# 山东大学 计算机科学与技术 学院

# 计算机体系结构 课程实验报告

学号: 202200130048 姓名: 陈静雯 班级: 6

实验题目:用 WinDLX 模拟器执行程序 求最大公约数

#### 实验目的:

通过本实验,熟练掌握 WinDLX 模拟器的操作和使用,清楚 WinDLX 五段流水线在执行具体程序时的流水情况,熟悉 DLX 指令集结构及其特点。

#### 硬件环境:

Windows

## 软件环境:

0tvdm

实验程序: 给出的 gcm. s

## 实验程序解读:

## 主程序部分(main)

- 1. main+0x4: jal InputUnsigned
  - 。 跳转目标: InputUnsigned 子程序
  - 。 操作:调用无符号整数输入函数,将第一个输入值存入 r1。
- 2. main+0x8: BID 0x00201020
  - 。 操作: 将 r1 的值复制到 r2 (add r2, r1, r0), 用于暂存第一个输入值。
- 3. main+0xc: addr1, 0, 0x100e
  - 。 操作:设置 r1 指向第二个输入提示符 (Prompt2 的地址),准备第二次输入。
- 4. main+0x10: jal InputUnsigned
  - 。 跳转目标: InputUnsigned 子程序
  - 操作:读取第二个输入值到 r1,此时 r1 和 r2 分别存储两个输入数。

#### GCD 计算循环(gcm. Loop)

- 5. gcm. Loop: seq r3, r1, r2
  - 操作: 比较 r1 和 r2 是否相等,结果存入 r3(相等为 1,否则为 0)。
- 6. 0x00000118: bnez r3, Result
  - 跳转目标: Result
  - 条件: 若 r1 == r2, 跳转到结果输出部分。
- 7. 0x0000011c: sgt r3, r1, r2
  - 操作: 检查 r1 是否大于 r2, 结果存入 r3 (大于为 1, 否则为 0)。
- 8. 0x00000120: bnez r3, r1Greater
  - 跳转目标: r1Greater
  - 条件: 若 r1 > r2, 跳转到 r1Greater 执行减法。

## 减法分支逻辑

- 9. r2Greater: sub r2, r2, r1
  - 操作: r2 = r2 r1(当 r2 > r1 时执行)。
- 10. 0x00000128: j gcm. Loop
  - 。 跳转目标: gcm. Loop
  - 操作:跳回循环开头继续比较。
- 11. r1Greater: sub r1, r1, r2
  - 操作: r1 = r1 r2(当 r1 > r2 时执行)。
- 12. 0x00000130: j gcm. Loop
  - 跳转目标: gcm. Loop
  - 。 **操作**: 跳回循环开头继续比较。

## 结果输出(Result)

- 13. Result: sw PrintValue[r0], r1
  - 。 操作: 将结果 r1 存入 PrintValue 的内存地址。
- 14. Result+0x4: addr1, 4, 0, 0x1028
  - 操作:设置 r14 指向输出参数表 PrintfPar。
- 15. Result+0x8: trap 0x5
  - 操作:调用 printf 输出结果,但因参数错误可能输出错误值。
- 16. Result+0xc: trap 0x0
  - 。 操作:终止程序。

