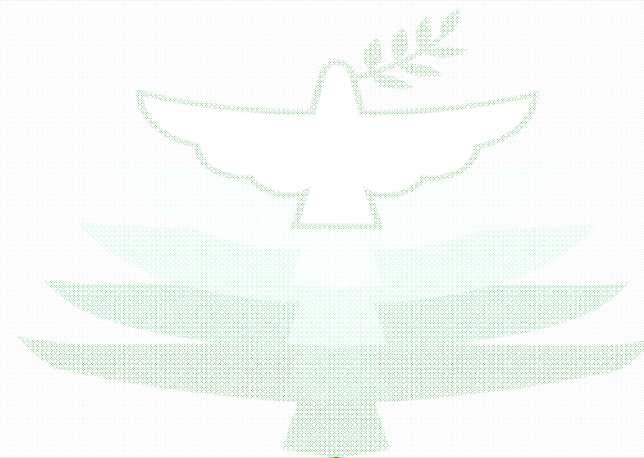


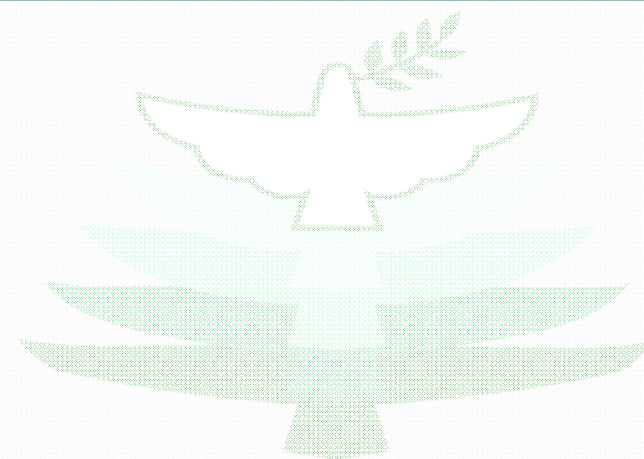
## □ 计算机的工作原理与硬件体系结构

### 加法指令实现原理

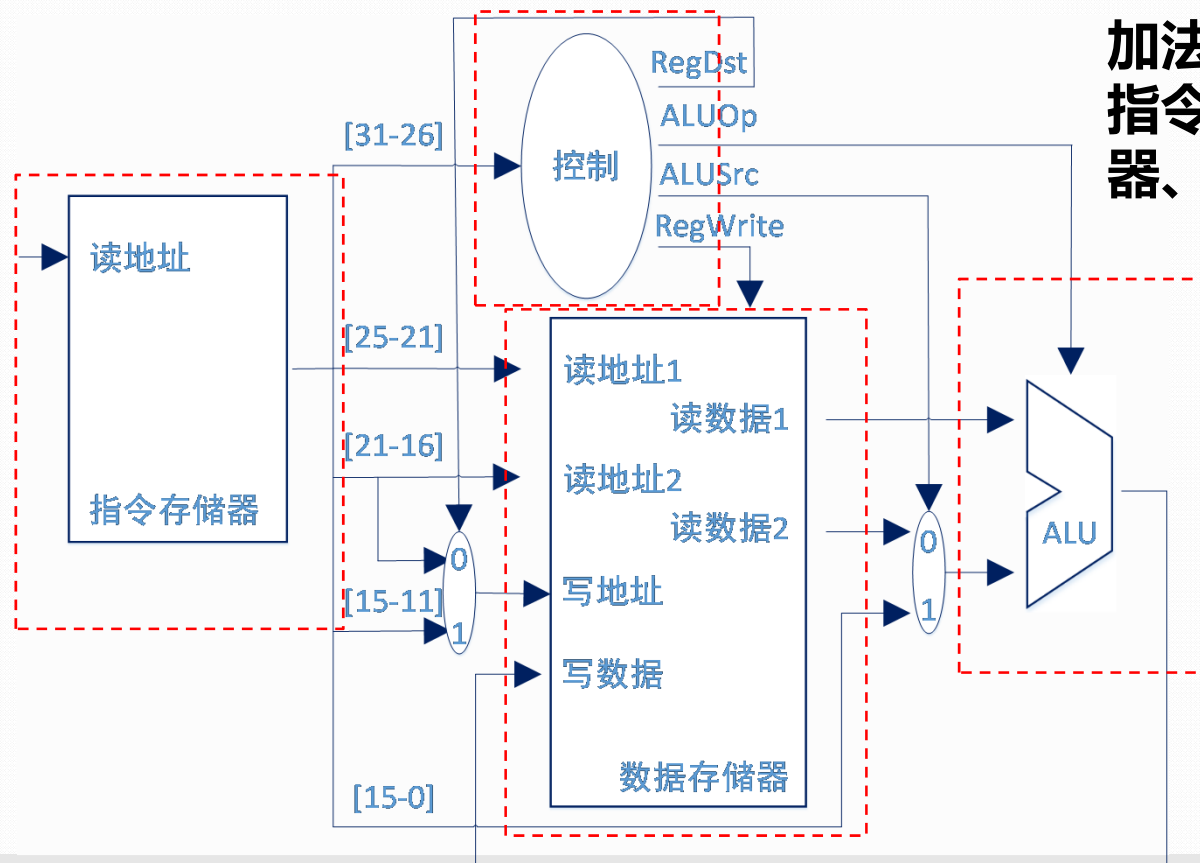


## □ 计算机的工作原理与硬件体系结构

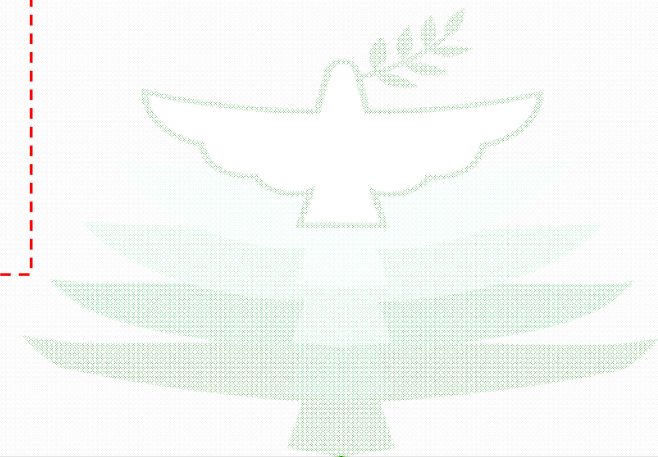
