TMS470R1x Direct Memory Access (DMA) Controller Reference Guide

Literature Number: SPNU194
November 2002

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

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

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

Reproduction of 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. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

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

Mailing Address:

Texas Instruments
Post Office Box 655303
Dallas, Texas 75265

REVISION HISTORY

REVISION	DATE	NOTES
*	11/02	*Initial Version*

Contents

Figures

Tables

Direct Memory Access (DMA) Controller

The direct memory access (DMA) controller transfers data between memory locations. The data transfers take place in parallel with CPU activity, maximizing system performance.

Topic **Page**

1 Overview

The direct memory access (DMA) controller is used to transfer data between any locations in the TMS470 memory map except the system control registers - for more information on this, see the *TMS470R1x System Module Reference Guide* (literature number SPNU189).

The DMA controller is available in two configurations:

- ❏ 32 control packets and 16 channels
- ❏ 16 control packets and 8 channels

The DMA controller supports the following key features:

- ❏ Transfers in byte, half-word, or word
- ❏ Transfers in request mode or non-request mode
- ❏ Status of DMA completion through interrupt or polling
- ❏ Channel-level and global-level configuration
- ❏ Data chaining

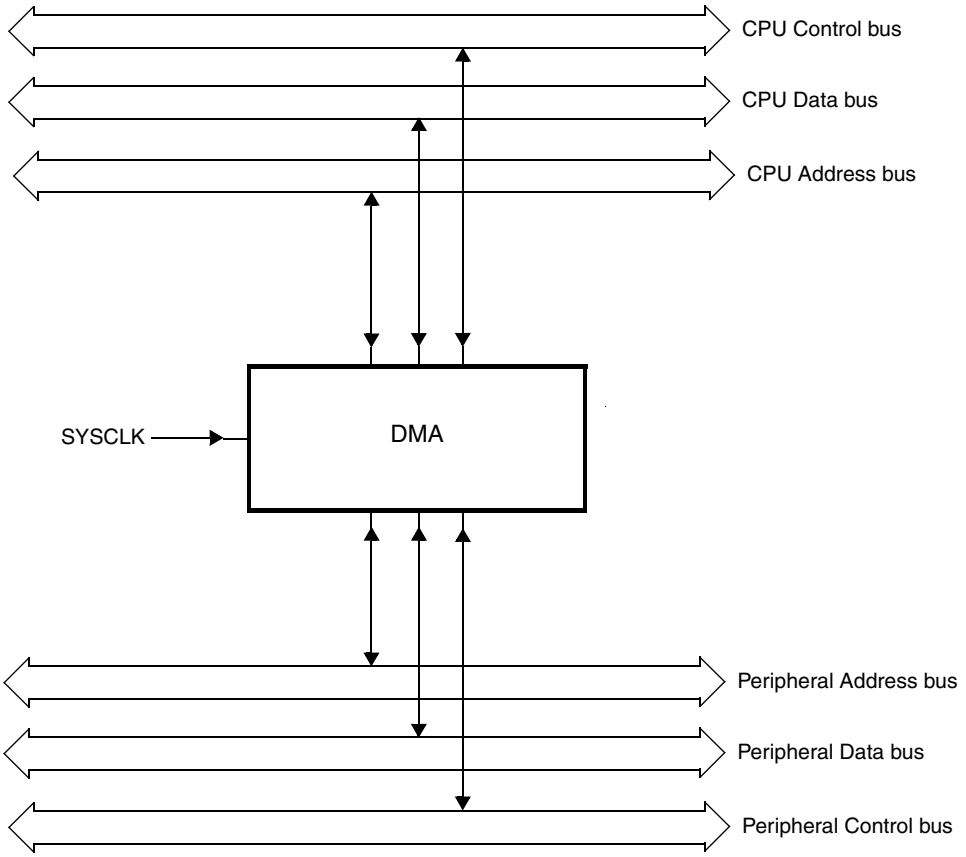
Note:

The information presented in this guide applies to a DMA configuration with 32 control packets and 16 channels. The DMA contoller configuration is device specific. Refer to the device data sheet for more information.

2 Functional Description of the DMA Controller

The DMA controller connects to the CPU bus and the peripheral bus, as shown in Figure 1, allowing the DMA controller to transfer data between any locations in the TMS470 memory map (except for the system control registers).

Figure 1. DMA Controller System Level Diagram



Data can be transferred concurrently with CPU transactions as long as there is no resource conflict (such as may occur when both the CPU and DMA controller attempt to access the same bus). The main advantage of DMA controller is that it can complete data transfers independent of the CPU, so that the CPU time is not spent on data transfers. As the data transfers are not interrupt driven, the system performance is maximized.

3 DMA Controller Registers

The DMA controller registers can be grouped into two sets - control packets and DMA channel and global registers. (See Figure 2)

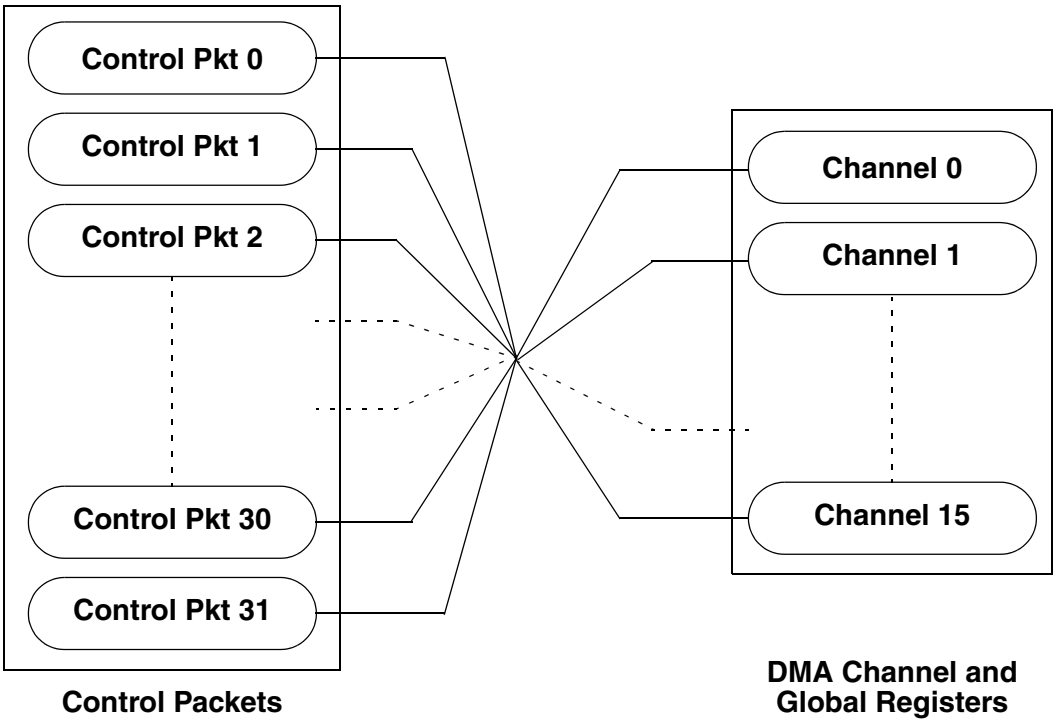
Control Packets (a set of four registers each) that configure the data transfers in regard to:

- ❏ access type (i.e. byte, half-word, word)
- ❏ the number of DMA transactions to perform, etc.

