Freescale Semiconductor
Application Note

AN3221
Rev. 0, 01/2006

MPC5200B SDRAM Initialization and Configuration

by: Lamar Hansford
Infotainment, Multimedia,and Telematics Division

This document is presented for the purpose of developing knowledge in the area of the MPC5200B DRAM interface. Both the hardware and software aspects of configuring SDRAM and DDR DRAM devices are provided here. The MPC5200B is based on a an MPC603e PowerPC™ core.

1 Overview

The MPC5200B SDRAM interface module is a flexible controller enabling systems using either SDR and DDR devices. The memory subsystem on the MPC5200B has the following features.

- Two chip selects
- Up to 256 MB programmable space on each chip select
- Support for 2 and 3 CAS latency SDR
- Support for 2 & 2.5 CAS latency DDR
- Programmable timing interface which allows support for memories from multiple vendors

Table of Contents

2 MPC5200B Memory Interface & Architecture

2.1 Clock Generation

When configuring the DRAM interface, it is critical that developers understand how the DRAM clock (mem_clk) is generated.

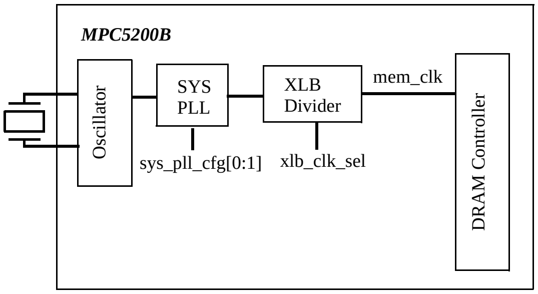

Figure 1. DRAM Clock Generation

In the diagram above, the signals sys_pll_cfg[0:1] and xlb_clk_sel are controlled at boot time through the CDM power on reset configuration register–MBAR + 0x0204, bits [24:26].

2.2 MPC5200B Bus Architecture

The MPC5200B DRAM controller address and data is derived from the XL Bus driven from the PowerPC system. Care must be taken to ensure that PowerPC notation is translated to conventional notation when considering the external interface[1].

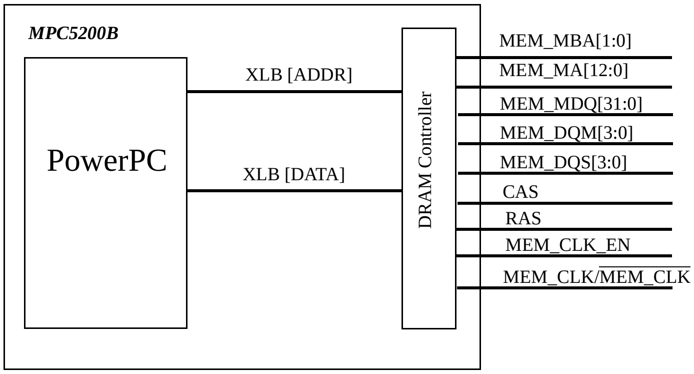

Figure 2. DRAM Signals

[1] In PowerPC notation, the bit 0 is the MSB, and bit 31 is the LSB.

2.2.1 DRAM External Signals

Table 1. SDRAM External Signals

External Signal Name	Direction	Description	Derived From
MEM_CLK	O	Memory clock–Frequency is the same as the internal XL bus clock. Maximum allowed value is 132 MHz.	xlb_clock[1]
MEM_CLK	O	Inverted memory clock–Used for DDR-SDRAM devices. Internally generated “DQS” for SDR-SDRAM devices.	xlb_clock
MEM_CLK_EN	O	Memory clock enable (CKE)–When low, the SDRAM is disabled. Used to switch memory into and out of self-refresh/power-down modes.	
$\overline{\text{MEM\_CS}}$[1:0]	O	Memory chip select.	See Section 3.2.1, “SDRAM Chip Select Configuration Registers”
MEM_RAS	O	Memory row address select–Asserts on read and write operations to the SDRAM memory space.	See Table 2
MEM_CAS	O	Memory Column Address Select–Asserts on read and write operations to the SDRAM memory space.	See Table 2
MEM_WE	O	Memory write enable–Asserts on write operation to the SDRAM memory space.	See Table 2
MEM_MA[12:0]	O	Memory multiplexed address[2]	See Table 10
MEM_MBA[1:0]	O	Memory bank address	xlb_addr[20:21]
MEM_DQM[3:0]	O	Memory data mask–Addressing = 0:3 0 Data byte read/write is enabled 1 Data byte read/write is inhibited In all modes the memory controller ignores mask bits when reading data.	xlb_addr[1:0]
MEM_MDQ[31:0]	I/O	Memory data. addressing	xlb_data[31:0][3]
MEM_MDQS[3:0]	I/O	Memory data strobe, DDR only–Addressing = 0:3.	
Note: For 16-bit mode external pulldown devices are required on MEM_MDQS[1:0].			

NOTES:

[1] xlb_clock is controlled through the reset configuration word.

[2] MEM_MA[10] is used as a control signal per DRAM specifications.

[3] When a port size of 16-bits is selected, data bits are mapped from the upper half-word, or xlb_data[31:16].

2.2.2 DRAM Command Generation

The MPC5200B generates DRAM commands on the memory interface according to JEDEC specifications. This specification dictates that the $\overline{\text{CS}}$, $\overline{\text{CAS}}$, $\overline{\text{RAS}}$, and $\overline{\text{WE}}$ signals can be driven to place the DRAM into a corresponding mode which allows read, write, precharge, and internal setup mode (others may be available depending on manufacturer). These commands are used by the memory controller during normal operation of the device without user intervention. The list of supported commands can be seen in Section 2.2.2.1, “Supported DRAM Commands”.

