CoreLink™ DMC-400 Dynamic Memory Controller

Revision: r1p2

Technical Reference Manual

ARM®

ARM DDI 0466E (ID120513)

CoreLink DMC-400 Dynamic Memory Controller
Technical Reference Manual

Release Information

The following changes have been made to this book.

Change history

Date	Issue	Confidentiality	Change
30 June 2011	A	Non-Confidential	First release for r0p0
05 April 2012	B	Non-Confidential	First release for r0p1
13 July 2012	C	Non-Confidential	First release for r1p0
18 January 2013	D	Non-Confidential	First release for r1p1
09 May 2013	E	Non-Confidential	First release for r1p2

Proprietary Notice

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by ARM and the party that ARM delivered this document to.

Product Status

The information in this document is final, that is for a developed product.

Web Address

`http://www.arm.com`

Contents
CoreLink DMC-400 Dynamic Memory Controller Technical Reference Manual

Preface
About this book vi
Feedback ix

Chapter 1 Introduction
1.1 About the DMC-400 1-2
1.2 Compliance 1-3
1.3 Features 1-4
1.4 Interfaces 1-5
1.5 Configurable options 1-6
1.6 Test features 1-7
1.7 Product documentation, design flow, and architecture 1-8
1.8 Product revisions 1-9

Chapter 2 Functional Description
2.1 DMC-400 interfaces 2-2
2.2 DMC-400 operation 2-6

Appendix A Signal Descriptions
A.1 System clock, reset and DFT signals A-2
A.2 ACE-Lite signals A-3
A.3 QVN signals A-6
A.4 APB signals A-7
A.5 LPI signals A-8
A.6 DFI signals A-9
A.7 Miscellaneous signals A-13

Appendix B Revisions

Preface

This preface introduces the *CoreLink DMC-400 Dynamic Memory Controller Technical Reference Manual*. It contains the following sections:

- *About this book* on page vi
- *Feedback* on page ix.

About this book

This *Technical Reference Manual* (TRM) is for the CoreLink DMC-400 *Dynamic Memory Controller* (DMC-400).

Product revision status

The r*n*p*n* identifier indicates the revision status of the product described in this book, where:

r*n* Identifies the major revision of the product.

p*n* Identifies the minor revision or modification status of the product.

Intended audience

This book is written for system designers, system integrators, and programmers who are designing or programming a *System-on-Chip* (SoC) device that uses the DMC-400. The DMC-400 provides an interface between the *Advanced Coherency Extensions* (ACE-Lite™) system bus and the external, off-chip, memory devices.

Using this book

This book is organized into the following chapters:

Chapter 1 *Introduction*

Read this for an introduction to the DMC-400 and its features.

Chapter 2 *Functional Description*

Read this for an overview of the major functional blocks and the operation of the DMC-400.

Appendix A *Signal Descriptions*

Read this for a description of the input and output signals.

Appendix B *Revisions*

Read this for a description of the technical changes between released issues of this book.

Glossary

The *ARM Glossary* is a list of terms used in ARM documentation, together with definitions for those terms. The *ARM Glossary* does not contain terms that are industry standard unless the ARM meaning differs from the generally accepted meaning.

See *ARM Glossary*, http://infocenter.arm.com/help/topic/com.arm.doc.aeg0014-/index.html.

Conventions

Conventions that this book can use are described in:

- *Typographical conventions* on page vii
- *Timing diagrams* on page vii
- *Signals* on page vii.

Typographical conventions

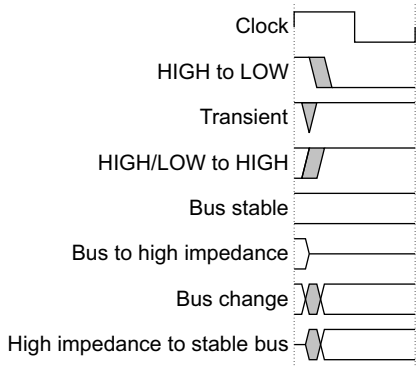
The following table describes the typographical conventions:

Style	Purpose
italic	Introduces special terminology, denotes cross-references, and citations.
bold	Highlights interface elements, such as menu names. Denotes signal names. Also used for terms in descriptive lists, where appropriate.
`monospace`	Denotes text that you can enter at the keyboard, such as commands, file and program names, and source code.
`mono`space	Denotes a permitted abbreviation for a command or option. You can enter the underlined text instead of the full command or option name.
`monospace italic`	Denotes arguments to monospace text where the argument is to be replaced by a specific value.
`monospace bold`	Denotes language keywords when used outside example code.
<and>	Encloses replaceable terms for assembler syntax where they appear in code or code fragments. For example: `MRC p15, 0 <Rd>, <CRn>, <CRm>, <Opcode_2>`
SMALL CAPITALS	Used in body text for a few terms that have specific technical meanings, that are defined in the *ARM glossary*. For example, IMPLEMENTATION DEFINED, IMPLEMENTATION SPECIFIC, UNKNOWN, and UNPREDICTABLE.

Timing diagrams

The figure named *Key to timing diagram conventions* explains the components used in timing diagrams. Variations, when they occur, have clear labels. You must not assume any timing information that is not explicit in the diagrams.

Shaded bus and signal areas are undefined, so the bus or signal can assume any value within the shaded area at that time. The actual level is unimportant and does not affect normal operation.



Key to timing diagram conventions

Signals

The signal conventions are:

Signal level The level of an asserted signal depends on whether the signal is active-HIGH or active-LOW. Asserted means:

- HIGH for active-HIGH signals
- LOW for active-LOW signals.

Lower-case n At the start or end of a signal name denotes an active-LOW signal.

Additional reading

This section lists publications by ARM and by third parties.

See Infocenter, http://infocenter.arm.com, for access to ARM documentation.

ARM publications

This book contains information that is specific to this product. See the following documents for other relevant information:

- *CoreLink DMC-400 Dynamic Memory Controller Technical Reference Manual Supplement* (ARM DSU 0016)
- *CoreLink DMC-400 Dynamic Memory Controller Implementation Guide* (ARM DII 0257)
- *CoreLink DMC-400 Dynamic Memory Controller Integration Manual* (ARM DII 0258)
- *CoreLink DMC-400 Dynamic Memory Controller Supplement to AMBA® Designer ADR-400 User Guide* (ARM DSU 0015)
- *AMBA 3 APB™ Protocol Specification* (ARM IHI 0024)
- *CoreLink QVN Protocol Specification* (ARM IHI 0063)
- *AMBA AXI and ACE™ Protocol Specification AXI3™, AXI4™, and AXI4-Lite™, ACE and ACE-Lite™* (ARM IHI 0022)
- *TrustZone® Address Space Controller Technical Reference Manual* (ARM DDI 0431).

