MSC8122RMAD
Rev. 10, 12/2007

MSC8122

Reference Manual Addendum

This document provides updates to revision 3 of the *MSC8122 Reference Manual (MSC8122RM)*. The changes are organized by the chapters that are affected.

Contents

1 About This Book

☞ In the *Other MSC8122 Documentation* section on page xxvii, delete the first bullet.

2 MSC8122 Overview

☞ Replace the 3-Mode Signal Multiplexing row in **Table 1-3** on page 1-3 with the following:

3-Mode Signal Multiplexing	• 64-bit DSI and 32-bit system bus. • 32-bit DSI and 64-bit system bus. • 32-bit DSI and 32-bit system bus.

☞ Replace the Time-Division Multiplexing (TDM) row in **Table 1-5** on page 1-4 with the following:

Time-Division Multiplexing (TDM)	Up to four independent TDM modules, each with the following features: • Optional operating configurations: — Totally independent receive and transmit channels, each having one data line, one clock line, and one frame sync line — Four data lines with one clock and one frame sync shared among the transmit and receive lines. — Glueless interface to E1/T1 framers and switches as well as to common buses such as ST-BUS. • Hardware A-law/µ-law conversion • Up to 62.5 Mbps per TDM (62.5 MHz bit clock if one data line is used, 31.25 MHz if two data lines are used, 15.625 MHz if four data lines are used). • Up to 256 channels. • Up to 16 MB per channel buffer (granularity 8 bytes), where A/µ law buffer size is double (granularity 16 byte). • Receive buffers share one global write offset pointer that is written to the same offset relative to their start address. • Transmit buffers share one global read offset pointer that is read from the same offset relative to their start address. • All channels share the same word size. • Two programmable receive and two programmable transmit threshold levels with interrupt generation that can be used, for example, to implement double buffering. • Each channel can be programmed to be active or inactive. • 2-, 4-, 8-, or 16-bit channels are stored in the internal memory as 2-, 4-, 8-, or 16-bit channels, respectively. • The TDM Transmitter Sync Signal (TxTSYN) can be configured as either input or output. • Frame Sync and Data signals can be programmed to be sampled either on the rising edge or on the falling edge of the clock. • Frame sync can be programmed as active low or active high. • Selectable delay (0–3 bits) between the Frame Sync signal and the beginning of the frame. • MSB or LSB first support.

☞ In **Table 1-5** on page 1-6, replace the 15th bullet in the Ethernet Controller row with the following:

Ethernet Controller	• Ethernet PHY can be exposed either on GPIO pins or on the high ms bits of the DSI/system bus when the DSI and system bus are both 32 bits wide.

☞ In **Section 1.2.8** on page 1-21, change the second sentence in the first paragraph to the following:

It can also connect to multiple framers and switches as well as to common buses such as the ST-BUS.

3 External Signals

☞ Replace the GPIO29 row in **Table 3-7** on page 3-26 with the following:

GPIO29	Input/ Output	**General-Purpose Input Output 29** One of 32 GPIO signals used as GPIO or as one of two dedicated inputs or one of two dedicated outputs. For details, refer to the *MSC8122 Reference Manual* GPIO programming model.
CHIP_ID3	Input	**Chip ID 3** Determines the chip ID of the MSC8122 DSI. It is sampled on the rising edge of $\overline{\text{PORESET}}$ signal.
ETHTX_EN	Output	**Ethernet Transmit Enable** Used to enable the Ethernet transmit controller for MII and RMII modes.

☞ Replace the $\overline{\text{TRST}}$ row in **Table 3-9** on page 3-27 with the following:

$\overline{\text{TRST}}$	Input	**Test Reset** Asynchronously initializes the test controller; must be asserted during power up.

4 Reset

☞ Change the next to last line in **Table 5-2** on page 5-2 to the following:

SC140 Extended Cores Reset (PC points to boot starting address)	Yes	Yes	Yes	Yes

☞ Add the following note after the first paragraph in **Section 5.2** on page 5-4:

Note: Do not use the default HRCW which clears the HRCW[DLLDIS] bit. Because the MSC8113 does not support DLL operation, make sure that the HRCW[DLLDIS] bit is always set after reset.