# 用ALU、指令存储器、数据存储器 实现加法



### ■ 3.5 加法数据连路图



加法数据连路图由四部分组成：  
指令存储器、数据存储器、控制  
器、ALU单元

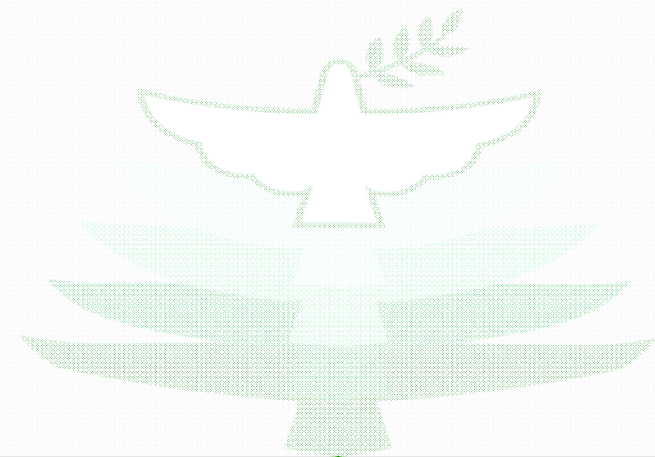
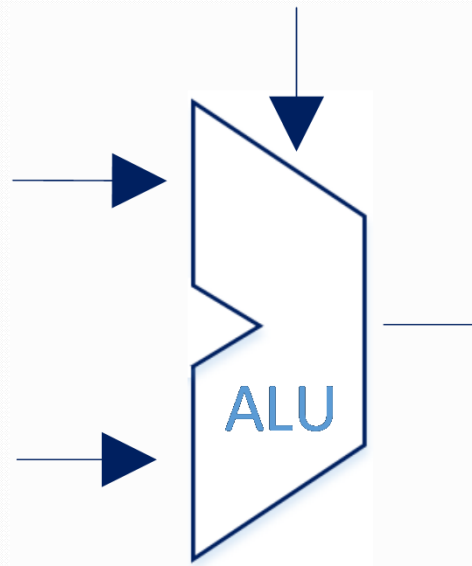