Other publications

This section lists relevant documents published by third parties:

- *DDR PHY Interface (DFI) Specification*, version 2.1.1, http://www.ddr-phy.org
- *JEDEC STANDARD DDR3 SDRAM Specification, JESD79-3D*, http://www.jedec.org
- *JEDEC STANDARD DDR2 SDRAM Specification, JESD79-2E*, http://www.jedec.org
- *JEDEC STANDARD LPDDR2 SDRAM Specification, JESD209-2 LPDDR2*, http://www.jedec.org.

Feedback

ARM welcomes feedback on this product and its documentation.

Feedback on this product

If you have any comments or suggestions about this product, contact your supplier and give:

- The product name.
- The product revision or version.
- An explanation with as much information as you can provide. Include symptoms and diagnostic procedures if appropriate.

Feedback on content

If you have comments on content then send an e-mail to `errata@arm.com`. Give:

- the title
- the number, ARM DDI 0466E
- the page numbers to which your comments apply
- a concise explanation of your comments.

ARM also welcomes general suggestions for additions and improvements.

Note

ARM tests the PDF only in Adobe Acrobat and Acrobat Reader, and cannot guarantee the quality of the represented document when used with any other PDF reader.

Chapter 1
Introduction

This chapter introduces the DMC-400 and contains the following sections:

- *About the DMC-400* on page 1-2
- *Compliance* on page 1-3
- *Features* on page 1-4
- *Interfaces* on page 1-5
- *Configurable options* on page 1-6
- *Test features* on page 1-7
- *Product documentation, design flow, and architecture* on page 1-8
- *Product revisions* on page 1-9.

1.1 About the DMC-400

The DMC-400 is an *Advanced Microcontroller Bus Architecture* (AMBA) compliant *System-on-Chip* (SoC) peripheral developed, tested, and licensed by ARM. It is a high-performance, area-optimized, memory controller that is compatible with the AMBA ACE-Lite protocol. It supports the following memory devices:

- *Double Data Rate 2* (DDR2) *Synchronous Dynamic Random Access Memory* (SDRAM)
- *Low-Power Double Data Rate 2* (LPDDR2)-S2 SDRAM
- LPDDR2-S4 SDRAM
- *Double Data Rate 3* (DDR3) SDRAM
- Low-voltage DDR3 SDRAM.

Figure 1-1 shows an example system.

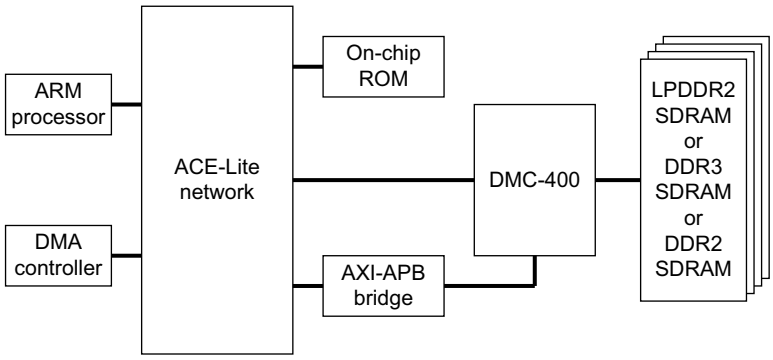


Figure 1-1 Example system

The DMC-400 enables data transfer between the SoC and the DRAM devices external to the chip. It connects to the on-chip system through one or more ACE-Lite interfaces. It connects to the DRAM devices through its memory interface block, and the DFI interface.

1.2 Compliance

The DMC-400 is compliant with the:

- ACE-Lite protocol
- AMBA3 APB protocol
- JEDEC LPDDR2 JESD209-2 standard
- JEDEC DDR3 JESD79-3D standard
- JEDEC DDR2 JESD79-2E standard
- DFI v2.1.1 specification.

1.3 Features

The DMC-400 supports the following features:

- soft macrocell available in Verilog
- configurable hardware to support area and performance optimization
- multiple ACE-Lite system interfaces
- multiple outstanding transactions
- system *Quality of Service* (QoS) and request arbitration, to achieve low-latency transfers and the optimal use of memory bandwidth
- protection against transaction barriers
- configurable bit width for the ACE-Lite system interfaces
- separate read acceptance capability and write acceptance capability
- AMBA ACE-Lite exclusive access transfers
- synchronous n:1 clocking between ACE-Lite and APB interfaces
- connection to different types of memory device. See *About the DMC-400* on page 1-2.

  Note

  You can configure only one type of memory device at any one time.

- multiple memory interfaces
- optimal use of the external memory bus
- DRAM power saving
- programmable external memory width
- a configurable number of memory chip-selects for each memory interface
- memory *Error-Correcting Code* (ECC) for 64-bit and 32-bit SDRAM
- interfaces to the *Physical Layer* (PHY) using the DFI2.1.1interface.

1.4 Interfaces

The DMC-400 provides the following external interfaces:

- APB3 interface, to configure and control the DMC-400
- ACE-Lite slave interface, to transfer memory data to or from an AMBA master
- *QoS Virtual Network* (QVN) extension to the ACE-Lite interface
- DFI-compatible PHY interface, to transfer data to or from the external memory devices
- clock and reset interface
- debug and profile interface
- low-power control interface
- training interface.

See *DMC-400 interfaces* on page 2-2.

1.5 Configurable options

The DMC-400 has the following configurable options:

- number of ACE-Lite system interfaces
- for all ACE-Lite system interfaces:
  - data bus width
  - address bus width
  - ID bus width
  - read acceptance capability
  - read hazard acceptance capability
  - read hazard buffer implementation
  - include or exclude QVN.
- number of memory channels
- for all memory interfaces:
  - read queue depth
  - write buffer depth
  - write buffer implementation
  - number of memory chips
  - DFI data width
  - maximum DFI burst length
  - enable or disable memory *Single-Error Correction and Double-Error Detection* (SECDED).

See *Configurability* on page 2-8.

