

计算机原理配套教程

计算机设计与实验

邓昌延 著

西南交通大学出版社

前 言

为了配合计算机原理课程的理论教学,加强计算机组成部件及整机实验,作者在多年从事计算机原理教学工作的基础上写作了本书,开发了小型化的计算机原理实验套件。本书和实验套件作为计算机原理课程的教学改革成果,奉献给计算机专业的广大教师和学生。

本书和实验套件的研制目的有两个:一是解决计算机原理教学中的模型机问题;二是培养学生用大规模PLD器件设计复杂数字系统的能力。

为了达到第一个目的,本书以实验计算机的设计、实验和制作为主线,硬件方面要求完成算术逻辑运算器、CPU寄存器组、微程序控制器、主存储器、键盘/显示器(I/O设备)和I/O接口电路的设计与实验;软件方面要求完成指令系统、微程序和监控程序的设计与实验。在此基础上,通过整机调试实验,让学生亲手设计、制作出一台完整的计算机,从而建立起整机概念。

为了达到第二个目的,本书选用不同厂家、不同特色的大规模PLD器件来设计实验计算机的功能部件(运算器、寄存器组、控制器和I/O接口电路)。使学生熟悉几大公司大规模PLD器件的结构特点和开发技术,为他们今后设计复杂的大型数字系统奠定坚实的基础。

全书安排了8个课程设计和5个实验,有助于提高学生的综合设计能力和实验、制作技术。为了降低设计和实验的难度,减轻学生负担,在保证实验计算机结构完整的前提下,对其功能和结构进行了简化,以便让学生不花太多的时间和精力就能设计、制作和调试成功一台简单而完整的计算机。

根据我们的教学实践经验,建议在进行计算机原理教学的同时,开设32学时的“计算机原理实验”课。使用计算机原理实验套件,学生投入适当的课外学时,就能在一学期内完成运算器、寄存器组、微程序控制器和监控程序的设计和实验。整机实验和基于RISC的CPU优化设计,可以安排在假期或实习期间进行,放手让学生自己去完成。

本书的配套教材包括:①计算机原理课程设计指导书;②计算机原理实验指导书;③《计算机设计与实验》教师参考书。

由于作者水平所限,书中难免存在错误和缺点,敬请读者批评指正。

与本书套配的计算机原理实验套件包括:①通用PLD实验板,可进行各种数字电路实验和计算机组成部件的独立实验;②CPU板,键盘/显示器板,见第十章图10.1和图10.2;③调试板,用于微程序和监控程序的装入和调试;④PLD编程电缆和微型实验电源;⑤实验器盒(体积:175mm×120mm×35mm)。

实验器套件成本在800元左右,每套售价在1000元以内。

电话:(028)7600743(O)、7603632(H) 传真:(028)7600743

E-mail: cydeng @ center2. swjtu. edu. cn 网址: <http://www.swjtu.edu.cn>

作 者

1999年8月

目 录

第一章 实验计算机的结构	1
1.1 实验计算机的性能指标	1
1.2 实验计算机的结构	1
1.3 中央处理器CPU的内部结构	2
第二章 运算器设计与实验	4
2.1 运算器的功能	4
2.2 运算器的设计	4
设计指导一：推导逻辑函数	5
设计指导二：选择合适的PLD器件	6
设计资料一：ATV2500	8
设计资料二：MACH211	8
设计资料三：pLSI1016/ispLSI1016	14
设计指导三：PLD设计文件的编制	20
课程设计一 8位算术逻辑运算器的设计	23
实验一 8位算术逻辑运算器实验	24
问题与思考	25
第三章 寄存器组设计与实验	26
3.1 寄存器的功能和操作方式	26
3.2 寄存器组的设计	27
设计指导一：推导逻辑函数	27
设计指导二：选择合适的PLD器件	29
设计资料一：EPM7096	30
设计资料二：pLSI/ispLSI1032	34
设计指导三：PLD设计文件的编制	35
问题与思考	36
3.3 运算器和寄存器组的集成设计	37
课程设计二 寄存器组设计	39

实验二 寄存器组实验	39
第四章 指令系统设计	40
4.1 指令格式	40
4.2 寻址方式	40
4.3 指令系统设计	40
4.4 指令编码	43
课程设计三 指令编码	45
问题与思考	45
思路引导	46
第五章 微程序设计	48
5.1 CPU 控制信号	48
5.2 指令流程分析	49
问题与思考	52
5.3 微指令编码	53
5.4 微指令地址的形成	54
问题与思考	55
思路引导	55
5.5 微程序设计	56
设计指导	56
课程设计四 微程序设计	56
第六章 控制器设计与实验	59
6.1 微程序控制器的结构	59
6.2 指令译码与非法操作码处理	60
6.3 微地址形成电路	60
问题与思考	65
6.4 中断控制电路	65
6.5 微指令译码	67
6.6 时序分配与指令周期波形图(CPU 定时图)	67
6.7 时钟电路与时序分配电路	78
6.8 微程序控制器设计	81
设计指导一 推导逻辑函数	81
问题与思考	86
设计指导二 PLD 设计文件的编制	86