## ■ ALU

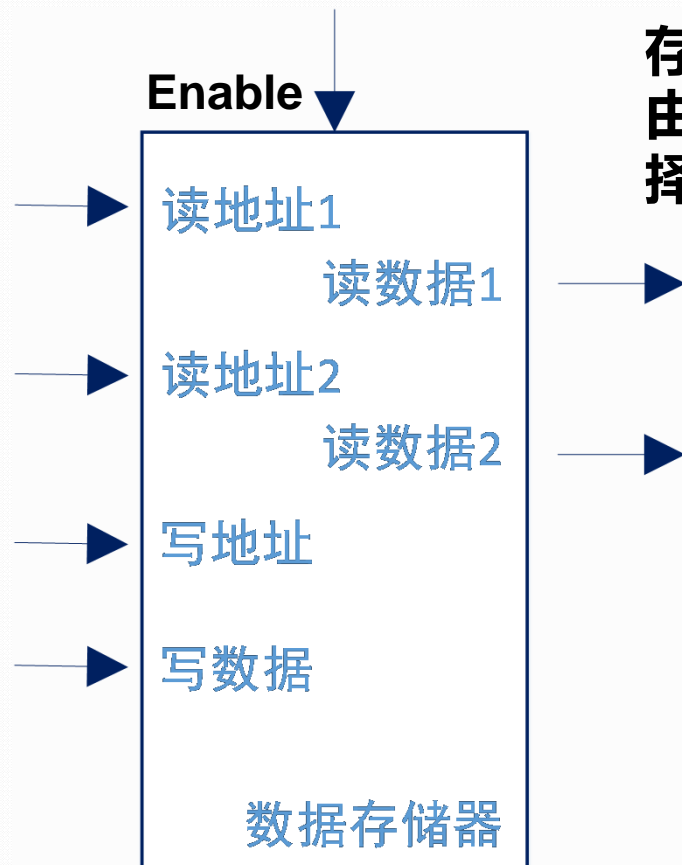
**ALU具有2个输入，每个输入为一排数**

**1个输出，输入数据操作后的结果**

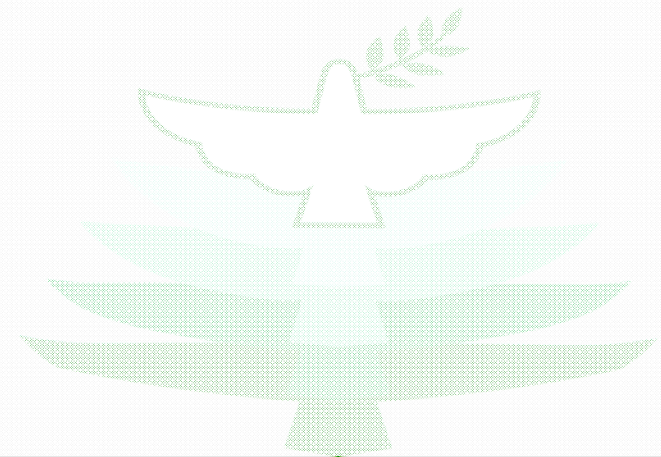
**由控制线决定加减法运算**



## ■ 存储器

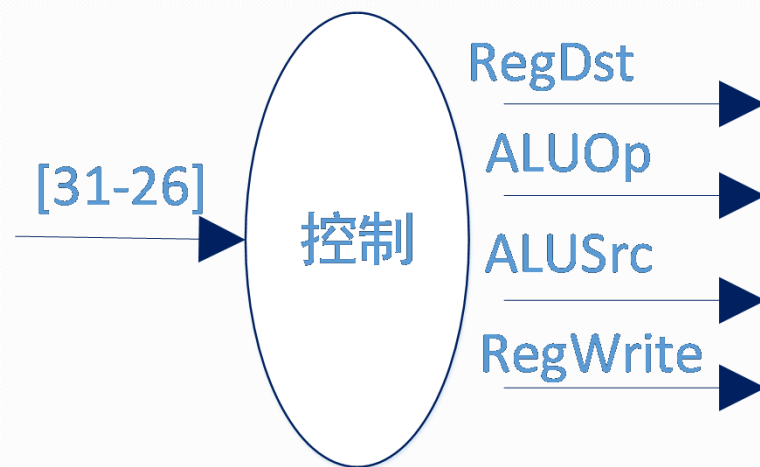