1.6 Test features

The DMC-400 provides:

- integration test logic for integration testing
- a debug and profile interface to enable you to monitor transaction events.

1.7 Product documentation, design flow, and architecture

The DMC-400 documentation is as follows:

Technical Reference Manual and Technical Reference Manual Supplement

The TRM and TRM Supplement describe the functionality and the effects of functional options on the behavior of the DMC-400. These are required at all stages of the design flow. The choices that you make in the design flow can mean that some behavior that the TRM describes is not relevant. If you are programming the DMC-400 then contact:

- the implementer to determine:
  - the build configuration of the implementation
  - what integration, if any, was performed before implementing the DMC-400.
- the integrator to determine the pin configuration of the device that you are using.

The TRM Supplement is a confidential book that is only available to licensees.

Implementation Guide

The *Implementation Guide* (IG) describes:

- the available build configuration options and related issues if you select them
- how to configure the *Register Transfer Level* (RTL) with the build configuration options
- the processes to sign off the configured design.

The ARM product deliverables include reference scripts and information about how you use them to implement your design.

The IG is a confidential book that is only available to licensees.

Integration Manual

The *Integration Manual* (IM) describes how to integrate the DMC-400 into a SoC. It includes a description of the signals that the integrator must tie-off to configure the macrocell for the required integration. Some of the integration is affected by the configuration options you use when you implement the DMC-400.

The IM is a confidential book that is only available to licensees.

Supplement to AMBA Designer ADR-400 User Guide

The supplement describes how to use AMBA Designer to:

- configure and build a DMC-400
- generate its RTL
- reconfigure the DMC-400.

1.8 Product revisions

This section describes the differences in functionality between product revisions of the DMC-400:

r0p0 First release.

r0p0 - r0p1 No differences in functionality.

r0p1 - r1p0 Addition of the QVN.
ECC support for 32-bit SDRAM.

r1p0 - r1p1 Extended the range of the `t_rdlvl_rr` and `t_wrlvl_ww` parameters.

r1p1 - r1p2 Addition of clr_ex_mon pin.

Chapter 2
Functional Description

This chapter describes how the DMC-400 operates. It contains the following sections:

- *DMC-400 interfaces* on page 2-2
- *DMC-400 operation* on page 2-6.

2.1 DMC-400 interfaces

Figure 2-1 shows the interfaces of the DMC-400.

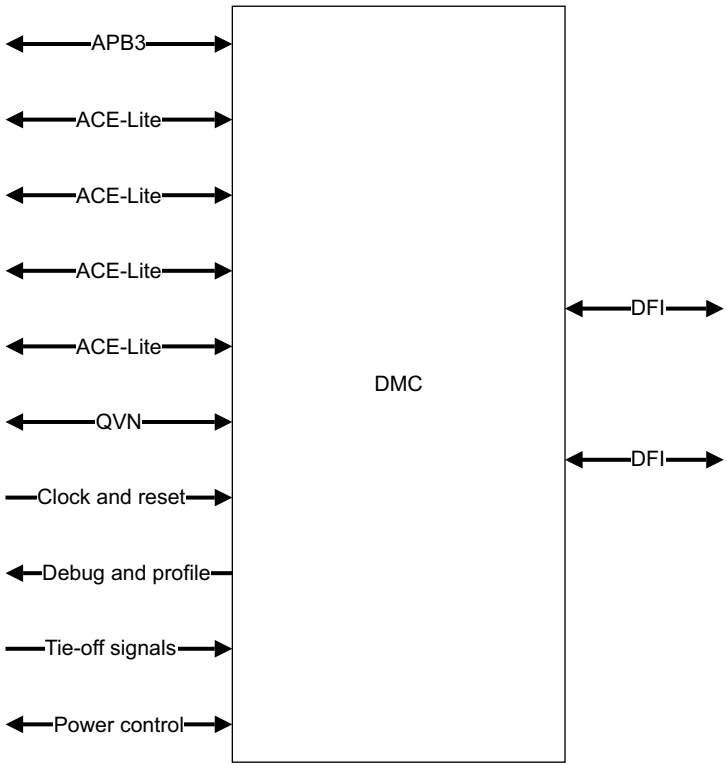


Figure 2-1 DMC-400 interfaces

The DMC-400 provides the following interfaces:

- *APB3 interface*
- *ACE-Lite interface* on page 2-3
- *PHY interface* on page 2-3
- *Clocks and resets* on page 2-4
- *Debug and profile interface* on page 2-5
- *Hardware power control interface(s)* on page 2-5.

2.1.1 APB3 interface

You can access the configuration and control registers of the DMC-400 through its APB3 interface. Use these registers for:

- initialization
- configuration
- control
- power-management.

The APB3 interface address map has a range of 4KB. You can use it to access the DMC-400 control registers. The address map is split into regions. Table 2-1 shows the function of each region.

Table 2-1 APB3 register regions

Region base address	Region function
0x000	Memory controller configuration
0x100	Memory channel configuration
0x200	PHY control
0x300	Timing configuration
0x400	User configuration
0xE00	Integration test
0xF00	Peripheral ID

See the *AMBA 3 APB Protocol Specification*. Also see Appendix A *Signal Descriptions*.

Note

The DMC-400 responds with a **PSLVERR** error response when the APB master does any of the following:

- requests a read access to either a RO register or a RW register when the DMC-400 is in a state where this is not allowed
- requests a write access to either a WO register or a RW register when the DMC-400 is in a state where this is not allowed
- requests the DMC to perform an illegal state transition.

2.1.2 ACE-Lite interface

The DMC-400 uses the standard ACE-Lite interface. See the *AMBA AXI and ACE Protocol Specification AXI3, AXI4, and AXI4-Lite, ACE and ACE-Lite*. Also see Appendix A *Signal Descriptions*.

The signal names for each system interface include the _<*index*> suffix, where _<*index*> is the decimal system address. For a DMC-400 with a single system interface the suffix is _0.

2.1.3 QVN

You can configure the DMC-400 to exclude or include the RTL for QVN functionality.

If you include the QVN RTL the DMC-400 supports up to four QVNs, QVN0 to QVN3.

2.1.4 PHY interface

The PHY interface connects the DMC-400 to the memory devices. The interface conforms to the DFI 2.1.1 interface standard. See the *DDR PHY Interface Specification* version 2.1.1.

