

## 实验题目

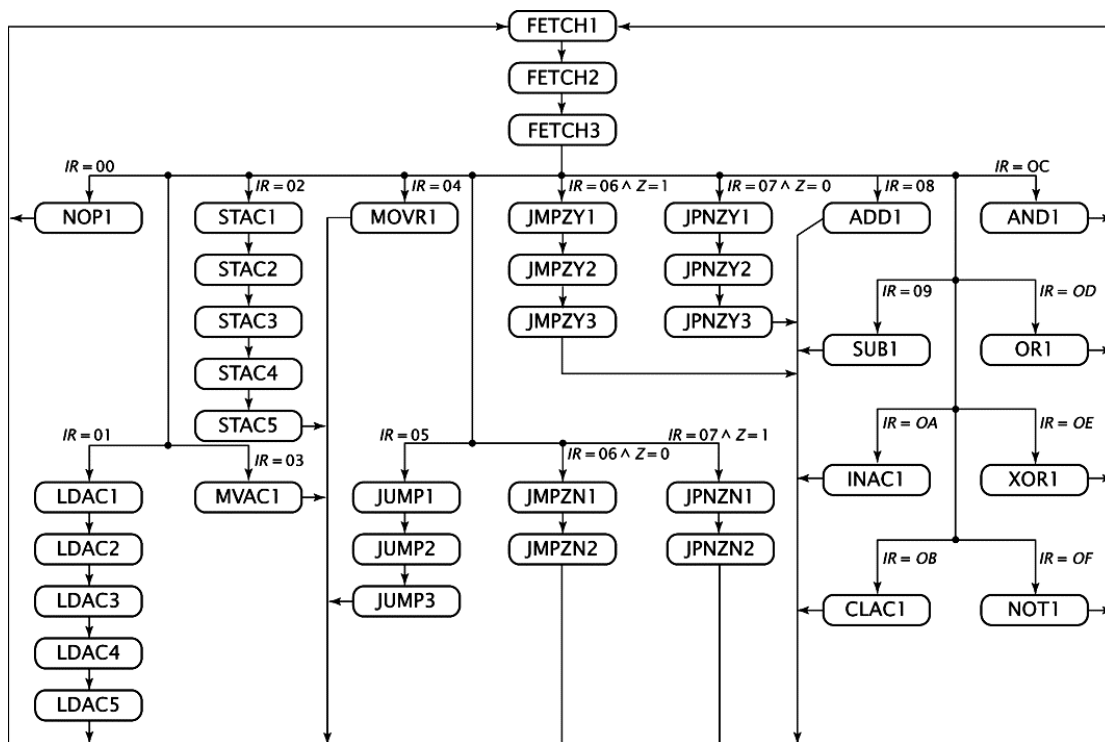
设计并实现简单 cpu 并完成 1-5 的累加

## 实验要求

1. 64 字节存储器，每个存储单元 8 为宽
2. 16 位地址，8 位数据
3. 具有 8 位累加器 AC，寄存器 R，零标志位 Z，16 位地址寄存器 AR，8 位指令寄存器 IR，8 位临时寄存器 TR

指令	指令码	操作
NOP	0000 0000	无
LDAC	0000 0001 $\Gamma$	$AC \leftarrow M[\Gamma]$
STAC	0000 0010 $\Gamma$	$M[\Gamma] \leftarrow AC$
MVAC	0000 0011	$R \leftarrow AC$
MOVR	0000 0100	$AC \leftarrow R$
JUMP	0000 0101 $\Gamma$	GOTO $\Gamma$
JMPZ	0000 0110 $\Gamma$	IF (Z = 1) THEN GOTO $\Gamma$
JPNZ	0000 0111 $\Gamma$	IF (Z = 0) THEN GOTO $\Gamma$

