Register & Memory Map

Node Virtual Memory Map

Region	Contents	hex address
		0xFFFF FFFF
R 7	RESERVED	0xF840 0000
R 7	OS Kernel	0xF800 0000
R 7	local stack ↓ / global stack ↑	0xE000 0000
R 6	Control and Data Regs (← 0xD000 0000)	0xC000 0000
R 5	heap (← 0xB000 0000)	0xA000 0000
R 5 / R 4	VU Data	
R 4	stack (← 0x9000 0000)	0x8000 0000
R 3	heap (← 0x7000 0000)	0x6000 0000
R 3 / R 2	VU Inst	
R 2	stack (← 0x5000 0000)	0x4000 0000
R 0/1	global heap ↓	0x2010 0000
R 0/1	NI Registers	0x2000 0000
R 0/1	local heap ↑	
R 0/1	user variables	
R 0/1	user program	0x0000 2000
R 0/1	unmapped	0x0000 0000

VU Regions

Region	hex offset
	0x1FFF FFFF
RESERVED	0x1E00 0000
RESERVED	0x1C00 0000
VU 6/7	0x1A00 0000
VU 2/3	0x1800 0000
VU 4/5	0x1600 0000
VU 0/1	0x1400 0000
RESERVED	0x1200 0000
ALL VU's	0x1000 0000
VU 7	0x0E00 0000
VU 3	0x0C00 0000
VU 6	0x0A00 0000
VU 2	0x0800 0000
VU 5	0x0600 0000
VU 1	0x0400 0000
VU 4	0x0200 0000
VU 0	0x0000 0000

Sample VU Region

Area	hex offset
	0x1FF FFFF
RESERVED	0x0180 0000
Control Registers (Supervisor Area)	0x0100 0000
Data Registers (128 Words)	0x0080 0000
Control Registers (User Area)	0x0000 0000

Control Registers (user and supervisor)

	Register	Permission	hex offset
			0x1FF
	RESERVED		0x178
	dp_instruction_ext_enb	sr	0x174
	dp_instruction_ext	sr	0x170
	RESERVED		0x16C
	dp_busy		0x168
*	dp_stack_limits	srw	0x164
*	dp_heap_limits	srw	0x160
	RESERVED		0x15C
*	dp_current_element	sr	0x158
	dp_bad_instruction_low	sr	0x154
	dp_bad_instruction_high	sr	0x150
	dp_bad_address_low	sr	0x14C
*	dp_bad_address_high	sr	0x148
*	dp_interrupt_clear_green		0x144
*	dp_interrupt_clear	sw	0x140
*	dp_interrupt_enable_green		0x13C
*	dp_interrupt_enable	srw	0x138
*	dp_interrupt_cause_green	sw	0x134
*	dp_interrupt_cause	srw	0x130
	RESERVED		0x12C
	dp_sync		0x128
*	dp_status	sw	0x124
*	dp_status_enable		0x120
	dp_vector_mask_direction		0x11C
*	dp_vector_mask_mode		0x118
	dp_vector_mask_buffer		0x114
	dp_vector_mask		0x110
	dp_stride_rs1		0x10C
	dp_stride_memory		0x108
	dp_vector_length		0x104
*	dp_alu_mode		0x100
	RESERVED		0x07C
*	dp_mbus_port	sr	0x078
	RESERVED		0x070
	dp_jtag_request	srw	0x068
*	dp_mc_refresh	sr	0x060
	dp_mc_sync_refresh_logic	sw	0x058
	dp_led	sw	0x050
	dp_jtag_backdoor	srw	0x048
*	dp_forced_error	srw	0x040
	dp_mc_bad_syndrome	srw	0x038
	dp_memory_access_model	srw	0x030
	dp_dp_interrupt_color_mode	srw	0x028
	dp_error_reset	sw	0x020
	dp_error_debug	sw	0x018
*	dp_mc_status	srw	0x010
*	dp_transaction_timer	srw	0x008
*	dp_mc_mode	srw	0x000

Virtual Address Decoding: (binary)

R	R	R	D	D	D	D	C	C	C	C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31			28				24				20				16				12				8				4				0

R R R = Region select bits
D D D D = VU Region select bits
C C C C = Control register area select bits
0 0 0 0 = Offset into selected region

Key:

- Boxes indicate related registers
- * Indicates register with subfields (See listings on reverse side)
- sr sw srw Indicates supervisor read/write permission only

Register & Memory Map

Register: dp_mc_mode

Field Name:	Pos:	Size:
dp_mc_mode_correct	0	1
dp_mc_mode_timing	1	2
dp_mc_mode_size	3	3
dp_mc_mode_psize	6	2
dp_mc_mode_pnum	8	2
dp_mc_mode_reserved	10	1
dp_mc_mode_refresh	11	1
dp_mc_mode_preset	12	13

Register: dp_transaction_timer

Field Name:	Pos:	Size:
dp_transaction_timer_preset	0	16
dp_transaction_timer_mode	16	1