DMA Channel and Global Registers that configure:

- ❏ channel specific controls for the transfer
  - ■ how the DMA controller responds to emulation mode.
  - ■ what level interrupt to generate, etc.
- ❏ global controls for the transfer
  - ■ how the DMA controller behaves during an interrupt.
  - ■ how the DMA controller prioritizes the channels, etc.

Figure 2. DMA Channel - Control Packet Association



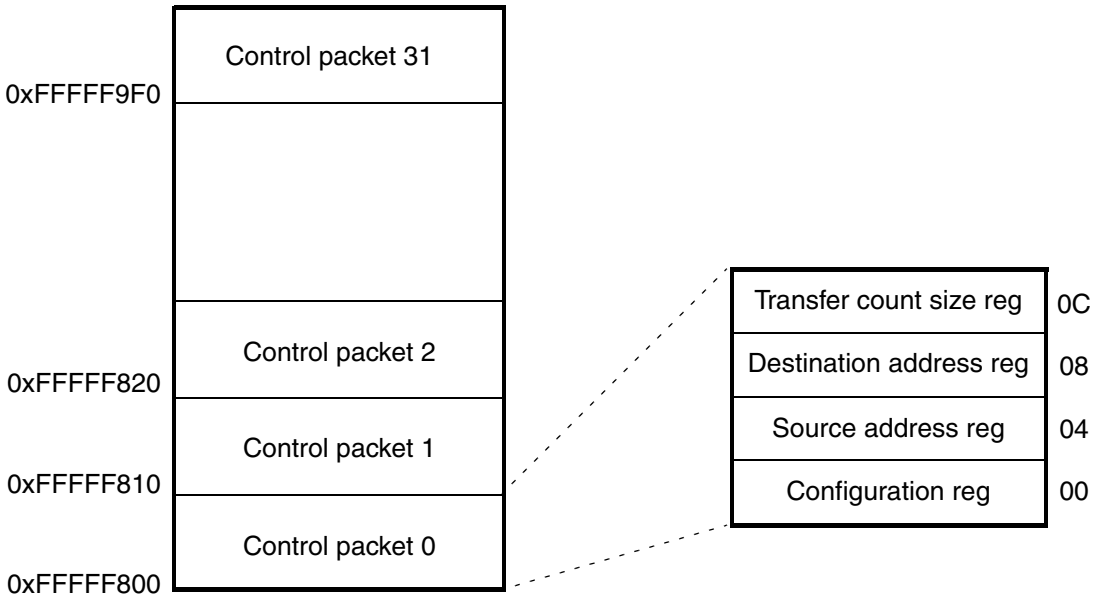
3.1 Control Packet Registers

There are 32 control packets (0 to 31) in the DMA command buffer RAM. Each control packet consists of the following four 32-bit fields:

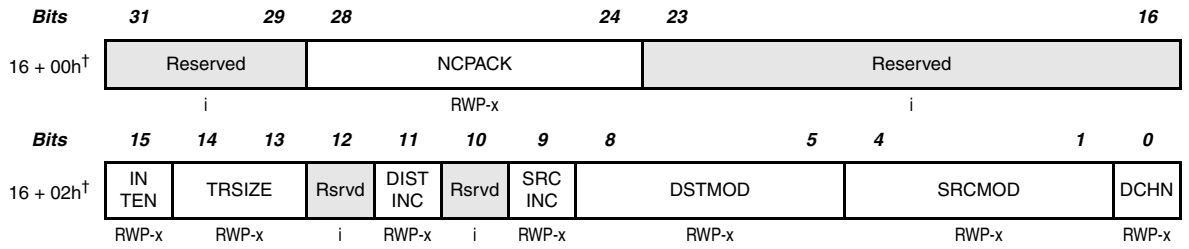
- ❏ DMAC (DMA Configuration) register configures the data transfer;
- ❏ DMASA (DMA Source Address) register contains the source address;
- ❏ DMADA (DMA Destination Address) register contains the destination address;
- ❏ DMATC (DMA Transfer Count) register contains the DMA controller transaction counter.

Each DMA channel needs at least one control packet for the channel setup. The control packets have to be programmed before enabling the DMA channel. If data chaining (DMAC.1) is enabled, the selected channel uses more than one control packet for consecutive block transfers (number of transactions performed until the transfer count becomes zero). See section 3.1.4. Before switching to another channel, the DMA controller saves back the current control packet registers. The DMA command buffer is placed in the system control registers at address range 0xFFFFF800 through 0xFFFFF9FF, as shown in Figure 3. Each control packet is aligned in a four-word boundary (only word writes are supported).

Figure 3. DMA Command Buffer Memory Map



3.1.1 DMA Configuration (DMAC) Register



RWP = Read in all modes, write in privilege mode only; i= Read values are indeterminate, -x = unknown after reset

Bits 31:29 **Reserved**

Bits 28:24 **NCPACK:** Next Control Packet

The next control packet to be used for block transfer after completion of the current one. This data field is valid only when data chaining (DMAC.0) is enabled for the corresponding channel. The table below shows the control packet number referenced by the value in this field.

Table 1. Next Control Packet Bits Encoding

NCPACK.4:0	Control Packet
00000	Control packet 0
00001	Control Packet 1
:	:
:	:
11110	Control Packet 30
11111	Control Packet 31

Bits 23:16 **Reserved**

Bit 15 **INTEN:** Interrupt Enable

INTEN = 1, the DMA controller generates an interrupt after the block transfer counter (DMATC.15:0) reaches zero for the corresponding DMA channel. If data chaining (DMAC.0) is enabled, for each control packet the interrupt is generated when its corresponding transfer counter (DMATC 15:0) reaches zero. The interrupt is deasserted when the CPU reads the DMA interrupt offset register.

INTEN = 0, an interrupt is not generated after a block transfer by this DMA channel is complete.

Bits 14:13 **TRSIZE:** Transfer Access Size

The DMA transaction size for the corresponding DMA channel.

00 =An 8-bit read/write
01 =A 16-bit read/write
10 =A 32-bit read/write
11 =Reserved

Bit 12 **Reserved**

Bit 11 **DSTINC:** Destination Address Increment

DSTINC = 0, the destination address remains constant after the DMA write is performed. This allows the data to be written to the same location.

DSTINC = 1, the destination address is incremented after the write is performed. The address is incremented depending on the transfer access size (TRSIZE) – one for byte, two for half-word, and four for word. If the destination address is expansion bus peripheral, the destination address is always incremented by four, irrespective of the transaction size (TRSIZE).

Bit 10 **Reserved**

Bit 9 **SRCINC:** Source Address Increment

SRCINC = 0, the source address remains constant after the DMA read is performed.

SRCINC = 1, the source address is incremented after the read operation from the source module. The address is incremented depending on the transfer access size (TRSIZE) – one for byte, two for half-word, and four for word. If the destination address is peripheral, the source address is always incremented by four, irrespective of the transaction size (TRSIZE).