Note

The DMC-400 does not support the following DFI signals:

- **dfi_ctrlupd_req**
- **dfi_ctrlupd_ack**.

However, you can connect to them with the DMC-400 **User** signals

The signal names for each memory interface include the _<*index*> suffix, where _<*index*> is the decimal channel address. For a DMC-400 with a single memory interface the suffix is _0.

The DMC-400 supports the Update Request handshake that the DFI PHY initiates. The PHY, or an agent working on behalf of the PHY, can request an update through the DFI interface by asserting the **dfi_phyupd_req** signal. The DMC-400 supports the low-power interface that the DFI specification defines. The DMC-400 also supports the training interface that the DFI specification defines. See *Training mode* on page 2-9.

2.1.5 Clocks and resets

Table 2-2 shows the clock and reset signals of the DMC-400 when you configure it to have the maximum number of system interfaces, and the maximum number of memory interfaces.

Table 2-2 Clock and reset signals

Signal name	Width	Direction	Description
dmc_clk	1	Input	Clock for the DMC-400
pclk	1	Input	Clock for the APB interface clock domain
dmc_resetn	1	Input	Reset for the DMC-400
presetn	1	Input	Reset for APB interface domain

Clocks

The DMC-400 clock and the APB interface clock must be phase-aligned and synchronous. The DMC-400 clock must run at a frequency higher than, or equal to the frequency of the APB clock. If you want the APB interface to run at a slower rate than **dmc_clk**, you can reduce the **pclk** rate using **pclken**.

Note

You must only change the frequency of the DMC-400 clock when both of the following conditions are true:

- all memory chips are in self-refresh state or in *Deep Power Down* (DPD) state
- the DMC-400 status registers indicate that the memory interface is in the quiescent state.

Reset

Use the **dmc_resetn** signal to reset the DMC-400. You must assert it for longer than two **dmc_clk** clock periods. You can assert **dmc_resetn** asynchronously to **dmc_clk**. You must deassert **dmc_resetn** synchronously with the rising edge of **dmc_clk**.

Use the **presetn** signal to reset the APB interface of the DMC-400. You must assert the **presetn** signal for longer than two **pclk** clock periods. You can assert **presetn** asynchronously to **pclk**. You must deassert **presetn** synchronously with the rising edge of **pclk.**

Note

To assert any DMC-400 reset signal you must set it LOW.

2.1.6 Debug and profile interface

The debug and profile interface enables you to monitor events that occur inside the DMC-400. You can use these events to debug and to monitor the performance of the DMC-400.

2.1.7 Hardware power control interface(s)

The DMC-400 supports the standard AXI low-power interface. See the *AMBA AXI and ACE™ Protocol Specification AXI3™, AXI4™, and AXI4-Lite™, ACE and ACE-Lite™*.

2.2 DMC-400 operation

This section describes:

- *Memory initialization*
- *Memory support*
- *Memory access*
- *Power control* on page 2-7
- *Configurability* on page 2-8
- *Security* on page 2-9
- *Training mode* on page 2-9.

2.2.1 Memory initialization

You can initialize the DRAM memory devices through the APB3 interface.

2.2.2 Memory support

The DMC-400 supports multiple memory channels. Each channel can support the following memory types:

- LPDDR2-S2
- LPDDR2-S4
- DDR3
- DDR2
- low-voltage DDR3.

The DMC-400 is a configurable peripheral. It enables you to:

- manage the distribution of traffic through several memory interfaces
- minimize the overhead and cyclic dependency between channels
- use its flexibility to create an efficient memory subsystem
- manage the operation and maintenance of multi-port memory devices.

You can use each chip-select signal to select a group of devices, that are bound, to form a single data interface.

2.2.3 Memory access

The DMC-400 connects to the system through one or more ACE-Lite interfaces. The DMC-400 supports all the features of the ACE-Lite specification. The features include:

- barrier transactions
- cache maintenance operations
- QoS.

Addressing

The DMC-400 maps the ACE-Lite system address to a physical DRAM address. The DRAM address consists of a bank number, a row number, and a column number. The DMC-400 supports configurable and programmable controls for mapping the ACE-Lite system address to the physical DRAM address.

Note

DRAM devices have page sizes of 1KB, 2KB, or 4KB. The ACE-Lite protocol does not permit a transaction to cross a 4KB boundary.

The DMC-400 provides programmable controls to map the system address onto the physical chips and channels.

Burst control

The DMC-400 formats all ACE-Lite transactions into memory bursts.

The DMC-400 supports sequential burst addressing and sequential wrapped burst addressing of the DRAM.

Responses

The DMC-400 always responds to an ACE-Lite access with an OKAY, EXOKAY, or a DECERR response. It never issues a SLVERR response.

Exclusive access

The DMC-400 provides eight exclusive access monitors for each memory interface. It tracks exclusive accesses at a granularity of the configured DMC-400 burst.

Barriers and cache maintenance operations

The DMC-400 supports barriers and cache maintenance operations to guarantee the correct ordering of memory accesses.

QoS signals

To determine the quality of service value that the system requests for its memory access, the DMC-400 uses the **arqos** and **awqos** signals of the ACE-Lite address.

Error-correcting code

The DMC-400 provides SECDED code protection for 32-bit and 64-bit SDRAM interface accesses. It combines the codes with the data of write transactions, and checks the codes for data read transactions.

The DMC-400 uses a programmable interrupt to report the data errors that it corrects. You can read the information of a transaction that the DMC-400 corrects through an APB register in the DMC-400. You can clear the interrupt through an APB register.

The DMC-400 reports uncorrected errors through another programmable interrupt. You can clear the interrupt through an APB register. You can read the information of the transaction through an APB register.

If either interrupt overflows the interrupt system, the DMC-400 triggers another interrupt to notify you that this has happened. You can clear this interrupt through an APB register.

2.2.4 Power control

The DMC-400 and the DRAM have several operating states. Each operating state consumes different amounts of power. The DMC-400 enables you to control its operating state and the operating state of the DRAM. Therefore you can reduce the power consumption of the DMC-400, the PHY, the interface, and the DRAMs by using the following methods:

- *Hardware-controlled power management* on page 2-8
- *Clock frequency adjustment* on page 2-8
- *DMC-400 controlled power management* on page 2-8.

Hardware-controlled power management

The DMC-400 provides hardware-controlled power-management of each memory interface through the PHY low-power request interface. This interface enables power-management hardware to remove power from idle memory interfaces and associated logic.