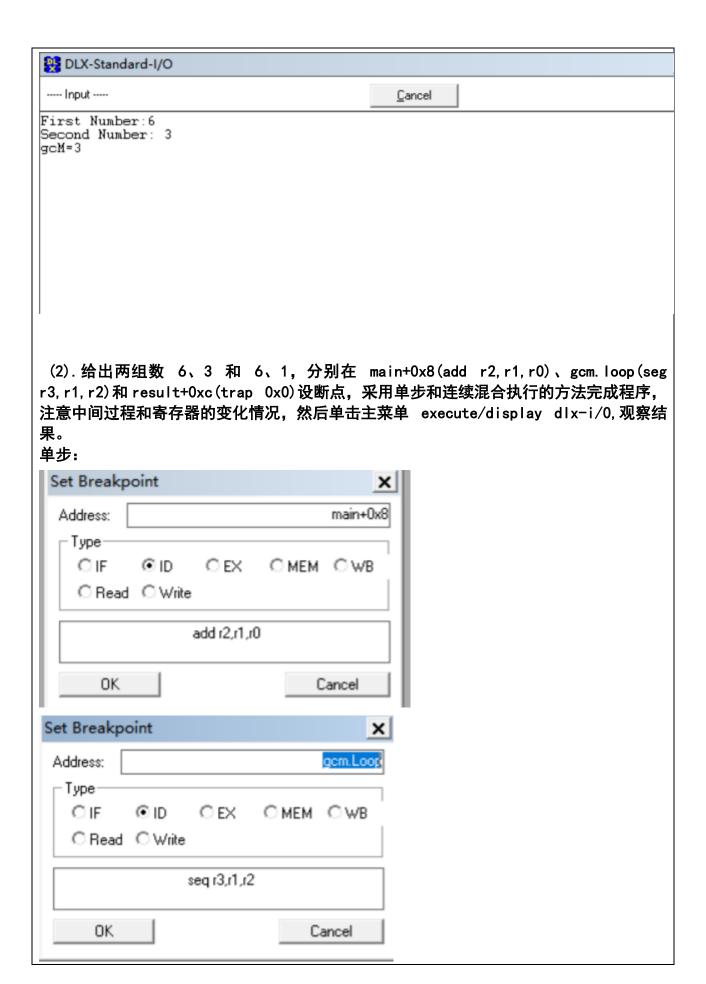
## 输入处理(input. Loop)

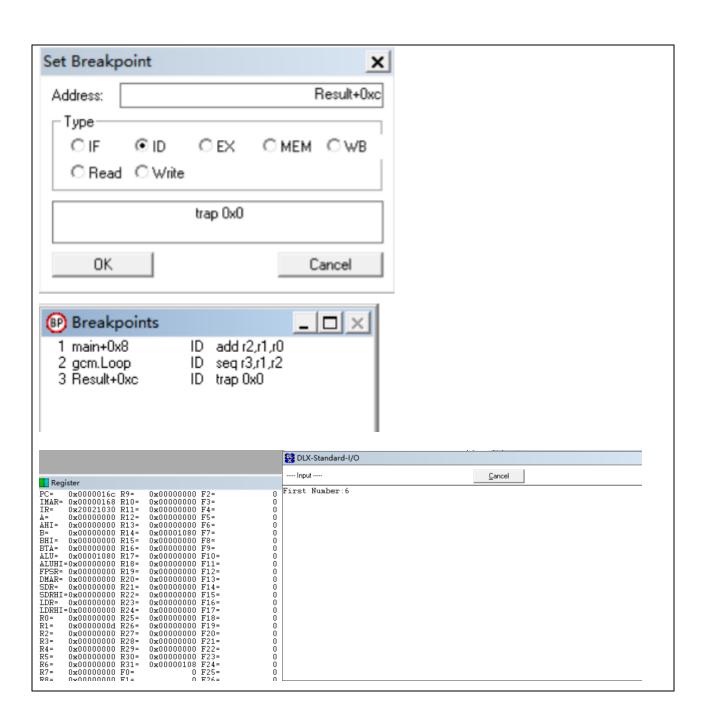
- 17. input. Loop: Ibur3, 0x0fz
  - 。 操作: 从输入缓冲区读取一个字符到 r3。
- 18. 0x00000178: seg r5, r3, 0xa
  - 操作: 检查字符是否为换行符(ASCII 0xA)。
- 19. 0x0000017c: bnez r5, Finish
  - 跳转目标: Finish
  - 条件: 若遇到换行符, 结束输入循环。
- 20. 0x00000180: sub r3, r3, 0x30
  - 操作:将 ASCII 字符转换为数字(减去 '0' 的 ASCII 码)。
- 21. 0x00000184: multu r1, r1, r4
  - 。 操作:将当前数值乘以 10。
- 22. 0x00000188: add r1, r1, r3
  - 。 操作: 累加新读取的数字到 r1。