课程设计五 微程序控制器设计	88
实验三 微程序控制器实验	88
6.9 CPU 的集成设计	92
设计方法总结	94
第七章 主存储器设计	96
7.1 存储器字长	96
7.2 存储容量与地址分配	96
7.3 存储器与CPU 的时序配合	98
第八章 键盘显示器及接口电路设计	101
8.1 键盘/显示器的结构	101
8.2 显示器驱动电路	103
8.3 键盘/显示器扫描工作原理	104
8.4 断点及单步执行控制电路	105
8.5 时钟及复位电路	106
8.6 中断向量电路	107
课程设计六 键盘/显示器印刷电路板设计	107
设计资料一: 键盘/显示器印刷电路板几何尺寸	108
设计资料二: TPIC6B273 功率锁存器	108
第九章 监控程序设计	109
9.1 监控程序的结构	109
9.2 监控程序的存储组织	109
9.3 监控程序功能分析及流程图设计	110
9.4 监控程序设计	118
问题与思考	133
课程设计七 监控程序设计	133
第十章 计算机整机调试实验	134
10.1 硬件调试	134
10.2 软件硬件联合调试	136
实验四 整机实验	137
科学实验与素质修养	138
第十一章 基于RISC 的CPU 优化设计	140
11.1 RISC 结构的基本特点	140

11.2 优化后的CPU 内部结构	141
11.3 寄存器组的优化设计.....	142
11.4 内部推拉式堆栈及其加速原理.....	145
11.5 指令系统设计.....	146
11.6 组合逻辑控制器设计.....	149
11.7 优化设计的效果分析.....	157
课程设计八 RISC 结构CPU 的单片集成设计	157
实验五 RISC CPU 的整机实验.....	158
附录 CPU 初始引导及键盘/显示器扫描程序	159
参考文献.....	163

第一章 实验计算机的结构

学习计算机原理,亲手设计和制作一台简单而完整的实验计算机,对于掌握计算机的基本结构和工作原理是十分重要的。

实验计算机主要用于教学实验目的,因此不要求有较强的功能和较高的性能,而应以结构完整、功能简单为准。

结构完整,硬件方面指的是计算机的五大功能部件(运算器、控制器、存储器、输入设备和输出设备)必须齐备;软件方面则要求指令系统、微程序和系统监控程序不可缺少。这样才能使实验者对计算机系统有全面的认识,从而建立起整机概念。

功能简单,指的是运算器和指令系统简单,字长较短(比如采用8位字长)。这样可以简化控制器,使整机结构简明、脉络清晰,便于突出主要矛盾,掌握基本原理。

设计工作的第一步,就是在上述原则下确定实验计算机的性能指标和基本结构。

1.1 实验计算机的性能指标

1. 字长:8位。

8位字长决定了算术逻辑运算器(ALU)、通用寄存器、存储器和数据总线的位宽均是8位的。

2. 主存储器容量:64K字节。

主存容量就是CPU直接寻址的内存空间。64K字节的寻址范围决定了地址总线为16位。程序计数器PC和堆栈指针寄存器SP也是16位的。但在实际配置存储器时,可以远小于64K字节。

3. 具有堆栈结构和中断功能。

4. I/O结构:存储器映像I/O,I/O设备与存储器统一编址。

5. I/O设备:输入设备——小键盘,输出设备——数码显示器。

6. 指令系统:24~32条指令,包括数据传送指令、算术逻辑运算指令和CPU控制指令。

7. 系统软件:监控程序,用于管理存储器和I/O设备,实现人机对话功能。

8. 运算速度:实验计算机对运算速度要求不高。根据目前能买到的廉价PLD器件,容易实现1MHz的时钟频率,即时钟周期1 μ s,机器周期4 μ s,指令周期8~24 μ s。每秒可运行4万~12万条指令。

1.2 实验计算机的结构

通用计算机目前仍然采用Von. Neuman结构,它由运算器、控制器、存储器、输入设备和输出设备五大基本部件组成,称为存储程序计算机,如图1.1所示。这是以存储器为中心的结构,图中的控制器和运算器早期被安装在主机柜的中心部位(现在则被集成在同一块集成电路芯片上),称为中央处理器(CPU)。

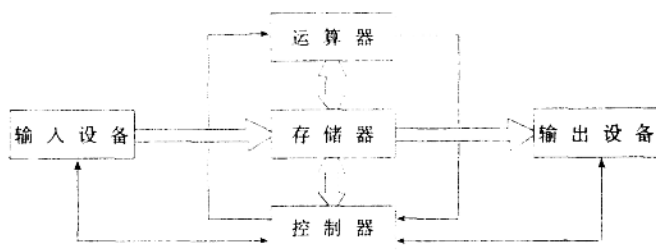


图1.1 数字计算机的一般结构

图1.2 示出实验计算机的结构,它由中央处理器、存储器、I/O 设备和总线构成。存储器包括只读存储器ROM 和读写存储器RAM。ROM 用于存放监控程序, RAM 用于存放用户程序和数据。I/O 设备包括输入设备键盘和输出设备数码显示器。所有设备都挂接在同一条总线上,称为单总线结构。

