only access to external world from I-side).	R	0
E	0	Enable the DSRAM0.	R/W	0

2.8 DSRAM1 Configuration Register (0x1e000024)

The DSRAM1 is controlled by the *DSRAM1 Configuration* register located at physical address 0x1e000024. It contains a 22-bit field for setting the D-side physical base address. This is the physical starting address at which the DSRAM1 will be decoded, and it must be loaded with an address that is aligned with the block size of the DSRAM1. The *Enable* (*E*) bit enables the decoding of CPU loads and stores to DSRAM1. The read-only *Redirection* (*R*) bit indicates a D-side access has been redirected to I-side.

Figure 6 shows the format of the *DSRAM1 Configuration* register; Table 6 describes the *DSRAM1 Configuration* register fields.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Figure 7 DSRAM1 Configuration Register Format

Table 7 DSRAM1 Configuration Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
I-Side Base Address	31:10	I-side physical base address (must be aligned with block size),	R/W	0
0	9:2	Must be written with zero; returns zero on read.	0	0
R	1	Redirect D-side if NOT hitting D-side address range (if only access to external world from I-side).	R	0
E	0	Enable the DSRAM1.	R/W	0

2.9 DSRAM0 Mask Register (0x1e000030)

The *DSRAM0 Mask* register is located at physical address 0x1e000030. It contains a 22-bit read-only mask used by software to mask and align the D-Side Address in the *DSRAM0 Configuration* register.

Figure 8 shows the format of the *DSRAM0 Mask* register; Table 8 describes the *DSRAM0 Mask* register fields.

Figure 8 DSRAM0 Mask Register Format

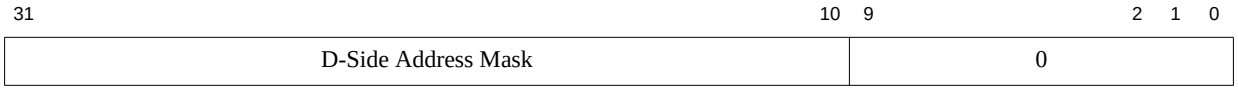


Table 8 DSRAM0 Mask Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
D-Side Address Mask	31:10	D-side address mask 32K=0xffff8000, 64K=0xffff0000.	R	0
0	9:0	Returns zero on read.	0	0

2.10 DSRAM1 Mask Register (0x1e000034)

The *DSRAM1 Mask* register is located at physical address 0x1e000034. It contains a 22-bit read-only mask used by software to mask and align the D-Side Address in the *DSRAM1 Configuration* register.

Figure 8 shows the format of the *DSRAM1 Mask* register; Table 8 describes the *DSRAM1 Mask* register fields.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Figure 9 DSRAM1 Mask Register Format

Table 9 DSRAM1 Mask Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
D-Side Address Mask	31:10	D-side address mask 32K=0xffff8000, 64K=0xffff0000.	R	0
0	9:0	Returns zero on read.	0	0

2.11 Prefetch-Buffer Configuration Register (0x1e000008)

The Prefetch-Buffer is controlled by the *Prefetch-Buffer Configuration* register located at physical address 0x1e000008. It contains a 22-bit field for setting the I-side physical base address. This is the physical starting address at which the Prefetch-Buffer will be decoded, and it must be loaded with an address that is aligned with the block size of the Prefetch-Buffer. The *Enable* (*E*) bit enables the decoding of CPU fetches from Prefetch-Buffer. The *Redirection* (*r*) bit enables CPU loads and stores to Prefetch-Buffer so instructions may be copied to it. Figure 10 shows the format of the *Prefetch-Buffer Configuration* register; Table 10 describes the *Prefetch-Buffer Configuration* register fields.

Figure 10 IPrefetch-Buffer Configuration Register Format

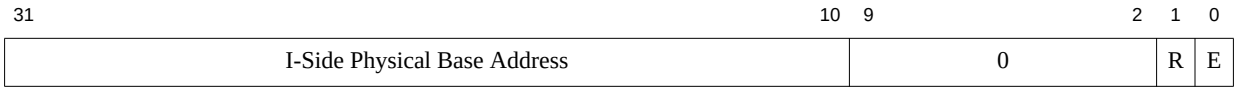


Table 10 Prefetch-Buffer Configuration Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
I-Side Base Address	31:10	I-side physical base address (must be aligned with block size).	R/W	0
0	9:2	Must be written with zero; returns zero on read.	0	0
R	1	Redirect D-side if hitting I-side address range (and not hitting D-side address range).	R/W	
E	0	Enable the Prefetch-Buffer.	R/W	0

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

2.12 Prefetch-Buffer Mask Register (0x1e000018)

The *Prefetch-Buffer Mask* register is located at physical address 0x1e000018. It contains a 22-bit read-only mask used by software to mask and align the D-Side Address in the *Prefetch-Buffer Configuration* register.

Figure 11 shows the format of the *Prefetch-Buffer Mask* register; Table 11 describes the *Prefetch-Buffer Mask* register fields.

Figure 11 Prefetch-Buffer Mask Register Format

Table 11 Prefetch-Buffer Mask Register Field Descriptions

Fields		Description	Read / Write	Reset State
Name	Bits			
D-Side Address Mask	31:10	D-side address mask 32K=0xffff8000, 64K=0xffff0000.	R	0
0	9:0	Returns zero on read.	0	0

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

3 References

This appendix lists other documents available from MIPS Technologies, Inc. that are referenced elsewhere in this document. These documents may be included in the `$MIPS_HOME/$MIPS_CORE/doc` area of a typical *Core-Name* soft or hard core release, or in some cases may be available on the MIPS web site, http://www.mips.com.

