Lab5 Notes

zjx@ustc.edu.cn

2019.5.9

分布式双端口存储器

- **CPU端口**

a: 读/写地址

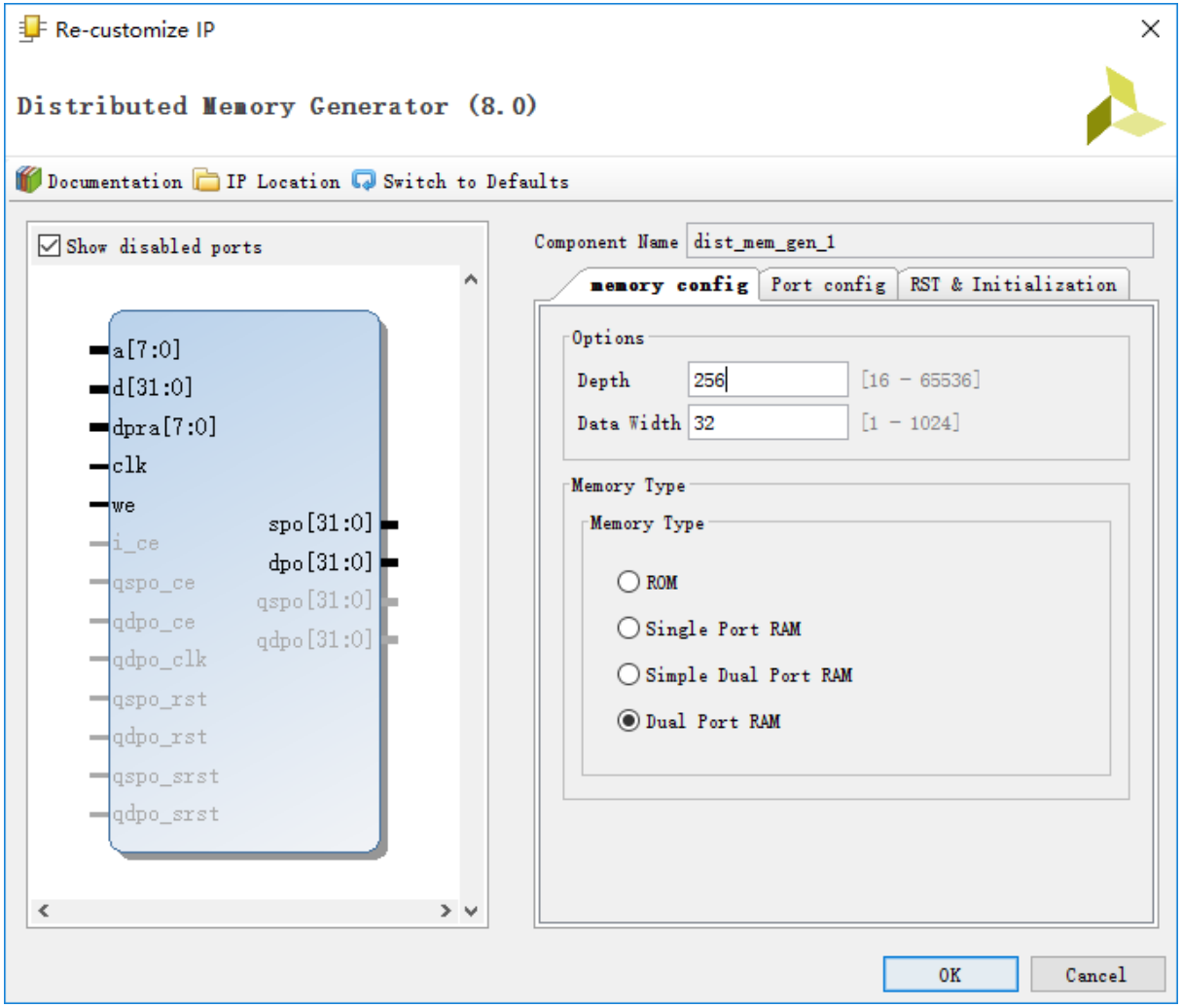
d: 写数据

spo： 读数据

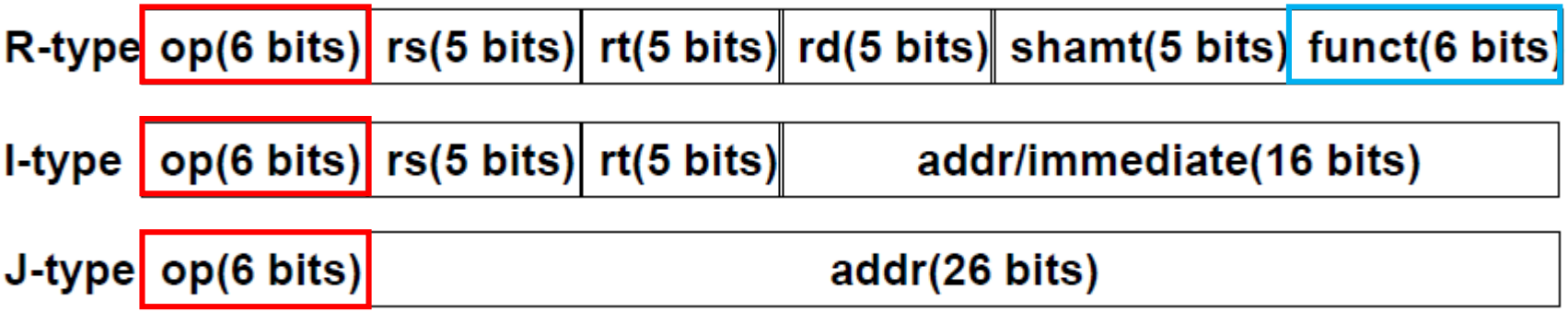
- **调试端口**

dpra: 读地址

dpo： 读数据



Encoding of the Opcode Field



opcode		*bits 28..26*							
		0	1	2	3	4	5	6	7
bits 31..29		000	001	010	011	100	101	110	111
0	000	*SPECIAL* δ	*REGIMM* δ	J	JAL	BEQ	BNE	BLEZ	BGTZ
1	001	ADDI	ADDIU	SLTI	SLTIU	ANDI	ORI	XORI	LUI
2	010	*COP0* δ	*COP1* δ	*COP2* θδ	*COP1X*[1] δ	BEQL ϕ	BNEL ϕ	BLEZL ϕ	BGTZL ϕ
3	011	β	β	β	β	*SPECIAL2* δ	JALX ε	ε	*SPECIAL3*[2] δΘ
4	100	LB	LH	LWL	LW	LBU	LHU	LWR	β
5	101	SB	SH	SWL	SW	β	β	SWR	CACHE
6	110	LL	LWC1	LWC2 θ	PREF	β	LDC1	LDC2 θ	β
7	111	SC	SWC1	SWC2 θ	*	β	SDC1	SDC2 θ	β

SPECIAL Opcode Encoding of Function Field

R-type	op(6 bits)	rs(5 bits)	rt(5 bits)	rd(5 bits)	shamt(5 bits)	funct(6 bits)

function		*bits 2..0*							
		0	1	2	3	4	5	6	7
bits 5..3		000	001	010	011	100	101	110	111
0	000	SLL[1]	*MOVCI* δ	*SRL* δ	SRA	SLLV	*	*SRLV* δ	SRAV
1	001	JR[2]	JALR[2]	MOVZ	MOVN	SYSCALL	BREAK	*	SYNC
2	010	MFHI	MTHI	MFLO	MTLO	β	*	β	β
3	011	MULT	MULTU	DIV	DIVU	β	β	β	β
4	100	ADD	ADDU	SUB	SUBU	AND	OR	XOR	NOR
5	101	*	*	SLT	SLTU	β	β	β	β
6	110	TGE	TGEU	TLT	TLTU	TEQ	*	TNE	*
7	111	β	*	β	β	β	*	β	β

测试代码

```
j   _start
.data
.word   0,8,1,6,0xfffffff8,1,3,5,0   #编译成机器码时编译器会在前面多加个0
_start:
addi $t0,$0,3        #t0=3
addi $t1,$0,5        #t1=5
addi $t2,$0,1        #t2=1