2.2.2.1 Supported DRAM Commands

When performing DRAM transactions, the memory controller generates the corresponding SDRAM command. Table 2 lists SDRAM commands supported by the memory controller. These commands are automatically generated by the controller any time read and write operations occur to the selected DRAM chip select range. Additionally the memory controller will generate refresh commands according to the parameters programmed in the ref_interval bit field of the Memory Controller Control Register–MBAR + 0x0104 of the MPC5200B.

Table 2. SDRAM Commands

Function	Symbol	CKE[1]	CS[2]	RAS	CAS	WE	BA[1:0][3]	A10[4]	Other A
Command inhibit	INH	H	H	X	X	X	X	X	X
No operation	NOP	H	L	H	H	H	X	X	X
Read	READ	H	L	H	L	H	V	L	V
Write	WRIT	H	L	H	L	L	V	L	V
Row and bank active	ACT	H	L	L	H	H	V	V	V
Burst terminate	BST	H	L	H	H	L	X	X	X
Precharge all banks	PALL	H	L	L	H	L	X	H	X
Load mode register	LMR	H	L	L	L	L	LL	V	V
Load extended mode register	LEMR	H	L	L	L	L	LH	V	V
CBR auto refresh	AREF	H	L	L	L	H	X	X	X
Self refresh	SREF	H→L	L	L	L	H	X	X	X
Power down	PDWN	H→L	H	X	X	X	X	X	X

Note:

- H = High
- L = Low
- V = Valid
- X = Don’t care

NOTES:

[1] MEM_CLK_EN

[2] MEM_CS[0:1]

[3] MEM_MBA[1:0]

[4] MEM_MA[10]

3 Software Configuration

Software configuration of the DRAM memory interface is performed by following 4 basic steps.

1. GPIO configuration
2. Memory map and chip select configuration
3. DRAM interface configuration
4. DRAM vendor specific configuration

This software initialization should be combined into the standard boot flow and should be addressed in separate phases. The list above represents four different software modules.

3.1 GPIO Configuration

By default “chip select 1” is configured as a general purpose IO pin. To use this pin as a chip select, it must be reconfigured as a chip select. This is accomplished by programming bit 0 (MSB) of the GPS port configuration register–MBAR + 0x0B00–to 1 (reset value is 0).

3.1.1 GPS Port Configuration Register

Table 3. GPS Port Configuration Register

MBAR offset	Name	Description
0x0B00	GPS_PORCFG	GPIO port configuration register

	msb 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	CS1	LPTZ	ALTs		CS7	CS6	ATA		IR_USB_CLK	IRDA			Ether			
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31 lsb
R	PCI_DIS	USB_SE	USB		PSC3				Rsvd	PSC2			Rsvd	PSC1		
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 4. GPS Port Configuration Field Description (Sheet 1 of 3)

Bit	Name	Description
0	CS1	Memory chip select bit 0 gpio_wkup_6 1 mem_cs1 (second SDRAMC chip select) on gpio_wkup_6 pin
1	LPTZ	LocalPlus non-muxed TSIZ bit 0 gpio_wkup_7 and test_sel_1 1 TSIZ 1 on gpio_wkup_7 and TSIZ 2 on test_sel_1
2:3	ALTs	Alternatives[1] 00No Alternatives: CAN1/2 on PSC2 according to PSC2 setting; SPI on PSC3 according to PSC3 setting. 01 ALT CAN position: CAN1 on I2C1, CAN2 on Tmr0/1 pins[2] 10 ALT SPI position: SPI on Tmr2/3/4/5 pins[1] 11 Both on ALT
4	CS7	0 Interrupt GPIO on PSC3_5[3] 1 CS7 on PSC3_5

Table 4. GPS Port Configuration Field Description (Sheet 2 of 3)

Bit	Name	Description
5	CS6	0 Interrupt GPIO on PSC3_4[3] 1 CS6 on PSC3_4
6:7	ATA	Advanced technology attachment 00 No ATA chip selects, csb_4/5 used as normal chip select 01 ATA cs0/1 on csb_4/5 10 ATA cs0/1 on i2c2 clk/io 11 ATA cs0/1 on Tmr0/1[2]
8	IR_USB_CLK	Infrared USB clock 0 IrDA/USB 48MHz clock generated internally. Pin is GPIO. 1 IrDA/USB clock is sourced externally, input only
9:11	IRDA	Infrared data association 000 All IrDA pins are GPIOs 001 Reserved 010 Reserved 011 Reserved 100 Reserved 101 UART (without CD) / IrDA 110 Reserved 111 CODEC (without MCLK) / IrDA
12: 15	Ether	Ethernet 0000 All 18 Ethernet pins are GPIOs 0001 USB2 on Ethernet[4] 0010 Ethernet 10Mbit (7-wire) mode 0011 Ethernet 7-wire and USB2[4] 0100 Ethernet 100Mbit without MD 0101 Ethernet 100Mbit with MD 011X Reserved 1000 Ether 7-wire, UARTe, J1850 1001 Ether 7-wire, J1850 1010 Two UARTs, J1850 1011 One UARTe, J1850 1100 J1850 1101 Reserved 111X Reserved
16	PCI_DIS	0 PCI controller enabled 1 PCI controller disabled When large Flash or most graphics modes are enabled on the LocalPlus bus interface, the PCI interface cannot be used (PCI control signals are used to support these modes). When these modes are enabled (see LocalPlus control registers), the PCI controller must be disabled to prevent interference. If these modes are enabled at boot, this bit will come out of reset set to 1. If these modes are not enabled at boot, this bit will come out of reset set to 0.