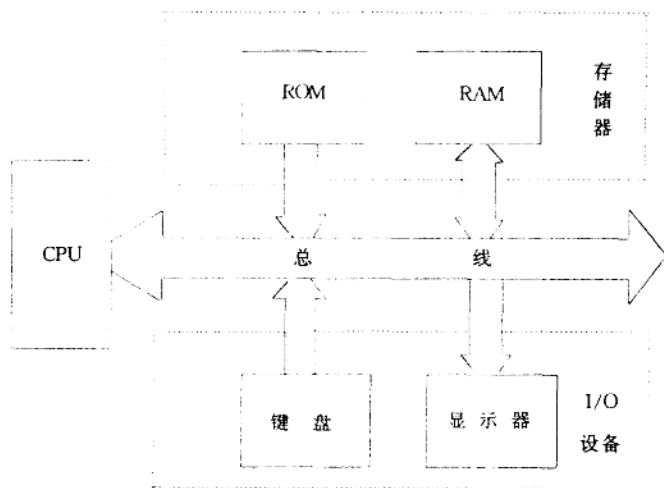


图1.2 实验计算机的结构

1.3 中央处理器CPU 的内部结构

CPU 内部结构的复杂程度取决于计算机的性能指标,图1.3 示出满足实验计算机性能指标的CPU 内部结构。CPU 内除运算器和控制器外,还包括寄存器组和总线接口部件。寄存器组包括8 位通用寄存器(R_0 、 R_1 、 R_2 、 R_3)和16 位专用寄存器(程序计数器PC 和堆栈指针寄存器SP)。总线接口部件包括地址锁存器、数据和控制信号缓冲器,它们完成CPU 内外信号转接和总线驱动工作。