ADD	0000 1000	$AC \leftarrow AC + R$ , IF ( $AC + R = 0$ ) THEN $Z \leftarrow 1$ ELSE $Z \leftarrow 0$
SUB	0000 1001	$AC \leftarrow AC - R$ , IF ( $AC - R = 0$ ) THEN $Z \leftarrow 1$ ELSE $Z \leftarrow 0$
INAC	0000 1010	$AC \leftarrow AC + 1$ , IF ( $AC + 1 = 0$ ) THEN $Z \leftarrow 1$ ELSE $Z \leftarrow 0$
CLAC	0000 1011	$AC \leftarrow 0$ , $Z \leftarrow 1$
AND	0000 1100	$AC \leftarrow AC \wedge R$ , IF ( $AC \wedge R = 0$ ) THEN $Z \leftarrow 1$ ELSE $Z \leftarrow 0$
OR	0000 1101	$AC \leftarrow AC \vee R$ , IF ( $AC \vee R = 0$ ) THEN $Z \leftarrow 1$ ELSE $Z \leftarrow 0$
XOR	0000 1110	$AC \leftarrow AC \oplus R$ , IF ( $AC \oplus R = 0$ ) THEN $Z \leftarrow 1$ ELSE $Z \leftarrow 0$
NOT	0000 1111	$AC \leftarrow AC'$ , IF ( $AC' = 0$ ) THEN $Z \leftarrow 1$ ELSE $Z \leftarrow 0$



(指令执行的框架图)