Bits 8:5 **DSTMOD:** Destination Module

The destination module where the data is to be written. Table 2 describes the encoding of these bits.

Bits 4:1 **SRCMOD:** Source Module

Indicates the source module from where the data is read. Table 2 describes the encoding of these bits.

Table 2. DSTMOD and SRCMOD Bit Encoding

Bit Encoding	Description
0000	Memory module controlled by fine memory select 0.
0001	Memory module controlled by fine memory select 1.
0010	Memory module controlled by fine memory select 2.
0011	Memory module controlled BY fine memory select 3.
0100	Memory module controlled by fine memory select 4.
0101	Memory module controlled by fine memory select 5.
0110	Memory module controlled by coarse memory select 0.
0111	Memory module controlled by coarse memory select 1.
1000	Memory module controlled by coarse memory select 2.
1001	Memory module controlled by coarse memory select 3.
1010	Memory module controlled by coarse memory select 4.
1011	Memory module controlled by coarse memory select 5.
1100	Memory module with own address decoding on the CPU bus.
1101	Reserved
1110	Reserved
1111	Expansion bus peripherals

Bit 0 **DCHN:** Data Chaining Enable

If DCHN = 1 and the control packet's CPS (DMACPSR.31:0) bit = 1, on completion of the current block transfer, the DMA controller uses the control packet pointed by NCPACK for the next block transfer. If the CPS bit of the new control packet pointed by NCPACK is set to zero, the DMA controller waits until the application program sets it to one.

3.1.2 DMA Source Address (DMASA) Register

Bits	31	16
16 + 00h†	SOURCEADDR[31:16]	
	RWP-x	

Bits	15	0
16 + 02h†	SOURCEADDR[15:0]	
	RWP-x	

RWP = Read in all modes, write in privilege mode only; i= Read values are indeterminate, -x = unknown after reset

Bits 31:0 **SOURCEADDR.31:0:** Source Address

The address from where the data is to be read.

After each DMA read is performed, the source address is modified according to the value in the SRCINC bit (DMACW.9). The value is updated in this register only under the following circumstances:

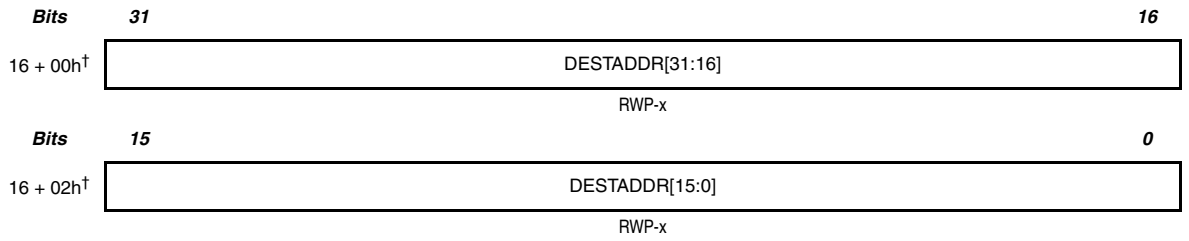
- ❏ upon completion of the block transfer
- ❏ when the DMA controller switches to another channel
- ❏ or, when the program sets the DMA_STOP bit (DMAGD.1) or DMA_HALT bit (DMAGD.0)

Note:

In peripheral addressing, bits 0 to 32 form the source address. However, in memory addressing, offset is sufficient, meaning that though the complete address is given, only the offset will be considered (ignoring the higher bits).

Addresses are aligned according to the transfer access size (DMAC.14:13). The address is word aligned for 32-bit transfers (lower two bits – bits 0 and 1 – are masked) and half-word aligned for 16-bit transfers (bit 0 is masked).

3.1.3 DMA Destination Address (DMADA) Register



RWP = Read in all modes, write in privilege mode only; i= Read values are indeterminate, -x = unknown after reset

Bits 31:0 **DESTADDR.31:0:** Destination Address

The address where the data is to be written.

After each DMA write is performed, the destination address is modified according to the value in the DISTINC bit (DMACW.11). The value is updated in this register only under the following circumstances:

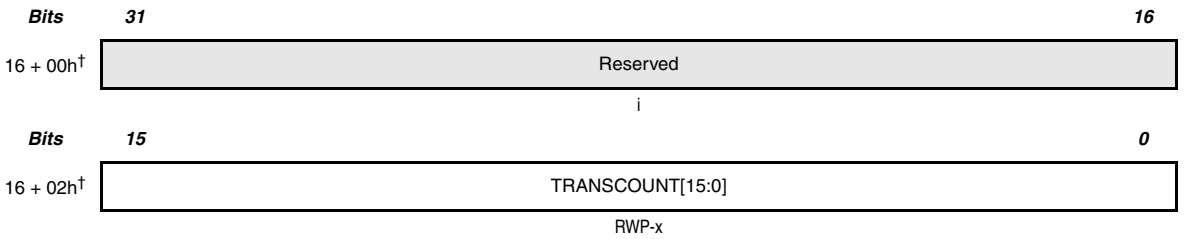
- ❏ upon completion of the block transfer
- ❏ when the DMA controller switches to another channel
- ❏ or, when the program sets the DMA_STOP bit (DMAGD.1) or DMA_HALT bit (DMAGD.0)

Note:

In peripheral addressing, bits 0 to 32 form the destination address. However, in memory addressing, offset is sufficient, meaning that though the complete address is given, only the offset will be considered (ignoring the higher bits).

Addresses are aligned according to the transfer access size (DMAC.14:13). The address is word aligned for 32-bit transfers (lower two bits – bits 0 and 1 – are masked) and half-word aligned for 16-bit transfers (bit 0 is masked).

3.1.4 DMA Transfer Count (DMATC) Register



RWP = Read in all modes, write in privilege only, i = Read values are indeterminate, -x = Unknown after reset

Bits 31:16 **Reserved**

Bits 15:0 **TRANSCOUNT.15:0:** Transfer Count

The number of DMA transactions (one read operation from the source and one write operation to the destination) to be completed. After each transaction the transfer count is decremented, and is updated only under the following circumstances:

- ❏ upon completion of the block transfer (TRANSCOUNT.15:0 reaches zero)
- ❏ when the DMA controller switches to another channel
- ❏ or, when the program sets the DMA_STOP bit (DMAGD.1) or DMA_HALT bit (DMAGD.0)

See Table 3 for transfer count and transfer access size (TRSIZE) bit encoding.

Table 3. Transfer Count Bit Encoding

	Block Size of the Data to be Transferred		
Bit Encoding	**TRSIZE = 00**	**TRSIZE = 01**	**TRSIZE = 10**
0000h	0	0	0
0001h	1 byte	1 half-word	1 word
:	:	:	:
:	:	:	:
:	:	:	:
FFFEh	65534 bytes	65534 half-words	65534 words
FFFFh	65535 bytes	65535 half-words	65535 words

3.2 DMA Control Registers

This section provides a bit-by-bit description of the control registers.

