

作业 1

袁晨圃 2023K8009929012

1 (1.7): 解释概念：主机、CPU、主存、存储单元、存储元件、存储基元、存储元、存储字、存储字长、存储容量、机器字长、指令字长

解：

- 主机：主存+CPU
- CPU：中央处理器，ALU+CU
- 主存：能直接与 CPU 交互的存储程序和数据的元件
- 存储单元：存储器中最小的可寻址单位
- 存储元件：可存储 1 位 0/1 的元件
- 存储基元/存储元：存储元件
- 存储字长：一个存储单元中的位数
- 存储容量：存储单元个数 × 存储字长
- 机器字长：CPU 一次能处理的二进制数的位数
- 指令字长：CPU 处理的指令位数，是字节的整数倍

2 (1.9): 画出主机框图，分别以存数指令 STA M 和加法指令 ADD M (M 均为主存地址) 为例，在图中按序标出完成该指令（包括取指阶段）的信息流程（如^①→）。假设主存容量为 256M × 32 位，在指令字长、存储字长、机器字长相等的条件下，指出图中各寄存器的位数。

解：

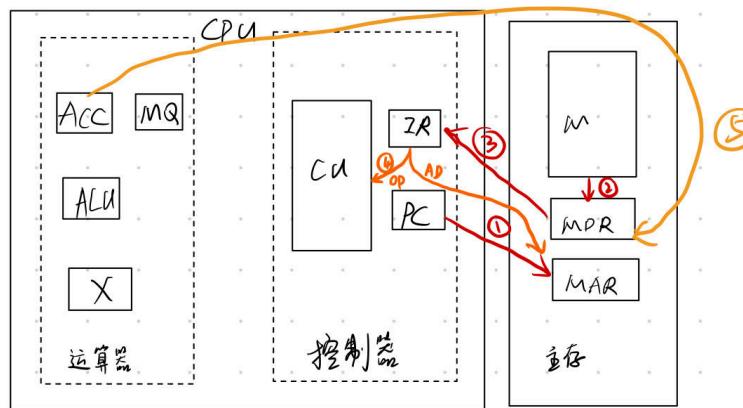


Figure 1: STA M 流程

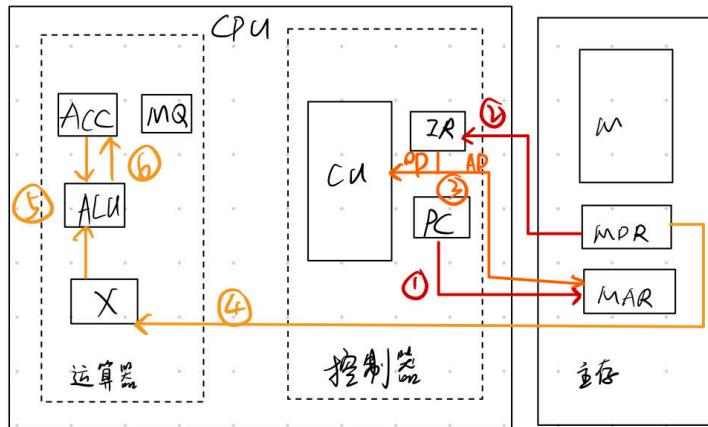


Figure 2: ADD M 流程

寄存器位数：

32 位寄存器：IR、MDR、ACC, X

28 位寄存器：MAR、PC

3 (1.10): 根据迭代公式 $\sqrt{x} = \frac{1}{2} \left(y_n + \frac{x}{y_n} \right)$ 设初态 $y_0 = 1$ ，要求精度为 ε ，试编制求 \sqrt{x} 的解题程序，并结合所编程序简述计算机的解题过程。

解：

```

main:
    li x1, 0.5 ; coefficient
    li x2, $x ; x
    li x3, 1.0 ; y0
    mv x4, x2 ; x / y0
    j while_entry
while_entry:
    ; calculate x3_new = x1 * (x3 + x4), i.e. 0.5(y_n+x/y_n)
    add x3, x3, x4
    mul x3, x3, x1
    ; calculate x4_new = x2 / x3, i.e. x / y_n
    div x4, x2, x3
    ; calculate x5 = abs(x4_new - x3_new)
    sub x5, x4, x3
    ; calculate abs x5
    bge x5, x5, positive
    neg x5, x5
    j positive
positive:
    ; compare x5 and $epsilon$
    li x6, $epsilon
    ; if x5 >= $epsilon$, then continue loop, else break
    bge x5, x6, while_body
    j while_end
while_body:
    j while_entry
while_end:

```

```
; calculate answer = x1 * (x3 + x4), i.e. 0.5(y_n+x/y_n)
add x3, x3, x4
mul x3, x3, x1
; store answer
mv a0, x3
ret
```

指令系统：使用 RISC-V 汇编，但假设寄存器和指令都可以存储计算浮点数

\$x 和 \$epsilon 处是题目给出的 x 和 ε

计算流程：上下界分别是 y_n 和 $\frac{x}{y_n}$ （或反过来），当上下界的差小于 ε 时，答案就是 $\frac{1}{2} \left(y_n + \frac{x}{y_n} \right)$ ，反之 $y_{n+1} = \frac{1}{2} \left(y_n + \frac{x}{y_n} \right)$ ，继续下一步迭代