Register file structure: dmawr.pdf
Created by jdesilet on 2019/05/06 15:58:21

1. Main Parameters

Register file endianness: little endian
Address bus width: 9 bits
Data bus width: 64 bits

2. Memory Map

Section name	Address(es) / Address Ranges	Register name
HEADER	0x000	fstruc
	0x008	fid
	0x010	fsize
	0x018	fctrl
	0x020	foffset
	0x028	fint
	0x030	fversion
	0x038	frsvd3
USER	0x040	dstfstart
	0x048	dstlnpitch
	0x050	dstlnsize
	0x058	dstnbline
	0x060	DSTFSTART1
	0x068	DSTFSTART2
	0x070	DSTFSTART3
	0x078	DSTCTRL
	0x080	DSTCLRPTRN

Section: HEADER

Address Range: [0x000 - 0x038]

fstruc Function STRUCture

Address: section "HEADER" base address + 0x000

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
tag(23:16)							
23	22	21	20	19	18	17	16
tag(15:8)							
15	14	13	12	11	10	9	8
tag(7:0)							
7	6	5	4	3	2	1	0
mjver(3:0)				mnver(3:0)			

tag (23:0)	Structure valid TAG	
STATIC	This field is a tag identifying the beginning of a register section associated with every IP block.	
Value at Reset:	0x4d5458	
Possible Values:	0x4D5458	

mjver (3:0)	Standard header structure MaJor VERsion	
STATIC	This field reflects any major changes in the register structure associated with the standard 32 byte function header. Zero value is reserved for field overflow.	
Value at Reset:	0x1	
Possible Values:	Any Value	

mnver (3:0)	Standard header structure MiNor VERsion	
STATIC	This field reflects any minor changes in the register structure associated with the standard 32 bytes function header. (Software backward compatible)	
Value at Reset:	0x4	
Possible Values:	Any Value	

Address: section "HEADER" base address + 0x008

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
fid(15:8)							
23	22	21	20	19	18	17	16
fid(7:0)							
15	14	13	12	11	10	9	8
CAPABILITY(15:8)							
7	6	5	4	3	2	1	0
CAPABILITY(7:0)							

fid (15:0)	Function IDentification	
STATIC	This field indicates the IPs unique ID	
Value at Reset:	0xc011	
Possible Values:	Any Value	

CAPABILITY (15:0)	CAPABILITY list	
STATIC	This 16 bit value reflects the functionality and features that can be present or not depending on the configuration chosen for the function IP block.	
Value at Reset:	0x0	
Possible Values:	0x0	No specific capability field
	0x1 - 0xFFFF	User defined capabilities.

Address: section "HEADER" base address + 0x010

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
fullsize(15:8)							
23	22	21	20	19	18	17	16
fullsize(7:0)							
15	14	13	12	11	10	9	8
usersize(15:8)							
7	6	5	4	3	2	1	0
usersize(7:0)							

fullsize (15:0)	FULL register SIZE	
STATIC	This field indicates the amount of QWORD (= 8bytes = int64 = long long) that is mapped on the register address space by the entire register structure associated with this function. Maximum size is 256MB. Granularity of 512 bytes.	
Value at Reset:	0x40	
Possible Values:	0x0	This is a reserved value indicating that the field does not actually report the size of the entire structure. That value is likely to be retrieved in the HDF section of the IP.
	0x1 - 0x7	Reserved values.
	0x8 - 0xFFFF	The field reports the size of the

usersize (15:0)	SPECific register SIZE	
STATIC	This field indicates the amount of QWORD (= 8bytes = int64 = long long) that is mapped on the register address space by the specific register section ("user" registers) that is associated with the control and functionality of the IP block.	
Value at Reset:	0x8	
Possible Values:	Any Value	

