Application Report
SPRABX1A–August 2016–Revised December 2016

Quality of Service (QoS) Knobs for DRA74x, DRA75x and TDA2x Family of Devices

Rahul Gulati, Prashant Karandikar, Mihir Mody, Vasant Easwaran, Prithvi Shankar Y.A.

ABSTRACT

This application report lists various quality-of-service (QoS) knobs that are implemented in DRA74x, DRA75x and TDA2x system-on-chip (SoC) family of devices. These QoS knobs aid to optimize overall system performance while running several concurrent application scenarios.

Contents

Trademarks

1 Introduction

The DRA74x, DRA75x and TDA2x family of devices have multiple interconnect initiator ports that can access external memory interface controller (EMIF) and other peripherals available on the SoC. At a high level, these initiators can be classified into various categories as listed below:

- Hard real-time initiators like video input port (VIP), display subsystem (DSS)
- Soft real-time initiators like IVA subsystem, 3D graphics accelerator (GPU) and video processing engine (VPE)
- Processor subsystems like dual Cortex®-M4 IPU subsystem, dual Cortex-A15 MPU subsystem, DSP subsystem and Embedded Vision Engine (EVE)
- DMA category of initiators like enhanced DMA (EDMA) subsystem
- Debug category of initiators like CS_DAP, IEEE1500P

NOTE: For the availability of these peripherals and initiators (as not all devices within a family support every peripheral and initiator), see the device-specific data sheets.

In order to provide a fair share of bandwidth and optimal latency to intended initiators for concurrent use case scenarios, various QoS mechanisms have been implemented on DRA74x, DRA75x and TDA2x family of devices. The QoS implementation on these families of devices spans across various initiators, L3 network on chip interconnect and the memory subsystem.

Figure 1 provides a high-level overview of various QoS mechanisms available in DRA74x, DRA75x and TDA2x family of devices.

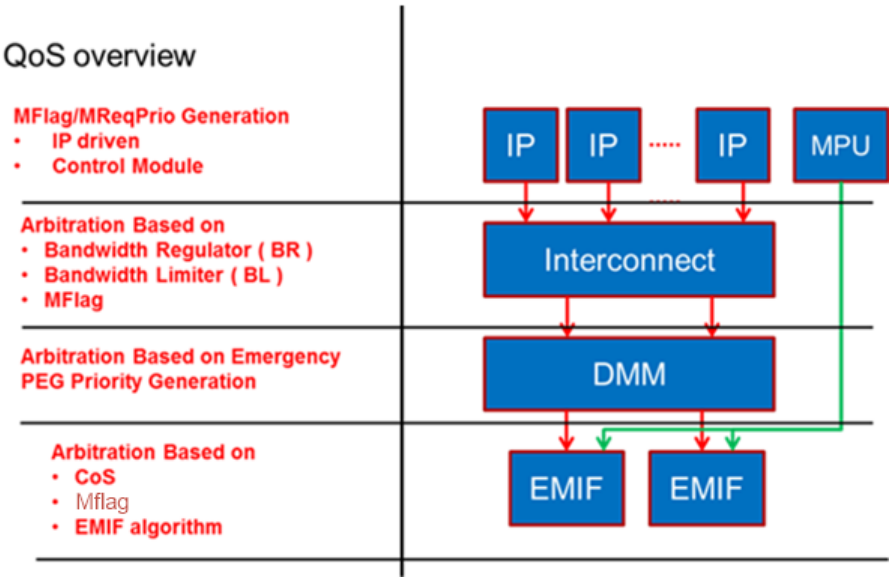

Figure 1. High Level Summary of QoS Mechanisms in DRA74x, DRA75x and TDA2x Family of Devices

As is evident from Figure 1, the implementation for the various QoS mechanisms on DRA74x, DRA75x and TDA2x family of devices spans across various initiators, the L3 network on chip interconnect (L3 NoC), the memory subsystem comprising of Dynamic Memory Manager (DMM) and EMIF

NOTE: The level 3 (L3) interconnect is an instantiation of the Network On Chip (NoC) interconnect from Arteris®, Inc.

All materials and trademarks are used under license from Arteris, Inc. For additional information, see the Arteris reference manuals, or contact Arteris, Inc.

2 Controlling Arbitration Through MFlag and MReqPriority Configuration

The L3 network on chip interconnect accepts signals that can influence the NoC's arbitration mechanisms. The software programmable configuration used to influence the arbitration within the L3 NoC on the DRA74x, DRA75x and TDA2x family of devices is referred to as 'Mflag'. Depending upon the implementation, this signal may either be driven directly by the initiators or through a configuration control provided in the device configuration/control module.

Section 2.1 discusses the list of initiators capable of driving this Mflag signal internally. Section 2.2 illustrates the list of initiators for which this signal can be driven through the device configuration and control module.

2.1 Initiator Driven Mflag

As mentioned in Section 2, this sub-section illustrates the list of initiators capable of driving the Mflag signal dynamically in order to influence the arbitration within the L3 interconnect.

