

MIPS32 监控程序模拟实验

2017011341, 陈旭

2019 年 9 月

1 熟悉环境

1.1 启动模拟器

- 运行... 启动 MIPS32 模拟器;
- 运行... 连接 MIPS32 模拟器, 然后在弹出的命令行中执行相关指令。

1.2 A 指令

用于向内存中写入汇编代码, 使用方法: 在命令行中输入 A, 并回车, 接着输入需要写入的内存地址, 然后即可开始编写汇编代码, 以空行结束。

图 1: A 指令示例

1.3 G 指令

编译并运行指定地址上的汇编代码, 使用方法: 在命令行输入 G, 并回车, 接着输入汇编代码所在的内存地址。

图 2: G 指令示例

1.4 D 指令

查看内存上的数据, 使用方法: 在命令行输入 D, 并回车, 接着输入需要查看的内存地址, 以及查看的长度。

图 3: D 指令示例

2 实验内容

2.1 斐波那契数列

2.1.1 实验说明

求前 10 个 Fibonacci 数，将结果保存到起始地址为 0x80400000 的 10 个字中，并用 D 命令检查结果正确性。

2.1.2 实验思路

实验主体是实现一个循环，在循环中通过三个变量之间的轮转来求斐波那契数列，同时将数写入内存中，每次循环内存地址和计数变量都增加（内存地址每次增加 4），计数变量到 10 时退出循环。

2.1.3 实验代码

```
1 li $t0, 0x1
2 li $t1, 0x1
3 li $t3, 0x0
4 li $t4, 0xA
5 li $a0, 0x80400000
6 sw $t0, 0($a0)
7 addio $a0, $a0, 4
8 addu $t2, $t0, $t1
9 move $t0, $t1
10 move $t1, $t2
11 addiu $t3, $t3, 0x1
12 bne $t3, $t4, -28
13 nop
14 jr $31
15 nop
```

2.1.4 实验结果

图 4: 任务 1 实验结果

2.2 可见字符

2.2.1 实验说明

输出所有可见字符（0x21~0x7F）到终端。

2.2.2 实验思路

实验主体是实现一个循环，从 0x21 开始将对应的字符输出，每次循环对输出的 ASCII 码进行累加，大于 0x7F 则退出。

2.2.3 实验代码

```
1  li $t0, 0x1
2  li $t1, 0x1
3  li $t3, 0x0
4  li $t4, 0xA
5  li $a0, 0x80400000
6  sw $t0, 0($a0)
7  addio $a0, $a0, 4
8  addu $t2, $t0, $t1
9  move $t0, $t1
10 move $t1, $t2
11 addiu $t3, $t3, 0x1
12 bne $t3, $t4, -28
13 nop
14 jr $31
15 nop
```

2.2.4 实验结果

图 5: 任务 2 实验结果

2.3 斐波那契数

2.3.1 实验说明

求第 60 个 Fibonacci 数，将结果保存到起始地址为 0x80400000 的 8 个字节中，并用 D 命令检查结果正确性。

2.3.2 实验思路

整体思路同任务 1，难点在于数值较大，计算时需要将数字拆分为高 32 位和低 32 位分别计算，特别处理的地方在于低位计算溢出时需要向高位进位，即高位需要加上 1。

2.3.3 实验代码

```
1  li $t0, 0x1
2  li $t1, 0x1
3  li $t3, 0x0
4  li $t4, 0xA
5  li $a0, 0x80400000
6  sw $t0, 0($a0)
7  addio $a0, $a0, 4
8  addu $t2, $t0, $t1
9  move $t0, $t1
10 move $t1, $t2
11 addiu $t3, $t3, 0x1
12 bne $t3, $t4, -28
13 nop
14 jr $31
15 nop
```

2.3.4 实验结果

图 6: 任务 3 实验结果

3 实验总结

本次实验内容对于平常的编程来说都是属于比较简单的题目，而实验目的主要也是为了熟悉 MIPS32 指令，在实验的同时也自己琢磨了一些 Debug 的方法，相信这次实验对之后工作会有很大的帮助。