#### 实验步骤与内容:

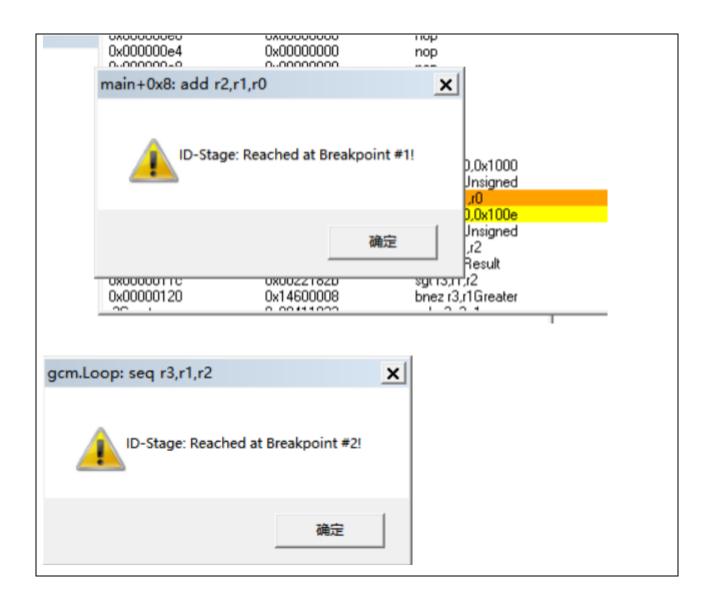
(1) 用 WinDLX 模拟器执行程序 gcm. s 。

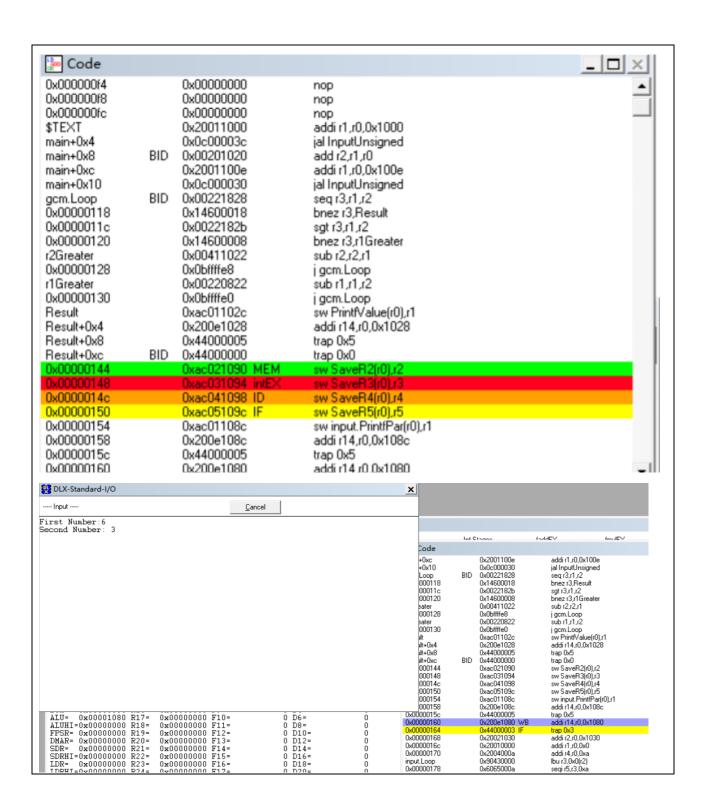
该程序从标准输入读入两个整数,求他们的 greatest common measure,然后将结果写到标准输出。 该程序中调用了 input.s 中的输入子程序。