Clock frequency adjustment

You can change the DMC-400 clock frequency. See *Clocks* on page 2-4.

DMC-400 controlled power management

The DMC-400 supports several features to manage the power dissipation of each memory device in the memory subsystem. These programmable features enable the DMC-400 to automatically manage the entry and exit of the DRAM low-power states when the DRAM is idle.

2.2.5 Configurability

This section describes the configuration parameters that you can set for the DMC-400.

Note

If the DMC-400 cannot map an ACE-Lite burst to a single DMC-400 burst, it splits the ACE-Lite burst into multiple DMC-400 bursts that in turn consume multiple slots. Therefore the terms burst and hazards refer to DMC-400 bursts and DMC-400 hazards, and not ACE-Lite bursts and ACE-Lite hazards.

System configuration parameters

Table 2-3 shows the system configuration parameters that you can set.

Table 2-3 System configuration parameters

Description	Range of values	Default value
The bit-width of the system ID bus	4-24	8
The bit-width of the system interface address bus	32, 40, 64	32
The bit-width of the system interface data bus	64, 128, 256	64
The number of read bursts that each system interface can accept	16, 32, 64	32
The number of read ID hazards that each system interface can issue before stalling more requests	8, 16	8
How you implement the read hazard buffer	RAM, synthesized registers	synthesized registers
Whether to include logic to support Virtual Networks	False, True	False

Memory configuration parameters

Table 2-4 shows the memory configuration parameters that you can set.

Table 2-4 Memory configuration parameters

Description	Range of values	Default value
The bit-width of the DFI data bus	32, 64, 128	64
The number of chip-selects on each memory interface	1, 2	1
The number of memory bursts the write buffer can hold	16, 32, 64	32
How you implement the write buffer	RAM, synthesized registers	synthesized registers
The number of memory bursts the read queue can hold	16, 32, 64	32
The number of DFI data beats in each queued burst	4, 8	8
Whether to include logic to support Single Error Correct, Double Error Detect ECC protection on the external memory. ECC is only supported for DFI widths of 128 bits or 64 bits	False, True	False

File selection configurable parameters

You must select the following configurable parameters by choosing the correct Verilog file:

- number of system interfaces, the permitted values are 1, 2, or 4
- number of memory channels, the permitted values are 1 or 2.

2.2.6 Security

The DMC-400 can protect areas of memory from unwanted memory map aliasing but it does not support any type of security control.

Note

- To define regions of either the ACE-Lite address map or the APB address map as secure, ARM recommends that you use the *TrustZone address space Controller* (TZC).
- If you define any memory region as secure, you must also define the APB interface as secure.

2.2.7 Training mode

The DFI specification defines evaluation modes for all of the following:

- write leveling training
- data eye training
- gate training.

Table 2-5 shows the evaluation modes that the DMC-400 supports.

Table 2-5 DFI evaluation modes

Mode	Description
No training in the PHY	-
PHY evaluation	The DMC-400 controls the training logic. The PHY determines the correct delay values.
PHY independent	The PHY controls all training operations.
Memory Controller (MC) evaluation[a]	-

a. The DMC-400 does not support this evaluation mode.

Appendix A
Signal Descriptions

This appendix describes the signals that the DMC-400 provides. It contains the following sections:

- *System clock, reset and DFT signals* on page A-2
- *ACE-Lite signals* on page A-3
- *QVN signals* on page A-6
- *APB signals* on page A-7
- *LPI signals* on page A-8
- *DFI signals* on page A-9
- *Miscellaneous signals* on page A-13.

A.1 System clock, reset and DFT signals

Table A-1 shows the clock, reset and DFT signals for the DMC-400. The DMC-400 also uses clock and reset signals for the programmers view. See *APB signals* on page A-7.

Table A-1 System clock, reset and DFT signals

Signal	Type	Description
dmc_clk	Input	Main clock source for DMC-400
dmc_resetn	Input	Main reset for DMC-400
dftse	Input	DFT scan enable
rst_bypass	Input	DFT reset bypass

A.2 ACE-Lite signals

Table A-2 shows the ACE-Lite signals. See the *AMBA AXI and ACE™ Protocol Specification AXI3™, AXI4™, and AXI4-Lite™, ACE and ACE-Lite™*.

Table A-2 ACE-Lite interface signals

Signal	Type	Description
awid[a]	Input	Write address ID. This signal is the identification tag for the write address group of signals.
awaddr[a]	Input	Write address. The write address bus defines the address of the first transfer of a write burst transaction. The associated control signals determine the addresses of the remaining transfers in the burst.
awregion	Input	Region identifier. Enables a single physical interface on a slave to be used for multiple logical interfaces.
awdomain	Input	This signal indicates the shareability domain of a write transaction. The DMC-400 does not functionally use this signal.
awsnoop	Input	This signal indicates the transaction type for shareable write transactions.
awbar	Input	This signal indicates a write barrier transaction.
awlen	Input	Burst length. The burst length gives the number of transfers in a burst. This information determines the number of data transfers associated with the address.
awsize	Input	Burst size. This signal indicates the size of each transfer in the burst. Byte lane strobes indicate which byte lanes to update.
awburst	Input	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
awlock	Input	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
awcache	Input	Cache type. This signal indicates the bufferable, cacheable, write-through, write-back, and allocate attributes of the transaction.
awprot	Input	Protection type. This signal indicates the normal, privileged, or secure protection level of the transaction and whether the transaction is a data access or an instruction access.
awqos	Input	Quality of Service. This signal is used to provide a QoS identifier for each write transaction.
awvalid	Input	Write address valid. This signal indicates that valid write address and control information are available.
awready	Output	Write address ready. This signal indicates that the slave is ready to accept an address and associated control signals.
wdata[a]	Input	Write data.
wstrb[a]	Input	Write strobes. This signal indicates which byte lanes to update in memory.
wlast	Input	Write last. This signal indicates the last transfer in a write burst.
wvalid	Input	Write valid. This signal indicates that valid write data and strobes are available.
wready	Output	Write ready. This signal indicates that the slave can accept the write data.
bid[a]	Output	Response ID. The identification tag of the write response. The **bid** value must match the **awid** value of the write transaction to which the slave is responding.