Table 4. GPS Port Configuration Field Description (Sheet 3 of 3)

Bit	Name	Description
17	USB_SE	USB single ended mode The USB interface is able to support both differential and single-ended modes. This bit allows the USB I/O interface to be programmed to single-ended mode. Differential mode supplies TXP/TXN and RXP/TXN. Single ended mode supplies TXP/TX_SE0 and RXP/RX_SE0. This bit controls *all* USB ports (that is, they are not individually programmable). Default is differential mode. 0 Differential mode (Default after reset) 1 Single ended mode
18:19	USB	00 4 GPIOs and 1 **interrupt GPIO** 01 USB 10 Two UARTs 11 Reserved
20:23	PSC3	Programmable serial controller 3 0000 All PSC3 pins are GPIOs 0001 USB2 on PSC3, no GPIOs available[4] 001X Reserved 0100 UART functionality without CD 0101 UARTe functionality with CD 0110 CODEC3 functionality 0111 CODEC3 functionality (with MCLK) 100X SPI 101X Reserved 1100 SPI with UART3 1101 SPI with UART3e 111X SPI with CODEC3
24	—	Reserved
25:27	PSC2	Programmable serial controller 2 000 All PSC2 pins are GPIOs 001 CAN1&2 on PSC2 pins[4] 01X AC97 functionality 100 UART functionality without CD 101 UARTe functionality with CD 110 CODEC2 functionality(without MCLK) 111 CODEC2 functionality (with MCLK)
28	—	Reserved
29:31	PSC1	Programmable serial controller 1 00X All PSC1 pins are GPIOs 01X AC97 functionality 100 UART functionality without CD 101 UARTe functionality with CD 110 CODEC1 functionality (without MCLK) 111 CODEC1 functionality (with MCLK)

NOTES:

[1] ALT SPI cannot exist with any SPI on PCS3.

[2] ALT CAN cannot exist with ATA on Tmr0/1, and not with CAN on PSC2.

[3] PSC3_4 and PSC3_5 default to zero (interrupt gpio) after reset; however, if the PSC3 is programmed to USB2 mode, RXP and RXN will be on these pins. If PSC is programmed to UARTe mode, CD will be on the PSC3_4 pin.

[4] USB cannot exist on both Either and PSC3.

3.2 Memory Map And Chip Select Configuration

The chip select configuration register can be found in the bank of registers located at MBAR + 0x34. These two registers control the configuration on the start address of each DRAM chip select and the size of the DRAM.

3.2.1 SDRAM Chip Select Configuration Registers

Table 5. SDRAM Chip Select Configuration Register

MBAR offset	Name	Description
0x0034	SDRAM Chip Select 0	Contains the base addresses and configurations for SDRAM's connected to the SDRAM controller.
0x0038	SDRAM Chip Select 1	

	msb 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	Base XLB Address												Reserved			
W																
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31 lsb
R	Reserved											SDRAM Size				
W																
RESET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Description
0:11	Base XLB Address	Start address for memory
12:26	Reserved	These bits are reserved
27:31	SDRAM size	Should be set to size of SDRAM at corresponding SDRAM chip select. Settings are included in the following table.
Note: The Base XL bus address must be consistent with the SDRAM size		

SDRAM Memory Size	SDRAM size bit setting
11111[1]	4GB[1]
11110[1]	2GB[1]
11101[1]	1GB[1]
11100[1]	512MB[1]

SDRAM Memory Size	SDRAM size bit setting
11011	256MB
11010	128MB
11001	64MB
11000	32MB
10111	16MB
10110	8MB
10101	4MB
10100	2MB
10011	1MB
00001-10010	Reserved
0000	Disable

NOTES:

[1] Unsupported values

3.3 DRAM Interface Configuration

The hardware interface for the SDRAM is configured through registers starting at MBAR + 0x100. These registers enable selection of:

- SDR / DDR type selection
- Refresh configuration
- Row/column configuration
- Hi-Z operation
- Timing configuration

Timing parameters in configuration registers 1 and 2 must be configured based on the DRAM manufacturers data sheet. These parameters ensure that the delays between transactions are not violated by the MPC5200B. For an electrical description of these settings, please refer to Section 4, “Memory Timing Parameters”.

When configuring the MPC5200B DRAM interface, perform the following 3 steps in this order:

1. Write the “SDelay” register
2. Configure timing (Section 3.3.2, “Configuration Register 1—MBAR + 0x0108 & Section 3.3.3, “Configuration Register 2—MBAR + 0x010C)
3. Enable and configure the DRAM control (Section 3.3.4, “Control Register—MBAR + 0x0104)

Following the above sequence ensures critical DRAM timing, and as a result, refresh parameters will be written correctly prior to the enabling of the DRAM.

3.3.1 SDelay—MBAR + 0x0190

Table 6. SDelay Register

	msb 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	reserved															
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31 lsb
R	reserved												sdelay			
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Description
28:31	sdelay	Special configuration delay–Must be written to 0x00000004.

The SDelay should be written to a value of 0x00000004. It is required to account for changes caused by normal wafer processing parameters.

3.3.2 Configuration Register 1—MBAR + 0x0108

Table 7. Memory Controller Configuration Register 1

	msb 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	srd2rwp				swt2rwp				rd_latency				Rsvd	act2rw		
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31 lsb
R	Rsvd	pre2act			ref2act				Rsvd	wr_latency			Reserved			
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Description[1]
0:3	srd2rwp	Single read to read/write/precharge delay, This is specified in MEM_CLK cycles.
4	—	Reserved
4:7	swt2rwp	Single write to read/write/precharge delay. This is specified in MEM_CLK cycles.