Address: section "HEADER" base address + 0x018

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
Reserved		snppdg	active	Reserved		snpsht	abort
23	22	21	20	19	18	17	16
ipferr(7:0)							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
Reserved							

snppdg *RO*	SNapSHot PenDinG This field is asserted whenever the SNPSHT bit is being set if the ABORT signal is not asserted. This bit stays asserted to a logic 1 until the arrival of the event to which the snapshot was referring to (usually a stream start-of-frame event).	
Possible Values:	0x0	Do nothing
	0x1	A snapshot command is pending

active *RO*	Active This field indicates that a transfer is currently active. This means the PU is currently processing data. Note that on the occurrence of an ABORT event, this bit must stay active until the PU has reached an idle state.	
Possible Values:	0x0	Do nothing
	0x1	A transfer is active

snpsht *WO/AutoClr*	SNaPSHoT This bit field sets the processing unit in a state in which it can process a forthcoming stream, usually a frame. This bit sets the snapshot pending signal allowing acceptation of the next input stream of data. Writing zero has no effect. Should be read as zero.	
Possible Values:	0x0	Do nothing
	0x1	Snapshot

abort *RW*	ABORT process Asynchronously or synchronously resets all the logic related to a pending or active transaction.	
Value at Reset:	0x0	
Possible Values:	0x0	Do nothing
	0x1	Reset logic.

ipferr (7:0) *STATIC*	IP Function ERRor This bit field is used to report error code. Specific error codes have not been defined yet.	
Value at Reset:	0x0	
Possible Values:	0x0	No error reported
	0x1 - 0xFF	User defined error numbers.

Address: section "HEADER" base address + 0x020

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
ioctloff(15:8)							
23	22	21	20	19	18	17	16
ioctloff(7:0)							
15	14	13	12	11	10	9	8
useroff(15:8)							
7	6	5	4	3	2	1	0
useroff(7:0)							

ioctloff (15:0) *STATIC*	IO ConTroL register OFFset This field indicates the distance in QWORD ((= 8bytes = int64 = long long) from the function header base address to the beginning of the Stream Output Port IO control register section. If no Stream Output port is declared the value of this field is 0x0.	
Value at Reset:	0x0	
Possible Values:	0x0	No Stream Output port is declared for this Processing Unit
	0x1 - 0xFFFF	Distance in QWORD from the header base address to the first register of the first stream output port register.

useroff (15:0) *STATIC*	SPECific register offset This field indicates distance in QWORD ((= 8bytes = int64 = long long) from the function header base address to the beginning of the user specific register section. If no specific register section is defined the value of this field is 0x0.	
Value at Reset:	0x8	
Possible Values:	0x0	No User specific register section is defined.
	0x1 - 0xFFFF	Distance in QWORD from the header base address to the first register of us

Address: section "HEADER" base address + 0x028

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved	lsbof(6:0)						
7	6	5	4	3	2	1	0
Reserved					ipevent(2:0)		

lsbof (6:0) *RO*	LSB OFFSET This field indicates the position mapping of the first interrupt event in the interrupt forwarding vector.	
Possible Values:	0x0 - 0x7D	LSB Offset assigned by the top level generator.
	0x7E	Reserved
	0x7F	Interrupt not mapped.
	Others	

ipevent (2:0) *STATIC*	IP (number of) EVENT This field indicates the amount of distinct event(s) (start-of-frame, end-of-frame, etc.) this IP would reported to the interrupt manager module. A maximum of 7 events can be declared.	
Value at Reset:	0x1	
Possible Values:	0x0	No interrupt implemented
	0x1	1 Interrupt implemented
	0x2	2 Interrupts implemented
	0x3	3 Interrupts implemented
	0x4	4 Interrupts implemented
	0x5	5 Interrupts implemented
	0x6	6 Interrupts implemented
	0x7	7 Interrupts implemented

Address: section "HEADER" base address + 0x030

Description:
This register specify the revision of the current Processing Unit (PU). The Full revision number of the PU is retrieved from the three fields IPMJVER.IPMNVER.IPHWVER.

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
SUBFID(7:0)							
15	14	13	12	11	10	9	8
ipmjver(3:0)				ipmnver(3:0)			
7	6	5	4	3	2	1	0
iphwver(4:0)						Reserved	

SUBFID (7:0)	SUB Function ID	
RO	Reserved	
Possible Values:	0x0	1 x 128-bit sdram bank
	0x1	2 x 32-bit sram banks
	0x2	4 x 32-bit sram banks
	0x3	1 x 256-bit sdram bank
	0x4	1 x 512-bit sdram bank
	0x5	1 x 512-bit sdram bank, 4 contexts
	0x6	1 x 128-bit sdram bank, 2 contexts

ipmjver (3:0)	IP register structure MaJor VERsion	
STATIC	This field reflects any major changes in the register structure associated with the IP's specific registers. Zero value is reserved for field overflow	
Value at Reset:	0x2	
Possible Values:	0x0	Field Overflow
	0x1 - 0xF	Major revision number

