

其次,在有的 EPLD 中,将每一组乘积项分作两部分,产生两个与或逻辑函数,然后通过编程使这两部分既可以单独地送到输出逻辑电路,又可以组合在一起产生一个项数更多的与或逻辑函数,如图 8.5.2 所示。Atmel公司生产的 ATV750 就采用了这种阵列结构。

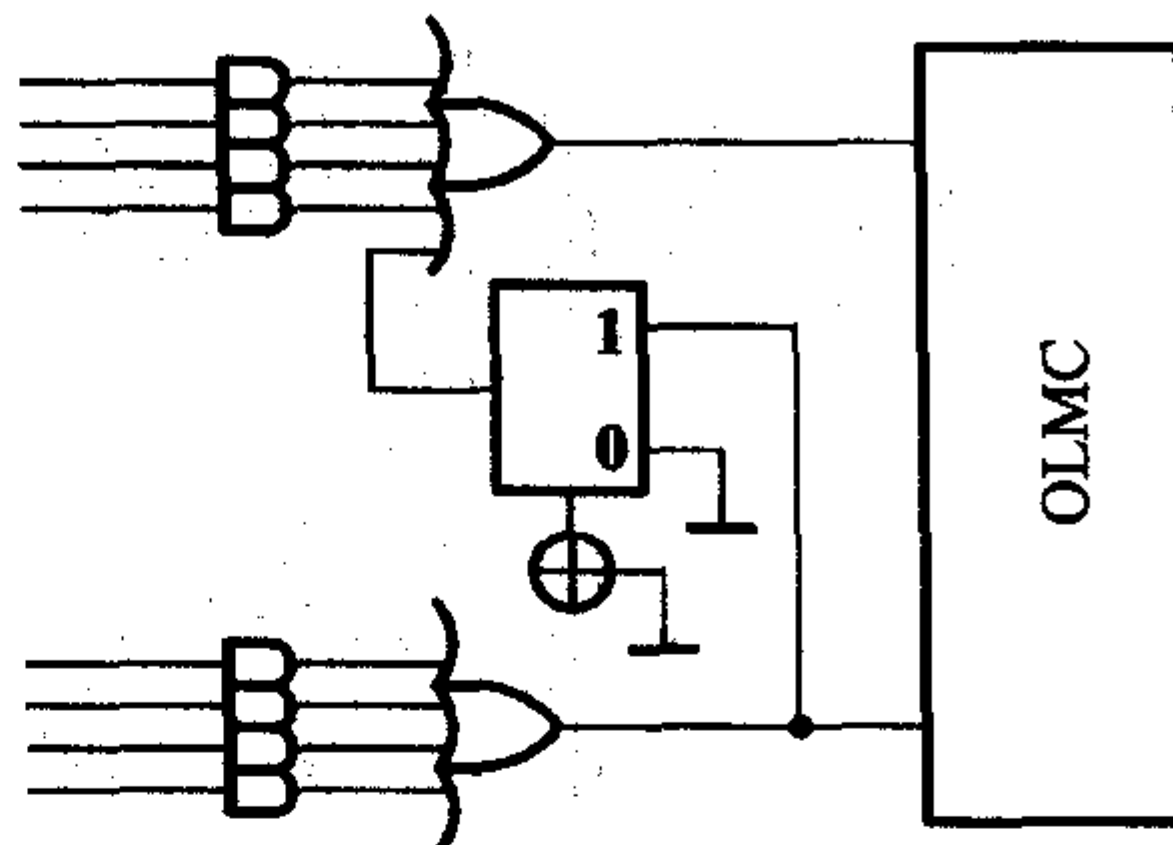


图 8.5.2 每组乘积项分为两部分的可编程结构

另外一种改进的与-或阵列是图 8.5.3 所示的乘积项共享的可编程结构。其中每组乘积项都分成两部分,通过编程可以将这两部分相加,产生一个含有 8 个乘积项的与或逻辑函数,也可以分别为相邻一组所共享,与相邻一组乘积项共同组成一个项数更多的与或逻辑函数。在图 8.5.3 给出的情况下,虽然每一组乘积项本身为 8 项,但通过对 4 个编程单元的编程可以产生包含 4、8、12、16 项的与或逻辑函数。可见,这种可编程结构能使与逻辑阵列的乘积项得到充分的利用。采用这种与-或阵列结构的 EPLD 有 Altera 公司的 EP512 等。