☞ Replace the 01 row in **Table 5-5** on page 5-4 with the following:

01	Reset configuration write through the 60x-compatible system bus. MSC8122 is a configuration slave. If the HRCW is not written during 1024 CLKIN cycles, it gets a default value of all zeros. **Note:** Always ensure that a valid configuration is written to the HRCW through the system bus. The default configuration is not valid.

☞ Delete **Section 5.5.1** and renumber the succeeding subsections.

☞ Replace the **DLLDIS** row in **Table 5-8** with the following:

DLLDIS 27	0	**DLL Disable** Defines whether the DLL mechanism is disabled. See **Section 7.3**, *Clock Configuration*. **Note:** The MSC8122 does not support DLL operation. Always write a 1 to this bit to configure the device correctly.	0 No DLL bypass. 1 DLL bypass.

5 Clocks

☞ Change the third bullet in **Section 7.2.1** on page 7-3 to the following:

- CLKIN mode. This is the recommended clock scheme for high-frequency synchronous memory interface (SDRAM).

6 Memory Map

☞ In **Table 8-2** on page 8-8, replace the rows for memory locations 00EFFE40 to 00EFFEDF with the following:

00EFFE40	EDCA0_CTRL	EDCA0 Control Register	2
00EFFE44	EDCA1_CTRL	EDCA1 Control Register	2
00EFFE48	EDCA2_CTRL	EDCA2 Control Register	2
00EFFE4C	EDCA3_CTRL	EDCA3 Control Register	2
00EFFE50	EDCA4_CTRL	EDCA4 Control Register	2
00EFFE54	EDCA5_CTRL	EDCA5 Control Register	2
00EFFE58–00EFFE5F	Reserved		
00EFFE60	EDCA0_REFA	EDCA0 reference value A	4
00EFFE64	EDCA1_REFA	EDCA1 reference value A	4
00EFFE68	EDCA2_REFA	EDCA2 reference value A	4
00EFFE6C	EDCA3_REFA	EDCA3 reference value A	4
00EFFE70	EDCA4_REFA	EDCA4 reference value A	4
00EFFE74	EDCA51_REFA	EDCA5 reference value A	4
00EFFE78–00EFFE7F	Reserved		
00EFFE80	EDCA0_REFB	EDCA0 reference value B	4
00EFFE84	EDCA1_REFB	EDCA1 reference value B	4
00EFFE88	EDCA2_REFB	EDCA2 reference value B	4
00EFFE8C	EDCA3_REFB	EDCA3 reference value B	4
00EFFE90	EDCA4_REFB	EDCA4 reference value B	4
00EFFE94	EDCA5_REFB	EDCA5 reference value B	4
00EFFE98–00EFFEBF	Reserved		
00EFFEC0	EDCA0_MASK	EDCA0 Mask Register	4
00EFFEC4	EDCA1_MASK	EDCA1 Mask Register	4
00EFFEC8	EDCA2_MASK	EDCA2 Mask Register	4
00EFFECC	EDCA3_MASK	EDCA3 Mask Register	4
00EFFED0	EDCA4_MASK	EDCA4 Mask Register	4
00EFFED4	EDCA5_MASK	EDCA5 Mask Register	4
00EFFED8–00EFFEDF	Reserved		

7 Extended Core

☞ In **Table 9-10** on page 9-21, replace the **PFOFF** row with the following:

PFOFF 11	0	**Prefetch** Enables/disables Prefetch mode.	0 Prefetch mode enabled (default after boot). 1 Prefetch mode disabled (value after reset).

8 System Bus

☞ Replace **Table 13-11** on page 13-22 with the following:

Table 13-11. Transfer Code Encoding

TC[0–2]	System Bus	Local Bus
000	Reserved	System-Local bus bridge
001	DSI	DSI
010	Reserved	TDM
011	Ethernet Controller	Ethernet Controller
100	Reserved	Reserved
101	SC140 cores	Reserved
110	DMA	DMA
111	DMA	DMA

9 Direct Slave Interface (DSI)

☞ On page 14-2, change the second note to the following two notes:

Note: For the description of the DSI external signals, see **Chapter 3**, *External Signals.*

Note: In systems that include a host device that connects to the MSC8122 through the DSI and the MSC8122 connects to other MSC8122 or MSC8126 devices through the system bus, all devices on the system bus must allow 64-bit read access for reads initiated by the host device. This is because any read access by the host through the DSI is translated by the MSC8122 DSI block as a 64-bit read access.