Table 4. DMA Control Registers Summary

Address	Mnemonic	Name	Description	Page
0xFFFFFE80	DMACC0	DMA channel control 0	Control for channels 0-7: (a) interrupt level (b) suspend enable (c) request enable	16
0xFFFFFE84	DMACC1	DMA channel control 1	Control for channels 8-15	17
0xFFFFFE88	DMAS	DMA status	Flag register for pending interrupts and transfer complete	18
0xFFFFFE8C	DMAIO0	DMA interrupt offset 0	Offset register for interrupts with interrupt offset level 0	19
0xFFFFFE90	DMAIO1	DMA interrupt offset 1	Offset register for interrupts with interrupt offset level 1	20
0xFFFFFE94	DMACPS	DMA control packet status	A set bit indicates that the data in the control packet is updated	21
0xFFFFFE98	DMACPSC	DMA control packet status clear		22
0xFFFFFE9C	DMAGC	DMA global control	Global control: (a) halt on IRQ (b) halt on FIQ (c) channel priority (d) bus master switch size (e) channel service size	23
0xFFFFFEA0	DMAGD	DMA global disable	STOP and HALT	25
0xFFFFFEA4	DMAAC	DMA active channel	Reads which channel is active	26
0xFFFFFEA8	DMACCP0	DMA current control packet 0	Current control packets for channels 0-3 (a) channel enable (b) current control packet for channel	27
0xFFFFFEAC	DMACCP1	DMA current control packet 1	Current control packets for channels 4-7	27
0xFFFFFEB0	DMACCP2	DMA current control packet 2	Current control packets for channels 8-11	27
0xFFFFFEB4	DMACCP3	DMA current control packet 3	Current control packets for channels 12-15	27

Table 5. DMA Control Registers

Offset Address	Register	31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0
0xFFFF FE80h	DMA CC0	IL7	SEN7	RQEN7	Rsrvd	CH6_CNTL				CH5_CNTL				CH4_CNTL			
		CH3_CNTL				CH2_CNTL				CH1_CNTL				CH0_CNTL			
0xFFFF FE84h	DMA CC1	CH15_CNTL				CH14_CNTL				CH13_CNTL				CH12_CNTL			
		CH11_CNTL				CH10_CNTL				CH9_CNTL				CH8_CNTL			
0xFFFF FE88h	DMAS	Reserved														TC	PINT
		IF[15:0]															
0xFFFF FE8Ch	DMA IO0	Reserved															
		Reserved								Offset-0							
0xFFFF FE90h	DMA IO1	Reserved															
		Reserved								Offset-1							

Table 5. DMA Control Registers (Continued)

Offset Address	Register	31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0
0xFFFF FE94h	DMA CPS	CPS[31:16]															
		CPS[15:0]															
0xFFFF FE98h	DMA CPSC	CPSC[31:16]															
		CPSC[15:0]															
0xFFFF FE9Ch	DMA GC	Reserved															
		Reserved									HIRQ	HFIQ	CHPR	BMSS	CHSS		
0xFFFF FEA0h	DMA GD	Reserved															
		Reserved														DMA STOP	DMA HALT
0xFFFF FEA4h	DMA AC	Reserved															
		Reserved													ACT_CHN		

Table 5. DMA Control Registers (Continued)

Offset Address	Register	31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0
0xFFFF FEA8h	DMA CCP0	Rsrvd	DMAEN3	Rsrvd	CCPACK3					Rsrvd	DMAEN2	Rsrvd	CCPACK2				
		Rsrvd	DMAEN1	Rsrvd	CCPACK1					Rsrvd	DMAEN0	Rsrvd	CCPACK0				
0xFFFF FEACh	DMA CCP1	Rsrvd	DMAEN7	Rsrvd	CCPACK7					Rsrvd	DMAEN6	Rsrvd	CCPACK6				
		Rsrvd	DMAEN5	Rsrvd	CCPACK5					Rsrvd	DMAEN4	Rsrvd	CCPACK4				
0xFFFF FEB0h	DMA CCP2	Rsrvd	DMA EN11	Rsrvd	CCPACK11					Rsrvd	DMA EN10	Rsrvd	CCPACK10				
		Rsrvd	DMAEN9	Rsrvd	CCPACK9					Rsrvd	DMAEN8	Rsrvd	CCPACK8				
0xFFFF FEB4h	DMA CCP3	Rsrvd	DMA EN15	Rsrvd	CCPACK15					Rsrvd	DMA EN14	Rsrvd	CCPACK14				
		Rsrvd	DMA EN13	Rsrvd	CCPACK13					Rsrvd	DMA EN12	Rsrvd	CCPACK12				

3.2.1 DMA Channel Control 0 (DMACC0) Register

This channel control register is used to configure the interrupt level, suspend enable/disable, and request enable or disable for the channels 0 to 7.

Bits	31	30	29	28	27			24	23			20	19			16
0xFFFF-FE80h	IL7	SEN7	RQ EN7	Rsrvd	CH6_CNTL				CH5_CNTL				CH4_CNTL			
	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i
Bits	**15**			**12**	**11**			**8**	**7**			**4**	**3**			**0**
	CH3_CNTL				CH2_CNTL				CH1_CNTL				CH0_CNTL			
	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i

RWP = Read in all modes, write in privilege mode only, i = Read values are indeterminate, *-n* = Value after reset

This register is divided into eight groups of four bits each, to configure the channels from 7 to 0. The four bits are IL*x*, SEN*x*, REQEN*x* (where *x* represents the channel number), and a RESERVED bit, consecutively from MSB, as shown above for channel 7 (bits 31:28). These bits are explained in detail below.

Bit 31 **ILx:** Interrupt level for the channel.

The DMA controller can generate interrupts either on DMA interrupt line 0 or on DMA interrupt line 1 at the completion of the control packet when INTEN(DMAC.15) is set.

0 = DMA module generates interrupt on DMA interrupt line 0
1 = DMA module generates interrupt on DMA interrupt line 1

See the device data sheet to find which interrupt channels (CIM/IEM) correspond to the DMA interrupt lines 0 and 1.

Bit 30 **SENx:** Suspend enable for the channel.

When the software debugger hits a break point, the CPU sends a suspend signal to the modules. The SEN bit controls how the DMA channel treats the suspend signal.

0 = DMA controller does not perform any DMA transfers through this channel during suspend mode.
1 = DMA controller continues to do the DMA transfers in suspend mode as in normal mode.

Bit 29 **RQENx:** Request enable for the channel.

0 = DMA transaction is performed independent of the DMA request. Normally, this mode is used in memory-to-memory data transfers.

1 = Each DMA transaction (read data from source address (DMASA) and write to destination address (DMADA)) is performed only when the DMA controller receives a DMA request. This is normally used for peripheral-to-memory or memory-to-peripheral data transfers.

Note:

The DMA transaction is performed when the bus is available to the DMA controller and the corresponding DMA channel is enabled (DMACCPx).

Bit 28 **Reserved.**

3.2.2 DMA Channel Control 1 (DMACC1) Register

This channel control is used to configure the interrupt level, suspend enable/disable, and request enable or disable for channels 8 through 15.