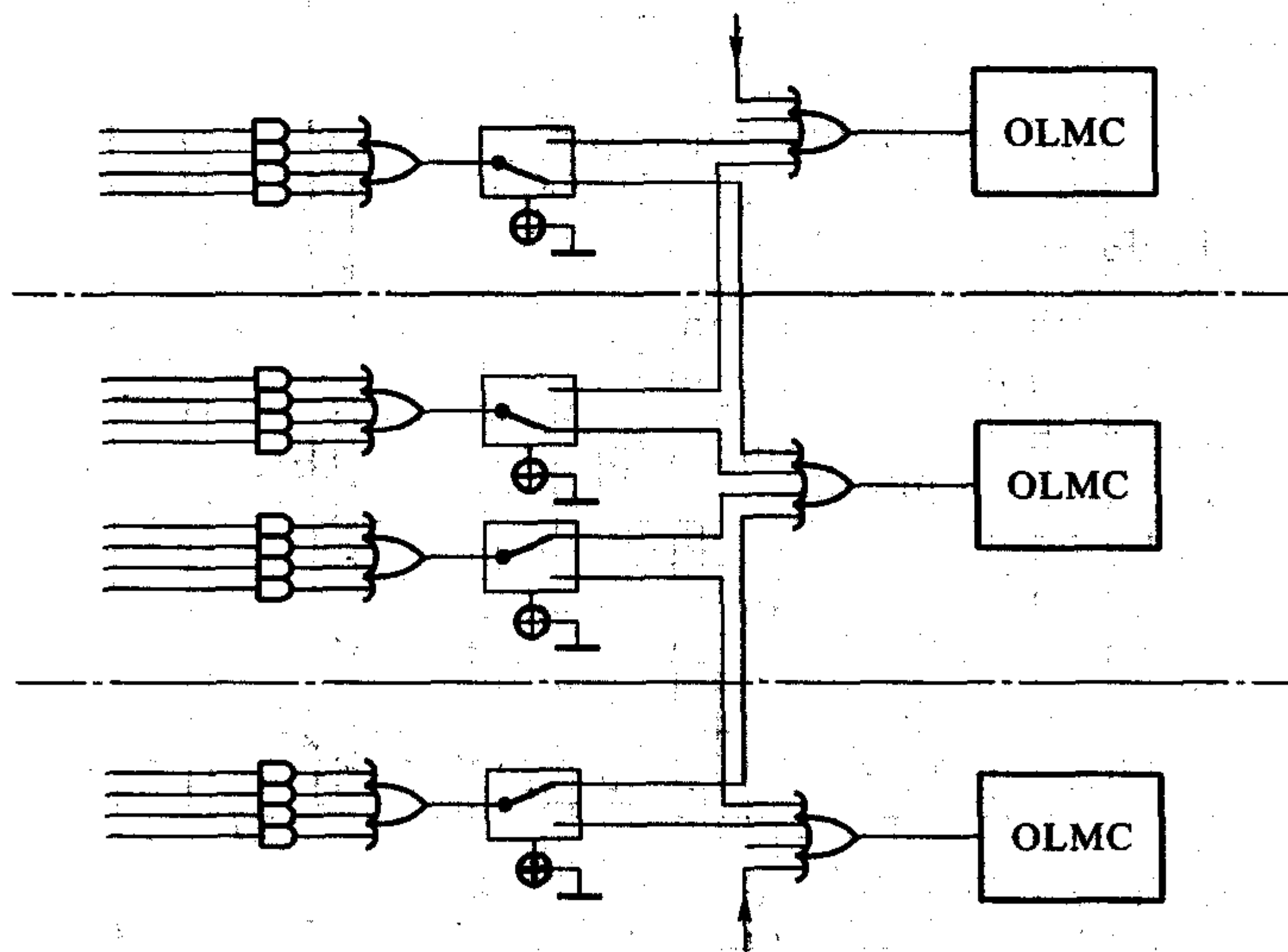


图 8.5.3 与-或逻辑阵列的乘积项共享结构

*8.5.3 EPLD 的输出逻辑宏单元(OLMC)

EPLD 的输出电路结构和 GAL 相似,也采取了可编程的输出逻辑宏单元 OLMC。通过编程的方法能将 OLMC 设置成各种不同的工作状态。而且,由于

增加了对 OLMC 中触发器的预置和置零功能,因而具有更大的使用灵活性。

不同型号 EPLD 的 OLMC 在电路结构上也各不相同,但从预置和置零的工作方式上可以分为两大类,一类为同步工作方式,另一类为异步工作方式。

例如,AT22V10 中的 OLMC 就属于同步工作方式。图 8.5.4 是 AT22V10 的 OLMC 电路结构图,它由 D 触发器、异或门和两个 2 选 1 数据选择器 MUX1、MUX2 构成。

