

Homework 3

1 改造处理器

这条指令读一个指令，选择一个寄存器，产生一个立即数，计算和，选择一个寄存器，作为内存地址，写内存

需要在 `Reg[]` 的出口 `DataA`, `DataB` 产生的地方增加一个多路复用器，选择 `DataA`, `DataB` 的源寄存器，例如选择 `DataA` 来自 `Reg[rs2]`，那么 `DataB` 就来自 `Reg[rs1]`（也不是改 `Reg[]` 内部）

这样一来还需要在这里加一个控制信号（记为 `ctrl1`），控制 MUX，控制信号为 1，则 `DataA=Reg[rs1]`，`DataB=Reg[rs2]`；控制信号为 0，则相反

还需要在 `DMEM` 之前加一个 MUX，选择 `ADDR` 和 `Dataw` 的来源，控制信号（记为 `ctrl2`）为 1，`Addr` 来自 `ALU`，`Dataw` 来自 `DataB`

对于这个指令，`ctrl1=0`，`ctrl2=0`，`Reg[rs2]` 和 `idata` 相加，得到结果传到 `Dataw` 门口，`Reg[rs1]` 传到 `Addr` 门口，下个时钟周期写入

2 流水线处理器

因为分支延迟导致阻塞的时钟周期为 1 个

`bne t0, s5, exit` 因为 `t0` 的计算产生了一个周期的阻塞

因为 `bne t0, s5, exit` 计算 `PC` 的值又产生了一个周期的延迟才执行 `add s3, s3, s4` 或 `exit`

3 指令调度

调整后的代码如下

```
ld [sp+4] -> r2
ld [sp+8] -> r1
ld [sp+16] -> r2
add r1, r2 -> r1
ld [sp+20] -> r1
st r1 -> [sp+0]
sub r2, r1 -> r1
st r1 -> [sp+12]
```

一个周期都不用停