Register: dp_mc_status

Field Name:	Pos:	Size:
dp_mc_status_reserved	0	3
dp_mc_status_scount	3	8

Register: dp_forced_error

Field Name:	Pos:	Size:
dp_forced_error_ecc	0	8
dp_forced_error_mode	8	1

Register: dp_mc_refresh

Field Name:	Pos:	Size:
dp_mc_refresh_timer	16	13

Register: dp_mbus_port

Field Name:	Pos:	Size:
dp_mbus_mvend	0	4
dp_mbus_mrev	4	4
dp_mbus_mdev	8	8
dp_mbus_tmc	16	16

Register: dp_alu_mode

Field Name:	Pos:	Size:
dp_alu_mode_round	0	2
dp_alu_mode_fast	2	1

Register: dp_stride_memory

Field Name:	Pos:	Size:
dp_stride_memory_value	2	25

Register: dp_vector_mask_mode

Field Name:	Pos:	Size:
dp_vector_mask_mode_mem_cond	0	8
dp_vector_mask_mode_alu_cond	8	1

Registers: dp_status, dp_status_enable

(same bit positions, all flags)

Field Name:	Pos:
dp_status_inexact	0
dp_status_divide_by_zero	1
dp_status_underflow	2
dp_status_overflow	3
dp_status_invalid_operation	4
dp_status_integer_overflow	5
dp_status_negative_unsigned	6
dp_status_denorm_input	7
dp_status_zero_result	8
dp_status_pos_result	9
dp_status_neg_result	10
dp_status_int_carry	11
dp_status_infinite_result	12
dp_status_nan_result	13
dp_status_denorm_result	14
dp_status_unordered	15
dp_status_under	16
dp_status_deno	17

Register: dp_bad_address_high

Field Name:	Pos:	Size:
dp_bad_address_high	0	4
dp_bad_address_transaction_type	4	4
dp_bad_address_transaction_size	8	3

Register: dp_current_element

Field Name:	Pos:	Size:
dp_id_lsb	0	1
dp_element	1	4

Registers: dp_interrupt_{op}

(same bit positions, all flags)

Field Name:	Pos:
dp_debug_trap	0
dp_invalid_register_address	1
dp_invalid_memory_address	2
dp_invalid_local_access	3
dp_register_write_collision	4
dp_invalid_opcode	5
dp_memory_sco	6
dp_memory_single	7
dp_internal_fault	8
dp_memory_error	9
dp_jtag_interrupt	10
dp_invalid_remote_access (no enable)	11
ni_cause/clear_dr_count_negative	12

MBUS Transaction Fields

Register: dp_bad_address_high

Field Name: dp_bad_address_high

4 m.s. bits of physical address

Field Name: dp_bad_address_type

value:	transaction type:
0	write
1	read
2	coherent invalidate
3	coherent read
4	coherent write and invalidate
5	coherent read and invalidate
6-15	reserved

Field Name: dp_bad_address_type

value:	transaction type:
0	byte
1	halfword
2	word
3	doubleword
4	16 bytes
5	32 bytes
6	64 bytes
7	128 bytes

Registers: dp_interrupt_{op}_green

(same bit positions, all flags)

Field Name:	Pos:
dp_alu_inexact	0
dp_alu_divide_by_zero	1
dp_alu_underflow	2
dp_alu_overflow	3
dp_alu_invalid	4
dp_alu_integer_overflow	5
dp_alu_negative_unsigned	6
dp_alu_denorm_input	7
dp_set_enb	8

