MIPS模拟器实验报告

实验要求

以程序模拟MIPS/RISC-V运行，功能包括：
汇编器：将汇编程序转换成机器码。能有较灵活的格式，可以处理格式指令、表达式、有出错信息。
汇编反汇编：汇编指令与机器码的相互转换。
模拟器：根据机器码模拟执行可以运行简单汇编程序。

1、模拟器运行界面设计：可以命令行或窗口界面。可以执行指令的汇编、反汇编，可以单步执行指令观察寄存器、内存的变化。（命令行版可参考DEBUG）
2、指令伪指令的汇编反汇编：将汇编指令转换成二进制机器码，能够处理标号、变量。
3、MMU存储器管理单元：存储器存取模拟。大头小头，对齐不对齐，Cache，虚拟存储。
4、指令：常用5~10条；

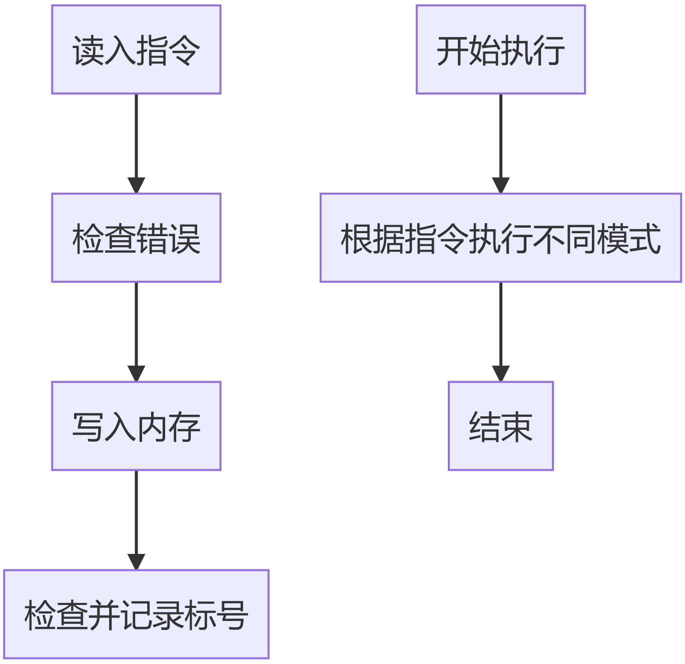
命令：-->R-看寄存器，-->D-数据方式看内存，-->U-指令方式看内存，-->A-写汇编指令到内存，

-->T-单步执行内存中的指令

实现功能

1. 读取文件中指令
2. 支持label和指令间跳转
3. 支持查看寄存器、内存、单步执行、全部执行
4. 支持运行中写入指令（仅一条）
5. 实现了小头模式读内存
6. 支持add, and, or, addi, sub, lw, sw, beq, j指令的实现
7. 有报错信息

程序架构



MipsCPU 类

```cpp
class MipsCPU {
public:
    const int MAXSIZE = 4096;
    int PC, IR;
    int *Rgf;
    int *memory;
    map<string, int> rgf;
    map<int, const char*> num2rgf;
    map<string, int> label;
    int size;
    int reg2num(string s)
    int str2num(string s)
    int unasm(string in) //反汇编
    bool load() //载入指令
    void disp() //显示寄存器
    void disp_inst()//显示指令
    void disp_num(int mode,int num) //小端输出
    void disp_memo() //展示内存
    void exec(int mode)//执行指令
};
```

测试过程

测试代码：

```
addi $t1, $zero, 3
loop:
add $s0, $t1, $s0
addi $t0, $t0, 1
addi $t2, $zero, 3
beq $s0, $t1, Corret
Incorret:
addi $a0, $zero, 1
j end
Corret:
add $a0, $zero, $s0
j end
end:
addi $s2, $zero, 16708
sub $s2, $s2, $a0
sw $s2, 96($zero)
lw $s1, 96($zero)
```

测试过程：

-u查看指令

```
-u
13
0000:0x20090003 ADDi    $t1, $zero, 3
0001:0x01308020 ADD     $s0, $t1, $s0
0002:0x21080001 ADDi    $t0, $t0, 1
0003:0x200A0003 ADDi    $t2, $zero, 3
0004:0x12090002 BEQ     $t1, $s0, 2
0005:0x20040001 ADDi    $a0, $zero, 1
0006:0x08000009 J       9
0007:0x00102020 ADD     $a0, $zero, $s0
0008:0x08000009 J       9
0009:0x20124144 ADDi    $s2, $zero, 16708
000A:0x02449022 SUB     $s2, $s2, $a0
000B:0xAC120060 SW      $s2, 96($zero)
000C:0x8C110060 LW      $s1, 96($zero)
```