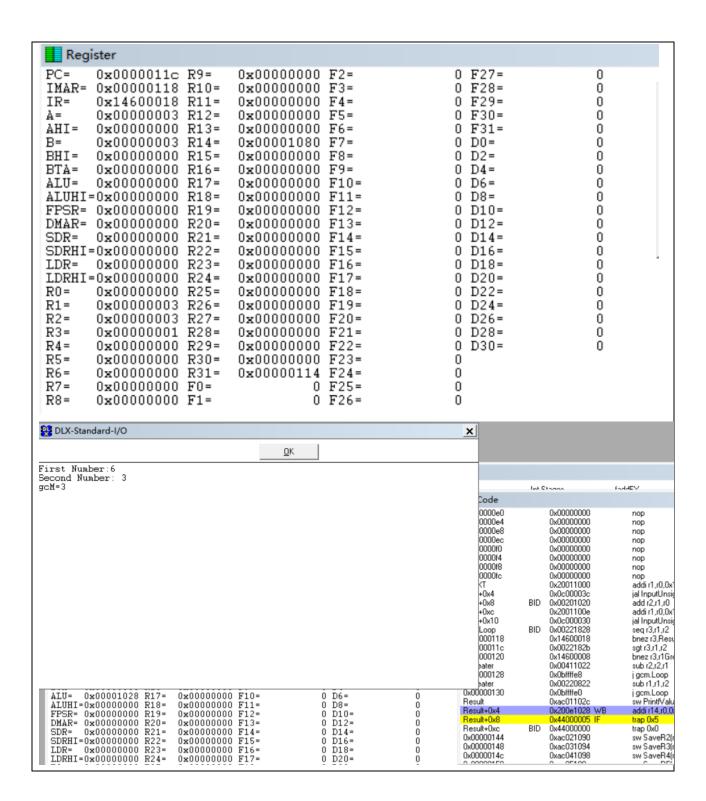


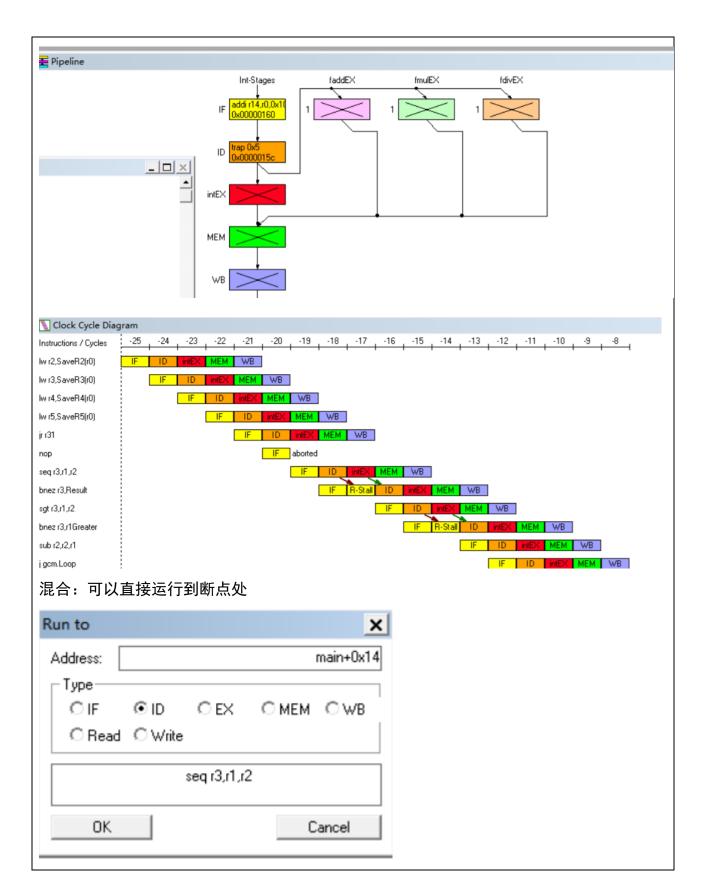


```
Register
PC=
      0x00000170 R9=
                         0x00000000 F2=
                                                      0
                                                       Ŧ
                                                      0 I
IMAR=
      0x0000016c R10=
                         0x00000000 F3=
IR=
      0x20010000 R11=
                         0x00000000 F4=
                                                      0
                                                        Ε
      0x00000000 R12=
                                                      0
                                                        Ε
A=
                         0x00000000 F5=
                                                        Ε
AHI=
      0x00000000 R13=
                         0x00000000 F6=
                                                      0
B=
      0x00000000 R14=
                         0x00001080 F7=
                                                      0 I
BHI=
      0x00000000 R15=
                         0x00000000 F8=
                                                      0
                                                       Ι
BTA=
      0x00000000 R16=
                         0x00000000 F9=
                                                      0
                                                       Ι
ALU=
      0x00000000 R17=
                         0x00000000 F10=
                                                      0
                                                        Ι
ALUHI=0x00000000 R18=
                         0x00000000 F11=
                                                      0
                                                       Ι
FPSR= 0x00000000 R19=
                         0x00000000 F12=
                                                      0 L-
DMAR= 0x00000000 R20=
                         0x00000000 F13=
                                                      0
                                                       D12=
SDR=
      0x00000000 R21=
                         0x00000000 F14=
                                                      0
                                                       D14=
SDRHI=0x00000000 R22=
                         0x00000000 F15=
                                                      0 D16=
      0x00000000 R23=
                         0x00000000 F16=
                                                      0
                                                        D18=
LDR=
LDRHI=0x00000000 R24=
                         0x00000000 F17=
                                                      0
                                                        D20=
R0 =
      0x00000000 R25=
                                                      0 D22=
                         0x00000000 F18=
R1=
      0x00000002 R26=
                         0x00000000 F19=
                                                      0 D24=
R2=
      0x00000000 R27=
                         0x00000000 F20=
                                                      0 D26=
R3=
      0x00000000 R28=
                         0x00000000 F21=
                                                      0 D28=
      0x00000000 R29=
                         0x00000000 F22=
R4=
                                                      0 D30=
R5=
      0x00000000 R30=
                         0x00000000 F23=