图1.3 中的各种控制信号及其作用,将在后续各章中逐步介绍。

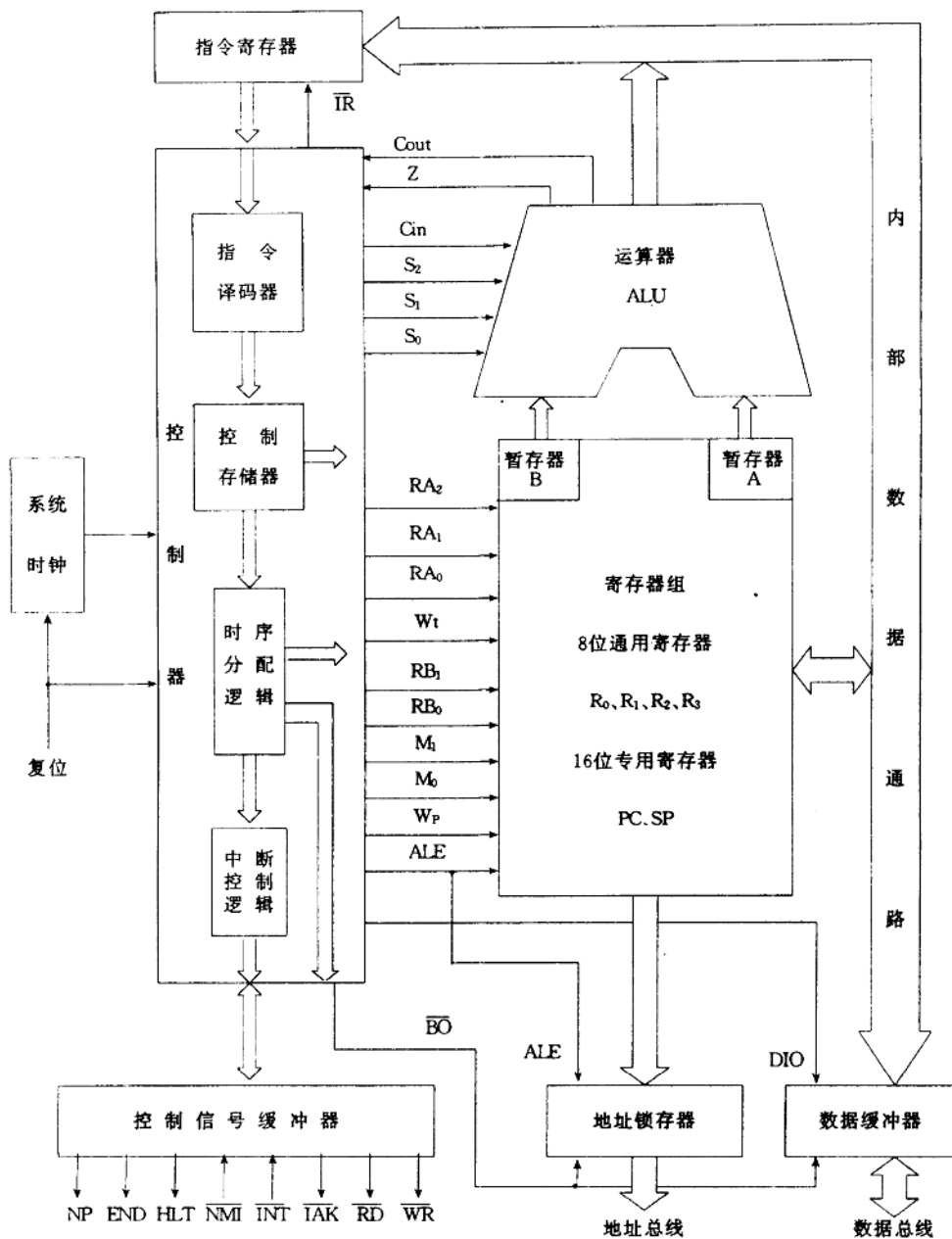


图1.3 实验计算机CPU内部结构

第二章 运算器设计与实验

运算器是CPU执行算术和逻辑运算的部件。运算器具有哪些运算功能,就决定了CPU能执行哪些运算指令。为了简化运算电路,本运算器只设计了基本的逻辑运算、算术运算和移位操作功能,没有设计乘除法运算功能。

2.1 运算器的功能

8位算术逻辑运算器,并行运算,超前进位(4位分组,组内超前进位,组间串行进位)。图2.1是运算器的示意图。

在单独设计运算器时,使用的控制信号与图1.3中的信号不同。运算器的两个输入操作数(A和B)由信号 S_A 和 S_B 写入暂存器A和暂存器B。当 S_A 产生负脉冲时,输入数据写入暂存器A;当 S_B 产生负脉冲时,输入数据写入暂存器B。输出F通过三态缓冲器与数据总线连接。运算器中还包括一个标志寄存器(或者称为状态字寄存器),用于存放运算结果的状态标志,通常作为某些转移指令的执行条件。这里只用了最常用的两个状态标志:一是进位标志 $Cout$,有进位 $Cout=1$,无进位 $Cout=0$;二是零标志Z,运算结果为零,则 $Z=1$,否则 $Z=0$ 。

运算器的功能如表2.1所示。 S_2, S_1, S_0 为运算功能选择信号, Cin 为进位输入,A加B加 Cin 是带进位位的算术加法运算。从表2.1可以看到,当 $S_2 S_1 S_0=111$ 时,运算器不执行任何操作,不输出 F ,所以可用 $S_2 S_1 S_0$ 作 F 的输出三态门控制信号。

本运算器与专用ALU电路74LS181、74LS381和74LS681相比,功能要简单些,以便简化运算电路,降低对PLD器件的资源需求。

表2.1 运算器功能表

S_2	S_1	S_0	输出F
0	0	0	0 (清零)
0	0	1	AB (与)
0	1	0	$A+B$ (或)
0	1	1	$A \oplus B$ (异或)
1	0	0	A加B加 Cin (带进位加)
1	0	1	A带进位位左移
1	1	0	A带进位位右移
1	1	1	无操作

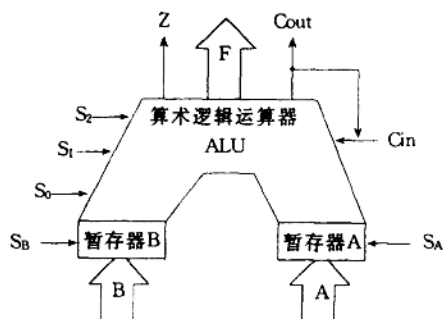


图2.1 运算器示意图

2.2 运算器的设计

用可编程逻辑器件(PLD)设计和实现运算器,一般包括以下步骤:推导逻辑函数,选择合适的PLD器件,编制PLD设计文件,上机编译、仿真,然后写入PLD器件。

设计指导一 推导逻辑函数

根据图2.1和表2.1所示的功能,需要推导下列逻辑函数:

$$F_i = f(A_i, B_i, C_{in}, S_2, S_1, S_0), \quad i = 0, 1, 2, \dots, 7,$$

$$C_i = g(A_i, B_i, C_{in}, S_2, S_1, S_0),$$

$$Z = h(F_i).$$

1. 对于算术加法运算A加B加Cin,根据1位全加器的真值表,可以得到下列计算公式:

$$F_i = A_i \bar{B}_i \bar{C}_{i-1} + \bar{A}_i B_i \bar{C}_{i-1} + \bar{A}_i \bar{B}_i C_{i-1} + A_i B_i C_{i-1},$$

$$= P_i \bar{C}_{i-1} + P_i C_{i-1},$$

$$C_i = A_i B_i + (A_i \oplus B_i) C_{i-1},$$

$$= A_i B_i + P_i C_{i-1}.$$

其中 $P_i = A_i \oplus B_i$.