Bits	*31*			*28*	*27*			*24*	*23*			*20*	*19*			*16*
0xFFFF-FE84h	CH15_CNTL				CH14_CNTL				CH13_CNTL				CH12_CNTL			
	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i
Bits	*15*			*12*	*11*			*8*	*7*			*4*	*3*			*0*
	CH11_CNTL				CH10_CNTL				CH9_CNTL				CH8_CNTL			
	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i	RWP-0	RWP-0	RWP-0	i

RWP = Read in all modes, write in privilege mode only, i = Read values are indeterminate, *-n* = Value after reset

For details on the bits, see section 3.2.1, DMA Channel Control 0 (DMACC0) Register, on page 16.

3.2.3 DMA Status (DMAS) Register

This register contains the interrupt flags for all the channels, a pending interrupt flag, and a transfer complete flag.

Bits	31													18	17	16
0xFFFF-FE88h	Reserved														TC	PINT
	i														RCP-0	R-0
Bits	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	IF15	IF14	IF13	IF12	IF11	IF10	IF9	IF8	IF7	IF6	IF5	IF4	IF3	IF2	IF1	IF0
	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0	RWP-0

RCP = Read in all modes, clear in privilege mode only, RWP = Read in all modes, write in privilege mode only, R = Read, i = Read values are indeterminate, *-n* = Value after reset

Bits 31:18 **Reserved.**

Bit 17 **TC:** Transfer complete flag.

Set to one when the transfer count (DMATC) reaches zero for any control packet assigned to any channel. Application program needs to clear this bit to 0 in order to receive the subsequent DMA transfer completion status.

Bit 16 **PINT:** Pending interrupt. This bit is read-only.

The bit-wise-or of interrupt flag bits (IF.15:0).

0 = Interrupt condition has not occurred
1 = Interrupt condition has occurred

Bits 15:0 **IF.15:0:** Interrupt flag for channel 15 through channel 0.

The DMA controller sets this bit to 1 and generates an interrupt (for the channel) when the following conditions are met:

- ❏ the interrupt enable bit in the configuration register (DMAC.15 in the control packet corresponding to the channel) is set to 1.
- ❏ the block transfer fo that control packet is complete.

The DMA controller generates the interrupt for the control packet assigned to the channel. If data chaining is enabled, an interrupt is generated for each control packet in the data chain that is assigned to the channel.

When the DMA interrupt offset register is read, the interrupt flag bit (IF.15:0) corresponding to the offset value (DMAIOx.7:0) is cleared. The mapping of

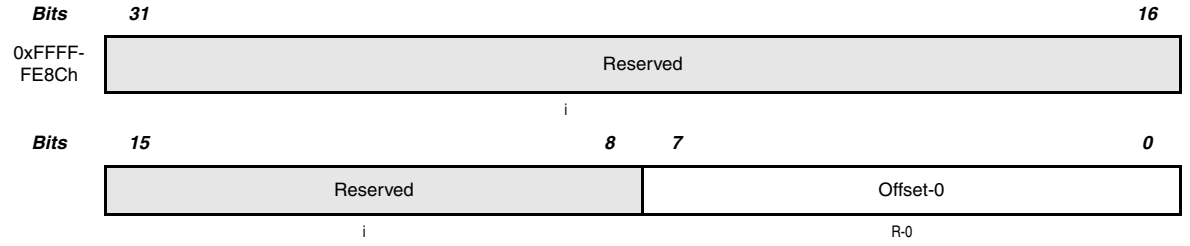
offset values and the channel numbers are given in Table 6. These bits can also be cleared by writing 0. Writing 1 will not have any effect.

Note:

DMAIO0 and DMAIO1 correspond to the DMA interrupt lines 0 and 1, respectively.

3.2.4 DMA Interrupt Offset 0 (DMAIO0) Register

This register provides the offset value for the highest priority pending interrupt.



R = Read in all modes, i = Read values are indeterminate, *-n* = Value after reset

Bits 31:8 **Reserved.**

Bits 7:0 **OFFSET-0:** DMA interrupt offset for DMA interrupt level 0.

These eight read-only bits represent the interrupt offset value driven by the DMA controller. These bits have the value generated by the highest priority channel with pending interrupts and whose interrupt level (IL) bits are programmed to zero. The interrupt priority is fixed among the DMA channels, as shown in Table 6. The table also shows the offset value driven by each individual channel. On reading this register for the current offset value, the register gets updated to the next pending DMA channel interrupt's offset.

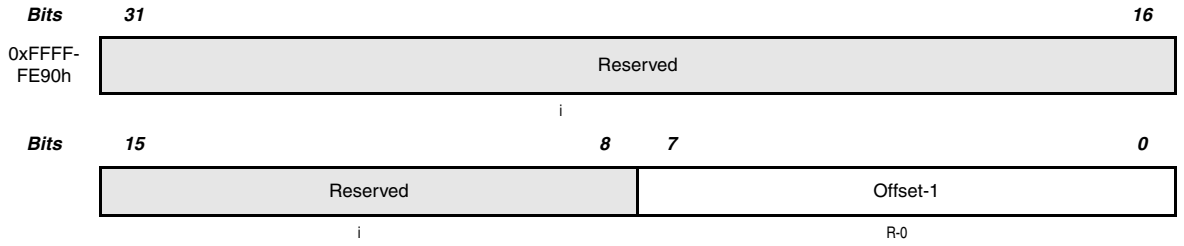
Table 6. Offset and the Corresponding DMA Channels

Offset-0/Offset-1	Highest Priority Pending Interrupt Channel
00000000	No pending interrupts.
00000001	DMA channel0
00001111	DMA channel14

Offset-0/Offset-1	Highest Priority Pending Interrupt Channel
00010000	DMA channel15

3.2.5 DMA Interrupt Offset 1 (DMAIO1) Register

This register provides the offset value for the highest priority pending interrupt in the DMA interrupt level 1.



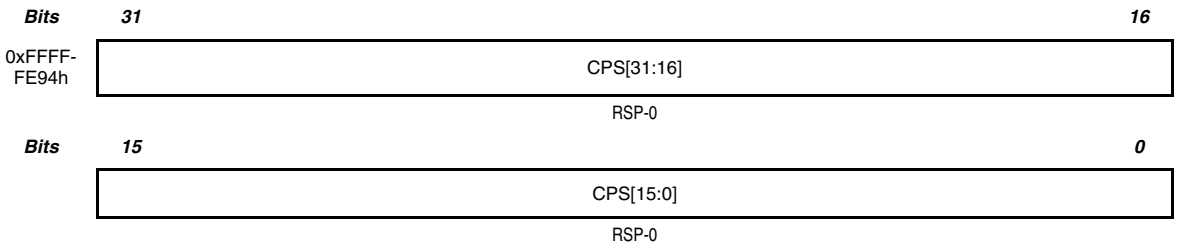
R = Read in all modes, i = Read values are indeterminate, *-n* = Value after reset

Bits 31:8 **Reserved.**

Bits 7:0 **OFFSET-1:** DMA interrupt offset for DMA interrupt level 1.