**存储器由触发器控制数据读取，触发器由4个与、非门实现，通过数据链路选择器，存储器即可实现对地址的访问。**

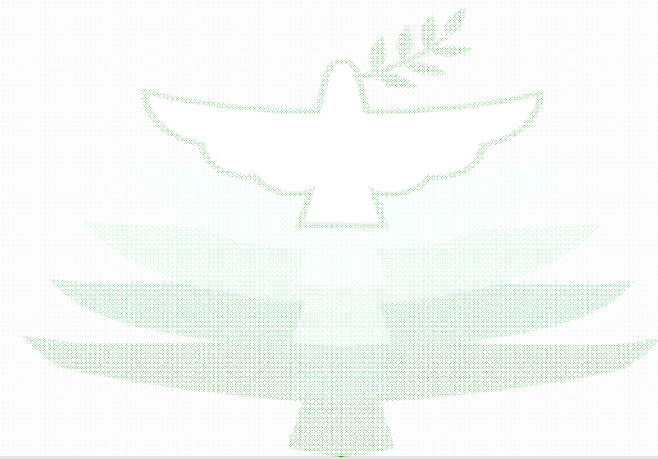


## ■ 控制器

### 控制器示意图

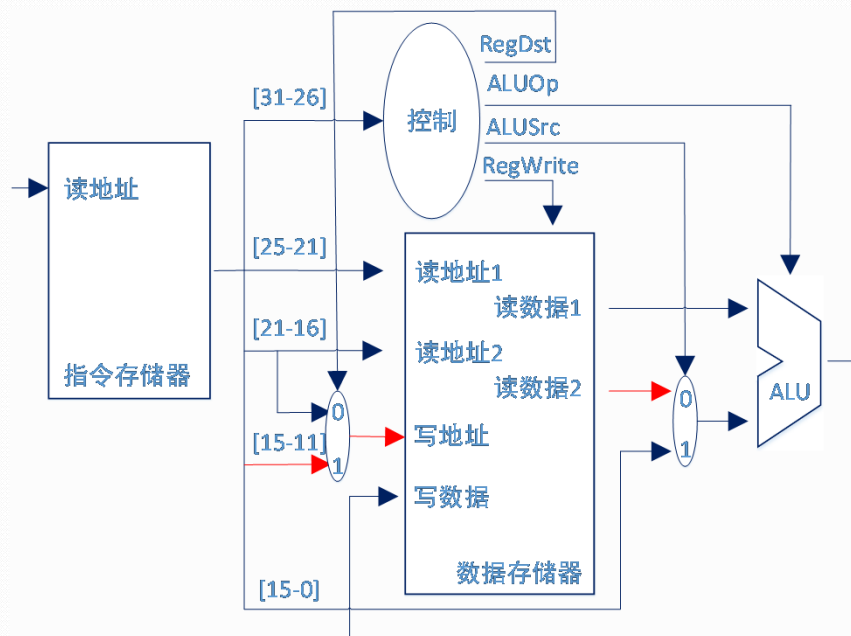


[31-26] : 指令操作的前6位  
4个输出 : RegDst、ALUOp、ALUSrc  
、RegWrite :