表2.2 1位全加器真值表

输入			输出	
A _i	B _i	C _{i-1}	F _i	C _i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

2. 对于移位功能,一般是在运算器的输出端加移位寄存器来实现。本设计使用PLD器件,可将移位功能纳入运算器内,不必再加移位寄存器。实现移位功能的原理很简单,对于输入数据A,不难写出移位操作的表达式。

当 $S_2 S_1 S_0 = 101$ 时,执行带进位左移操作:

$$[C_7] \leftarrow [A_7 \leftarrow A_0] \leftarrow 0$$

$A_6 \sim A_0$ 依次左移一位,最高位 A_7 移入进位位 C_7 ,末位 A_0 补0。因此有:

$$F_i = A_{i-1}, \quad i = 1, 2, \dots, 7, \quad F_0 = 0, \quad C_7 = A_7.$$

当 $S_2 S_1 S_0 = 110$ 时,执行带进位右移操作:

$$[C_7] \rightarrow [A_7 \rightarrow A_0] \rightarrow \text{丢掉}$$

$A_7 \sim A_1$ 依次右移一位, C_7 移入 A_7 位,末位 A_0 原来的数丢掉。故有:

$$F_i = A_{i+1}, \quad i = 0, 1, \dots, 6, \quad F_7 = C_7.$$

3. 将运算功能选择信号(变量) $S_2 S_1 S_0$ 加入逻辑函数中,写出 $C_7 \sim C_0$ 、 $F_7 \sim F_0$ 和 Z 的表达式。

例1 进位位表达式:因为采用4位分组,组内超前进位、组间串行进位,所以 C_0 、 C_1 、 C_2 、 C_3 以Cin为进位输入; C_4 、 C_5 、 C_6 、 C_7 以 C_3 为进位输入。

$$\therefore C_1 = S_2 \bar{S}_1 \bar{S}_0 (A_1 B_1 + P_1 C_{in}),$$

$$\therefore C_0 = S_2 \bar{S}_1 \bar{S}_0 (A_0 B_0 + P_0 C_{in}),$$

$$C_1 = S_2 \bar{S}_1 \bar{S}_0 (A_1 B_1 + P_1 A_0 B_0 + P_1 P_0 C_{in}),$$

.....

$$C_7 = S_2 \bar{S}_1 \bar{S}_0 (A_7 B_7 + P_7 A_6 B_6 + P_7 P_6 A_5 B_5 + P_7 P_6 P_5 A_4 B_4 + P_7 P_6 P_5 P_4 C_3) \\ + S_2 \bar{S}_1 S_0 A_7.$$

注意:当 $S_2 S_1 S_0 = 101$ 时,执行带进位左移操作, A_7 移入进位位,故有 $S_2 \bar{S}_1 S_0 A_7$ 这一项。

例2 和表达式:根据表2.1,“清零”功能的表达式为: $F_i = \bar{S}_2 \bar{S}_1 \bar{S}_0 \cdot 0 = 0$,故可省去这一项。“无操作”实际上不执行任何运算,不输出 F_i 、 Z 和 C_{out} 。因此, $S_2 S_1 S_0 (=111)$ 这一项也可

省去,所以 F_i 的逻辑函数只有6项。

$$\therefore F_i = \bar{S}_2 \bar{S}_1 S_0 A_i B_i + \bar{S}_2 S_1 \bar{S}_0 (A_i + B_i) + \bar{S}_2 S_1 S_0 P_i + S_2 \bar{S}_1 \bar{S}_0 (P_i \bar{C}_{i-1} + \bar{P}_i C_{i-1}) \\ + S_2 \bar{S}_1 S_0 A_{i-1} (i \neq 0) + S_2 S_1 \bar{S}_0 A_{i+1} (i \neq 7).$$

$$\therefore F_0 = \bar{S}_2 \bar{S}_1 S_0 A_0 B_0 + \bar{S}_2 S_1 \bar{S}_0 (A_0 + B_0) + \bar{S}_2 S_1 S_0 P_0 + S_2 \bar{S}_1 \bar{S}_0 (P_0 \bar{C}_{in} + \bar{P}_0 C_{in}) \\ + S_2 S_1 \bar{S}_0 A_1.$$

.....

$$F_7 = \bar{S}_2 \bar{S}_1 S_0 A_7 B_7 + \bar{S}_2 S_1 \bar{S}_0 (A_7 + B_7) + \bar{S}_2 S_1 S_0 P_7 + S_2 \bar{S}_1 \bar{S}_0 (P_7 \bar{C}_6 + \bar{P}_7 C_6) \\ + S_2 \bar{S}_1 S_0 A_6 + S_2 S_1 \bar{S}_0 C_7.$$

说明: ① 当 $S_2 S_1 S_0 = 101$ 时,执行带进位左移操作, F_0 位补0,所以 $F_0 = S_2 \bar{S}_1 S_0 \cdot 0 = 0$,故省去这一项。

② 当 $S_2 S_1 S_0 = 110$ 时,执行带进位右移操作,将 C_7 移入 F_7 ,所以 F_7 的表达式中有 $S_2 S_1 \bar{S}_0 C_7$ 这一项。

例3 零标志表达式: $Z = \bar{F}_7 + \bar{F}_6 + \cdots + \bar{F}_0 = \bar{F}_7 \bar{F}_6 \cdots \bar{F}_0$.