## 分析指令

指令会经过取指，译码，执行的过程

### 必要指令

所有指令都需要经过相同的取指和译码阶段

FETCH1:  $AR \leftarrow PC$

FETCH2:  $DR \leftarrow M, PC \leftarrow PC + 1$

FETCH3:  $IR \leftarrow DR, AR \leftarrow PC$

### 指令执行

1-5 累加的可能会用到的指令: NOP, ADD, CMP, JMP, MOV, load 指令等。

### 建立数据通路

实现数据通路的两种方式: 硬布线逻辑, 微程序控制

硬布线逻辑

时序逻辑和组合逻辑产生控制信号

微程序控制

使用存储器查找表方式来输出控制信号

数据通路的实现逻辑: 在这里使用硬布线逻辑实现相对的简单的 CPU 数据通路的两种连接方式: 总线、直连

在所有需要传送数据的部件之间创建一条数据通路

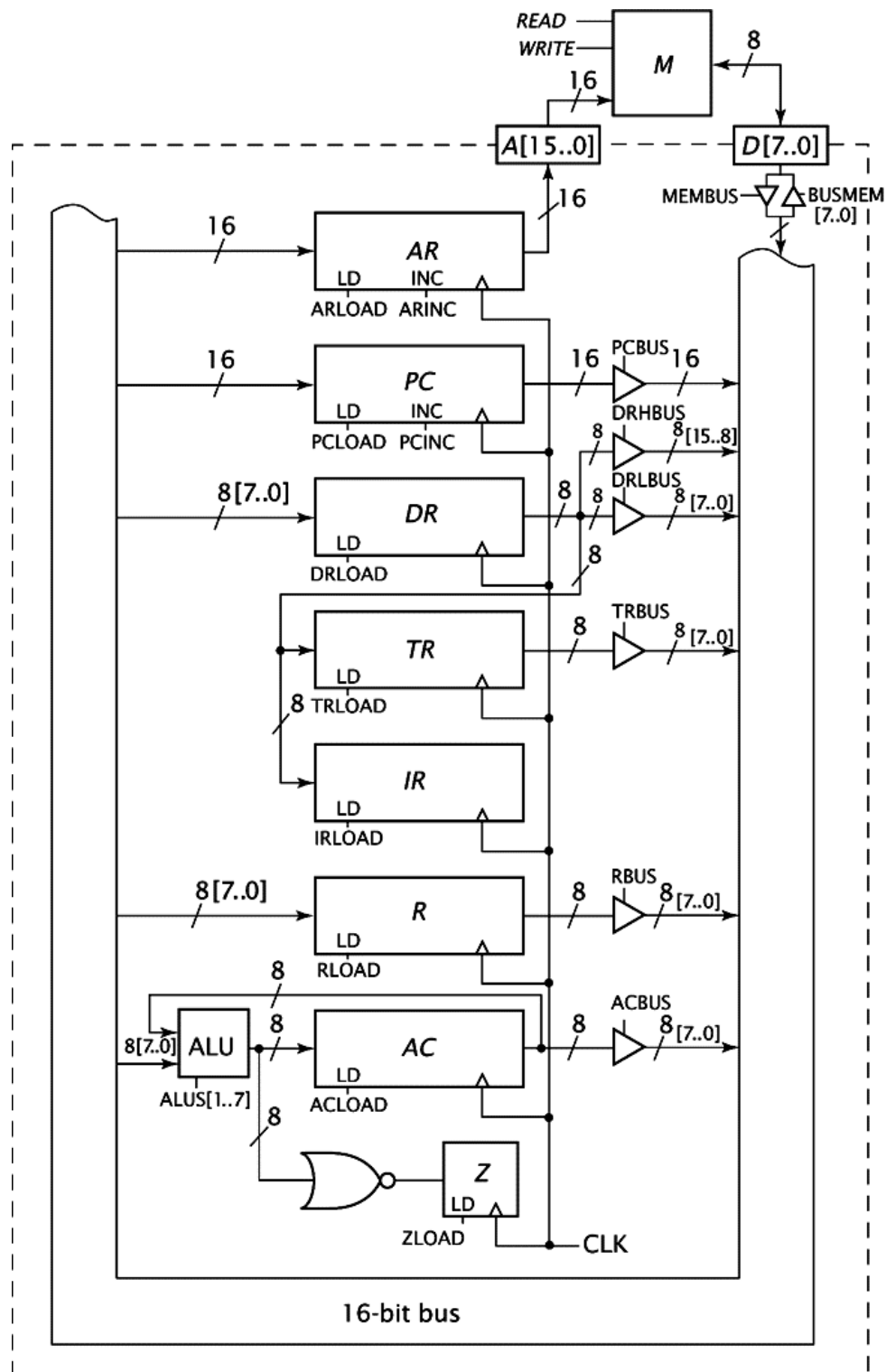
直连方式

在所有需要传送数据的部件之间创建一条数据通路

总线方式

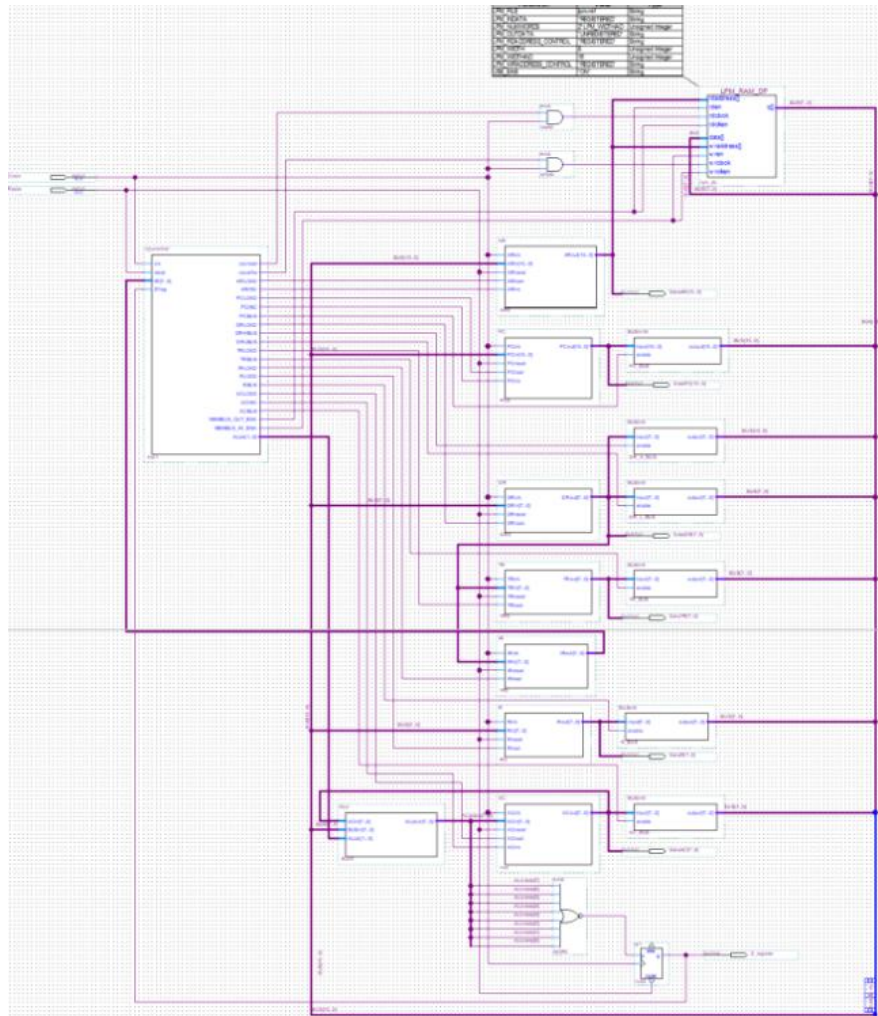
在 CPU 内部创建一条总线，各个部件之间使用总线传递数据 数据通路的连接方式：在这里使用总线的方式连接数据通路。

(实验要求用数据通路



## 设计并验证

最终布线:



## 仿真结果