The VIP and DSS are the two initiators capable of driving 'Mflag' dynamically. More details regarding how to configure this for the VIP and the DSS are provided in Section 2.1.1 and Section 2.1.2, respectively.

2.1.1 Video Input Port

The Video Input Port (VIP) L3 initiator port (VPDMA) arbitrates between multiple DMA sources within the video input port (VIP) based on FIFO levels of DMA channels connected to VPDMA. Priority escalation mechanism implemented within the video input port subsystem is based on overflow threshold and FIFO margin.

The following is a summary of priority and 'Mflag' levels provided by the VIP:

- High priority (MFlag = 3) when FIFO margin is below 25 %
- Medium priority (MFlag = 1) when FIFO margin is between 25 % and 50 %
- Low priority (MFlag = 0) when FIFO margin is above 50 %

NOTE: Note that the VIP 'Mflag' influences the arbitration within the L3 network on chip interconnect.

Additionally, the VIP subsystem also generates 'MReqPriority' based upon a programmed descriptor configuration. Please note that this 'MReqPriority' configuration influences the arbitration mechanism in the memory subsystem only and has no influence on arbitration that takes place within the L3 network on chip interconnect.

NOTE: For details regarding the video input port data transfer descriptor, see the device-specific technical reference manual.

Table 1. VIP Data Packet Descriptor Word 3 Details

Bits	Name	Description
31:27	Packet Type	Host Packet Descriptor Type = 0xa
26	Mode	0 = Normal
		1 = TILED
25	Direction	Inbound = 0
		Outbound = 1
24:16	Channel	Channel that this descriptor describes
15	Reserved	Reserved for future use
11:9	Priority	Only Bit 9 and Bit 11 are used to set the priority. Bit 10 is ignored.
		Highest = 0
		Lowest = 3
		By default, hardware assigns priority = 3. This priority level is used in the arbitration between the masters (for DDR access).
8:0	Next Channel	Next channel to execute on a line or the channel to use in the generated write descriptor

2.1.2 Display Subsystem (DSS)

The display subsystem comprises of four read pipelines viz, Graphics, Vid1, Vid2, and Vid3 and 1 write pipeline viz, WB.

NOTE: For the availability of pipelines within the display subsystem (as not all devices within a family support all the pipelines), see the device-specific data sheet and technical reference manual.

The display subsystem is capable of driving the 'Mflag' signal if any of read pipelines have been configured as high priority and FIFO levels for the configured pipeline are below the configured threshold values.

The pipelines Vid1 to Vid3 have a 32 KB FIFO each and the graphics pipeline has a 16-KB FIFO. The FIFO thresholds are measured in terms of 16-byte word.

Example 1 depicts a typical sequence to enable the display subsystem (DSS) 'Mflag' and to configure various pipeline threshold values.

NOTE: To configure the display subsystem 'Mflag' and pipeline threshold values, see the appropriate starterware APIs. For more information on starterware APIs, contact your TI sales representative. Additionally, for the display subsystem registers mapping summary, see the device-specific data sheet and technical reference manual.

Example 1. Typical Sequence to Enable Display Subsystem 'Mflag' and Configure Various Pipeline Threshold Values

```
// Enable MFlag Generation DISPC_GLOBAL_MFLAG_ATTRIBUTE
WR_REG32(0x5800185C,0x2);
// Set Video Pipe as High Priority DISPC_VIDx_ATTRIBUTES[23]
WR_REG32(0x580010CC,RD_REG32(0x580010CC) | 1<<23);
WR_REG32(0x580010CC,RD_REG32(0x5800115C) | 1<<23);
WR_REG32(0x580010CC,RD_REG32(0x58001370) | 1<<23);
// Set Graphics Pipe as High Priority DISPC_GFX_ATTRIBUTES[23]
WR_REG32(0x580010CC,RD_REG32(0x580010A0) | 1<<14);

// GFX threshold 75 % HT , 50 % LT
WR_REG32(0x58001860 , 0x03000200);
// VIDx threshold 75 % HT , 50 % LT
WR_REG32(0x58001864, 0x06000400);
WR_REG32(0x58001868, 0x06000400);
WR_REG32(0x5800186C, 0x06000400);
```

Recommended settings for high and low threshold are 75% and 50 % respectively. It is generally observed that having a low threshold at higher level degrades performance for other initiators in a concurrent use case scenario. Note that the above is just guidance and the system integrator is expected to determine the correct threshold configuration based on the overall system use case scenario.

2.2 Control Module Driven Mflag for Various Initiators

The 'Mflag' for the following initiators can be driven through the Control Module:

- MPU
- DSP1_CFG
- DSP2_CFG
- IPU1
- IPU2
- PRUSS1_PRU0
- PRUSS1_PRU1
- PRUSS2_PRU0
- PRUSS2_PRU1
- SATA
- MMC1
- MMC2
- USB1
- USB2
- USB3
- USB4
- DAP
- IEEE1500

Table 2 summarizes the control module register details for configuring 'Mflag' for various initiators in the system.