These eight read-only bits represent the interrupt offset value driven by the DMA controller. These bits have the value generated by the highest priority channel with pending interrupts and whose interrupt level (IL) bits are programmed to one. The interrupt priority is fixed among the DMA channels as shown in Table 6. The table also shows the offset value driven by each individual channel. On reading this register for the current offset value, the register gets updated to the next pending DMA channel interrupt's offset.

3.2.6 DMA Control Packet Status (DMACPS) Register



RSP = Read in all modes, set in privilege mode only; i = Read values are indeterminate, -*n* = Value after reset

Bits 31:0 **CPS.31:0:** Control packet status bits

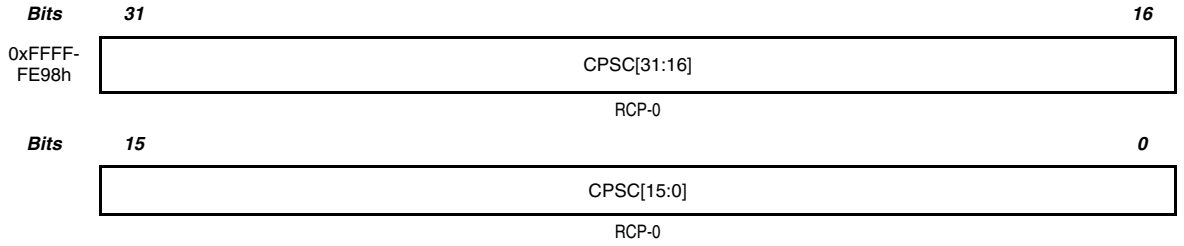
The status of the corresponding control packet (CPS.31:0).

1 = Indicates to the DMA controller that the corresponding control packet is updated and will be used for transfer.

0 = Indicates the completion of DMA transfer (using the updated control packet).

After programming the control packet with the channel setup information, the program needs to set the corresponding bit in this register to one. This indicates to the DMA controller that the control packet is updated and will be used for transfer. This bit will be cleared by the DMA controller either upon completion of the block transfer or when zero is written in the corresponding bit in the DMACPSC register (see section 3.2.7). Bits can be set by writing ones into it. Writing zeros has no effect.

3.2.7 DMA Control Packet Status Clear (DMACPSC) Register



RCP = Read in all modes, clear in privilege mode only; i = Read values are indeterminate, *-n* = Value after reset

Bits 31:0 **CPSC.31:0:** Clear control packet status bits

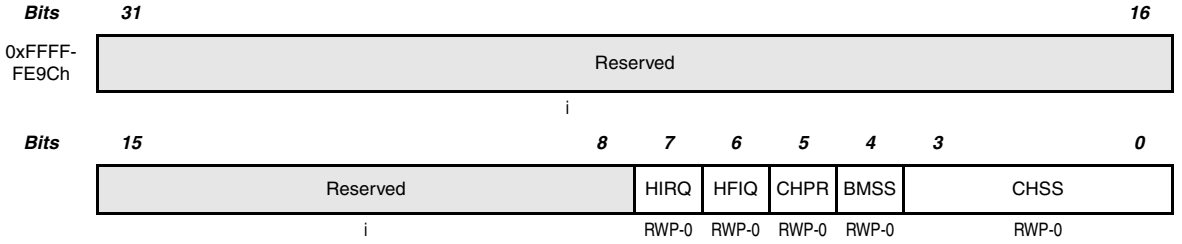
Clears the corresponding status bit.

This register is a mirror image of the DMACPS register, but with the read/clear function. To clear the status bit (DMACPS.32:0), the corresponding bit in this register has to be cleared. Bits can be cleared by writing zeros into it; writing one has no effect.

3.2.8 DMA Global Control (DMAGC) Register

This register defines the following:

- ❏ DMA controller behavior when an interrupt is received.
- ❏ DMA controller behavior when bus conflicts occur.
- ❏ channel prioritization scheme.



RWP = Read in all modes, write in privilege mode only; i = Read values are indeterminate, *-n* = Value after reset

Bits 31:7 **Reserved.**

Bit 7 **HIRQ:** Halt on IRQ mode entry.

When HIRQ=1, the DMA controller halts whenever the CPU enters IRQ mode (i.e. whenever it receives an IRQ interrupt). The following conditions apply:

- ❏ When an IRQ occurs, and the DMA is in the middle of a DMA transaction, the controller finishes the current transaction and sets the DMA_HALT bit (DMAGD.0). The controller does not perform any more DMA transactions until the DMA_HALT bit (DMAGD.0) is cleared.
- ❏ If the DMA is not performing any DMA transactions when an IRQ occurs, the controller sets the DMA_HALT bit (DMAGD.0) and does not perform any DMA transactions until the DMA_HALT bit (DMAGD.0) is cleared.

When HIRQ=0, the DMA controller operation is not affected by an IRQ.

Bit 6 **HFIQ:** Halt on FIQ mode entry.

When HFIQ=1, the DMA controller halts whenever the CPU enters FIQ mode (i.e. whenever it receives an FIQ interrupt). The following conditions apply:

- ❏ When an FIQ occurs, and the DMA is in the middle of a DMA transaction, the controller finishes the current transaction and sets the DMA_HALT bit

(DMAGD.0). The controller does not perform any more DMA transactions until the DMA_HALT bit (DMAGD.0) is cleared.

- If the DMA is not performing any DMA transactions when and FIQ occurs, the controller sets the DMA_HALT bit (DMAGD.0) and does not perform DMA transactions until the DMA_HALT bit (DMAGD.0) is cleared.

When HFIQ=0, the DMA controller operation is not affected by an FIQ.

Bit 5 **CHPR:** Channel priority.

The priority scheme for the DMA channels when multiple channels have pending DMA transactions. When CHPR=0, the DMA controller will be configured for a fixed priority scheme in which channel 0 has highest priority and channel15 has lowest priority. When CHPR=1, the DMA controller will be configured to the rotating priority scheme in which the most recently serviced channel will be placed at the bottom of the priority list.

Bit 4 **BMSS:** Bus master switch size.

The maximum transactions allowed for one bus master when both the CPU and DMA simultaneously request bus ownership. Table 7 shows the number of transactions allowed for the CPU and DMA before bus ownership is switched.

Table 7. BMSS Encoding - Bus Ownership and Number of Transactions

	Number of Transactions	
BMSS Encoding	**CPU**	**DMA**
1	1	4
0	1	1

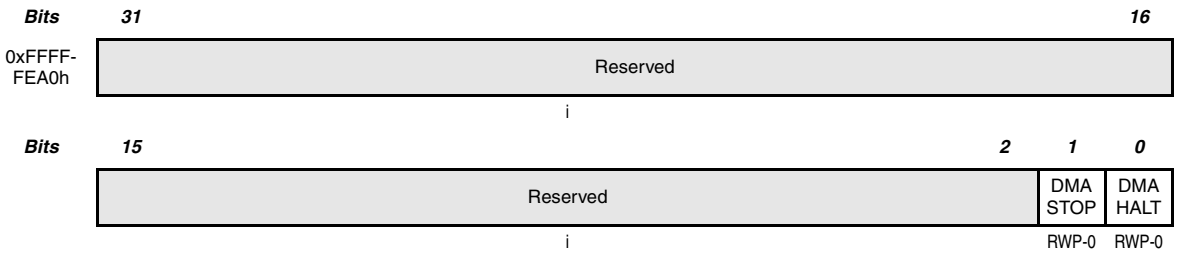
Bits 3:0 **CHSS:** DMA channel service size.

