**关于测试多周期CPU的简单方法**

**（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** | | | | | |
| **op（6）** | **rs(5)** | **rt(5)** | **rd(5)/immediate (16)** | **16进制数代码** | |
| **0x00000000** | addi $1,$0,8 | 000010 | **00000** | **00001** | **0000 0000 0000 1000** | **=** | 08010008 |
| **0x00000004** | ori $2,$0,2 | 010010 | 00000 | 00010 | 0000 0000 0000 0010 |  | 48020002 |
| **0x00000008** | xori $3,**$2**,8 | 010011 | 00010 | 00011 | 0000 0000 0000 1000 |  | 4C430008 |
| **0x0000000C** | sub $4,**$3**,$1 | 000001 | 00011 | 00001 | 0010 0000 0000 0000 |  | 04612000 |
| **0x00000010** | and $5,**$4**,$2 | 010000 | 00100 | 00010 | 0010 1000 0000 0000 |  | 4082280 |
| **0x00000014** | **sll $5,$5,2** | 011000 | 00000 | 00101 | 0010 1000 1000 0010 |  | 60005882 |
| **0x00000018** | **beq $5,$1,-2(=,转14)** | 110100 | 00101 | 00001 | 1111 1111 1111 1110 |  | D0A1FFFE |
| **0x0000001C** | **jal 0x0000050** | 111010 | 00000 | 00000 | 0000 0000 0001 0100 |  | E8000014 |
| **0x00000020** | **slt $8,$13,$1** | 100111 | 01101 | 00001 | 0100 0000 0000 0000 |  | 9DA14000 |
| **0x00000024** | **addi $14,$0,-2** | 000010 | 00000 | 01110 | 1111 1111 1111 1110 |  | 080EFFFE |
| **0x00000028** | **slt $9,$8,$14** | 100111 | 01000 | 01110 | 0100 1000 0000 0000 |  | 9D0E4800 |
| **0x0000002C** | **slti $10,$9,2** | 100110 | 01001 | 01010 | 0000 0000 0000 0010 |  | 992A0002 |
| **0x00000030** | **slti $11,$10,0** | 100110 | 01010 | 01011 | **0000 0000 0000 0000** |  | 994B0000 |
| **0x00000034** | add $11,**$11**,$10 | 000000 | 01011 | 01010 | **0101 1000 0000 0000** |  | 016A5800 |
| **0x00000038** | **bne $11,$2,-2 (≠,转34)** | 110101 | 01011 | 00010 | **1111 1111 1111 1110** |  | D562FFFE |
| **0x0000003C** | addi $12,$0,-2 | 000010 | 00000 | 01100 | **1111 1111 1111 1110** |  | 080CFFFE |
| **0x00000040** | addi $12,$12,1 | 000010 | 01100 | 01100 | **0000 0000 0000 0001** |  | 098C0001 |
| **0x00000044** | **bltz $12,-2 (<0,转40)** | 110110 | 01100 | 00000 | **1111 1111 1111 1110** |  | D980FFFE |
| **0x00000048** | andi $12,$2,2 | 010001 | 00010 | 01100 | **0000 0000 0000 0010** |  | 444C0002 |
| **0x0000004C** | **j 0x000005C** | 111000 | 00000 | 00000 | **0000 0000 0001 0111** |  | E0000017 |
| **0x00000050** | sw $2,4($1) | 110000 | 00001 | 00010 | 0000 0000 0000 0100 |  | C020004 |
| **0x00000054** | lw $13,4($1) | 110001 | 00001 | 01101 | 0000 0000 0000 0100 |  | C42D0004 |
| **0x00000058** | **jr $31** | 111001 | 11111 | 00000 | 0000 0000 0000 0000 |  | E7E0000 |
| **0x0000005C** | **halt** | 111111 | 00000 | 00000 | 0000000000000000 | **=** | FC000000 |
|  |  |  |  |  |  |  |  |

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。