ipmnver (3:0) *STATIC*	IP register structure MiNor VERsion This field reflects any minor changes in the register structure associated with the IP's specific registers.	
Value at Reset:	0x2	
Possible Values:	0x0	Initial version
	0x1	
	0x2	
	Others	Reserved

iphwver (4:0) *RO*	IP HardWare VERsion This field indicates the functional hardware revision associated with the current IP build. The hardware version is incremented for any new build that includes a functional change (bug fixes, performance issue, functionality, etc.). This field is reset to zero whenever the register structure minor or major field is incremented.	
Possible Values:	Any Value	

frsvd3 Function ReSerVed 3

Address: section "HEADER" base address + 0x038

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
frsvd3(31:24)							
23	22	21	20	19	18	17	16
frsvd3(23:16)							
15	14	13	12	11	10	9	8
frsvd3(15:8)							
7	6	5	4	3	2	1	0
frsvd3(7:0)							

frsvd3 (31:0)	Function Reserved 3
STATIC	Function header reserved expansion register.
Value at Reset:	0x0

Section: USER

Address Range: [0x040 - 0x080]

dstfstart — Destination frame start register.

Address: section "USER" base address + 0x000

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved				fstart(35:32)			
31	30	29	28	27	26	25	24
fstart(31:24)							
23	22	21	20	19	18	17	16
fstart(23:16)							
15	14	13	12	11	10	9	8
fstart(15:8)							
7	6	5	4	3	2	1	0
fstart(7:0)							

fstart (35:0)	Frame start address
RW	This field indicates the byte position in memory where the received image will be written (first byte of the first image's line). The frame start can map to everywhere in the 4GB range. This field is always available in every flavor of the DMARD TU
Value at Reset:	0x0

dstlnpitch — Destination pitch register.

Address: section "USER" base address + 0x008

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved				lnpitch(35:32)			
31	30	29	28	27	26	25	24
lnpitch(31:24)							
23	22	21	20	19	18	17	16
lnpitch(23:16)							
15	14	13	12	11	10	9	8
lnpitch(15:8)							
7	6	5	4	3	2	1	0
lnpitch(7:0)							

lnpitch (35:0) *RW*	Pitch Sets the distance between the beginnings of any two adjacent rows of the image to transfer, in bytes.
Value at Reset:	0x0
Possible Values:	Any Value

dstlnsize — Destination row size register.

Address: section "USER" base address + 0x010

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
Reserved	lnsize(30:24)						
23	22	21	20	19	18	17	16
lnsize(23:16)							
15	14	13	12	11	10	9	8
lnsize(15:8)							
7	6	5	4	3	2	1	0
lnsize(7:0)							

lnsize (30:0)	Row size	
RW	Sets the size of the row to of the image to transfer, in bytes. Cropping will occur whenever this value is less than the actual number of rows in the input image.	
Value at Reset:	0x0	
Possible Values:	Any Value	

dstnbline — Destination number of row register.

Address: section "USER" base address + 0x018

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
Reserved	nbline(30:24)						
23	22	21	20	19	18	17	16
nbline(23:16)							
15	14	13	12	11	10	9	8
nbline(15:8)							
7	6	5	4	3	2	1	0
nbline(7:0)							

nbline (30:0)	Number of rows
RW	Sets the number of rows to transfer. Cropping will occur whenever this value is less than the actual number of lines in the input image.
Value at Reset:	0x0

DSTFSTART1 DeSTination Frame START register context 1

Address: section "USER" base address + 0x020

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved				FSTART(35:32)			
31	30	29	28	27	26	25	24
FSTART(31:24)							
23	22	21	20	19	18	17	16
FSTART(23:16)							
15	14	13	12	11	10	9	8
FSTART(15:8)							
7	6	5	4	3	2	1	0
FSTART(7:0)							

FSTART (35:0)	Frame START address of the context number 1
RW	This field indicates the byte position in memory where the received image will be written (first byte of the first image's line). The frame start can map to everywhere in the 4GB range. This field is always available in every flavor of the DMARD TU
Value at Reset:	0x0