NOTE: For the availability of these peripherals and initiators (as not all devices within a family support every peripheral/initiator), see the device-specific data sheet and technical reference manual.

Table 2. Configuration Register Details for Driving 'Mflag' for Various Initiators for L3 Arbitration

Initiator	Register Name	Physical Address	BitField Name	BitField
MPU	CTRL_CORE_L3_INITIATOR_PRESSURE_1	0x4A00243C	MPU_L3_PRESSURE	[27:26]
DSP1_CFG	CTRL_CORE_L3_INITIATOR_PRESSURE_1	0x4A00243C	DSP1_CFG_L3_PRESSURE	[18:17]
DSP2_CFG	CTRL_CORE_L3_INITIATOR_PRESSURE_1	0x4A00243C	DSP2_CFG_L3_PRESSURE	[10:9]
IPU1	CTRL_CORE_L3_INITIATOR_PRESSURE_2	0x4A002440	IPU1_L3_PRESSURE	[13:12]
IPU2	CTRL_CORE_L3_INITIATOR_PRESSURE_2	0x4A002440	IPU2_L3_PRESSURE	[10:9]
PRUSS1_PRU0	CTRL_CORE_L3_INITIATOR_PRESSURE_2	0x4A002440	PRUSS1_PRU0_L3_PRESSURE	[7:6]
PRUSS1_PRU1	CTRL_CORE_L3_INITIATOR_PRESSURE_2	0x4A002440	PRUSS1_PRU1_L3_PRESSURE	[4:3]
PRUSS2_PRU0	CTRL_CORE_L3_INITIATOR_PRESSURE_2	0x4A002440	PRUSS2_PRU0_L3_PRESSURE	[1:0]
PRUSS2_PRU1	CTRL_CORE_L3_INITIATOR_PRESSURE_3	0x4A002444	PRUSS2_PRU1_L3_PRESSURE	[27:26]
SATA	CTRL_CORE_L3_INITIATOR_PRESSURE_5	0x4A00244C	SATA_L3_PRESSURE	[4:3]
MMC1	CTRL_CORE_L3_INITIATOR_PRESSURE_5	0x4A00244C	MMC1_L3_PRESSURE	[1:0]
MMC2	CTRL_CORE_L3_INITIATOR_PRESSURE_6	0x4A002450	MMC2_L3_PRESSURE	[18:17]
USB1	CTRL_CORE_L3_INITIATOR_PRESSURE_6	0x4A002450	USB1_L3_PRESSURE	[16:15]
USB2	CTRL_CORE_L3_INITIATOR_PRESSURE_6	0x4A002450	USB2_L3_PRESSURE	[13:12]
USB3	CTRL_CORE_L3_INITIATOR_PRESSURE_6	0x4A002450	USB3_L3_PRESSURE	[10:9]
USB4	CTRL_CORE_L3_INITIATOR_PRESSURE_6	0x4A002450	USB4_L3_PRESSURE	[7:6]
DAP	CTRL_CORE_L3_INITIATOR_PRESSURE_6	0x4A002450	CS_DAP_L3_PRESSURE	[4:3]
IEEE1500	CTRL_CORE_L3_INITIATOR_PRESSURE_6	0x4A002450	IEEE1500_L3_PRESSURE	[1:0]

Note that the default value of the bit fields is 0 and the bit fields can be programmed to a value of 0x0 (= lowest), 1 or 3 (=highest). Configuring a value of 2 would have same effect as 3. It is expected that this configuration is performed during system startup by the application and is not modified during runtime.

3 QoS Knobs Implemented Within the L3 Network on Chip Interconnect

This section discusses the QoS knobs implemented within the L3 network on chip interconnect for the DRA74x, DRA75x and TDA2x family of devices.

The two components within the L3 network on chip interconnect that aid to optimize overall system performance on this family of devices are:

- Bandwidth regulators
- Bandwidth limiters

3.1 Bandwidth Regulator

In order to ensure the required average target bandwidth to an initiator, the system integrators can make use of the bandwidth regulators instantiated on key L3 network on chip interconnect initiator ports.

Bandwidth regulator mechanism provides priority within interconnect/L3 to achieve average bandwidth to a particular initiator port. The bandwidth regulator is implemented for EVE, DSP1, DSP2, IVA, GPU, BB2D, MMU2, GMAC and PCIe subsystems.

Example 2 illustrates a typical sequence to configure a typical bandwidth regulator.

Example 2. Typical Sequence to Configure a Bandwidth Regulator

```
set_bw_regulator(port,average_bw, time_in_us ,high level, low level) {
base_address = get_bw_reg_base_address(port);
WR_REG32(base_address+0x8,int(ceil(average_bw/8.3125));
WR_REG32(base_address+0xC,(time_in_us*average_bw));
WR_REG32(base_address+0x10,(high_level + (low_level <<2)));
WR_REG32(base_address+0x14,0x1);
}
    get_bw_reg_base_address(port) {
    if ( port == “EVE1_TC0” ) {    return
L3_NOC_AVATAR__DEBUGSS_CS_DAP_INIT_OCP_L3_NOC_AVATAR_CLK1_EVE1_TC0_BW_REGULATOR
        ;    }
}
```

NOTE: To configure the bandwidth regulators available for various L3 interconnect initiator ports, see the device-specific starterware APIs.