Bit	Name	Description[1]
8:11	rd_latency	Read CAS latency. For DDR: If CL==2, write 0x6 If CL==2.5, write 0x7 For SDR: If CL==2, write 0x2 If CL==3, write 0x3 **Note:** CL=2.5 is not supported for SDR.
12	—	Reserved
13:15	act2rw	Active to read/write delay. This is specified in MEM_CLK clock cycles.
16	—	Reserved
17:19	pre2act	Precharge to active or refresh delay. This is specified in MEM_CLK clock cycles.
20:23	ref2act	Refresh to active delay. This is specified in MEM_CLK clock cycles.
24	—	Reserved
25:27	wr_latency	Write latency. This is specified in 2xMEM_CLK clock cycles.
28:31	—	Reserved

NOTES:

[1] In SDR mode all delays are calculated in terms of MEM_CLK cycles. In DDR mode all delays are calculated in terms of 2xMEM_CLK cycles.

3.3.3 Configuration Register 2—MBAR + 0x010C

Table 8. Memory Controller Configuration Register 2

	msb 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	brd2rp				bwt2rwp				brd2wt				burst_length			
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31 lsb
R	Reserved															
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Description
0:3	brd2rp	Burst read to read/precharge delay. This is expressed in terms of MEM_CLK clock cycles.
4:7	bwt2rwp	Burst write to read/write/precharge delay. This is expressed in terms of MEM_CLK.
8:11	brd2wt	Burst read to read/write/precharge delay. This is expressed in terms of MEM_CLK

Bit	Name	Description
12:15	burst_length	This value is burst length - 1. The MPC5200B only supports a burst length of 8 (0x7).
16:31	—	Reserved

3.3.4 Control Register—MBAR + 0x0104

The 32-bit read/write control register controls specific operations and generates some SDRAM commands. This register is reset only by a power-up reset signal.

Table 9. Memory Controller Control Register

	msb 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R	mode _en	cke	ddr	ref _en	Rsvd			hi_ addr	Rsvd	drive _rule	ref_interval[0:5]					
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31 lsb
R	Reserved		mem_ ps	Rsvd	dqs_oe				Reserved							Rsvd
W														soft _ref	soft _pre	
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Description
0	mode_en	0 Mode register locked; cannot be written. 1 Mode register enabled; can be written.
1	cke	0 MEM_CLK_EN negated (low). 1 MEM_CLK_EN asserted (high). cke must be set to 1 to perform normal read and write operations. Set cke to 0 to put the memory in self refresh or powerdown mode.
2	ddr	0 SDR mode 1 DDR mode
3	ref_en	0 Automatic refresh disabled. 1 Automatic refresh enabled.
4:6	—	Reserved
7	hi_addr	Control the translation of internal address signals to the external pins. See Table 3.
8	—	Reserved (must be written 0)
9	drive_rule	0 Tri-state except to write mode: MPC5200B drives the MDQ and MDQS lines only when necessary to perform write commands. 1 Drive except to read mode: MPC5200B tri-states the MDQ and MDQS lines only when necessary to perform read commands. This mode prevents unterminated memory signals from floating for extended periods; however, terminated routing is always recommended over unterminated.
10:15	ref_interval[0:5]	The average periodic interval at which the controller generates refresh commands to memory; it is measured in increments of 64 x MEM_CLK period.

Bit	Name	Description
16:17	—	Reserved
18	mem_ps	Memory data port size 0 32-bit data bus 1 16-bit data bus
19	—	Reserved
20:23	dqs_oe[3:0]	Each bit individually controls one MEM_MDQS output. 0 The corresponding MEM_MDQS pin is ***never*** driven, regardless of memory operation and drive_rule. Always set to 0000 for SDR. 1 The corresponding MEM_MDQS pin ***can*** be driven, depending on memory operations and drive_rule. DDR only.
24:28	—	Reserved
29	soft_ref	0 No operation. 1 Generate a non-periodic Auto Refresh command as soon as possible. This is a write-only bit; always returns 0 on a read. A software requested refresh is completely independent of the periodic refresh interval counter. Software refresh is only possible when mode_en==1.
30	soft_pre	0 No operation 1 Generate a Precharge All command as soon as possible. This is a write-only bit; always returns 0 on a read. Software precharge is only possible when mode_en==1.
31	—	Reserved

3.3.5 Address Translation

In order for the MPC5200B to provide a flexible interface for a variety of memories, the registers provide two configuration settings for address, row, and column configurations.

Table 10 indicates how the internal XL bus address bits are translated internally to support sdram devices on the external electrical interface. This address translation is controlled through programming the hi_addr and mem_ps bits of the control register.

Table 10. Address Translation

hi_addr	mem_ps	MEM_MA[12:0][1]		MEM_BA[1:0]	Dim[2]
		Row Address Phase	Column Address Phase		
0	0	XLB_ADDR 0 [8:19] MEM_MA 12 [11:0]	XLB_ADDR 4 5 0 6 7 [22:29] MEM_MA 12 11 10 9 8 [7:0]	XLB_ADDR[20:21]	4k x 4k x 2 x 4
1	0	XLB_ADDR [7:19] MEM_MA [12:0]	XLB_ADDR 0 4 0 5 6 [22:29] MEM_MA 12 11 10 9 8 [7:0]	XLB_ADDR[20:21]	8k x 2k x 2 x 4
0	1	XLB_ADDR 0 [8:19] MEM_MA 12 11 10 9 8 [7:0]	XLB_ADDR 5 6 0 7 [22:30] MEM_MA 12 11 10 9 [8:0]	XLB_ADDR[20:21]	4k x 4k x 2 x 2
1	1	XLB_ADDR [7:19] MEM_MA [12:0]	XLB_ADDR 4 5 0 6 [22:30] MEM_MA 12 11 10 9 [8:0]	XLB_ADDR[20:21]	8k x 4k x 2 x 2

NOTES:

[1] Fields marked with a ‘0’ are always driven low.