4. 标志位刷新与保持: 运算器每执行一次运算都要产生零标志 Z ,在输出 F 时也要输出 Z 标志,因此可用 $\bar{S}_2 \bar{S}_1 S_0$ 信号刷新零标志 Z 。只有在执行加法运算和左移操作时才会产生新的进位标志,执行其它操作时进位标志保持不变,所以可用 $S_2 \bar{S}_1$ 信号输出 C_7 作为进位标志 C_{out} 。

进位标志 C_{out} 和零标志 Z 应保存在锁存器中,便于条件转移指令使用。

设计指导二 选择合适的PLD器件

通常根据以下条件来选择合适的PLD器件:

① 输入/输出引脚是否满足要求,② 逻辑宏单元的乘积项数是否足够,③ 器件价格,④ 资源利用率(一般应达到60%)。其中前两个条件是最基本的,后两个条件在于降低成本。

1. 计算I/O引脚

a. 输入信号 $A_7 \sim A_0, B_7 \sim B_0$, 16条引脚,若分时输入锁存,仅需8条引脚。 S_2, S_1, S_0, C_{in} , S_A 和 S_B ,6条引脚。

b. 输出信号: $F_7 \sim F_0, C_{out}, Z$, 10条引脚。

c. 反馈信号: $P_7 \sim P_0, C_7 \sim C_0, F_7 \sim F_0$, 24个。这些信号经过PLD逻辑宏单元反馈到与门阵列。

上述计算表明:这种运算器要选用有14~22个输入引脚、10个输出引脚的PLD器件,而且要有24条反馈路径。

2. 计算函数(F_i, C_i, P_i, Z)的乘积项数目。

F_i 包括8个乘积项; C_i 中乘积项最多的是 $C_7(C_{out})$,有6项;

P_i 只含2个乘积项; Z 只有1个乘积项($F_7 F_6 \cdots F_0$)。

由上可知,实现 F_i 需要8个宏单元(含8个乘积项的或阵列),实现 C_i 和 Z 需要9个宏单元(含2~6个乘积项的或阵列), P_i 需要8个宏单元(每个宏单元只用2个乘积项)。

3. 选择PLD器件

用PLD器件设计8位ALU,可供选用的器件很多。在器件的资源满足要求的前提下,尽量选用低功耗、低价格器件。由于对运算速度要求不高,可选用低速器件,价格会更低些。

根据上述分析,可选用AMD公司的MACH211器件设计8位运算器。

4. PLD器件的资源分配

MACH211器件内部资源丰富(见设计资料二),可以有多种资源分配方案,下面仅讨论一种方案。

MACH211的4个PAL块通过开关矩阵连接在一起,各PAL块具有相同的结构,在功能分配时可以相对独立。但必须注意,每个PAL块只能从开关矩阵中获得26个输入变量,这使设计受到一定限制,在资源分配时必须考虑这一点。

a. 引脚分配

MACH211有32个I/O引脚、4个专用输入引脚和2个时钟输入引脚。

用时钟输入引脚CLK₀和CLK₁作S_A和S_B的输入引脚。专用输入引脚I₀、I₁、I₂和I₃用作S₀、S₁、S₂和Cin的输入引脚。

用8个I/O引脚作8位数据(A和B的)输入引脚,用8个I/O引脚作运算器的输出(F的)引脚,再用2个I/O引脚分别作进位标志Cout(C₇)和零标志Z的输出引脚。还余下14个I/O引脚不用。

b. PAL块和宏单元的分配

MACH211有4个PAL块A、B、C、D。

用PAL块A的8个隐埋宏单元锁存输入数据A,8个输出宏单元形成函数P₁,并反馈到开关矩阵中。共使用16个宏单元、16个乘积项、16个输入。

用PAL块B的8个隐埋宏单元锁存输入数据B,用其中一个输出宏单元形成零标志Z,并输出到I/O引脚上。共使用9个宏单元、9个乘积项、16个输入。

用PAL块C实现C₀~C₃和F₀~F₃。需要的输入变量为A₀~A₃、B₀~B₃、P₀~P₃、S₀、S₁、S₂、Cin和反馈输入C₀、C₁、C₂,共19个输入,小于26个,满足PAL块输入变量的限制。共使用8个宏单元、46个乘积项、19个输入。

同理可用PAL块D实现C₄~C₇和F₄~F₇。

在C块和D块中,要将F₀~F₃和F₄~F₇通过I/O单元输出到I/O引脚上。这些I/O单元的输出三态门由乘积项S₃S₀控制。

上述的资源分配表明MACH211的资源满足8位运算器的设计要求,并略有剩余。

5. 计算资源利用率

MACH211的4个PAL块共有64个宏单元(64个触发器)、256个一般乘积项、16个专用(输出允许、异步预置和复位)乘积项和104个与阵列输入。

实际使用的资源为:宏单元41个、乘积项118个、与阵列输入70个。资源利用率如表2.3所示。

表2.3 MACH211的资源利用率

资源	MACH211提供数	使用数	利用率
专用输入引脚	6	6	100%
I/O引脚	32	18	56%
宏单元	64	41	62.5%
乘积项	256	118	46%
与阵列输入	104	70	67%

从表2.3可以看到,5项主要资源的平均利用率达到66%。到底要达到多高的资源利用率才表示器件选择合适呢?一般的看法是:对于定型的产品(功能扩展要求不高),PLD的资源利用率越高越好。对于研制中的产品(包括我们作实验用的器件),资源利用率达到60%就可以了,以便留有功能扩展的余地。