Table A-2 ACE-Lite interface signals (continued)

Signal	Type	Description
bresp	Output	Write response. This signal indicates the status of the write transaction. The permitted responses are OKAY, EXOKAY, SLVERR, and DECERR.
bvalid	Output	Write response valid. This signal indicates that a valid write response is available.
bready	Input	Response ready. This signal indicates that the master can accept the response information.
arid[a]	Input	Read address ID. This signal is the identification tag for the read address group of signals.
araddr[a]	Input	Read address. The read address bus gives the initial address of a read burst transaction. Only the start address of the burst is provided and the control signals that are issued alongside the address describe how the address is calculated for the remaining transfers in the burst.
arregion	Input	Region decode signal. The signal is provided alongside the transaction address. The signal enables a single physical interface on a slave to be used for multiple logical interfaces that reside in different locations in the system address map.
ardomain	Input	This signal indicates the shareability domain of a read transaction. The DMC-400 does not use this signal functionally.
arsnoop	Input	This signal indicates the transaction type for shareable read transactions.
arbar	Input	This signal indicates a read barrier transaction.
arlen	Input	Burst length. The burst length gives the exact number of transfers in a burst. This information determines the number of data transfers associated with the address.
arsize	Input	Burst size. This signal indicates the size of each transfer in the burst.
arburst	Input	Burst type. The burst type, coupled with the size information, details how the address for each transfer within the burst is calculated.
arlock	Input	Lock type. This signal provides additional information about the atomic characteristics of the transfer.
arcache	Input	Cache type. This signal provides additional information about the cacheable characteristics of the transfer.
arprot	Input	Protection type. This signal provides protection unit information for the transaction.
arqos	Input	Quality of Service. Signal used to provide a QoS identifier for each read transaction.
arvalid	Input	Read address valid. When HIGH this signal indicates that the read address and control information is valid and stable until the **arready** address acknowledge signal is HIGH.
arready	Output	Read address ready. This signal indicates that the slave is ready to accept an address and associated control signals.
rid[a]	Output	Read ID tag. This signal is the ID tag of the read data group of signals. The **rid** value is generated by the slave and must match the **arid** value of the read transaction to which it is responding.
rdata[a]	Output	Read data.
rresp	Output	Read response. This signal indicates the status of the read transfer. The permitted responses are OKAY, EXOKAY, SLVERR, and DECERR.

Table A-2 ACE-Lite interface signals (continued)

Signal	Type	Description
rlast	Output	Read last. This signal indicates the last transfer in a read burst.
rvalid	Output	Read valid. This signal indicates that the required read data is available and the read transfer can complete.
rready	Input	Read ready. This signal indicates that the master can accept the read data and response information.

a. Signal width is configuration dependent.

A.3 QVN signals

Table A-3 and Table A-4 show the QVN signals.

Table A-3 VN interface signals

Signal	Type	Description
vawvalid_ vn<x>[a]	Input	Token request valid. This signal indicates that the master requests a token.
vawready_ vn<x>	Output	Token request accepted. This signal indicates that the slave is ready to accept an address and associated control signals.
vawqos_ vn<x>	Input	Quality of service value. Non-standard AXI3 signal.
vwvalid_ vn<x>	Input	Token request valid. This signal indicates that the master requests a token.
vwready_ vn<x>	Output	Token request accepted. This signal indicates that the slave is ready to accept a write data transfer and associated control signals.
varvalid_ vn<x>	Input	Token request valid. This signal indicates that the master requests a token.
varready_ vn<x>	Output	Token request accepted. This signal indicates that the slave is ready to accept an address and associated control signals.
varqos_ vn<x>	Input	Quality of service value. Non-standard AXI3 signal.
qvn_prealloc_w	Input	Write address token pre-allocation. This bus is 4 bits wide. Each bit specifies the pre-allocation of the corresponding virtual network.
qvn_prealloc_r	Input	Read address token pre-allocation. This bus is 4 bits wide. Each bit specifies the pre-allocation of the corresponding virtual network.

a. <x> is the virtual network number for the signals in this table.

A.3.1 AXI QVN signals

Table A-4 AXI QVN signals

Signal	Type	Description
awvnet	Input	AW channel virtual network ID
wvnet	Input	W channel virtual network ID
arvnet	Input	AR channel virtual network ID

A.4 APB signals

Table A-5 shows the APB signals. See the *AMBA 3 APB™ Protocol Specification*.

Table A-5 APB interface signals

Signal	Type	Description
pclk	Input	Clock. The rising edge of **pclk** synchronizes all transfers on the APB.
presetn	Input	Reset. The APB reset signal is active LOW. This signal is normally connected directly to the system bus reset signal.
paddr	Input	Address. This is the APB address bus. It can be up to 32-bits wide and is driven by the peripheral bus bridge unit.
psel	Input	Select. The APB bridge unit generates this signal to each peripheral bus slave. It indicates that the slave device is selected and that a data transfer is required.There is a **psel** signal for each slave.
penable	Input	Enable. This signal indicates the second and subsequent cycles of an APB transfer.
pwrite	Input	Direction. This signal indicates an APB write access when HIGH and an APB read access when LOW.
prdata	Output	Read Data. The selected slave drives this bus during read cycles when **pwrite** is LOW. This bus can be up to 32-bits wide.
pwdata	Input	Write data. This bus is driven by the peripheral bus bridge unit during write cycles when **pwrite** is HIGH. This bus can be up to 32-bits wide.
pready	Output	Ready. The slave uses this signal to extend an APB transfer.
pslverr	Output	This signal indicates a transfer failure. APB peripherals are not required to support the **pslverr** pin. This is true for both existing and new APB peripheral designs. When a peripheral does not include this pin, the appropriate input to the APB bridge is tied LOW.
pclken	Input	This is an optional enable signal for **pclk** domain.

A.5 LPI signals

Table A-6 shows the *Low Power Interface* (LPI) signals. See the *AMBA AXI and ACE™ Protocol Specification AXI3™, AXI4™, and AXI4-Lite™, ACE and ACE-Lite™*.

Table A-6 LPI interface signals

Signal	Type	Description
csysreq	Input	System low-power request. The system clock controller uses this signal to request the DMC-400 to enter a low-power state.
csysack	Output	Low-power request acknowledgement. The DMC-400 uses this signal to acknowledge a system low-power request.
cactive	Output	Clock active. The DMC-400 uses this signal to indicate that it requires a clock signal: **1** Peripheral clock required. **0** Peripheral clock not required.