☞ In **Table 14-7** on page 14-30, replace the HTAAD row with the following:

HTAAD 4	0	**$\overline{\text{HTA}}$ Actively Driven** Indicates whether, at the end of the access, the $\overline{\text{HTA}}$ signal stops driving only after it is driven to a logic 1 at the end of the access. HTAAD is valid only for asynchronous accesses. See **Section 14.3.3** for details. **Note:** The reset value is 0—the device does not drive HTA high after a host access. If the system uses a pull-up resistor, you must set the HT$\overline{\text{AAD}}$ bit. Until HTAAD is set, the rising slope of $\overline{\text{HTA}}$ at the end of the access depends on the strength of the pull-up resistor; this results in a long delay for access termination.	0 $\overline{\text{HTA}}$ is released in logic 0.[1] 1 $\overline{\text{HTA}}$ is released in logic 1.[2]

10 Direct Memory Access (DMA) Controller

☞ Remove the note at the bottom of page 16-1. Refer to the MCS8122 Chip Errata for details.

☞ Add the following note to the end of **Section 16.2.2**:

Note: DMA channels are coupled in pairs (0 and 1, 2 and 3, up to 14 and 15). Do not use two coupled channels simultaneously for flyby or single access transactions.

☞ In **Section 16.2.4**, delete the last sentence on page 16-20.

☞ In **Section 16.2.4** on page 16-20, add the following **Example 16-4** after **Example 16-3**:

Example 16-4. Missing DMA Interrupt in Chained Buffer

Scenario: The DMA controller is activated with a chained buffer. Every buffer should generate an interrupt when it is done. The interrupt service routine (ISR) performed by the core clears the relevant status bit in the DSTR. If an additional buffer completes before the core clears the DSTR bit, the core does not receive the appropriate interrupt because it is a level interrupt.

Solution: Make sure that the ISR clears the DSTR bit as soon as possible. You can read the DCHCR after the interrupt is processed to determine whether another buffer was completed or if the DMA channel is still active.

☞ Replace **Table 16-4** on page 16-25 with the following:

Table 16-4. DCPRAM Values for a Chained Buffer and a Simple Buffer

BD	DCPRAM Parameters		Value	Description
0	BD_ADDR		0x1000	External memory buffer current address
	BD_ATTR	INTRPT	0x0	Do not generate an interrupt when buffer ends
		CONT	0x1	Continuous mode. Do not shut down the channel when size reaches zero
		NO_INC	0x0	Increment address after request is serviced
		NBD	0x1	When size reaches zero, next request calls buffer1
		TSZ	0x4	Maximum transfer size is one burst
		RD	0x1	Read buffer
	BD_BSIZE		0x20	Buffer base size of cyclic buffer
1	BD_ADDR		0x2000	External memory buffer current address
	BD_ATTR	INTRPT	0x0	Do not generate interrupt when the buffer ends. The channel generates the interrupt.
		CONT	0	Non-continuous mode. The channel is closed when the size reaches zero
		NO_INC	0x0	Increment address after request is serviced
		TSZ	0x4	Maximum transfer size is one burst
		RD	0x1	Read buffer
	BD_BSIZE		0x200	Buffer base size of cyclic buffer

☞ Add the following new section after **Section 16.3.1.2**:

16.3.1.3 DMA Arbitration Device Level Considerations

Any access issued by the DMA controller must pass two arbitration layers: the DMA arbitration and the bus controller arbitration. The first arbitration selects the DMA channel that generates the bus access. The second arbitration is the selection of the master by the bus arbitrator (DMA controller, TDM interface, Ethernet controller, DSI, and so on). The DMA arbiter selects the channel priority based on the values of DCHCRx[PRIO] and DPCR[AM]. The bus arbiter uses BD_ATTRx[BP] to select the channel priority on the bus.

You must assign correlating priorities to ensure correct operation of the transfers. The hierarchical arbitration may cause a high priority task to delay a low priority task pending in the bus arbiter.