设计资料一 ATV2500

ATMEL 公司的ATV2500 的引脚图如图2.2 所示。它有14 个专用输入引脚和24 个I/O 引脚。每个I/O 引脚对应一个逻辑宏单元,每个宏单元含两个触发器,三条反馈路径。每个宏单元有三个或门,各有4 个乘积项,共12 个乘积项。另有4 个乘积项用作两个触发器的清零和时钟信号。还有一个乘积项用作输出三态门控制。用于产生输出函数的乘积项总数为288 个(12×24),控制用乘积项120 个(5×24)。

图2.3 示出ATV2500 的宏单元结构。每个宏单元有三条反馈路径:当宏单元的I/O 引脚用作输入时,输入信号可直接馈入与阵列。如果用作输出引脚,输出信号在送到I/O 引脚的同时也反馈到与阵列。宏单元中两个触发器的输出 Q_1 和 Q_2 各有一条路径反馈到与阵列。

当宏单元设计为时序逻辑时,另一个触发器可用作埋入式触发器;如果宏单元设计为组合逻辑,则可留下两个埋入式触发器。从图2.3 中不难看出,宏单元的输出模式十分灵活,可以产生4~12 个乘积项的逻辑函数。这给设计带来了很大方便。

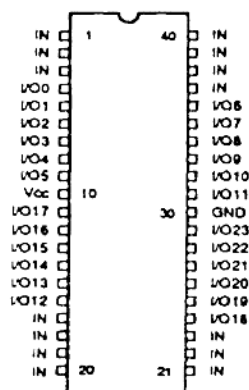


图2.2 ATV2500 引脚图

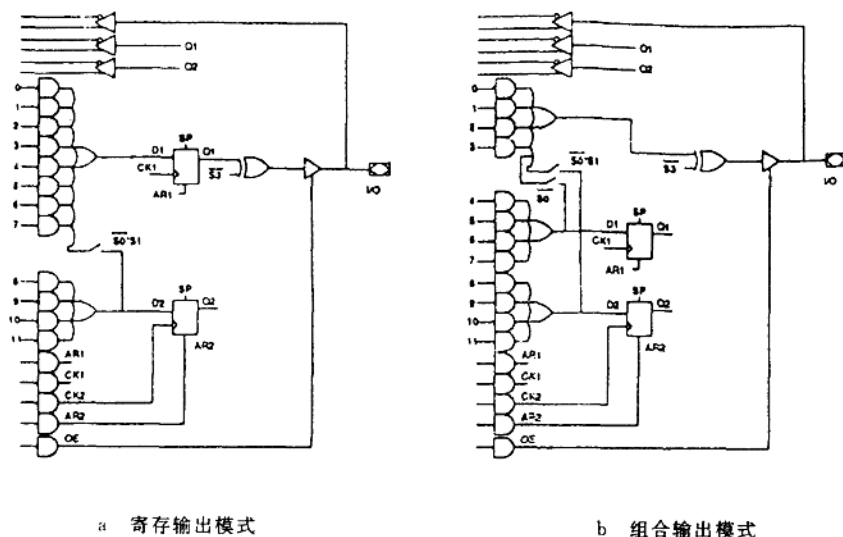


图2.3 ATV2500 的宏单元结构

设计资料二 MACH211

AMD 公司的MACH211 为44 脚PLCC 或CQFP 封装器件,引脚如图2.4 所示。它有32 个I/O 引脚,4 个专用输入引脚和2 个时钟输入引脚(可用作专用输入)。MACH211 的4 个PAL 块通过内部开关矩阵连在一起,如图2.5 所示。PAL 块的逻辑图示于图2.6。