                                                      0
R6=
      0x00000000 R31=
                         0x00000108 F24=
                                                      0
                                   0 F25=
R7=
                                                      Ū
      0x00000000 F0=
Register
PC=
      0x00000170 R9=
                         0x00000000 F2=
IMAR=
      0x0000016c R10=
                         0x00000000 F3=
IR=
      0x20010000 R11=
                         0x00000000 F4=
      0x00000000 R12=
                         0x00000000 F5=
A=
AHI=
      0x00000000 R13=
                         0x00000000 F6=
B=
      0x00000000 R14=
                         0x00001080 F7=
                                                      0 D
      0x00000000 R15=
                         0x00000000 F8=
BHI=
                                                  Matis Statis
BTA=
      0x00000000 R16=
                         0x00000000 F9=
                                                  Total:
ALU=
      0x00000000 R17=
                         0x00000000 F10=
                                                     69 Cyc
ALUHI=0x00000000 R18=
                         0x00000000 F11=
      0x00000000 R19=
                         0x00000000 F12=
                                                     ID exe
FPSR=
                                                     3 Instr
DMAR= 0x00000000 R20=
                         0x00000000 F13=
SDR=
      0x00000000 R21=
                         0x00000000 F14=
SDRHI=0x00000000 R22=
                         0x00000000 F15=
                                                  Hardwa:
      0x00000000 R23=
                         0x00000000 F16=
                                                     Memo
LDR=
                                                     faddE)
LDRHI=0x00000000 R24=
                         0x00000000 F17=
P0 =
      0~0000000
                  R25 =
                                                     fmulE>
                         0x00000000 F18=
R1=
                  R26=
      0 \times 000000002
                         0x00000000 F19=
                                                     fdivE×
R2=
                  R27=
                                                     Forwa
      0x00000006
                         0x000000000 F20=
      0×000000000 128=
R3=
                         0x00000000 F21=
R4=
      0x00000000 R29=
                                                  Stalls
                         0x00000000 F22=
                                                     RAW
R5=
      0x00000000 R30=
                         0x000000000 F23=
                                                        L[
R6=
      0x00000000 R31=
                         0x00000114 F24=
                                                        Br
R7=
      0x00000000 F0=
                                   0 F25=
R8=
      0x00000000 F1=
                                   0 F26=
                                                        FΙ
                                                     WAW
```









# B DLX-Standard-I/O

<u>0</u>K

First Number:6 Second Number: 1 gcM=1

## 结论分析与体会:

- 1. DLX 的固定 5 级流水线(IF/ID/EX/MEM/WB)要求程序员关注指令间的依赖性。例如,在 Ibu 后直接使用 r3 需插入 nop 或调整指令顺序以避免数据冒险。
- 2. 子程序调用(jal)与返回(jr r31)的机制需清晰掌握,避免因寄存器覆盖导致返回地址丢失。
- 3. 本次实验巩固了汇编语言编程能力,加深了对 DLX 架构及流水线机制的理解。通过调试与修复代码缺陷,不仅掌握了算法实现的底层细节,更培养了严谨的工程思维与问题解决能力。未来可进一步探索性能优化与功能扩展,提升代码的实用性。