Example 16-5. Multiple Device Arbitration

Scenario: Two DMA tasks (DMA_1 and DMA_2) are activated to use the device local bus. In addition, the TDM interface uses the local bus. The DMA controller uses fixed priority mode with the following setting:

— DMA_1: DCHCR[PRIO] = 0 (high), BD_ATTRx[BP] = 2 (high)

— DMA_2: DCHCR[PRIO] = F (low), BD_ATTRx[BP] = 0 (low)

The local bus arbiter uses the following priorities, from high to low:

— DMA_1, TDM, DMA_2

At some point during the operation, the DMA controller has no available data from DMA_1 and DMA_2 wins the arbitration. The DMA controller generates a low priority bus access based on the DMA_2 settings. At the same time, the TDM interface also attempts to access the local bus. The local bus arbitrator grants the bus to the TDM interface because its priority is higher than DMA_2. The TDM uses the bus. If, during the TDM transfers, DMA_1 has data to transfer, it tries to generate a bus access, but the DMA_2 access is still pending. This scenario causes the high priority DMA_1 activity to wait due to the pending low priority access by DMA_2. and the ongoing TDM access.

Solution: Assign the TDM a lower priority than DMA_2 to prevent the situation in the scenario from occurring.

☞ In **Section 16.3.2**, delete the last sentence on page 16-32 before the note at the end of the section.

☞ Replace **Table 16-9** and the subsequent bulleted list with the following:

Table 16-9. DCPRAM Bit Descriptions

Bits	Name	Description
0–31	BD_ADDR	**Buffer current address** Holds the buffer address pointer. If the buffer is cyclic, the original address value is restored when the BD_SIZE value reaches zero by decrementing BD_BSIZE from BD_ADDR. See **Section 16.2.4.2**, *Cyclic Buffer,* on page 16-22.
32–63	BD_SIZE	**Size of transfer left for the current buffer** Contains the remaining size of the buffer. This value decrements by the transfer block size each time the DMA controller issues a transaction, until it reaches zero. When BD_SIZE reaches zero, the original value is restored to the value of BD_BSIZE. Program BD_SIZE with a value larger than 0.
64–95	BD_ATTR	**Buffer attributes and temporary data** A 32-bit parameter that describes the attributes of the channel handling this buffer. See **Table 16-10**.
96–127	BD_BSIZE	**Buffer base size** Holds the base size of the buffer. if used, program BD_SIZE with a value greater than 0.

11 Interrupt Processing

☞ In **Table 17-9** on page 17-20, replace the 0x25 $\overline{\text{IRQ5}}$ row with the following:

0x25	$\overline{\text{IRQ5}}$	Ethernet controller Receive Inter Frame Gap Status Interrupt (RIFGSI) **Note:** RIFGSI is only used in SMII mode.	0x940

12 Debugging

☞ Replace **Figure 18-8** with the following:

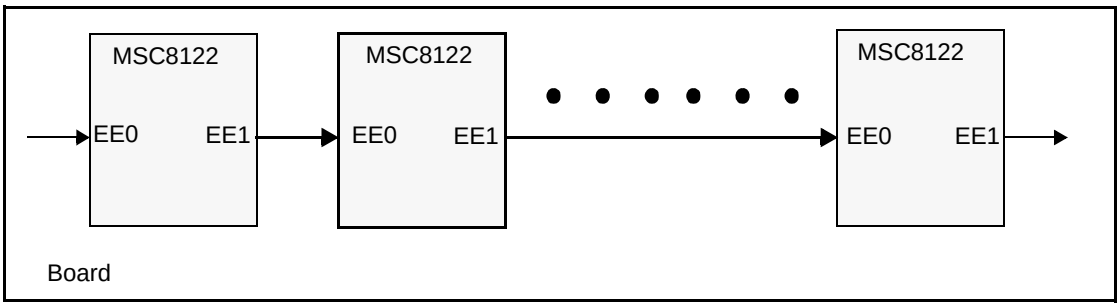


Figure 18-8. Board EE Signal Interconnectivity

13 Internal Peripheral Bus (IPBus)

☞ In **Section 19.1** on page 19-1, change the second sentence to the following:

The TDM bus connects gluelessly to most T1/E1 framers as well as to common buses such as the ST-BUS.