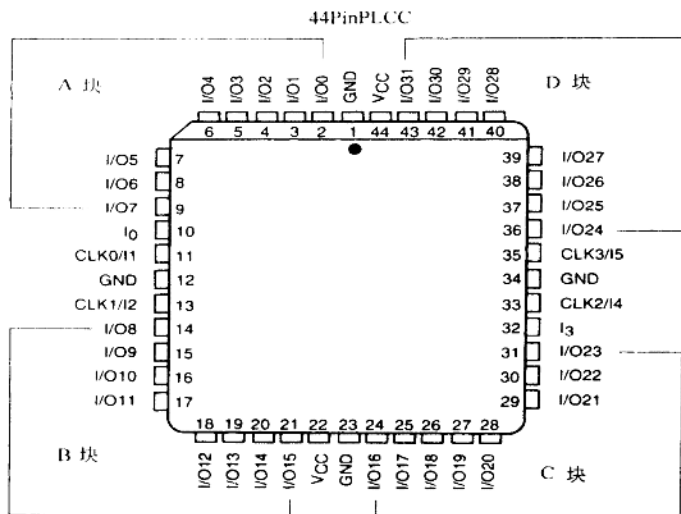


图2.4 MACH211 引脚图

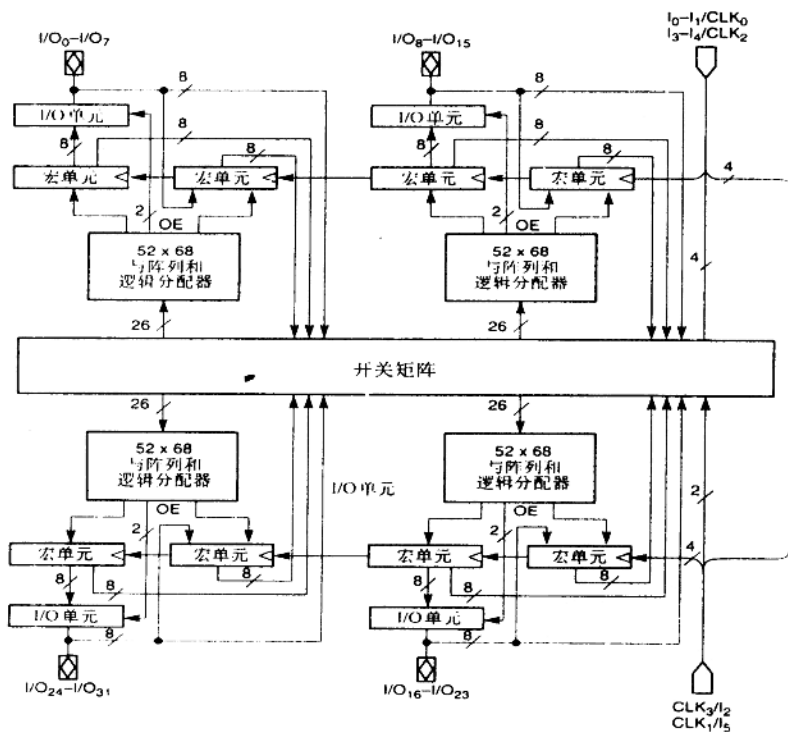


图2.5 MACH211 的结构框图

开关矩阵将26个输入引到PAL块的与阵列,可生成64个乘积项,每4个乘积项纳入一个或门,再进入逻辑分配器。图2.7示出的逻辑分配器表明,可以将1~4个或门的输出(4~16个乘积项)相或后再引入宏单元。

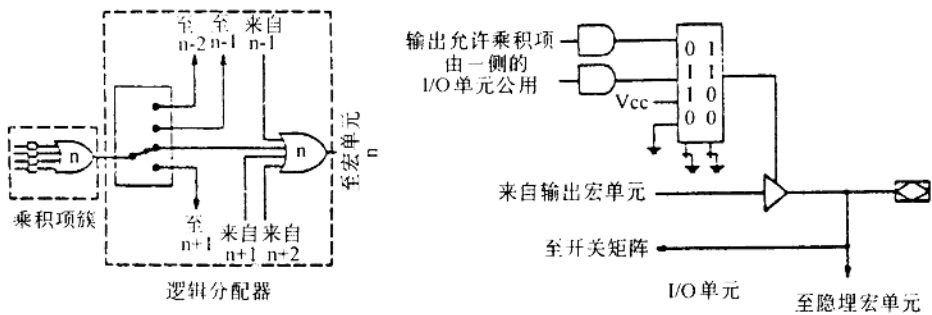


图2.7 MACH211的逻辑分配器和I/O单元

每个PAL块有16个宏单元,每个宏单元含一个触发器。这些宏单元分为两类:

一类称为输出宏单元,它的输入来自逻辑分配器,其输出既可连到I/O引脚上,也可反馈到开关矩阵(再引入与阵列),输出宏单元的结构示于图2.8。另一类称为隐埋式宏单元,它的输入来自逻辑分配器或者I/O引脚,其输出只能反馈到开关矩阵(再引入与阵列)。这类宏单元的结构示于图2.9。隐埋式宏单元和输出宏单元都有多种结构模式,分别示于图2.10和图2.11。

输出宏单元与I/O单元相连。I/O单元中包括一个输出三态门和一个多路选择开关,I/O单元如图2.7所示。多路选择开关决定是否允许输出,以及用什么信号控制输出三态门。

以上各种结构图表明,MACH211的内部资源十分丰富,结构也很灵活。但要用来设计8位运算器也必须对这些资源合理分配才能实现。

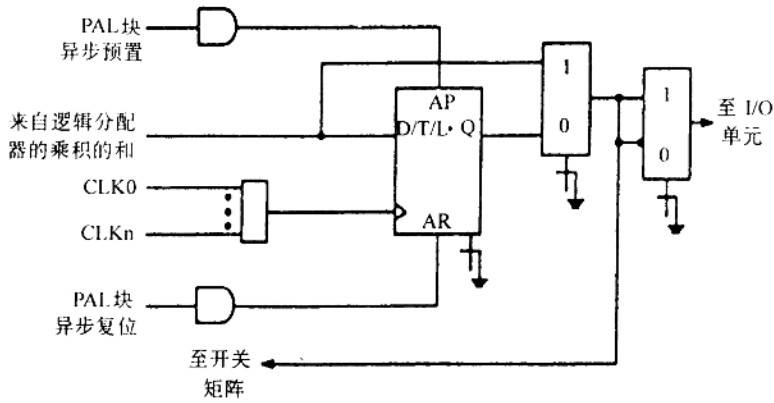


图2.8 MACH211的输出宏单元