A.6 DFI signals

Table A-7 shows the DFI signals. See the *DDR PHY Interface (DFI) Specification*.

Table A-7 DFI interface signals

Signal	Type	Description
dfi_rddata_valid[a]	Input	Read data valid indicator. The PHY asserts the **dfi_rddata_valid** signal with the read data for the number of cycles that it sends the read data. The timing is the same as for the **dfi_rddata** bus.
dfi_rddata[a]	Input	Read data bus. The DMC-400 expects to receive read data from the PHY within `tphy_rdlat` cycles after it asserts the **dfi_rddata_en** signal.
dfi_rdecc	Input	Read data ECC code bus. The timing is the same as for the **dfi_rddata** bus.
dfi_rddata_en[a]	Output	Read data enable. The DMC-400 must assert the **dfi_rddata_en** signal `trddata_en` cycles after it asserts the `read` command on the DFI control interface. It must remain valid for the duration of contiguous read data on the **dfi_rddata** bus.
dfi_rdecc_en	Output	Read data ECC code enable. The timing is the same as the **dfi_rddata_en** bus.
dfi_ras_n	Output	DFI row address strobe bus. These signals define the RAS information for all control commands to the DRAM devices.
dfi_cas_n	Output	DFI column address strobe bus. These signals define the CAS information for all control commands to the DRAM devices.
dfi_we_n	Output	DFI write enable bus. These signals define the WEN information for all control commands to the DRAM devices.
dfi_cke[a]	Output	DFI clock enable bus. These signals define the CKE information for all control commands to the DRAM devices.
dfi_cs_n	Output	DFI chip-select bus. These signals define the chip-select information for all control commands to the DRAM devices.
dfi_wrdata_cs_n	Output	PHY write training chip-select. Indicates which DRAM chip is being trained.
dfi_rddata_cs_n	Output	PHY read training chip-select. Indicates which DRAM chip is being trained.
dfi_cmd_addr	Output	DFI command/address bus. These signals define the CA information for the DRAM memory devices with multiplexed CA signalling, for example LPDDR2.
dfi_address	Output	DFI address bus. These signals define the address information for all control commands to the DRAM memory devices.
dfi_bank	Output	DFI bank bus. These signals define the bank information for all control commands to the DRAM devices.
dfi_odt[a]	Output	DFI on-die termination control bus. These signals define the ODT information for all control commands to the DRAM devices.
dfi_wrdata[a]	Output	Write data bus. The write data stream must begin `tphy_wrdata` cycles after the DMC-400 asserts the **dfi_wrdata_en** signal and continue for the number of cycles that the DMC-400 keeps the **dfi_wrdata_en** signal asserted. If the PHY requires notification of pending write data sooner, you can adjust the `tphy_wrdata` timing parameter to a higher value.
dfi_wrdata_mask[a]	Output	Write-data byte mask. The timing is the same as for the dfi_wrdata bus. For example, the **dfi_wrdata_mask [0]** signal defines masking for the **dfi_wrdata [7:0]** signals, the **dfi_wrdata_mask [1]** signal defines masking for the **dfi_wrdata [15:8]** signals. If the dfi_wrdata bus is not a multiple of 8 bits, then the most significant bit of the **dfi_wrdata_mask** signal corresponds to the most significant partial byte of data.

Table A-7 DFI interface signals (continued)

Signal	Type	Description
dfi_wrdata_en[a]	Output	Write data and data mask valid. These signals must be asserted `tphy_wrdata` cycles before the DMC-400 puts the data and data mask on the DFI interface. If the PHY requires notification of pending write data sooner, you can adjust the `tphy_wrdata` timing parameter to a higher value. The DMC-400 must send the **dfi_wrdata_en** signal `tphy_wrlat` cycles after the write command. When the DMC-400 asserts the **dfi_wrdata_en** signal, it must keep it asserted for the number of contiguous cycles that the write-data passes through the DFI write-data interface. A DFI term defines the width of the **dfi_wrdata_en** signal. There must be a single **dfi_wrdata_en** bit for each slice of memory data.
dfi_wrecc	Output	Write data ECC code bus. It has the same timing as the **dfi_wrdata** signal.
dfi_wrecc_mask	Output	Write data ECC code byte mask. It has the same timing as the **dfi_wrdata_mask** signal.
dfi_wrecc_en	Output	Write data ECC code and mask valid. It has the same timing as the **dfi_wrdata_en** signal.
dfi_dram_clk_disable[a]	Output	DRAM clock disable. When it is active, it indicates to the PHY that the clocks to the DRAM devices must be disabled so that the clock signals hold a constant value. When the **dfi_dram_clk_disable** signal is inactive, the DRAMs must be clocked normally.
dfi_phyupd_req	Input	PHY-initiated update request. The PHY uses the **dfi_phyupd_req** signal for a PHY-initiated update. It indicates that the PHY requires the DFI not to send control, read, or write commands or data for a specified period of time. The **dfi_phyupd_req** signal must remain asserted until both the DMC-400 acknowledges the request by asserting the **dfi_phyupd_ack** signal, and the update is complete. The DMC-400 must acknowledge this request.
dfi_phyupd_ack	Output	PHY-initiated update acknowledge. The DMC-400 uses the **dfi_phyupd_ack** signal for a PHY-initiated update. It indicates that the DFI is idle and remains so until the PHY deasserts the **dfi_phyupd_req** signal. While this signal is asserted, the DFI bus must remain idle except for transactions specifically associated with the update process.
dfi_phyupd_type	Input	PHY-initiated update select. The **dfi_phyupd_type** signal indicates which one of the 4 types of PHY update times is being requested by the **dfi_phyupd_req** signal.
dfi_clp_req	Output	Low-power opportunity request. The DMC-400 uses the **dfi_clp_req** signal to inform an ARM PHY that it can switch to a low-power state.
dfi_clp_ack	Input	Low-power acknowledge. The PHY asserts the **dfi_clp_ack** signal to acknowledge the DMC-400 low-power opportunity request. The PHY is not required to acknowledge this request. This signal corresponds to command signals and associated logic.
dfi_clp_wakeup	Output	Low-power wakeup time. The **dfi_clp_wakeup** signal indicates which one of the 16 wakeup times the MC is requesting for the PHY. This signal corresponds to command signals and associated logic.
dfi_rdlp_req	Output	Low-power opportunity request. The DMC-400 uses the **dfi_rdlp_req** signal to inform the PHY of an opportunity to switch to a low-power state. This signal corresponds to read data signals and associated logic.
dfi_rdlp_ack	Input	Low-power acknowledge. The PHY asserts the **dfi_rdlp_ack** signal to acknowledge the DMC-400 low-power opportunity request. The PHY is not required to acknowledge this request. This signal corresponds to read data signals and associated logic.
dfi_rdlp_wakeup	Output	Low-power wakeup time. The **dfi_rdlp_wakeup** signal indicates which one of the 16 wakeup times the MC is requesting for the PHY. This signal corresponds to read data signals and associated logic.
dfi_wrlp_req	Output	Low-power opportunity request. The DMC-400 uses the **dfi_wrlp_req** signal to inform the PHY of an opportunity to switch to a low-power state. This signal corresponds to write data signals and associated logic.