1. CASim™ Cycle Accurate Simulator for the MIPS32® M14K™ Core User's Guide
   MIPS Document: MD00697

1. CASim™ Cycle Accurate Simulator for the MIPS32® M14K™ Core Release Notes
   MIPS Document: MD00713

2. MIPS32® M14K™ Processor Core Datasheet
   MIPS Document: MD00666

3. MIPS32® M14K™ Processor Core Integrator’s Guide
   MIPS Document: MD00667

4. MIPS32® M14K™ Processor Core Software Users Manual
   MIPS Document: MD00668

5. SEAD™-3 Board Users Manual
   MIPS Document: MD00682

6. SEAD™-3 Board Getting Started
   MIPS Document: MD00687

7. YAMON™ Users Manual
   MIPS Document: MD00008

8. The GNU linker Manual “Using ld, version 2”

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

4 Revision History

Change bars (vertical lines) in the margins of this document indicate significant changes in the document since its last release. Change bars are removed for changes that are more than one revision old.

This document may refer to Architecture specifications (for example, instruction set descriptions and EJTAG register definitions), and change bars in these sections indicate changes since the previous version of the relevant Architecture document.

Revision	Date	Description
1.00	March 25, 2010	Initial external release.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Copyright © 2010 MIPS Technologies Inc. All rights reserved.

Copyright © 2010 MIPS Technologies, Inc. All rights reserved.

Unpublished rights (if any) reserved under the copyright laws of the United States of America and other countries.

This document contains information that is proprietary to MIPS Technologies, Inc. ("MIPS Technologies"). Any copying, reproducing, modifying or use of this information (in whole or in part) that is not expressly permitted in writing by MIPS Technologies or an authorized third party is strictly prohibited. At a minimum, this information is protected under unfair competition and copyright laws. Violations thereof may result in criminal penalties and fines.

Any document provided in source format (i.e., in a modifiable form such as in FrameMaker or Microsoft Word format) is subject to use and distribution restrictions that are independent of and supplemental to any and all confidentiality restrictions. UNDER NO CIRCUMSTANCES MAY A DOCUMENT PROVIDED IN SOURCE FORMAT BE DISTRIBUTED TO A THIRD PARTY IN SOURCE FORMAT WITHOUT THE EXPRESS WRITTEN PERMISSION OF MIPS TECHNOLOGIES, INC.

MIPS Technologies reserves the right to change the information contained in this document to improve function, design or otherwise. MIPS Technologies does not assume any liability arising out of the application or use of this information, or of any error or omission in such information. Any warranties, whether express, statutory, implied or otherwise, including but not limited to the implied warranties of merchantability or fitness for a particular purpose, are excluded. Except as expressly provided in any written license agreement from MIPS Technologies or an authorized third party, the furnishing of this document does not give recipient any license to any intellectual property rights, including any patent rights, that cover the information in this document.

The information contained in this document shall not be exported, reexported, transferred, or released, directly or indirectly, in violation of the law of any country or international law, regulation, treaty, Executive Order, statute, amendments or supplements thereto. Should a conflict arise regarding the export, reexport, transfer, or release of the information contained in this document, the laws of the United States of America shall be the governing law.

The information contained in this document constitutes one or more of the following: commercial computer software, commercial computer software documentation or other commercial items. If the user of this information, or any related documentation of any kind, including related technical data or manuals, is an agency, department, or other entity of the United States government ("Government"), the use, duplication, reproduction, release, modification, disclosure, or transfer of this information, or any related documentation of any kind, is restricted in accordance with Federal Acquisition Regulation 12.212 for civilian agencies and Defense Federal Acquisition Regulation Supplement 227.7202 for military agencies. The use of this information by the Government is further restricted in accordance with the terms of the license agreement(s) and/or applicable contract terms and conditions covering this information from MIPS Technologies or an authorized third party.

MIPS, MIPS I, MIPS II, MIPS III, MIPS IV, MIPS V, MIPS-3D, MIPS16, MIPS16e, MIPS32, MIPS64, MIPS-Based, MIPSsim, MIPSpro, MIPS Technologies logo, MIPS-VERIFIED, MIPS-VERIFIED logo, 4K, 4Kc, 4Km, 4Kp, 4KE, 4KEc, 4KEm, 4KEp, 4KS, 4KSc, 4KSd, M4K, M14K, 5K, 5Kc, 5Kf, 24K, 24Kc, 24Kf, 24KE, 24KEc, 24KEf, 34K, 34Kc, 34Kf, 74K, 74Kf, 74Kc, 1004K, 1004Kc, 1004Kf, R3000, R4000, R5000, ASMACRO, Atlas, "At the core of the user experience.", BusBridge, Bus Navigator, CLAM, CorExtend, CoreFPGA, CoreLV, EC, FPGA View, FS2, FS2 FIRST SILICON SOLUTIONS logo, FS2 NAVIGATOR, HyperDebug, HyperJTAG, JALGO, Logic Navigator, Malta, MDMX, MED, MGB, microMIPS, OCI, PDtrace, the Pipeline, Pro Series, SEAD, SEAD-2, SmartMIPS, SOC-it, System Navigator, and YAMON are trademarks or registered trademarks of MIPS Technologies, Inc. in the United States and other countries.

All other trademarks referred to herein are the property of their respective owners.

Template: nW1.03, Built with tags: 2B

Copyright © 2010 MIPS Technologies Inc. All rights reserved.