14 TDM Interface

☞ In the first part of **Chapter 20** on page 20-1, change the second sentence in the second paragraph to the following:

The TDM bus connects gluelessly to most T1/E1 framers as well as to common buses such as the ST-BUS.

☞ On page 20-2, change the first full sentence on line 4 by deleting “for transparent channels” before “is 16 MB”, so that the sentence reads:

Buffer size is 16 MB for transparent channels and 32 MB for A-law/μ–law channels.

☞ Add the following note at the end of **Section 20.2.4.3** on page 20-20:

Note: Error interrupts from the TDM are driven directly to the LIC by the TDMx Receive Error Interrupt and TDMx Transmit Error Interrupt bits (TDMxRER[RSE] and TDMxTER[TSE]) when the interrupts are enabled. Therefore, the interrupt handler should clear these bits by writing a 1 to them before clearing the LIC-related status register and before returning to normal operation (exiting the interrupt handler).

☞ In **Section 20.2.6.2** on page 20-23, change the third bullet about the receive channel to the following:

- *Receive Data Buffer Displacement*. TDMxRDBDR[RDBD] field, page 20-60. Adding this field to the first byte of receive data buffer n indicates the location to which the TDM will write next: RGBA << 16 + RCDBA + RDBD is the current write pointer to the receive data buffer n. In most cases, the RDBD can be used to indicate that data is written to the buffer and can be processed. However, in some cases in which the local bus is extremely busy and the TDM bus priority is low, the pointer may be updated before the data is actually written to internal memory. This typically affects the last channel transmitted. In all cases, when configured to reflect the last buffer completion, assertion of the receive buffer threshold interrupt indicates that the data was updated in memory.

☞ Add the following paragraph to the end of **Section 20.2.6.3** on page 20-26:

The TDM threshold interrupt can be programmed as pulse or level (TDMxRIR[RFTL] and TDMsdTIR[TFTL]). When level interrupt is used, you must clear the relevant bit (TDMxRER[RFTE/RSTE] and TDMxTER[TFTE/TSTE]) during interrupt handling. If the bits are not cleared, the TDM does not generate a new interrupt when it reaches the threshold the next time. Pulse interrupt mode does not require that these bits be cleared; the TDM continues to generate an interrupt toward the LIC every time it reaches the threshold. The LIC must be programmed to comply with the selected interrupt mode.

☞ In **Section 20.2.6.4** on page 20-27, replace the first paragraph with the following:

When the TDMxRFP[RUBM] bit is set (see page 20-46), the two receive channels are directed to one buffer in the local bus. The buffer parameters are stored in the TDMxRCPR0. The number of channels must be two (RNCF = 0x01), the number of active links must be one (RTSAL = 0b0000 or 0b0100 or 0b1100), and the number of bits per channel must be four, eight, or sixteen (RCS = 0b0011 or 0b0111. or 0b1111). The channel parameters of channels 0 and 1 are located in the TDMxRCPR0 register. Unified Buffer mode essentially creates a one-channel link that is typically used in point-to-point connections. When TDMxTFP[TUBM] =1, data is transmitted from one buffer into two transmit channels, each four, eight, or sixteen bits wide.

☞ In **Section 20.5** on page 20-31, change the first bullet in Step 3 to the following:

— Program the Transmit Sync Out (TSO) bit according the Transmit Sync signal direction (input or output) used in your system.

☞ In **Section 20.5** on page 20-32, change the fifth bullet in Step 3 to the following:

— Clear the Receive Data Edge bit (RDE = 0x0), so that the receive data is sampled on the positive edge.

☞ In **Section 20.7.2** on page 20-52, replace **Table 20-22** with the following:

Table 20-22. TDMxACR Bit Descriptions

Name	Reset	Description	Settings
— 0–29	0	Reserved. Write to zero for future compatibility.	
LTS 30	0	**Learn Transmit Sync** Determines whether the adaptation machine learns the transmit sync or the receive sync.	0 Adaptation machine learn the receive sync. 1 Adaptation machine learn the transmit sync.
AME 31	0	**Adaptation Machine Enable** Determines whether the adaptation machine is enabled or disabled.	0 Adaptation machine is disabled. 1 Adaptation machine is enabled

☞ In **Table 20-38** on page 20-64, replace the **RFTE** and **RSTE** rows with the following:

RFTE 30	0	**Receive First Threshold Event** This field is set when the first thresholds of all the received data buffers are filled with received data. The first threshold pointer is determined by the Receive Data Buffer First Threshold field (RDBFT). For details, see **Section 20.2.6.3**.	0 No receive first threshold event has occurred. 1 A receive first threshold event has occurred.
RSTE 31	0	**Receive Second Threshold Event** This field is set when the second thresholds of all the receive data buffers are filled with received data. The second threshold pointer is determined by the Receive Data Buffer Second Threshold. (RDBST) field. For details, see **Section 20.2.6**	0 No receive second threshold event has occurred. 1 A receive second threshold event has occurred.

☞ In **Table 20-41** on page 20-66, replace the **RENS** row with the following:

RENS 31	0	**Receive Enable Status** Indicates whether all the receiver parts are enabled/disabled. The propagation of the enable/disable may be delayed because of the different clocks domains. **Note:** If the serial clock is not toggling, this bit may not reflect updated values.	0 The receiver machine is disabled. 1 The receiver machine is enabled.

☞ In **Table 20-42** on page 20-67, replace the **TENS** row with the following:

TENS 31	0	**Transmit Enable Status** Indicates whether all the transmitter parts are enabled/disabled. The propagation of the enable/disable may be delayed because of the different clock domains. **Note:** If the serial clock is not toggling, this bit may not reflect updated values.	0 The transmit machine is disabled. 1 The transmit machine is enabled.

15 UART

☞ In **Section 21.1.1** on page 21-8, replace the second note on the page with the following:

Note: When the shift register is empty (the TC and TDRE flags are set), transmission starts no more than one bit time after the data register is written. If only the TC interrupt source is enabled (SCICR[TCIE] = 1, SCICR[TIE] = 0), then you must ensure at least one bit time interval between successive writes to the SCIDR to enable the transmitter software to write twice to the SCIDR per interrupt.

☞ In **Section 21.6** on page 21-25, replace the two notes before **Table 21-8** with the following:

Note: The formula for calculating the baud rate is: SCI baud rate = SCI system clock/(16 × SBR).

Note: The baud-rate generator is disabled until the SCICR[TE] bit or the SCICR[RE] bit is set for the first time after reset. The baud-rate generator is disabled when SBR = 0.

16 Timers

☞ In **Table 22-8** on page 22-12, change the **CYC** row to the following:

CYC 31	0	**Cyclic/One-Shot** Defines whether the Timer A*x* mode of operation is cyclic or one-shot. In One-Shot mode, the counter of Timer-n counts up until it equals the TCMPA*x*[COMPVAL] field and then stops counting. **Note:** There are only 2 ways to use a timer properly in one-shot mode: 1. Set the one-shot timer input clock to be the bus clock 2. The one-shot timer input clock must be higher than the bus clock frequency divided by 4. In Cyclic mode, the counter of Timer-n counts from 0 until TCMPA[COMPVAL], wraps back to 0 and continues counting.	0 One-Shot mode. 1 Cyclic mode.

17 Ethernet Controller

☞ Replace the MGTCS row in **Table 25-66** with the following:

MGTCS 29–31	0	**Management Clock Select** Determines the clock frequency of the management clock (EC_MDC). Its default value is 000.	000 BUSES_CLOCK/8 divided by 4. 001 BUSES_CLOCK/8 divided by 4. 010 BUSES_CLOCK/8 divided by 6. 011 BUSES_CLOCK/8 divided by 8. 100 BUSES_CLOCK/8 divided by 10. 101 BUSES_CLOCK/8 divided by 14. 110 BUSES_CLOCK/8 divided by 20. 111 BUSES_CLOCK/8 divided by 28.