[2] ROW x COL x BANK x WORDS

3.4 Vendor Specific Configuration

In the final stage of configuration the developer should configure the vendor specific settings in the DRAM mode registers. This may include programming for burst length, CAS settings, precharge during initialization, and locking of clocks. This is done by using the mode register as described in Section 3.4.1, “Mode Command Generation”.

3.4.1 Mode Command Generation

Mode commands may be generated by the DRAM controller through use of the mode register. When written, this register will generate a LMR/LEMR command (see Table 2). The data driven onto the bit shown in the register are then latched into the mode register of the external device to program memory specific settings without altering the contents of memory. Some settings which may be available by the manufacturer are as follows:

- Burst length
- CAS latency
- Burst type (sequential or interleaved)
- Clock functions (lock, etc)

See manufacturer’s data sheet for further details.

3.4.1.1 Mode Register—MBAR + 0x0100

Each time the 32-bit write-only mode register (mode[0:31]) is written (and cmd is set to 1), the controller generates command over the sdram interface.

Table 11. Memory Controller Mode Register

	msb 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R															Rsvd	
W	MEM_MBA [1:0]		MEM_MA[11:0]													cmd
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31 lsb
R	Reserved															
W																
RESET:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Description
0:1	MEM_MBA [1:0]	See the SDRAM data sheet. Select either the memory device mode register or the memory device extended mode register, if present.
2:13	MEM_MA[11:0]	See the SDRAM data sheet. MPC5200B supports the following: • Read CAS latency, SDR: 2, 3 • Read CAS latency, DDR: 2, 2.5 • Burst type: Sequential only • Burst length: 8 only • Other fields: As appropriate Specific bit allocation can vary from device to device. All devices in all $\overline{CS}$ spaces must have compatible format(s) because all are written at the same time with the same value.
14	—	Reserved
15	cmd	1 Generate a (extended) mode register set memory command. Apply to all $\overline{CS}$ at once. 0 Do not generate any memory command.
16:31	—	Reserved

To ensure that commands are generated properly care should be taken to ensure that signals are connected correctly.

4 Memory Timing Parameters[1]

Memory timing parameters are controlled through settings found in the memory controller register map. These are detailed in Section 3, "Software Configuration."

[1] For signals listed as 'command,' refer to Section 2.2.2.1, "Supported DRAM Commands" for electrical description.

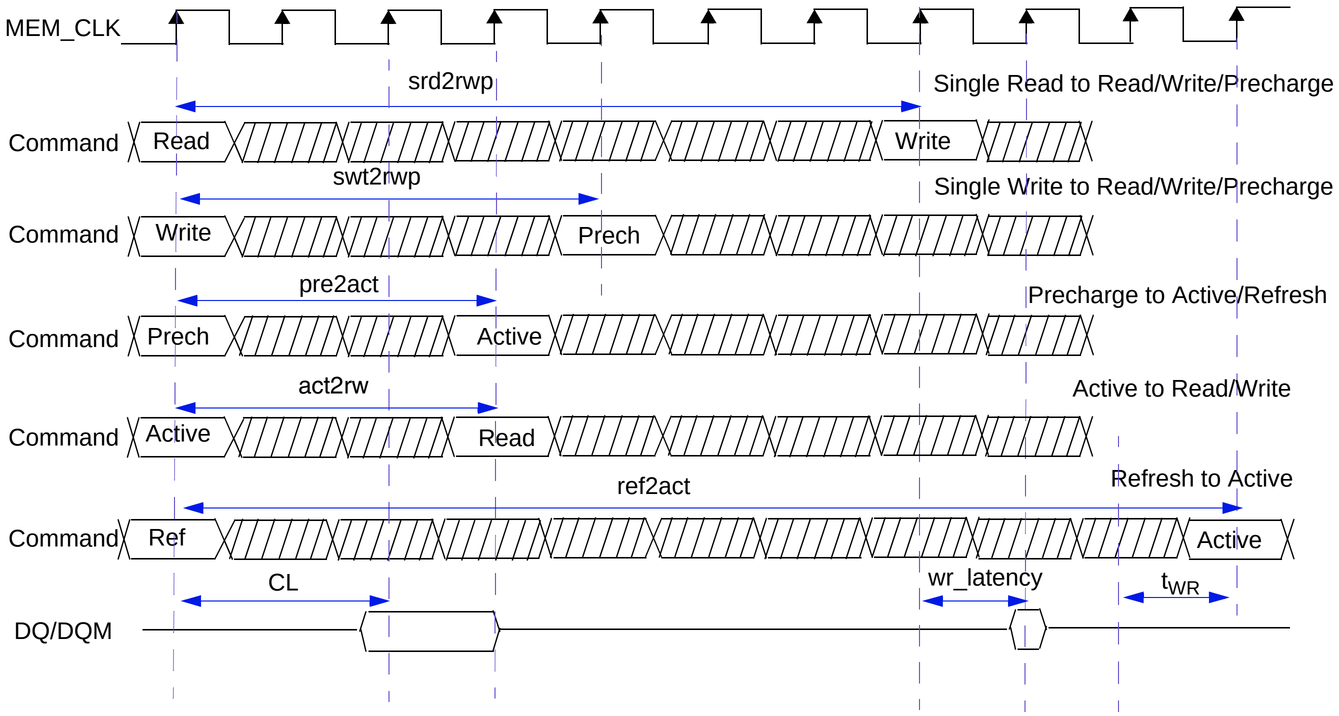

Figure 3. DDR Single Beat Timing[1]