add  $s0,$t1,$t0    #s0=t1+t0=8  测试add指令 正确继续执行
lw   $s1,12($0
bne  $s1,$s0,_fail  #不正确跳到fail
……
```

测试代码（续1）

```
sw   $t1,40($0)
lw   $s1,40($0)
beq  $t1,$s1,_sucess

_fail:
sw   $0,8($0)        #失败通过看存储器地址0x08里值，若为0
                     则测试不通过，最初地址0x08里值为0
j    _fail

_sucess:
sw   $t2,8($0)       #全部测试通过，存储器地址0x08里值为1
j   _sucess          #判断测试通过的条件是最后存储器地址
                     0x08里值为1，说明全部通过测试
```

COE文件

```
memory_initialization_radix  = 16;
memory_initialization_vector =
0800000b                                j   _start
                                        .word
00000000 00000000 00000008              0,8,1,6,0xfffffff8,1,3,5,0
00000001 00000006 fffffff8
00000001 00000003 00000005
00000000                                addi $t0,$0,3       #t0=3
                                        addi $t1,$0,5       #t1=5
20080003 20090005 200a0001              addi $t2,$0,1       #t2=1

01288020                                add  $s0,$t1,$t0
……
```

寄存器使用约定

REGISTER	NAME	USAGE
$0	$zero	常量0(constant value 0)
$1	$at	保留给汇编器(Reserved for assembler)
$2-$3	$v0-$v1	函数调用返回值(values for results and expression evaluation)
$4-$7	$a0-$a3	函数调用参数(arguments)
$8-$15	$t0-$t7	暂时的(或随便用的)
$16-$23	$s0-$s7	保存的(或如果用，需要SAVE/RESTORE的)(saved)
$24-$25	$t8-$t9	暂时的(或随便用的)
$28	$gp	全局指针(Global Pointer)
$29	$sp	堆栈指针(Stack Pointer)
$30	$fp	帧指针(Frame Pointer)
$31	$ra	返回地址(return address)

The End