

关于测试单周期 CPU 的简单方法（修正）

（特别说明：本表每个同学都必须建立，检查实验时，必须提供！）

1、测试程序段

地址	汇编程序	指令代码					运行结果
		op(6)	rs(5)	rt(5)	rd(5)/immediate (16)	16 进制数代码	
0x00000000	addiu \$1,\$0,8	001001	00000	00001	0000 0000 0000 1000	= 24010008	\$1=8
0x00000004	ori \$2,\$0,2	001101	00000	00010	0000 0000 0000 0010	34020002	\$2=2
0x00000008	add \$3,\$2,\$1	000000	00010	00001	00011 00000 100000	00411820	\$3=10
0x0000000C	sub \$5,\$3,\$2	000000	00011	00010	00101 00000 100010	00622822	\$5=8
0x00000010	and \$4,\$5,\$2	000000	00101	00010	00100 00000 100100	00a22024	\$4=0
0x00000014	or \$8,\$4,\$2	000000	00100	00010	01000 00000 100101	00824025	\$8=2
0x00000018	sll \$8,\$8,1	000000	00000	01000	01000 00001 000000	00084040	\$8=4,=8
0x0000001C	bne \$8,\$1,-2 (#,转 18)	000101	01000	00001	1111111111111110	1501fffe	跳转 18,不跳转
0x00000020	slti \$6,\$2,4	001010	00010	00110	0000000000000100	28460004	\$6=1
0x00000024	slti \$7,\$6,0	001010	00110	00111	0000000000000000	28c70000	\$7=0
0x00000028	addiu \$7,\$7,8	001001	00111	00111	0000000000001000	24e70008	\$7=8, \$7=16
0x0000002C	beq \$7,\$1,-2 (=,转 28)	000100	00111	00001	1111111111111110	10e1fffe	跳转 28, 不跳转
0x00000030	sw \$2,4(\$1)	101011	00001	00010	0000000000000100	ac220004	M[12:15]=2
0x00000034	lw \$9,4(\$1)	100011	00001	01001	0000000000000100	8c290004	\$9=2
0x00000038	addiu \$10,\$0,-2	001001	00000	01010	1111111111111110	240afffe	\$10 = -2
0x0000003C	addiu \$10,\$10,1	001001	01010	01010	0000000000000001	254a0001	\$10 = -1, \$10 = 0
0x00000040	bltz \$10,-2(<0,转 3c)	000001	01010	00000	1111111111111110	0540fffe	转 3c,不转
0x00000044	andi \$11,\$2,2	001100	00010	01011	0000000000000010	304b0002	\$11 = 2
0x00000048	j 0x00000050	000010	00000	00000	0000000000010100	08000014	跳转 50
0x0000004C	or \$8,\$4,\$2	000000	00100	00010	01000 00000 100101	00824025	跳过
0x00000050	NOP	000000	00000	00000	0000000000000000	= 00000000	NO OP

补充测试

地址	汇编程序	指令代码					运行结果
		op(6)	rs(5)	rt(5)	rd(5)/immediate (16)	16 进制数代码	
0x00000054	lui \$10,10	001111	00000	01010	0000000000001010	3c0a000a	\$10=0x000a0000
0x00000058	mult \$11,\$6	000000	01011	00110	0000000000011000	01660018	hi=0 lo=2
0x0000005C	mfhi \$12	000000	00000	00000	01100 00000 010000	00006010	\$12=0
0x00000060	mflo \$12	000000	00000	00000	01100 00000 010010	00006012	\$12=2
0x00000064	div \$11,\$6	000000	01011	00110	00000 00000 011010	0166001a	hi=2 lo=0
0x00000068	jal 0x1b	000011	00000	00000	0000000000011011	0c00001b	Pc=6c
0x0000006C	bltzal \$12 -2	000001	01100	10000	1111111111111110	0590fffe	Pc=70
0x00000070	blez \$12 -3	000110	01100	00000	1111111111111101	1980fffd	Pc=74

0x00000074	sllv \$13,\$12,\$6	000000	00110	01100	01101 00000 000100	00cc6804	\$13=4
0x00000078	addi \$14,\$0,132	001000	00000	01110	00000 00001 010100	200e0084	\$14=0x84
0x0000007C	jr \$14	000000	01110	00000	00000 00000 001000	01c00008	Pc=0x84
0x00000080	nop	000000	00000	00000	0000000000000000	00000000	被跳过
0x00000084	sra \$12,\$12,1	000000	00000	01100	01100 00001 000011	000c6043	\$12=1 \$12=0
0x00000088	bgtz \$12,-2	000111	01100	00000	1111111111111110	1d80fffe	跳转 84 不跳转
0x0000008C	addi \$15,\$0,-1	001000	00000	01111	1111111111111111	200fffff	\$15=-1
0x00000090	bgez \$15,-4	000001	01111	00001	1111111111111100	05e1fffc	不跳转
0x00000094	sb \$15,0(\$s1)	101000	10001	01111	0000000000000000	a22f0000	mem[0]=-1
0x00000098	sh \$15,2(\$s1)	101001	10001	01111	0000000000000010	a62f0002	mem[2:3]=-1
0x0000009C	lb \$16,0(\$s1)	100000	10001	10000	0000000000000000	82300000	\$16=-1
0x000000A0	lh \$16,2(\$s1)	100001	10001	10000	0000000000000010	86300002	\$16=-1
0x000000A4	slt \$16,\$15,\$0	000000	01111	00000	10000 00000 101010	01e0802a	\$16=1
0x000000A8	sltiu \$16,\$15,-5	001011	01111	10000	1111111111111011	2df0fffb	\$16=0
0x000000AC	sltu \$16,\$15,\$0	000000	01111	00000	10000 00000 101011	01e0802b	\$16=0
0x000000B0	addi \$8,\$0,8	001000	00000	01000	00000 00000 001000	20080008	\$8=8
0x000000B4	srl \$8,\$8,1	000000	00000	01000	01000 00001 000010	00084042	\$8=4
0x000000B8	srlv \$8,\$8,\$16	000001	01000	10001	00000 00000 000000	02084006	\$8=4
0x000000BC	bgezal \$8,0	000000	10000	01000	01000 00000 000110	05110000	Pc+4 \$31=0xC0
0x000000C0	addu \$9,\$8,\$13	000000	01000	01101	01001 00000 100001	010d4821	\$9=8
0x000000C4	subu \$9,\$8,\$13	000000	01000	01101	01001 00000 100011	010d4823	\$9=0
0x000000C8	nor \$9,\$8,\$13	000000	01000	01101	01001 00000 100111	010d4827	\$9= -5
0x000000CC	xor \$9,\$8,\$13	000000	01000	01101	01001 00000 100110	010d4826	\$9=-0
0x000000D0	sra \$9,\$8,\$13	000000	01000	01101	01001 00000 000111	01a84807	\$9= 0
0x000000D4	xori \$13,\$8,-1	001110	01000	01101	1111111111111111	390dffff	\$9 = -5
0x000000D8	syscall	000000	00000	00000	00000 00000 001100	0000000c	Pc+4
0x000000DC	halt	111111	00000	00000	0000000000000000	= FC000000	停机

- 1、将指令代码初始化到指令存储器中，直接写入。
- 2、初始化 PC 的值，也就是以上程序段首地址 PC=0x00000000，以上程序段从 0x00000000 地址开始存放。
- 3、运行 Xilinx Vivado 进行仿真，看波形。