Table A-7 DFI interface signals (continued)

Signal	Type	Description
dfi_wrlp_ack	Input	Low-power acknowledge. The the PHY asserts the **dfi_wrlp_ack** signal to acknowledge the DMC-400 low-power opportunity request. The PHY is not required to acknowledge this request. This signal corresponds to write data signals and associated logic.
dfi_wrlp_wakeup	Output	Low-power wakeup time. The **dfi_wrlp_wakeup** signal indicates which one of the 16 wakeup times the DMC-400 is requesting for the PHY. This signal corresponds to write data signals and associated logic.
dfi_rdlvl_req	Input	PHY-initiated read data eye training request.
dfi_rdlvl_cs_n	Input	PHY-initiated read training chip-select. This indicates which chip is requesting training.
dfi_rdlvl_en	Output	PHY data eye training logic enable. If the PHY initiates the training request, see the **dfi_rdlvl_req** signal, then the DMC-400 can use **dfi_rdlvl_en** to acknowledge the training request: **1** Training logic enabled. **0** Normal operation. The DMC-400 asserts this signal to trigger read training.
dfi_rdlvl_gate_req	Input	PHY-initiated read gate training request.
dfi_rdlvl_gate_en	Output	PHY gate training logic enable. If the PHY initiates the training request, see the **dfi_rdlvl_gate_req** signal, the DMC-400 uses **dfi_rdlvl_gate_en** to acknowledge that request: **1** Training logic enabled. **0** Normal operation. The DMC-400 asserts this signal to initiate read training.
dfi_rdlvl_resp	Input	Read training response. The response definition depends on the mode of operation and the memory type for the system: **PHY Evaluation mode for DDR3 memory systems** The response indicates that the PHY has completed read leveling and centered the DQS relative to the data or placed the gate within the DQS preamble. **PHY Evaluation mode for LPDDR2 memory systems** The response indicates that the PHY has completed data eye training or gate training and centered the DQS relative to the data or placed the gate within the DQS preamble.
dfi_wrlvl_req	Input	PHY-initiated write training request.
dfi_wrlvl_cs_n	Input	PHY-initiated write training chip-select. This indicates which chip is requesting training.
dfi_wrlvl_en	Output	PHY write training logic enable. If the PHY initiates the training request with the **dfi_wrlvl_req** signal, then the DMC-400 uses **dfi_wrlvl_en** to acknowledge that request: **1** Training logic enabled. **0** Normal operation. The DMC-400 asserts this signal to initiate data eye training.
dfi_wrlvl_strobe	Output	Write training strobe. This triggers the PHY write leveling strobe.
dfi_wrlvl_resp	Input	Write training response for PHY Evaluation mode. The PHY asserts this signal to indicate that it has completed write leveling and aligned the DQS relative to the memory clock. The DMC-400 uses this value to determine how to adjust the delay value.
dfi_ref_en	Input	Configuration tie-off pin for the refresh during training feature: **1** Refresh during training enabled. **0** Normal operation.

a. Signal width is configuration dependent.

A.7 Miscellaneous signals

There are a large number of ports on DMC-400 that are not contained in IP-XACT busdefs. You must stitch them with specific connections. These signals are subgrouped into related signal tables. Table A-8, to Table A-11 show the miscellaneous signals.

A.7.1 ECC interrupt signals

Table A-8 ECC interrupt signals

Signal	Type	Description
ecc_int	Output	Combined ECC interrupt
ecc_sec_int	Output	Single error corrected interrupt
ecc_ded_int	Output	Double error detected interrupt
ecc_overflow_int	Output	Error detection overflow interrupt

A.7.2 Debug and profile signals

Table A-9 Debug and profile signals

Signal	Type	Description
ev_bus_valid_s	Output	System interface PMU event valid bus
ev_bus_payload_s	Output	System interface PMU event payload bus
ev_bus_valid_m	Output	Memory interface PMU event valid bus
ev_bus_payload_m	Output	Memory interface PMU event payload bus

A.7.3 Wakeup signals

Table A-10 Wakeup signals

Signal	Type	Description
cwakeup_s	Input	Wakeup signal for the clock domain of the system interface and interconnect
cwakeup_m	Input	Wakeup signal for the clock domain of the interconnect, read buffer, write buffer, and memory interface

A.7.4 Miscellaneous signals

Table A-11 Ad hoc signals

Signal	Type	Description
awap	Input	Write auto-precharge policy signal
arap	Input	Read auto-precharge policy signal
clr_ex_mon	Output	Global exclusive monitor clear event

Table A-11 Ad hoc signals (continued)

Signal	Type	Description
user_status	Input	User-defined inputs
user_config0	Output	User-defined outputs
user_config1	Output	User-defined outputs

Appendix B
Revisions

This appendix describes the technical changes between released issues of this book.

Table B-1 Issue A

Change	Location	Affects
First release	-	-

Table B-2 Differences between issue A and issue B

Change	Location	Affects
DMC APB transaction response clarified	*APB3 interface* on page 2-2	All revisions
DMC ACE-Lite transaction response clarified	*Responses* on page 2-7	All revisions

Table B-3 Differences between issue B and issue C

Change	Location	Affects
QVN description added	*QVN* on page 2-3	r1p0
ECC functionality updated	*Error-correcting code* on page 2-7	r1p0

Table B-4 Differences between issue C and issue D

Change	Location	Affects
Updated the text for product revisions section	*Product revisions* on page 1-9	All revisions

Table B-5 Differences between issue C and issue D

Change	Location	Affects
Output pin added	*Miscellaneous signals* on page A-13	r1p2