The number of DMA transactions to be performed for the DMA channel before switching to another DMA channel. Table 8 shows the bit encoding for this field.

Table 8. Channel Service Size BIt Encoding for BMSS

CHSS[3:0]	No. of DMA transaction
0000	1
0001	2
:	:
:	:
1110	15
1111	16

3.2.9 DMA Global Disable (DMAGD) Register



RWP = Read in all modes, write in privilege mode only; i = Read values are indeterminate, *-n* = Value after reset

Bits 31:2 Reserved.

Bit 1 DMA_STOP.

When this bit is set to one, the DMA controller finishes the current transaction and updates the configuration (DMAC), source address (DMASA), destination (DMADA), and the transfer count (DMATC) registers of the control packet. The DMA controller does not perform any more DMA transactions until this bit is cleared. Once this bit is cleared, the DMA controller reloads the control packet and resumes its transactions.

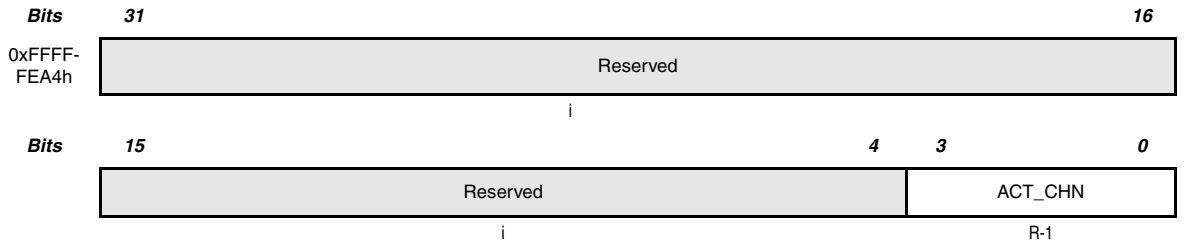
Bit 0 DMA_HALT.

When this bit is set to one, the DMA controller finishes the current transaction and updates the configuration (DMAC), source address (DMASA), destination (DMADA), and the transfer count (DMATC) registers of the control packet. The DMA controller does not perform any more DMA transactions until this bit is cleared. The DMA controller can enter halt mode under the following conditions:

- ❏ when this bit is set to one
- ❏ the CPU enters into IRQ mode when HIRQ is set (DMAGC.7)
- ❏ the CPU enters into FIQ mode when HFIQ is set (DMAGC.6)

When this bit is cleared, the DMA controller resumes the DMA transaction.

3.2.10 DMA Active Channel (DMAAC) Register



R = Read in all modes, i = Read values are indeterminate, *-n* = Value after reset

Bits 31:4 **Reserved.**

Bits 3:0 **ACT_CHN:** Active channel.

These four bits indicate the channel number through which the DMA controller is performing the current DMA transactions. Table 9 shows the bit encoding for these bits.

Table 9. Active Channel Bit Encoding

ACT_CHN[3:0]	Active DMA channel
0000	Channel 0
0001	Channel 1
:	:
:	:
1110	Channel 14
1111	Channel 15

3.2.11 DMA Channel Control Packet (DMACCP) Registers

A group of four registers (DMACCP0-DMACCP3) configures the DMA enable and control packet for the channels. Each register is divided into groups of 8 bits, each of which is used for one channel. Thus, DMACCP0's least significant byte is used for channel 0 and DMACCP0's most significant byte is used for channel 3, and so on.

DMA Current Control Packet 0 (DMACCP0)

Bits	31	30	29	28 ... 24	23	22	21	20 ... 16
0xFFFF-FEA8h	Rsrvd	DMA EN3	Rsrvd	CCPACK3	Rsrvd	DMA EN2	Rsrvd	CCPACK2
	i	RWP-0	i	RWP-0	i	RWP-0	i	RWP-0

Bits	15	14	13	12 ... 8	7	6	5	4 ... 0
	Rsrvd	DMA EN1	Rsrvd	CCPACK1	Rsrvd	DMA EN0	Rsrvd	CCPACK0
	i	RWP-0	i	RWP-0	i	RWP-0	i	RWP-0

RWP = Read in all modes, write in privilege mode only; i = Read values are indeterminate, *-n* = Value after reset

DMA Current Control Packet 1 (DMACCP1)

Bits	31	30	29	28 ... 24	23	22	21	20 ... 16
0xFFFF-FEACh	Rsrvd	DMA EN7	Rsrvd	CCPACK7	Rsrvd	DMA EN6	Rsrvd	CCPACK6
	i	RWP-0	i	RWP-0	i	RWP-0	i	RWP-0

Bits	15	14	13	12 ... 8	7	6	5	4 ... 0
	Rsrvd	DMA EN5	Rsrvd	CCPACK5	Rsrvd	DMA EN4	Rsrvd	CCPACK4
	i	RWP-0	i	RWP-0	i	RWP-0	i	RWP-0

RWP = Read in all modes, write in privilege mode only; i = Read values are indeterminate, *-n* = Value after reset

DMA Current Control Packet 2 (DMACCP2)

Bits	31	30	29	28 ... 24	23	22	21	20 ... 16
0xFFFF-FEB0h	Rsrvd	DMA EN11	Rsrvd	CCPACK11	Rsrvd	DMA EN10	Rsrvd	CCPACK10
	i	RWP-0	i	RWP-0	i	RWP-0	i	RWP-0

Bits	15	14	13	12 ... 8	7	6	5	4 ... 0
	Rsrvd	DMA EN9	Rsrvd	CCPACK9	Rsrvd	DMA EN8	Rsrvd	CCPACK8
	i	RWP-0	i	RWP-0	i	RWP-0	i	RWP-0

RWP = Read in all modes, write in privilege mode only; i = Read values are indeterminate, *-n* = Value after reset

DMA Current Control Packet 3 (DMACCP3)

Bits	31	30	29	28 ... 24	23	22	21	20 ... 16
0xFFFF-FEB4h	Rsrvd	DMA EN15	Rsrvd	CCPACK15	Rsrvd	DMA EN14	Rsrvd	CCPACK14
	i	RWP-0	i	RWP-0	i	RWP-0	i	RWP-0

Bits	15	14	13	12 ... 8	7	6	5	4 ... 0
	Rsrvd	DMA EN13	Rsrvd	CCPACK13	Rsrvd	DMA EN12	Rsrvd	CCPACK12
	i	RWP-0	i	RWP-0	i	RWP-0	i	RWP-0

RWP = Read in all modes, write in privilege mode only; i = Read values are indeterminate, *-n* = Value after reset

Bits 31 **Reserved.**

Bit 30 **DMENx:** DMA enable for the channel.

If the DMA channel is enabled (i.e.DMAENx=1), the DMA controller performs the DMA transactions through this channel.

If the DMA channel is disabled (i.e.DMAENx=0), the DMA controller does not perform any DMA transactions through this channel.