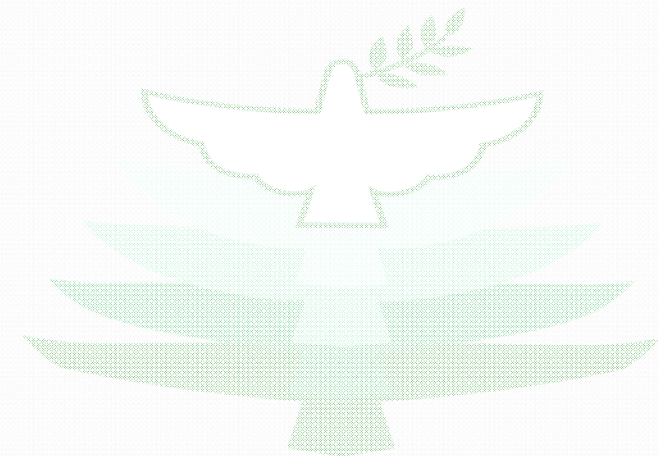


## ■ 例1

例1、 $c = a + b$ : 将地址a(00000)中存的数和地址b(00010)中存的数相加，存在地址c(00100)中



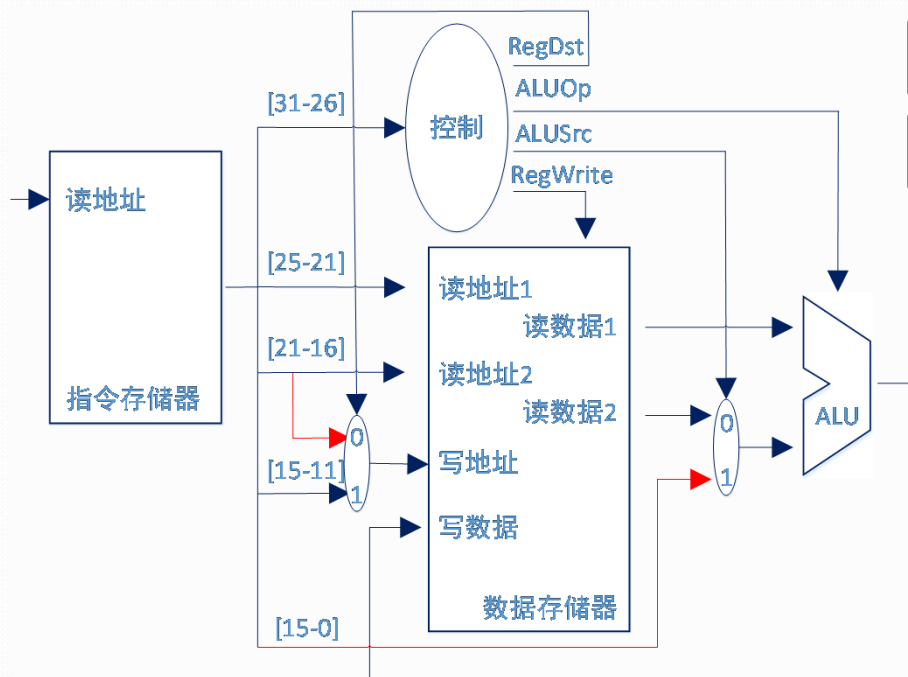
31	26	25	21	20	16	15	11	10	0
操作码		RS		RT		RD		other	
000000		00000		00010		00100		00000100000	