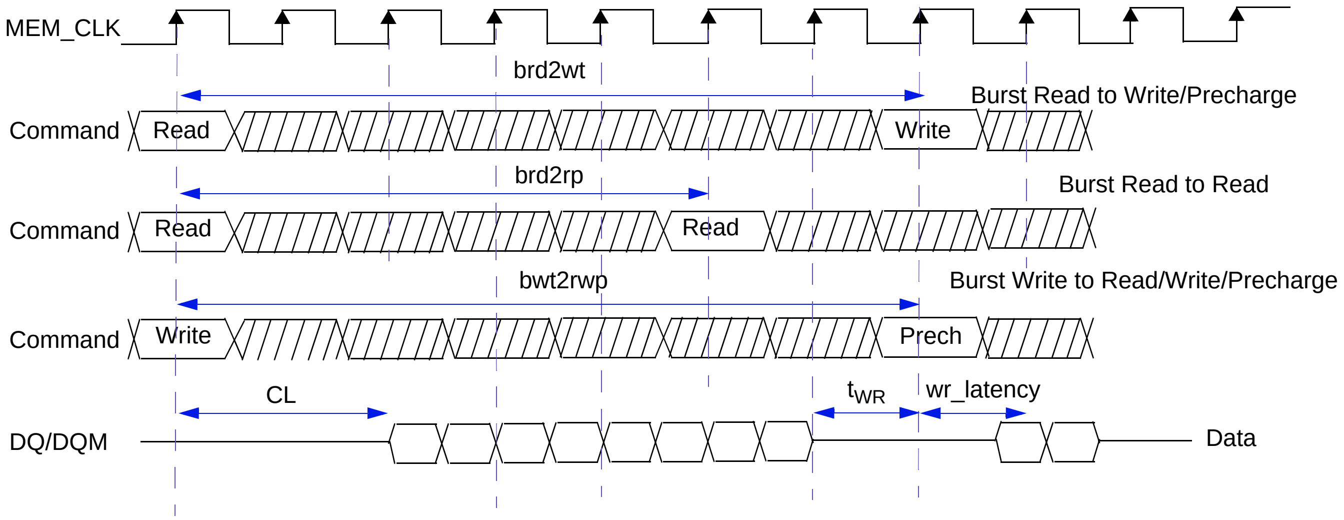

Figure 4. DDR Burst Timing[1]

[1] Timings values are relative to a specific command sequence. Care must be taken to select the appropriate command from the listed sequence.

5 Example Initialization Sequence

5.1 Example Micron Initialization

This example will use a MT46V16M16 memory available from Micron. The initialization description presented in Section 5.1.1, “Micron DDR Initialization Sequence,” is taken verbatim from the Micron data sheet. This is revised in Section 5.1.2, “MPC5200B Initialization Sequence for Micron DDR Memories,” with details on how each step is accomplished using the MPC5200B. The next section provides pseudocode for initializing the Micron device.

5.1.1 Micron DDR Initialization Sequence

Per the Micron data sheet. To ensure device operation the DRAM must be initialized as described below:

1. Simultaneously apply power to VDD and VDDQ.
2. Apply VREF and then VTT power.
3. Assert and hold CKE at a LVCMOS logic low.
4. Provide stable CLOCK signals.
5. Wait at least 200 µs.
6. Bring CKE high, and provide at least one NOP or DESELECT command. At this point, the CKE input changes from a LVCMOS input to a SSTL2 input only and will remain a SSTL_2 input unless a power cycle occurs.
7. Perform a PRECHARGE ALL command.
8. Wait at least tRP time. During this time, NOPs or DESELECT commands must be given.
9. Using the LMR command, program the extended mode register (E0 = 0 to enable the DLL and E1 = 0 for normal drive, or E1 = 1 for reduced drive. E2 through En must be set to 0, where n = most significant bit).
10. Wait at least tMRD time. Only NOPs or DESELECT commands are allowed.
11. Use the LMR command program and the mode register to set operating parameters and to reset the DLL. **Note:** At least 200 clock cycles are required between a DLL reset and any READ command.
12. Wait at least tMRD time. Only NOPs or DESELECT commands are allowed.
13. Issue a PRECHARGE ALL command.
14. Wait at least tRP time. Only NOPs or DESELECT commands are allowed.
15. Issue an AUTO REFRESH command. **Note:** This may be moved prior to step 13.
16. Wait at least tRFC time. Only NOPs or DESELECT commands are allowed.
17. Issue an AUTO REFRESH command. **Note:** This may be moved prior to step 13.
18. Wait at least tRFC time. Only NOPs or DESELECT commands are allowed.
19. Although not required by the Micron device, JEDEC requires a LMR command to clear the DLL bit (set M8 = 0). If an LMR command is issued, the same operating parameters should be utilized as in step 11.
20. Wait at least tMRD time. Only NOPs or DESELECT commands are allowed.

21. At this point, the DRAM is ready for any valid command. **Note:** 200 clock cycles are required between step 11 (DLL Reset) and any READ command. VDD and VDDQ Ramp Apply VREF and VTT CKE must be LVCMOS Low

5.1.2 MPC5200B Initialization Sequence for Micron DDR Memories

Translating the Micron data sheet into actions performed on the MPC5200B, the following steps are obtained:

1. Power-up the system. (Meet the requirements specified in the MPC5200B User’s Guide.)
2. Apply VREF, and then apply the VTT power.
3. CKE is automatically held at logic low until the DRAM controller is enabled.
4. mem_clock is automatically provided at reset.
5. Wait at least 200µs. MPC5200B initialization is included in this time.

   Initialize GPIOs (See Section 3.1, “GPIO Configuration.”)

   Initialize Chip Selects (See Section 3.2, “Memory Map And Chip Select Configuration.”)