Table 3 provides the base address details for various Bandwidth Regulators available on DRA74x, DRA75x and TDA2x family of devices

Table 3. Base Address Details for Various Bandwidth Regulators Available on DRA74x, DRA75x and TDA2x Family of Devices

BW Regulator Name	Physical Base Address
CLK1_2_MMU2_BW_REGULATOR	0x44803B00
CLK1_2_EVE1_TC0_BW_REGULATOR	0x44804200
CLK1_2_EVE2_TC0_BW_REGULATOR	0x44804300
CLK1_2_EVE3_TC0_BW_REGULATOR	0x44804400
CLK1_2_EVE4_TC0_BW_REGULATOR	0x44804500
CLK1_2_EVE1_TC1_BW_REGULATOR	0x44804600
CLK1_2_EVE2_TC1_BW_REGULATOR	0x44804700
CLK1_2_EVE3_TC1_BW_REGULATOR	0x44804800
CLK1_2_EVE4_TC1_BW_REGULATOR	0x44804900
CLK1_2_DSP2_EDMA_BW_REGULATOR	0x44804A00
CLK1_2_DSP1_EDMA_BW_REGULATOR	0x44804B00
CLK1_2_DSP1_MDMA_BW_REGULATOR	0x44804C00
CLK1_2_DSP2_MDMA_BW_REGULATOR	0x44804D00
CLK1_2_BB2D_P1_BW_REGULATOR	0x44804E00
CLK1_2_IVA1_BW_REGULATOR	0x44805000
CLK1_2_BB2D_P2_BW_REGULATOR	0x44805100
CLK1_2_GPU_P1_BW_REGULATOR	0x44805200
CLK1_2_GPU_P2_BW_REGULATOR	0x44805300
CLK1_2_PCIESS2_BW_REGULATOR	0x44805400
CLK1_2_PCIESS1_BW_REGULATOR	0x44805500
CLK1_2_GMAC_SW_BW_REGULATOR	0x44805600

NOTE: If an L3 initiator port has 2 ports then average bandwidth requirement can be split in half per port. It's possible to have a bandwidth imbalance across the 2 ports for an initiator. In such a case, bandwidth requirement may be set separately for each L3 interconnect initiator port.

3.2 Bandwidth Limiter

Bandwidth Limiter mechanism limits a particular initiator from consuming excessive bandwidth. This mechanism is implemented for EDMA, MMU1, VPE, GPU and BB2D.

Example 3 illustrates a typical sequence to configure a typical bandwidth limiter. Pseudo sequence to enable bandwidth limiter is as following.

Example 3. Typical Sequence to Configure a Bandwidth Limiter

```
set_bw_limiter(port,limit_bw) {
base_address = get_bw_limiter_base_address(port);
bandwidth = int(limit_bw / 8.3125);
bandwidth_int = ( bandwidth & 0xFFFFFFE0 ) >> 5;
bandwidth_frac = ( bandwidth & 0x1F );
WR_REG32(base_address+0x8,bandwidth_frac);
WR_REG32(base_address+0xC,bandwidth_int);
WR_REG32(base_address+0x10,0x0);
WR_REG32(base_address+0x14,0x1);
}
    get_bw_limiter_base_address(port) {
    if ( port == “VPE_P2” ) {    return
L3_NOC_AVATAR__DEBUGSS_CS_DAP_INIT_OCP_L3_NOC_AVATAR_CLK1_VPE_P2_BW_LIMITER
    ;    }
}
```

NOTE: To configure the bandwidth Limiters available for various L3 interconnect initiator ports, see the appropriate starterware APIs.

Table 4 provides the base address details for various Bandwidth Limiters available on DRA74x, DRA75x and TDA2x family of devices.

Table 4. Base Address Details for Various Bandwidth Limiters Available on DRA74x, DRA75x and TDA2x Family of Devices

Bandwidth Limiter Name	Physical Base Address
CLK1_2_MMU1_BW_LIMITER	0x44803A00
CLK1_2_TPTC1_RD_BW_LIMITER	0x44803C00
CLK1_2_TPTC2_RD_BW_LIMITER	0x44803D00
CLK1_2_TPTC1_WR_BW_LIMITER	0x44803E00
CLK1_2_TPTC2_WR_BW_LIMITER	0x44803F00
CLK1_2_VPE_P2_BW_LIMITER	0x44804000
CLK1_2_VPE_P1_BW_LIMITER	0x44804100
CLK1_2_BB2D_P1_BW_LIMITER	0x44805900
CLK1_2_BB2D_P2_BW_LIMITER	0x44805A00
CLK1_2_GPU_P1_BW_LIMITER	0x44805B00
CLK1_2_GPU_P2_BW_LIMITER	0x44805C00

NOTE: If an L3 initiator port has 2 ports then average bandwidth requirement can be split in half per port. It's possible to have a bandwidth imbalance across the 2 ports for an initiator. In such a case, bandwidth requirement may be set separately for each L3 interconnect initiator port.