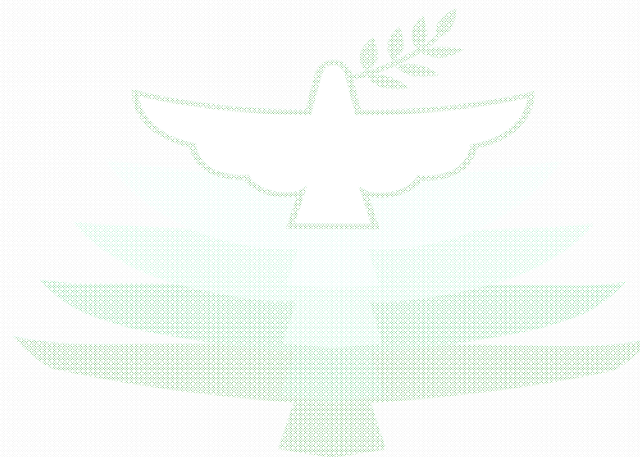


## ■ 例2

例2、 $c = a + FF00H$ : 将地址a(00000)中存的数和FF00H相加，结果存在地址c(00010)中

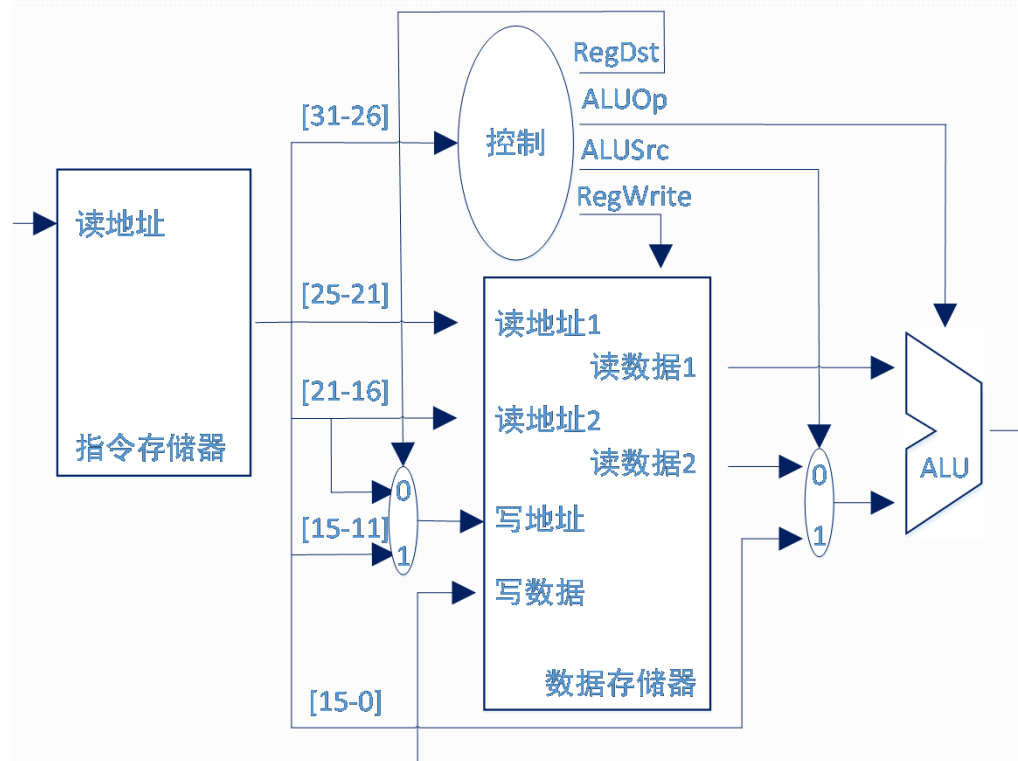


31	26	25	21	20	16	15	0
操作码		RS		RT		Immediate	
001100		00000		00010		1111111100000000	





## ■ 小结



与非门实现ALU  
与非门实现触发器  
与非门实现数据存储器  
与非门实现控制器