-a写入内存

```
-a
0x0000: ADDi $t1, $zero, 2
-u
13
0000:0x20090002 ADDi    $t1, $zero, 2
0001:0x01308020 ADD     $s0, $t1, $s0
0002:0x21080001 ADDi    $t0, $t0, 1
0003:0x200A0003 ADDi    $t2, $zero, 3
0004:0x12090002 BEQ     $t1, $s0, 2
0005:0x20040001 ADDi    $a0, $zero, 1
0006:0x08000009 J       9
0007:0x00102020 ADD     $a0, $zero, $s0
0008:0x08000009 J       9
0009:0x20124144 ADDi    $s2, $zero, 16708
000A:0x02449022 SUB     $s2, $s2, $a0
000B:0xAC120060 SW      $s2, 96($zero)
000C:0x8C110060 LW      $s1, 96($zero)
```

-t单步执行

```
-t
0x20090002      ADDi    $t1, $zero, 2
$zero = 00000000   $at = 00000000   $v0 = 00000000   $v1 = 00000000
  $a0 = 00000000   $a1 = 00000000   $a2 = 00000000   $a3 = 00000000
  $t0 = 00000000   $t1 = 00000002   $t2 = 00000000   $t3 = 00000000
  $t4 = 00000000   $t5 = 00000000   $t6 = 00000000   $t7 = 00000000
  $s0 = 00000000   $s1 = 00000000   $s2 = 00000000   $s3 = 00000000
  $s4 = 00000000   $s5 = 00000000   $s6 = 00000000   $s7 = 00000000
  $t8 = 00000000   $t9 = 00000000   $k0 = 00000000   $k1 = 00000000
  $gp = 00000000   $sp = 00000000   $fp = 00000000   $ra = 00000000
```

-r查看寄存器

```
-r
$zero = 00000000   $at = 00000000   $v0 = 00000000   $v1 = 00000000
  $a0 = 00000000   $a1 = 00000000   $a2 = 00000000   $a3 = 00000000
  $t0 = 00000000   $t1 = 00000002   $t2 = 00000000   $t3 = 00000000
  $t4 = 00000000   $t5 = 00000000   $t6 = 00000000   $t7 = 00000000
  $s0 = 00000000   $s1 = 00000000   $s2 = 00000000   $s3 = 00000000
  $s4 = 00000000   $s5 = 00000000   $s6 = 00000000   $s7 = 00000000
  $t8 = 00000000   $t9 = 00000000   $k0 = 00000000   $k1 = 00000000
  $gp = 00000000   $sp = 00000000   $fp = 00000000   $ra = 00000000
```

-p全部执行

```
-p
0x01308020      ADD     $s0, $t1, $s0
0x21080001      ADDi    $t0, $t0, 1
0x200A0003      ADDi    $t2, $zero, 3
0x12090002      BEQ     $t1, $s0, 2
0x00102020      ADD     $a0, $zero, $s0
0x08000009      J       9
0x20124144      ADDi    $s2, $zero, 16708
0x02449022      SUB     $s2, $s2, $a0
0xAC120060      SW      $s2, 96($zero)
0x8C110060      LW      $s1, 96($zero)
$zero = 00000000   $at = 00000000   $v0 = 00000000   $v1 = 00000000
  $a0 = 00000002   $a1 = 00000000   $a2 = 00000000   $a3 = 00000000
  $t0 = 00000001   $t1 = 00000002   $t2 = 00000003   $t3 = 00000000
  $t4 = 00000000   $t5 = 00000000   $t6 = 00000000   $t7 = 00000000
  $s0 = 00000002   $s1 = 00004142   $s2 = 00004142   $s3 = 00000000
  $s4 = 00000000   $s5 = 00000000   $s6 = 00000000   $s7 = 00000000
  $t8 = 00000000   $t9 = 00000000   $k0 = 00000000   $k1 = 00000000
  $gp = 00000000   $sp = 00000000   $fp = 00000000   $ra = 00000000
```

-d查看内存

```
-d
02 00 09 20 20 80 30 01
01 00 08 21 03 00 0A 20
02 00 09 12 01 00 04 20
09 00 00 08 20 20 10 00
09 00 00 08 44 41 12 20
22 90 44 02 60 00 12 AC
60 00 11 8C 00 00 00 00
```

报错

```
add $s0, $t1
~~~~~~~~~~~~^
ERROR: a register is missed here
```