DSTFSTART2 DeSTination Frame START register context 2

Address: section "USER" base address + 0x028

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved				FSTART(35:32)			
31	30	29	28	27	26	25	24
FSTART(31:24)							
23	22	21	20	19	18	17	16
FSTART(23:16)							
15	14	13	12	11	10	9	8
FSTART(15:8)							
7	6	5	4	3	2	1	0
FSTART(7:0)							

FSTART (35:0)	Frame START address of the context number 2
RW	This field indicates the byte position in memory where the received image will be written (first byte of the first image's line). The frame start can map to everywhere in the 4GB range. This field is always available in every flavor of the DMARD TU
Value at Reset:	0x0

DSTFSTART3 DeSTination Frame START register context 3

Address: section "USER" base address + 0x030

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved				FSTART(35:32)			
31	30	29	28	27	26	25	24
FSTART(31:24)							
23	22	21	20	19	18	17	16
FSTART(23:16)							
15	14	13	12	11	10	9	8
FSTART(15:8)							
7	6	5	4	3	2	1	0
FSTART(7:0)							

FSTART (35:0)	Frame START address of the context number 3
RW	This field indicates the byte position in memory where the received image will be written (first byte of the first image's line). The frame start can map to everywhere in the 4GB range. This field is always available in every flavor of the DMARD TU
Value at Reset:	0x0

Address: section "USER" base address + 0x038

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved							
7	6	5	4	3	2	1	0
BUFCLR	DTE3	DTE2	DTE1	DTE0	BITWDTH	NBCONTX(1:0)	

BUFCLR	BUFfer CLeaR	
RW	When set, this bit enables the buffer clear function. The destination memory is written with the pattern specified in the CLRPTRN register.	
Value at Reset:	0x0	
Possible Values:	0x0	Buffer clear is disabled
	0x1	Buffer clear is enabled

DTE3	Data Transfer Enable 3	
RW	This bit is used to enable the transfer of the component 3 during a multi context transfer.	
Value at Reset:	0x1	
Possible Values:	0x0	Component transfer disabled. The constant value 0x0 is sent instead of the pixel values for each pixel of the component 3.
	0x1	Component transfer is enabled.

DTE2	Data Transfer Enable 2	
RW	This bit is used to enable the transfer of the component 2 during a multi context transfer.	
Value at Reset:	0x1	
Possible Values:	0x0	Component transfer disabled. The constant value 0x0 is sent instead of the pixel values for each pixel of the component 2.
	0x1	Component transfer is enabled.

DTE1	Data Transfer Enable 1	
RW	This bit is used to enable the transfer of the component 1 during a multi context transfer.	
Value at Reset:	0x1	
Possible Values:	0x0	Component transfer disabled. The constant value 0x0 is sent instead of the pixel values for each pixel of the component 1.
	0x1	Component transfer is enabled.

DTE0	Data Transfer Enable 0	
RW	This bit is used to enable the transfer of the component 0 during a multi context transfer.	
Value at Reset:	0x1	
Possible Values:	0x0	Component transfer disabled. The constant value 0x0 is sent instead of the pixel values for each pixel of the component 0.
	0x1	Component transfer is enabled.

BITWDTH	BIT WiDTH	
RW	This field indicates the bit width of each buffer to write.	
Value at Reset:	0x0	
Possible Values:	0x0	8 bits per component
	0x1	16 bits per component

NBCONTX (1:0)	NumBer of CONTeXt	
RW	This field indicates the number of context to enable. A maximum of four contexts is supported. Unused contexts are disabled. This field is only available in DMAWR TU that support multi context.	
Value at Reset:	0x0	
Possible Values:	0x0	One context
	0x1	Two contexts
	0x2	Three contexts
	0x3	Four contexts

DSTCLRPTRN DeSTination CLeaR PaTteRN

Address: section "USER" base address + 0x040

63	62	61	60	59	58	57	56
Reserved							
55	54	53	52	51	50	49	48
Reserved							
47	46	45	44	43	42	41	40
Reserved							
39	38	37	36	35	34	33	32
Reserved							
31	30	29	28	27	26	25	24
CLRPTRN(31:24)							
23	22	21	20	19	18	17	16
CLRPTRN(23:16)							
15	14	13	12	11	10	9	8
CLRPTRN(15:8)							
7	6	5	4	3	2	1	0
CLRPTRN(7:0)							

CLRPTRN (31:0)	CLeaR PaTteRN
RW	Buffer clear pattern.
Value at Reset:	0x0