18 Programming Reference

☞ Replace the System Interface Unit (SIU) programming sheet on p. A-27 and the Direct Memory Access (DMA) BD_ATTR page 1 of 2 on page A-66 with the following two sheets, respectively:

System Interface Unit (SIU)

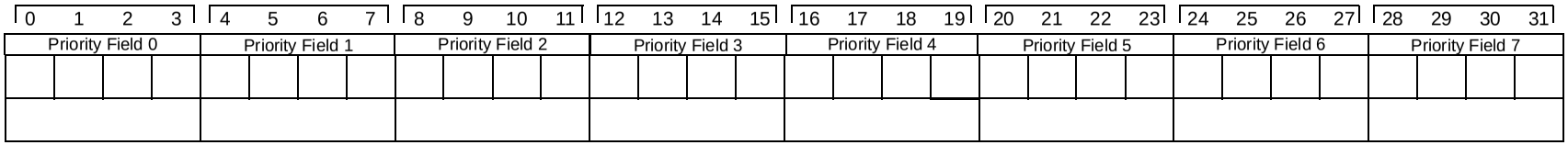
Note: LCL_ALRH and LCL_ALRL assign arbitration priorities for sixteen potential local bus masters. Priority 0 is the highest and Priority 15 is the lowest. The local bus master index number defines each master uniquely. Assign the priority for a local bus master by entering its index number in the appropriate field in LCL_ALRH or LCL_ALRL. The reset value is the recommended configuration.

Local Bus Master Indices (See LCL_ACR)	
0000	Reserved
0001	Reserved
0010	TDM
0011	Host bridge
0100	DSI
0101–1001	Reserved
1010	DMA high priority
1011	DMA middle priority
1100	DMA low priority
1101	Ethernet high priority
1110	Ethernet medium priority
1111	Ethernet low priority

LCL_ALRH

Local Bus Arbitration-Level Register High

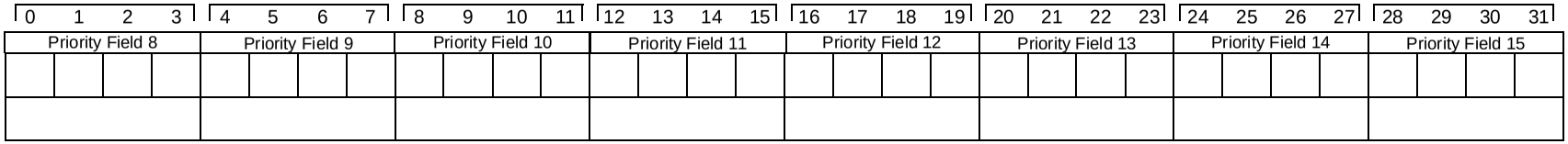
QBus/System Bus Address: _____ See **page 8-55**
DSI Address: 0x1D0038
Reset: 0x01234567; the Boot program changes this to 0x041A53B2
Read/Write