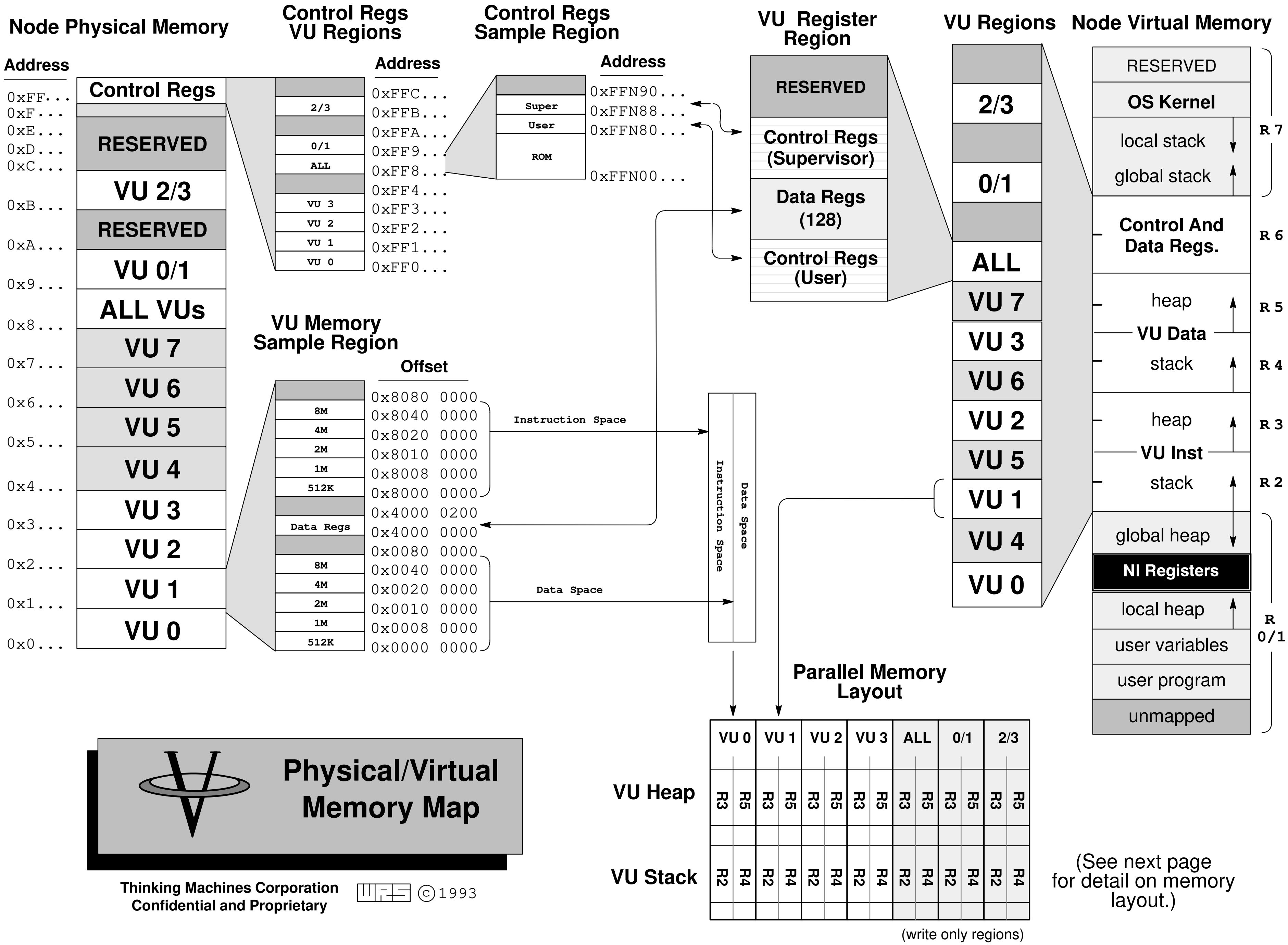
Register: dp_stack_limits

Field Name:	Pos:	Size:
dp_stack_limits_lower	0	15
dp_stack_limits_upper	16	15

Register: dp_heap_limits

Field Name:	Pos:	Size:
dp_heap_limits_lower	0	15
dp_heap_limits_upper	16	15

Register Subfields



Physical/Virtual Memory Map

Parallel Memory Addressing

(write only regions)

	VU Heap				VU Stack
VU 2/3	0xA800 0000 DPV_HEAP_DATA_2_AND_3	R5	Data Space	R4	0x8800 0000 DPV_STACK_DATA_2_AND_3
	0x6800 0000 DPV_HEAP_INST_PORT_2_AND_3	R3	Instruction Space	R2	0x4800 0000 DPV_STACK_INST_PORT_2_AND_#
VU 0/1	0xA400 0000 DPV_HEAP_DATA_0_AND_1	R5	Data Space	R4	0x8400 0000 DPV_STACK_DATA_0_AND_1
	0x6400 0000 DPV_HEAP_INST_PORT_0_AND+_1	R3	Instruction Space	R2	0x4400 0000 DPV_STACK_INST_PORT_0_AND_1
ALL VUs	0xB000 0000 DPV_HEAP_DATA_ALL	R5	Data Space	R4	0x9000 0000 DPV_STACK_DATA_ALL
	0x7000 0000 DPV_HEAP_INST_PORT_ALL	R3	Instruction Space	R2	0x5000 0000 DPV_STACK_INST_PORT_ALL
VU 3	0xAC00 0000 DPV_HEAP_DATA_3	R5	Data Space	R4	0x8C00 0000 DPV_STACK_DATA_3
	0x6C00 0000 DPV_HEAP_INST_PORT_3	R3	Instruction Space	R2	0x4C00 0000 DPV_STACK_INST_PORT_3
VU 2	0xA800 0000 DPV_HEAP_DATA_2	R5	Data Space	R4	0x8800 0000 DPV_STACK_DATA_2
	0x6800 0000 DPV_HEAP_INST_PORT_2	R3	Instruction Space	R2	0x4800 0000 DPV_STACK_INST_PORT_2
VU 1	0xA400 0000 DPV_HEAP_DATA_1	R5	Data Space	R4	0x8400 0000 DPV_STACK_DATA_1
	0x6400 0000 DPV_HEAP_INST_PORT_1	R3	Instruction Space	R2	0x4400 0000 DPV_STACK_INST_PORT_1
VU 0	0xA000 0000 DPV_HEAP_DATA_0	R5	Data Space	R4	0x8000 0000 DPV_STACK_DATA_0
	0x6000 0000 DPV_HEAP_INST_PORT_0	R3	Instruction Space	R2	0x4000 0000 DPV_STACK_INST_PORT_0