   Initialize DRAM interface (See Section 3.3, “DRAM Interface Configuration.”)
6. CKE will automatically be brought high when SDRAM is enabled. An NOP command will be issued by the controller.
7. Perform a PRECHARGE ALL command by writing the soft_pre bit of the SDRAM control register. (See Section 3.3.4, “Control Register—MBAR + 0x0104.)
8. Wait at least tRP[1] time. During this time, NOPs or DESELECT commands must be given. By default, the MPC5200B will issue these commands. By waiting the appropriate time period, this requirement will be satisfied.
9. Enable the DRAM DLL by writing the appropriate value to the mode register.[2]

   The LMR command enabling the DLL is programmed by writing a value of 0x40010000 to the mode register. (See Section 3.4.1, “Mode Command Generation.”)
10. Wait at least tMRD[1] time.
11. Reset the DLL, and program operational mode parameters by writing a value of 0x02C70000[1] mode register. (See Section 3.4.1, “Mode Command Generation.”)

    DLL reset

    2.5 CAS latency

    Sequential Burst

    8 word burst length
12. Wait at least tMRD[1].
13. Perform a PRECHARGE ALL command by writing the soft_pre bit of the SDRAM control register. (See Section 3.3.4, “Control Register—MBAR + 0x0104.”)
14. Wait at least tRP[1].

[1] See manufacturers data sheet

[2] The mode register cannot be written unless bit 0 of the Control register (Section 3.3.4, “Control Register—MBAR + 0x0104) is set.

15. Perform a AUTO REF by writing the soft_ref bit of the SDRAM control register. (See Section 3.3.4, “Control Register—MBAR + 0x0104.”)
16. Wait at least tRFC time[1].
17. Perform a AUTO REF by writing the soft_ref bit of the SDRAM control register. (See Section 3.3.4, “Control Register—MBAR + 0x0104.”)
18. Wait at least tRFC[1].
19. Although not required by the Micron device, JEDEC requires an LMR command to clear the DLL bit (set M8 = 0) by writing a value of 0x00C70000[1] to the mode register. (See Section 3.4.1, “Mode Command Generation.”)
20. Wait at least tMRD[1] time.
21. At this point, the DRAM is ready for any valid command. **Note:** 200 clock cycles are required between step 11 (DLL Reset) and any READ command. A delay may be added at this point to ensure that this constraint is met.

5.1.3 Example Pseudo Code

This code assumes MBAR has been already set to 0x10000000. In actual implementation, it will be more efficient for the boot loader to initialize the SDRAM using assembly language since a stack (required by C) is typically not available before SDRAM configuration.

5.1.3.1 Main initialization Sequence

```
start:
          /*
          *   1.   Configure GPIOs
          *       Some pins may have alternate functionality (DRAM1)
          *       therefore we will configure PIN muxing before
          *       prior to memory map configuration
          */
    bl gpio_pin_init

         /*
          *   2.   Configure Clocks (not required for DRAM configuration)
          */
    bl   clock_init

         /*
          *   3.   Configure DRAM space
          *
          */
    dram_init( );

/* Initializaiton complete */
```

5.1.3.2 GPIO initialization

```
/* GPIO offsets */
#define REL_GPIO_PORT_CFG 0xB00
#define REL_GPIO_PORT_ENABLES 0xB04
#define REL_GPIO_PORT_IO 0xB0C

/* Register Values */
#define GPIO_PORT_CONFIG_INIT             0x81000004

gpio_pin_init:
        /* Make sure r4 contains mbar */
        mfspr r4,SPR_MBAR

        /* Configure General Purpose Pins */
        lis r3,(GPIO_PORT_CONFIG_INIT)@h
        ori r3, r3, (GPIO_PORT_CONFIG_INIT)@l
        stw r3, REL_GPIO_PORT_CFG(r4)

        ...... /* Perform Intialization for board here */

        blr
```