4 Dynamic Memory Manager (DMM)

4.1 Priority Extension Generation (PEG) Priority

DMM implements PEG registers to pass 'MReqPriority' to EMIF controller. The incoming transactions are appended with the priority as per values configured by the users in DMM PEG Priority Registers.

4.1.1 DMM PEG Priority Registers

Figure 2. DMM_PEG_PRIO7: 0x4E00_063C

31 28	27 24	23 20	19 16	15 12	11 8	7 4	3 0
$PRIO_{63}$	$PRIO_{62}$	$PRIO_{61}$	$PRIO_{60}$	$PRIO_{59}$	$PRIO_{58}$	$PRIO_{57}$	$PRIO_{56}$
W7 P7	W6 P6	W5 P5	W4 P4	W3 P3	W2 P2	W1 P1	W0 P0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Figure 3. DMM_PEG_PRIO7: 0x4E00_0620

31 28	27 24	23 20	19 16	15 12	11 8	7 4	3 0
$PRIO_{7}$	$PRIO_{6}$	$PRIO_{5}$	$PRIO_{4}$	$PRIO_{3}$	$PRIO_{2}$	$PRIO_{1}$	$PRIO_{0}$
W7 P7	W6 P6	W5 P5	W4 P4	W3 P3	W2 P2	W1 P1	W0 P0

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

A 3 bit priority is set in Px bitfield and Wx bit is used as write enable for corresponding Px field.

Wx bitfield allows you to modify a single Px entry in a DMM PEG priority register without a read-modify-write sequence.

Table 5 illustrates the base address details for DMM PEG Priority Registers for DRA74x, DRA75x and TDA2x family of devices.

NOTE: For the availability of these peripherals and initiators (as not all devices within a family support every peripheral and initiator), see the device-specific data sheet.

Table 5. Base Address Details for DMM PEG Priority Registers for DRA74x, DRA75x and TDA2x Family of Devices

Initiator	Register Name	Register Address	BitField Name
MPU	DMM_PEG_PRIO0	0x4E00_0620	PRIO0
DEBUGSS_CS_DAP	DMM_PEG_PRIO0	0x4E00_0620	PRIO4
IEEE1500_2_OCP	DMM_PEG_PRIO0	0x4E00_0620	PRIO5
DSP1_MDMA	DMM_PEG_PRIO1	0x4E00_0624	PRIO8
DSP1_CFG	DMM_PEG_PRIO1	0x4E00_0624	PRIO9
DSP1_EDMA	DMM_PEG_PRIO1	0x4E00_0624	PRIO10
DSP2_EDMA	DMM_PEG_PRIO1	0x4E00_0624	PRIO11
DSP2_CFG	DMM_PEG_PRIO1	0x4E00_0624	PRIO12
DSP2_MDMA	DMM_PEG_PRIO1	0x4E00_0624	PRIO13
IVA	DMM_PEG_PRIO1	0x4E00_0624	PRIO14
EVE1 TC0	DMM_PEG_PRIO2	0x4E00_0628	PRIO16
EVE2 TC0	DMM_PEG_PRIO2	0x4E00_0628	PRIO17

Table 5. Base Address Details for DMM PEG Priority Registers for DRA74x, DRA75x and TDA2x Family of Devices (continued)