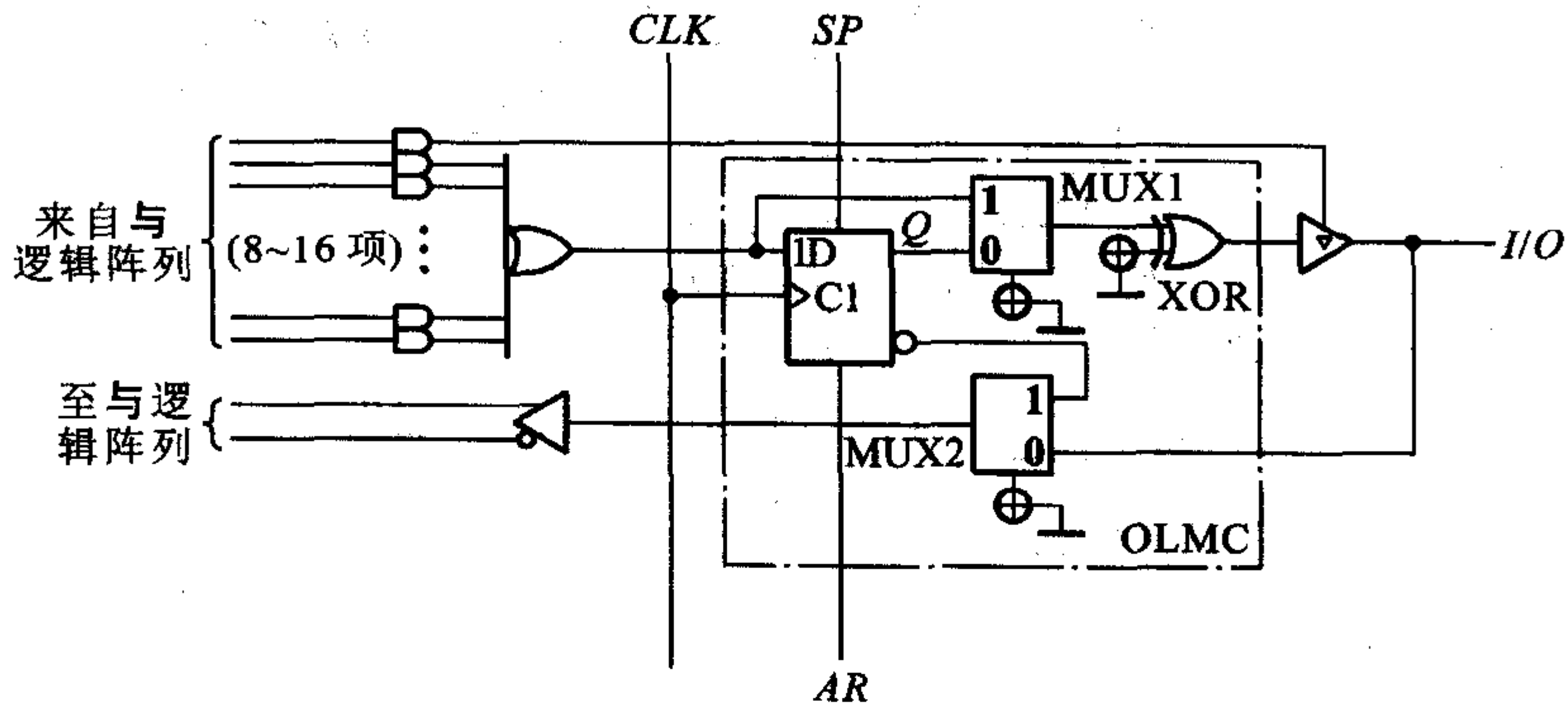


图 8.5.4 AT22V10 的 OLMC 电路结构图

MUX1 用于实现输出逻辑组态的选择。当控制 MUX1 的编程单元输出高电平时为组合逻辑输出;而当编程单元为低电平时为时序逻辑输出(或称为寄存器输出)。MUX2 用于反馈信号的选择。当控制 MUX2 的编程单元输出为高电平时,反馈信号取自触发器;而当编程单元输出为低电平时,反馈信号来自 I/O 端。

通过对异或门输入端编程实现对输出极性的选择。当异或门的可编程输入端接地时,输出高电平有效;当编程输入端为高电平时,输出低电平有效。三态缓冲器的控制信号由与逻辑阵列的一个可编程的乘积项给出。

由于触发器的异步置零信号 AR 和同步预置信号 SP 各由一个可编程的乘积项提供,这样就可以按照需要编程,以决定在什么情况下将触发器置 0 或置 1。

在 AT22V10 中,所有 OLMC 中触发器的时钟信号 CLK、异步置零信号 AR 和同步预置信号 SP 都是公用的,因此所有 OLMC 中的触发器无论在置零时还是在预置时都是同时动作的。