5.1.3.3 DRAM Initialization

```
/* SDRAM Extended Mode Write */
#define SDRAM_MODE_WR_M(extended_cmd, cmd)\
    ( ( extended_cmd << 17 ) | \
    ( cmd ))
```

```
/*****************************************************************
 *  Relative Register Definitions
 * These are needed because the definitions in the standard includes
 * are absolute.
*****************************************************************/
#define REL_SDRAM0_BASE 0x34
#define REL_SDRAM1_BASE 0x38

#define REL_SDRAM_MODE 0x100
#define REL_SDRAM_CTRL 0x104
#define REL_SDRAM_CONFIG1 0x108
#define REL_SDRAM_CONFIG2 0x10C
#define REL_SDRAM_SDELAY 0x190

/* vendor specific macros for mode register assignment */
#define MICRON_MODE_M(op_mode, cas, burst_type, burst_len, cmd)\
    ( ( 0<<30 ) | \
    ( op_mode << 25 ) | \
    ( cas << 22) | \
    ( burst_type << 21 ) | \
    ( burst_len << 18) | \
    ( cmd << 16) )

#define MICRON_EMODE_M(op_mode, ds, dll, cmd)\
    ( ( 1<<30 ) | \
    ( op_mode << 20 ) | \
    ( ds << 19 ) | \
    ( dll<< 18 ) | \
    ( cmd << 16 ) )

/* Register Values */
#define IPBI_SDRAM_CS0_CFG_INIT 0x0000001A
#define IPBI_SDRAM_CS1_CFG_INIT 0x00000000

#define CFG_SDRAM_CTRL 0x715F0F00
```

```
#define CFG_SDRAM_CFG1 0x73722930
#define CFG_SDRAM_CFG2 0x47770000

#define CFG_SDRAM_SDELAY 0x0x00000004
#define DRAM_PROG_DELAY 200

dram_init:
    /* Make sure r4 contains mbar */
    mfspr r4,SPR_MBAR

    /*
     *  This section programs the operational DRAM parameters
     *  (DRAM Size and address range)
     */

     /* DRAM0 CS Configuration */
    lis r3, (IPBI_SDRAM_CS0_CFG_INIT)@h
    ori r3, r3, (IPBI_SDRAM_CS0_CFG_INIT)@l
    stw r3, REL_SDRAM0_BASE(r4)

    /* DRAM1 CS Configuration */
    lis r3, (IPBI_SDRAM_CS1_CFG_INIT)@h
    ori r3, r3, (IPBI_SDRAM_CS1_CFG_INIT)@l
    stw r3, REL_SDRAM1_BASE(r4)

    /* DRAM CTRL Configuration - Unlock the register */
    SETBITS(REL_SDRAM_CTRL, BITS_DRAM_CTRL_UNLOCK)

    /* DRAM CFG1 Configuration */
    lis r3, (CFG_SDRAM_CFG1)@h
    ori r3, r3, (CFG_SDRAM_CFG1)@l
    stw r3, REL_SDRAM_CONFIG1(r4)

    /* DRAM CFG2 Configuration */
    lis r3, (CFG_SDRAM_CFG2)@h
```

```
ori r3, r3, (CFG_SDRAM_CFG2)@l
stw r3, REL_SDRAM_CONFIG2(r4)

/* DRAM CTRL Configuration - Configure the register */
lis r3, ( CFG_SDRAM_CTRL )@h
ori r3, r3, ( CFG_SDRAM_CTRL )@l
stw r3, REL_SDRAM_CTRL(r4)

/* DRAM TAP Delay Initial Setting */
lis r3, (CFG_SDRAM_SDELAY)@h
ori r3, r3, (CFG_SDRAM_SDELAY)@l
stw r3, REL_SDRAM_SDELAY(r4)

/*
 * Start Vendor Configuration here
 *  This section programs DRAM device through the 'mode'
 *  register (see manufacturers data sheet).
 */

/* Assume clocks have been stable for 200us */

/* precharge all banks */
SETBITS(REL_SDRAM_CTRL, BITS_DRAM_CTRL_SOFT_PRE)

/* DRAM Vendor Specific Configuration
 * - Enable DLL
 * - Use Normal Drive strength
 * MICRON_EMODE_M(op_mode, ds, dll, cmd)
*/
lis r3, MICRON_EMODE_M(0, 0, 1, 1)@h
ori r3, r3, MICRON_EMODE_M(0, 0, 1, 1)@l
stw r3, REL_SDRAM_MODE(r4)

/* DRAM Vendor Configuration
 * - Burst Size = 8
 * - CAS 2.5
 * - Sequential mode
 * - Reset DLL
 * MICRON_MODE_M(op_mode, cas, burst_type, burst_len, cmd)
*/
lis r3, MICRON_MODE_M(2, 6, 0, 3, 1)@h
ori r3, r3, MICRON_MODE_M(2, 6, 0, 3, 1)@h
stw r3, REL_SDRAM_MODE(r4)

/* Wait for DLL to lock */
DELAY( dram_delay, DRAM_PROG_DELAY )

/* precharge all banks */
SETBITS (REL_SDRAM_CTRL, BITS_DRAM_CTRL_SOFT_PRE)

/* DRAM - Perform a manual refresh */
SETBITS ( REL_SDRAM_CTRL, BITS_DRAM_CTRL_SOFT_REF )

/* DRAM Vendor Configuration
 * - Burst Size = 8
 * - CAS 2.5
 * - Sequential mode
 * - Reset DLL
 * MICRON_MODE_M(op_mode, cas, burst_type, burst_len, cmd)
*/
lis r3, MICRON_MODE_M(0, 6, 0, 3, 1)@h
ori r3, r3, MICRON_MODE_M(0, 6, 0, 3, 1)@h
stw r3, REL_SDRAM_MODE(r4)

/* DRAM Configuration Complete*/
blr
```

Appendix A
Micron MT46V16MT16 Programming Guide

This section outlines the mode and extended mode registers in the Micron MT46V16MT16 DDR device. The extended mode registers have been mapped from the electrical signals described by the Micron data sheet to the mode register described in Section 2.2.2, “DRAM Command Generation” & Section 3.4.1, “Mode Command Generation”.

Table A1. Normal Mode Programming - MBAR + 0x100

	msb 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R															Rsvd	
W	0	0	Operating Mode					CAS			Burst Type	Burst Length				cmd

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31 lsb
R	Reserved															
W																

Bit	Name	Description
2:6	operating mode	00000 Normal mode 00010 Reset DLL
7:9	CAS	CAS latency 000 Reserved 001 Reserved 010 2 011 Reserved 100 Reserved 101 Reserved 110 2.5 111 Reserved
10	burst type	0 Sequential 1 Interleaved
11:13	burst length	Burst length 000 Reserved 001 2 010 4 011 8 1xx Reserved
15	cmd	1 Generate a (extended) mode register set memory command. Applied to all $\overline{CS}$ at once. 0 Do not generate any memory command.
16:31	—	Reserved

Table A2. Extended Mode Programming - MBAR + 0x100

	msb 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R															Rsvd	
W	0	1	Operating Mode										DS	DLL		cmd

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31 lsb
R	Reserved															
W																

Bit	Name	Description
2:11	operating mode	000000000 Valid
12	DS	Drive Strength 0 Normal 1 Reduced
13	burst type	DLL Enable 0 Disabled 1 Enabled
15	cmd	1 Generate a (extended) mode register set memory command. Applied to all $\overline{\text{CS}}$ at once. 0 Do not generate any memory command.
16:31	—	Reserved

This page intentionally left blank.

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals”, must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. The described product is a PowerPC microprocessor core. The PowerPC name is a trademark of IBM Corp. and is used under license. All other product or service names are the property of their respective owners.