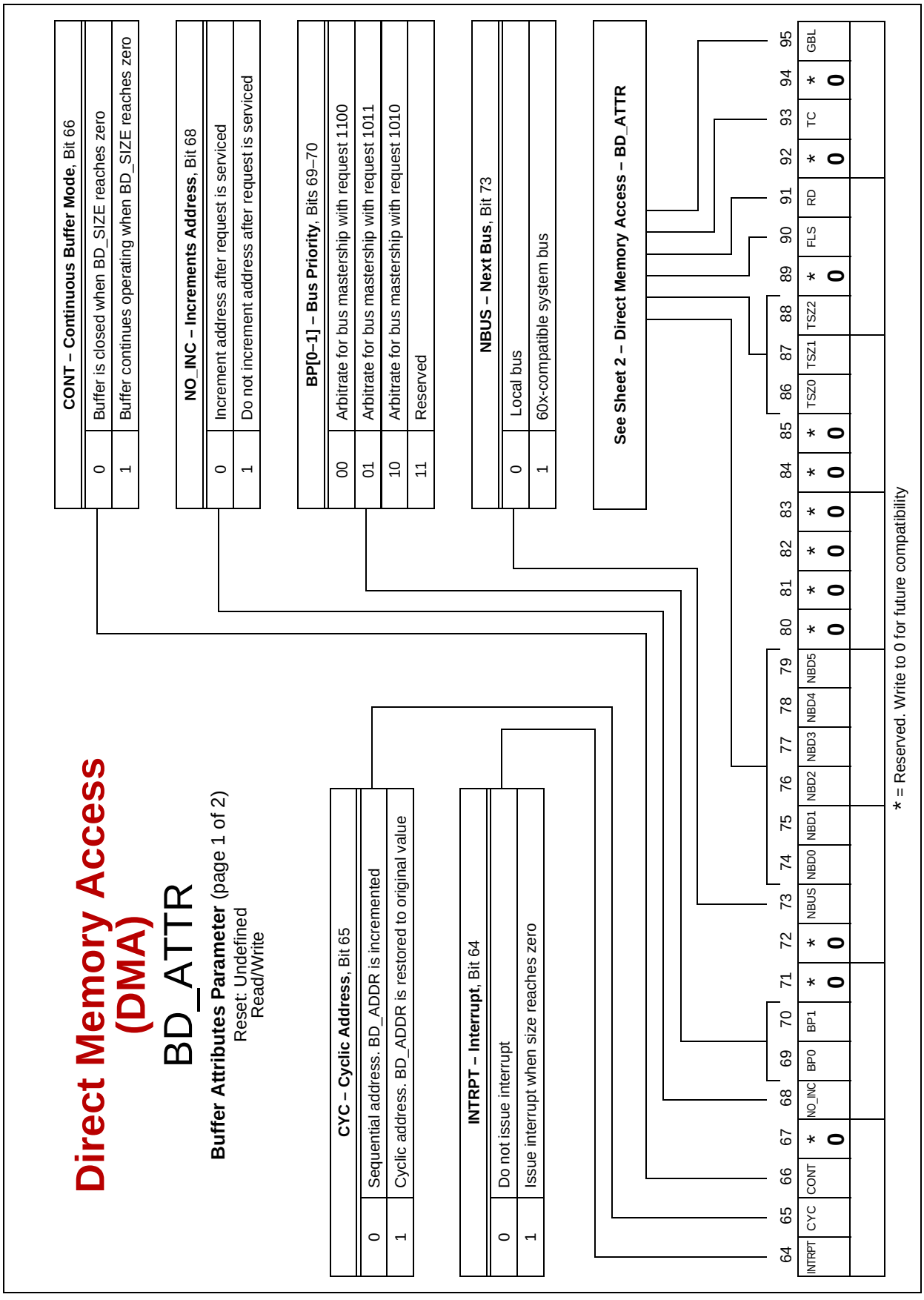


LCL_ALRL

Local Bus Arbitration-Level Register Low

QBus/System Bus Address: _____ See **page 8-55**
DSI Address: 0x1D003C
Reset: 0x89ABCDEF; the Boot program changes this to 0x6C789DEF
Read/Write




Direct Memory Access (DMA)
BD_ATTR
Buffer Attributes Parameter (page 1 of 2)
Reset: Undefined
Read/Write
CONT – Continuous Buffer Mode, Bit 66
0 Buffer is closed when BD_SIZE reaches zero
1 Buffer continues operating when BD_SIZE reaches zero
NO_INC – Increments Address, Bit 68
0 Increment address after request is serviced
1 Do not increment address after request is serviced
BP[0–1] – Bus Priority, Bits 69–70
00 Arbitrate for bus mastership with request 1100
01 Arbitrate for bus mastership with request 1011
10 Arbitrate for bus mastership with request 1010
11 Reserved
NBUS – Next Bus, Bit 73
0 Local bus
1 60x-compatible system bus
See Sheet 2 – Direct Memory Access – BD_ATTR
CYC – Cyclic Address, Bit 65
0 Sequential address. BD_ADDR is incremented
1 Cyclic address. BD_ADDR is restored to original value
INTRPT – Interrupt, Bit 64
0 Do not issue interrupt
1 Issue interrupt when size reaches zero
64 INTRPT
65 CYC
66 CONT
67 * 0
68 NO_INC
69 BP0
70 BP1
71 * 0
72 * 0
73 NBUS
74 NBD0
75 NBD1
76 NBD2
77 NBD3
78 NBD4
79 NBD5
80 * 0
81 * 0
82 * 0
83 * 0
84 * 0
85 * 0
86 TSZ0
87 TSZ1
88 TSZ2
89 * 0
90 FLS
91 RD
92 * 0
93 TC
94 * 0
95 GBL
* = Reserved. Write to 0 for future compatibility

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or
+1-480-768-2130
www.freescale.com/support

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)
www.freescale.com/support

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
+1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor
@hibbertgroup.com

Document Number: MSC8122RMAD
Rev. 10
07/2007

freescale™
semiconductor