Initiator	Register Name	Register Address	BitField Name
EVE3 TC0	DMM_PEG_PRIO2	0x4E00_0628	PRIO18
EVE4 TC0	DMM_PEG_PRIO2	0x4E00_0628	PRIO19
PRUSS1_PRU1	DMM_PEG_PRIO2	0x4E00_0628	PRIO20
PRUSS1_PRU2	DMM_PEG_PRIO2	0x4E00_0628	PRIO21
PRUSS2_PRU1	DMM_PEG_PRIO2	0x4E00_0628	PRIO22
PRUSS2_PRU2	DMM_PEG_PRIO2	0x4E00_0628	PRIO23
IPU1	DMM_PEG_PRIO3	0x4E00_062C	PRIO24
IPU2	DMM_PEG_PRIO3	0x4E00_062C	PRIO25
DMA_SYSTEM_RD	DMM_PEG_PRIO3	0x4E00_062C	PRIO26
DMA_SYSTEM_WR	DMM_PEG_PRIO3	0x4E00_062C	PRIO26
EDMA_TC0_WR	DMM_PEG_PRIO3	0x4E00_062C	PRIO28
EDMA_TC0_RD	DMM_PEG_PRIO3	0x4E00_062C	PRIO28
EDMA_TC1_WR	DMM_PEG_PRIO3	0x4E00_062C	PRIO29
EDMA_TC1_RD	DMM_PEG_PRIO3	0x4E00_062C	PRIO29
DSS	DMM_PEG_PRIO4	0x4E00_0630	PRIO32
MLB	DMM_PEG_PRIO4	0x4E00_0630	PRIO33
MMU1	DMM_PEG_PRIO4	0x4E00_0630	PRIO33
PCIESS_P1	DMM_PEG_PRIO4	0x4E00_0630	PRIO34
PCIESS_P2	DMM_PEG_PRIO4	0x4E00_0630	PRIO35
MMU2	DMM_PEG_PRIO4	0x4E00_0630	PRIO35
VIP1 P1	DMM_PEG_PRIO4	0x4E00_0630	PRIO36
VIP1 P2	DMM_PEG_PRIO4	0x4E00_0630	PRIO36
VIP2 P1	DMM_PEG_PRIO4	0x4E00_0630	PRIO37
VIP2 P2	DMM_PEG_PRIO4	0x4E00_0630	PRIO37
VIP3 P1	DMM_PEG_PRIO4	0x4E00_0630	PRIO38
VIP3 P2	DMM_PEG_PRIO4	0x4E00_0630	PRIO38
VPE P1	DMM_PEG_PRIO4	0x4E00_0630	PRIO39
VPE P2	DMM_PEG_PRIO4	0x4E00_0630	PRIO39
MMC1	DMM_PEG_PRIO5	0x4E00_0634	PRIO40
GPU_P1	DMM_PEG_PRIO5	0x4E00_0634	PRIO40
MMC2	DMM_PEG_PRIO5	0x4E00_0634	PRIO41
GPU_P2	DMM_PEG_PRIO5	0x4E00_0634	PRIO41
BB2D P1	DMM_PEG_PRIO5	0x4E00_0634	PRIO42
BB2D P2	DMM_PEG_PRIO5	0x4E00_0634	PRIO42
GMAC_SW	DMM_PEG_PRIO5	0x4E00_0634	PRIO43
USB1	DMM_PEG_PRIO5	0x4E00_0634	PRIO44
USB2	DMM_PEG_PRIO5	0x4E00_0634	PRIO45
USB3	DMM_PEG_PRIO5	0x4E00_0634	PRIO46
USB4	DMM_PEG_PRIO5	0x4E00_0634	PRIO47
SATA	DMM_PEG_PRIO6	0x4E00_0638	PRIO51
EVE1 TC1	DMM_PEG_PRIO6	0x4E00_0638	PRIO52
EVE2 TC1	DMM_PEG_PRIO6	0x4E00_0638	PRIO53
EVE3 TC1	DMM_PEG_PRIO6	0x4E00_0638	PRIO54
EVE4 TC1	DMM_PEG_PRIO6	0x4E00_0638	PRIO55

4.2 Mflag

The DMM offers two emergency vectors (2 x 64bits – one per TILER port). These emergency vectors are indexed by 6 MSBs of the connID. The mflag driven by certain IPs can be connected to the two emergency vectors, in order to prioritize the traffic of one TILER port over the other.

For DRA74x, DRA75x and TDA2x family of devices, the 'mflag' from the following L3 initiator ports are connected to the emergency vectors:

- DSS
- VIP

To enable the emergency feature on DMM, the 'ENABLE' bit in the DMM_EMERGENCY register needs to be set. The DMM_EMERGENCY Register is accessible at the following address: 0x4E00_0020.

For details regarding base address for DMM PEG priority registers, see Figure 4 and Table 6.

Figure 4. DMM_EMERGENCY Register

31	21	20	16
Reserved		WEIGHT	
R		R/W	

15	1	0
Reserved		ENABLE
RO		R/W

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 6. DMM_EMERGENCY Register Field Descriptions

Bit	Field	Value	Description
31-21	Reserved	0	Reserved
20-16	WEIGHT		Weight for the Lisa arbitration when any bit of the vector Mflag[63:0] is set . The recommendation is to set this field to 0x8 with ENABLE=1, after reset.
15-1	Reserved	0	Reserved
0	ENABLE	0	Emergency feature is disabled.
		1	Enable the emergency feature. Lisa arbitration priority is higher for the initiator that set emergency input of this initiator The recommendation is to enable the feature (=1) after reset.

5 External Memory Interface (EMIF)

The EMIF can be configured to either provide class of service or use mflag that is dynamically driven from several key L3 initiators – namely, VIP and DSS on the DRA74x, DRA75x and TDA2x family of devices. These two features (class of service and mflag) are mutually exclusive and are configurable using the MFLAG_OVERRIDE bit in the EMIF_READ_WRITE_EXECUTION_THRESHOLD configuration register. For details regarding this register, see Figure 5 and Table 7.

Figure 5. EMIF_READ_WRITE_EXECUTION_THRESHOLD Register

31	30	29 ... 16
MFLAG_OVERRIDE	RSVD	Reserved

15 ... 13	12 ... 8	7 ... 5	4 ... 0
Reserved	WR_THRSH	Reserved	RD_THRSH

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 7. EMIF_READ_WRITE_EXECUTION_THRESHOLD Register Field Descriptions

Bit	Field	Value	Description
31	MFLAG_ OVERRIDE	0	Use MFLAG
		1	Use Priority Class of Service
30	RSVD	0	Reserved
29-13	Reserved	0	Reserved - writes are ignored, always reads zeros.
12-8	WR_THRSH		Write Threshold. Number of SDRAM write bursts after which the EMIF arbitration will switch to executing read commands. The value programmed is always minus one the required number.
7-5	Reserved	0	Reserved - writes are ignored, always reads zeros.
4-0	RD_THRSH		Read Threshold. Number of SDRAM read bursts after which the EMIF arbitration will switch to executing write commands. The value programmed is always minus one the required number.