After completing the entire block transfer, if the data chaining (DMAC.0) is not enabled, then the DMA controller clears the corresponding DMAEN bit. If the data chaining (DMAC.0) is enabled, then the DMA controller continues to do the transfers until reaching the control packet in which the data chaining (DMAC.0) is not enabled, and at the end clearing the DMAEN bit.

Bits 28:24 **CCPACKx:** Current control packet for the channel.

The control packet to be used for setting up the particular channel. If data chaining is enabled for the selected channel, after the first block transfer is complete, the DMA controller updates these bits with the NCPACK bits (DMAC[28:24]) programmed in the configuration register of the control packet. The new setup information is then used for the next block transfer. Table 10 shows the bit encoding for this field.

Table 10. Current Control Packet Bit Encoding

CCPACK[4:0]	Control packet
00000	Control packet0
00001	Control packet 1
:	:
:	:
11110	Control packet 30
11111	Control packet 31

Appendix

Section A of the appendix provides detailed information about the DMA data transfer sequence. Section B of the appendix explains the step-by-step details of the DMA initialization with a DMA memory-memory transfer example, providing a reference on DMA initialization.

Topic **Page**

A DMA Data Transfer Sequence

The following sequence occurs during DMA transfer:

- ❏ If the DMA channel is programmed for a request mode transfer, the DMA controller waits until it detects a rising edge of the DMA_REQ signal before making the DMA transaction.
- ❏ If the DMA channel is programmed for a non-request mode transfer, the controller is ready to make the DMA transaction as soon as the application program enables the DMA channel.
  - ■ If multiple channels have pending DMA transfers, the DMA controller uses a rotating priority scheme to arbitrate between each channel then generates a bus request to the bus controller and waits until it gets ownership of the bus.
- ❏ The DMA controller drives the address bus with the source address and reads the data from the source module.
- ❏ The DMA controller writes the data to the destination address.
- ❏ At the end of the write transaction, the transfer counter is decremented by one. The source and destination addresses are modified according to the programmed option (incremented or kept constant).
- ❏ Internally, the service size counter in the DMA controller is incremented by one. If this counter value matches the programmed channel service size, and if any other channel is requesting a DMA transfer, the DMA controller arbitrates between the channels and services the next highest priority pending request. If the counter value is less than the service size, the DMA transfer is continued for the current channel.
- ❏ At the end of the block transfer (when the transfer counter reaches zero), the DMA controller sets the Transfer Complete flag bit. If the interrupt is enabled for that block transfer, the DMA controller sets the corresponding interrupt flag bit in the DMA status register and generates the interrupt. The DMA controller clears the control packet status bit to zero.
- ❏ If data chaining is enabled, the DMA controller continues by reading the next control packet as pointed by the current control packet's configuration word. If the new packet's status bit is zero (this means the information in this control packet is not updated), then the DMA controller waits until the program updates the control packet in the command buffer and sets the status bit to 1. Data chaining continues until the end of block transfer for the control packet in which the data chaining is disabled.
- ❏ If data chaining is not enabled for the DMA channel, after one block transfer is complete, the DMA channel goes idle until it is reprogrammed.

- After completing the entire block transfer, if the data chain is not enabled then the DMA controller clears the corresponding DMAEN bit such that no more DMA transactions are performed through that channel until application software re-programs the channel. If the data chaining is enabled, then the DMA controller continues to do the transfers until the control packet in which the data chaining is not enabled.

B DMA Initialization

This section details the order in which the DMA registers are to be initialized. The initialization can be carried in three sequential steps, namely, 1) initializing control packet registers, channel registers and global control registers, 2) updating control packet status and 3) enabling channel for DMA transfer as detailed below:

1) Initialize control packet registers, channel registers, and global control registers

■ Control packets initialization

Initialize the following registers for each control packet being used for DMA transfer:

a) DMA Configuration (DMAC) Register

b) DMA Source Address (DMASA) Register

c) DMA Destination Address (DMADA) Register

d) DMA Transfer Count (DMATC) Register

■ Channel initialization

Initialize DMA Channel Control 0 (DMACC0) and/or DMA Channel Control 1 (DMACC1) depending on the DMA channels selected for the transfer.

■ Global control initialization

The global control initialization involves filling the following registers:

a) DMA Global Control (DMAGC) Register

b) DMA Global Disable Register bits 0 and 1 (DMASTOP and HALT) to zero (reset state of this bits are zero).

2) Update control packet status

Fill the DMA Control Packet Status register (DMACPS) bits 31:0 with respect to the control packets being initialized for the transfer. This step should only be performed on completion of control packet initialization in step 1.

3) Enable channel for DMA transfer

Fill the DMA Channel Control Packet Registers (DMACCP 0-3). Here the initialized control packet will be assigned to the DMA channel and the channel is enabled for transfer. This step should only be performed after completing steps 1 and 2.

If any DMA transfer error occurs, before re-initializing the DMA, the following steps are to be carried out:

A) Set DMA STOP bit (DMAGD.1) to 1.

B) Clear the control packet status bit (DMACPS.31:0) by writing 0 to the corresponding bit in the control packet status clear (DMACPSC.31:0) register.

B.1 DMA Initialization Example

This section shows an example of *memory-to-memory* DMA transfer initialization for a particular configuration.

Consider the following DMA transfer configuration:

- Source RAM starting address = 0x4000, controlled by *fine memory select 2*.
- Destination RAM starting address = 0x8000, controlled by *fine memory select 3*.
- Transfer Count = 0x100
- Transfer size = 32 bit (word transfer), source and destination address incremented for each transfer
- No data chaining
- The DMA transfer is done through DMA channel 0 using Control packet 0
- The channel 0 is configured to:
    i) Generate an interrupt on *interrupt line 1* after completion of the DMA transfer
    ii) Non-request mode
    iii) Suspend Enable
- Halt on IRQ mode and FIQ mode are disabled
- Channel priority is fixed
- The maximum transactions allowed for one bus master when both the CPU and DMA simultaneously request bus ownership are one cycles each.
- DMA channel service size = 1
- DMA halt and DMA stop disabled

The DMA initialization for the given configuration is shown in Table 1.

Table 1. Address and Data Values for DMA Example Initialization

DMA Initialization Step	REGISTER NAME	ADDRESS	DATA
Initialize control packet registers, channel registers, and global control registers	DMA Configuration Register (DMAC)	0xFFFFF800	0x0000CA64 h
	DMA Source Address Register (DMASA)	0xFFFFF804	0x00004000 h
	Destination Address Register (DMADA)	0xFFFFF808	0x00008000 h
	DMA Transfer Count Register (DMATC)	0xFFFFF80C	0x00000100 h
	DMA Channel Control 0 (DMACC0)	0xFFFFFE80	0x0000000C h
	DMA Global Control (DMAGC)	0xFFFFFE9C	0x00000000 h
	DMA Global Disable (DMAGD)	0xFFFFFEA0	0x00000000 h
Update control packet status	DMA Control Packet Status (DMACPS)	0xFFFFFE94	0x00000001 h
Enable channel for DMA transfer	DMA Current Control Packet 0 (DMACCP0)	0xFFFFFEA8	0x00000040 h