5.1 Class of Service

The commands in the EMIF Command FIFO can be mapped to two classes of service namely: 1 and 2. The mapping of commands to a particular class of service can be done based on the priority or the master ID (also referred to as connID).

5.1.1 Priority Based Class of Service

The mapping based on priority can be done by setting the appropriate values in the EMIF_PRIORITY_TO_CLASS_OF_SERVICE_MAPPING register.

For EMIF1, this register is accessible at the base address 0x4C00 0100 and for EMIF2, this register is accessible at the base address 0x4D00 0100.

NOTE: For the availability of the number of EMIF instances on your device (as not all devices within a family support both instances), see the device-specific data sheets.

For details regarding this register, see Figure 6 and Table 8.

Figure 6. EMIF_PRIORITY_TO_CLASS_OF_SERVICE_MAPPING Register

31	30	16
PRI_COS_MAP_EN	Reserved	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRI_7_COS		PRI_6_COS		PRI_5_COS		PRI_4_COS		PRI_3_COS		PRI_2_COS		PRI_1_COS		PRI_0_COS	

LEGEND: R/W = Read/Write; R = Read only; *-n* = value after reset

Table 8. EMIF_PRIORITY_TO_CLASS_OF_SERVICE_MAPPING Register Field Descriptions

Bit	Field	Value	Description
31	PRI_COS_ MAP_EN		Set 1 to enable priority to class of service mapping. Set 0 to disable mapping.
30-16	Reserved	0	Reserved - writes are ignored, always reads zeros.
15-14	PRI_7_COS		Class of service for commands with priority of 7. Value can be 1, 2, or 3. Setting a value of 0 will have similar effects as a value of 3.
13-12	PRI_6_COS		Class of service for commands with priority of 6. Value can be 1, 2, or 3. Setting a value of 0 will have similar effects as a value of 3.
11-10	PRI_5_COS		Class of service for commands with priority of 5. Value can be 1, 2, or 3. Setting a value of 0 will have similar effects as a value of 3.
9-8	PRI_4_COS		Class of service for commands with priority of 4. Value can be 1, 2, or 3. Setting a value of 0 will have similar effects as a value of 3.
7-6	PRI_3_COS		Class of service for commands with priority of 3. Value can be 1, 2, or 3. Setting a value of 0 will have similar effects as a value of 3.
5-4	PRI_2_COS		Class of service for commands with priority of 2. Value can be 1, 2, or 3. Setting a value of 0 will have similar effects as a value of 3.
3-2	PRI_1_COS		Class of service for commands with priority of 1. Value can be 1, 2, or 3. Setting a value of 0 will have similar effects as a value of 3.
1-0	PRI_0_COS		Class of service for commands with priority of 0. Value can be 1, 2, or 3. Setting a value of 0 will have similar effects as a value of 3.

Each class of service has an associated latency counter (EMIF_COS_CONFIG[23:16] COS_COUNT_1 and EMIF_COS_CONFIG[15:8] COS_COUNT_2).

For EMIF1, this register is accessible at the base address 0x4C00 0124 and for EMIF2, this register is accessible at the base address 0x4D00 0124.

When the latency counter for a command expires, that is, it reaches the value programmed for the class of service that the command belongs to, that command is executed next. If there is more than one command that has expired latency counter, the command with the highest priority is executed first. One exception to this rule is, if the EMIF_COS_CONFIG[7:0] PR_OLD_COUNT value expires for the oldest command in the queue. That command is executed first irrespective of priority or class of service. This is done to prevent the continuous blocking effect.

The EMIF_COS_CONFIG[7:0] PR_OLD_COUNT value is used to identify when the oldest command in the command FIFO has timed out. At this point during the arbitration process, this oldest command is issued regardless of the priority of the other commands in the FIFO. This feature is disabled when writing 0x0 to the EMIF_COS_CONFIG[7:0] PR_OLD_COUNT bit field. After issuing the oldest command, the other remaining commands in the FIFO are reordered by age. The next oldest command in the FIFO is given highest priority again and issued after the EMIF_COS_CONFIG[7:0] PR_OLD_COUNT value expires. If a new value in the PR_OLD_COUNT bit field is written during counting, that is, before PR_OLD_COUNT expires, the counter keeps working but if this value is smaller the oldest command is issued sooner and if this value is larger the oldest command is issued later.

For details regarding this register, see Figure 7 and Table 9.

Figure 7. EMIF_COS_CONFIG Register

31 24	23 16	15 8	7 0
Reserved	COS_COUNT_1	COS_COUNT_2	PR_OLD_COUNT

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 9. EMIF_COS_CONFIG Register Field Descriptions

Bit	Field	Value	Description
30-24	Reserved	0	Reserved
23-16	COS_COUNT_1		Priority raise counter for class of service 1. Number of EMIF_FICLK cycles after which the EMIF momentarily raises the priority of the class of service 1 commands in the Command FIFO. A value of N will be equal to N x 16 clocks.
15-8	COS_COUNT_2		Priority raise counter for class of service 2. Number of EMIF_FICLK cycles after which the EMIF momentarily raises the priority of the class of service 1 commands in the Command FIFO. A value of N will be equal to N x 16 clocks.
7-0	PR_OLD_COUNT		Priority raise old counter. Number EMIF_FICLK cycles after which the EMIF momentarily raises the priority of the oldest command in the Command FIFO. A value of N will be equal to N x 16 clocks.

5.1.2 ConnID Based Class of Service

The mapping based on master ID can be done by setting the appropriate values of master ID and the masks in the EMIF_CONNECTION_ID_TO_CLASS_OF_SERVICE_1_MAPPING and EMIF_CONNECTION_ID_TO_CLASS_OF_SERVICE_2_MAPPING registers. There are 3 master ID and mask values that can be set for each class of service. In conjunction with the masks, each class of service can have a maximum of 144 master IDs mapped to it. For example, a master ID value of 0xFF along with a mask value of 0x3 will map all master IDs from 0xF8 to 0xFF to that particular class of service.

For EMIF1, this register is accessible at the base address 0x4C00 0104 and for EMIF2, this register is accessible at the base address 0x4D00 0104.

For details regarding this register, see Figure 8 and Table 10.

Figure 8. EMIF_CONNECTION_ID_TO_CLASS_OF_SERVICE_1_MAPPING Register

31	30 23	22 20	19 16
CONNID_COS_1_MAP_EN	CONNID_1_COS_1	CONNID_1_COS_1	CONNID_2_COS_1

15 12	11 10	9 2	1 0
CONNID_2_COS_1	MSK_2_COS_1	CONNID_3_COS_1	MSK_3_COS_1

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 10. EMIF_CONNECTION_ID_TO_CLASS_OF_SERVICE_1_MAPPING Register Field Descriptions

Bit	Field	Value	Description
31	CONNID_COS_1_MAP_EN		Set 1 to enable Connection ID to class of service 1 mapping. Set 0 to disable mapping.
30-23	CONNID_1_COS_1		Connection ID value 1 for class of service 1.

Table 10. EMIF_CONNECTION_ID_TO_CLASS_OF_SERVICE_1_MAPPING Register Field Descriptions (continued)

Bit	Field	Value	Description
22-20	MSK_1_COS_1		Mask for Connection ID value 1 for class of service 1.
		0	Disables masking
		1	Masks Connection ID bit 0
		2	Masks Connection ID bits 1:0
		3	Masks Connection ID bits 2:0
		4	Masks Connection ID bits 3:0
		5	Mask Connection ID bits 4:0
		6	Masks Connection ID bits 5:0
		7	Masks Connection ID bits 6:0
19-12	CONNID_2_ COS_1		Connection ID value 2 for class of service 1.
11-10	MSK_2_COS_1		Mask for Connection ID value 2 for class of service 1.
		0	Disables masking
		1	Masks Connection ID bit 0
		2	Masks Connection ID bits 1:0
		3	Masks Connection ID bits 2:0
9-2	CONNID_3_ COS_1		Connection ID value 3 for class of service 1.
1-0	MSK_3_COS_1		Mask for Connection ID value 3 for class of service 1. Value of 0 will disable masking. Value of 1 will mask Connection ID bit 0. Value of 2 will mask Connection ID bits 1:0. Value of 3 will mask Connection ID bits 2:0.
		0	Disables masking
		1	Masks Connection ID bit 0
		2	Masks Connection ID bits 1:0
		3	Masks Connection ID bits 2:0

The master ID mapping allows the same master ID to be put in both class of service 1 and 2. Also, a transaction might belong to one class of service if viewed by master ID and might belong to another class of service if viewed by priority. In these cases, the command will belong to both class of service. The EMIF will try executing the command as soon as possible, when the smaller of the two counters (EMIF_COS_CONFIG[23:16] COS_COUNT_1 and EMIF_COS_CONFIG[15:8] COS_COUNT_2) expires.

5.2 Mflag

When mflag is being used, the EMIF SYS port would be prioritized over the EMIF MPU port in case of mflag assertion.

For details regarding enabling mflag feature for EMIF, see Table 7.

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Original (August 2016) to A Revision	Page
• Update was made to Section 2.1.1.	3
• Updates were made in Table 3.	6
• Updates were made in Table 4.	7
• Renamed Section 4.2 from Emergency to Mflag.	10

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as “components”) are sold subject to TI’s terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI’s terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers’ products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers’ products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI’s goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or “enhanced plastic” are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have ***not*** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265