下面再以 ATV750 为例,说明一下异步工作方式 OLMC 的特点。图 8.5.5 是 ATV750 的 OLMC 电路结构图,它包含两个 D 触发器 FF₁ 和 FF₂、异或门 XOR 和一个 2 选 1 数据选择器 MUX。通过对 MUX 编程实现组合逻辑输出和寄存器

输出间的选择;通过对异或门输入端编程实现对输出极性的选择。

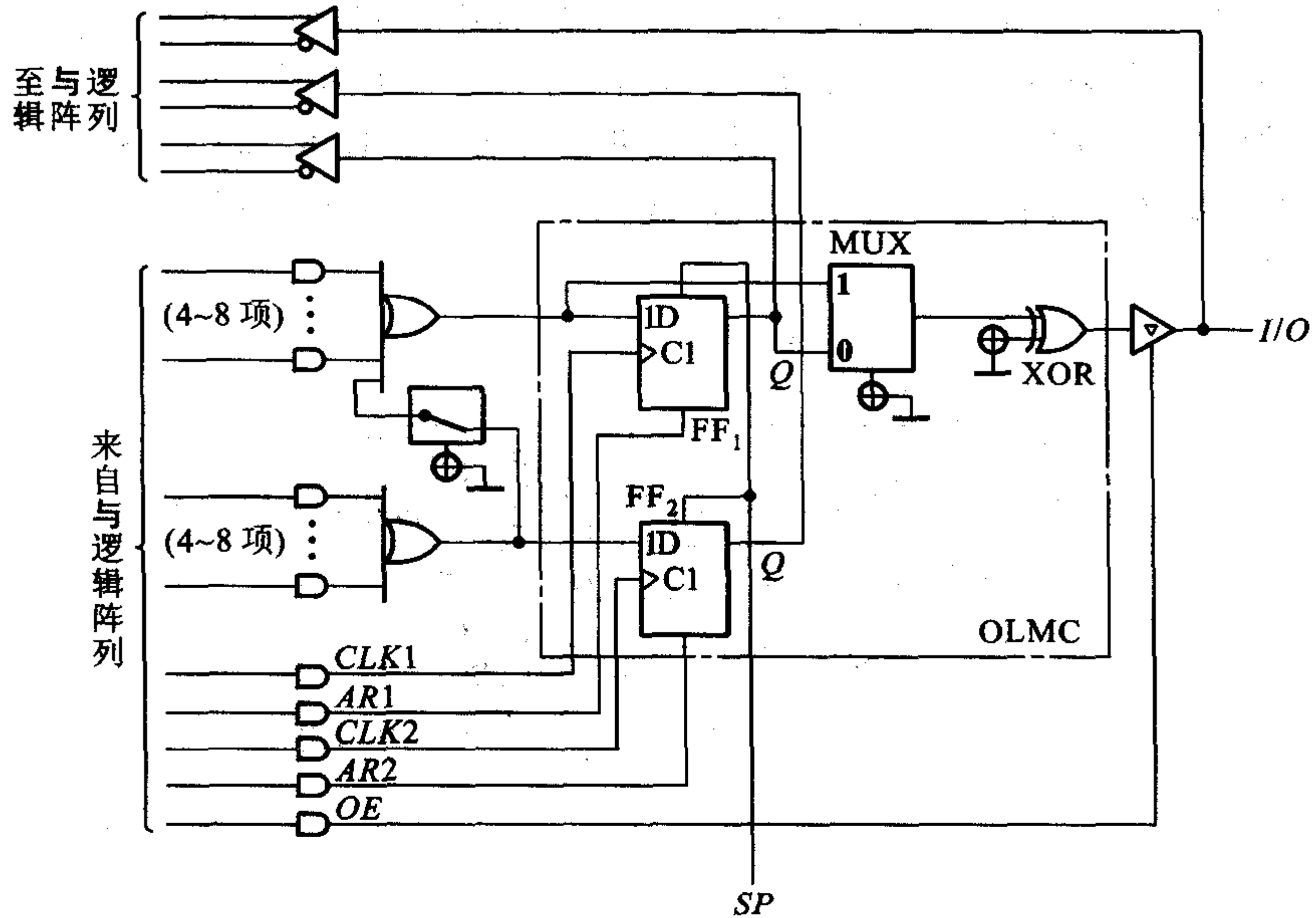


图 8.5.5 AT750 的 OLMC 电路结构图

这个电路最突出的特点在于每个触发器的时钟信号 CLK 、异步置零信号 AR 都是独立的,而且分别由与逻辑阵列的一个乘积项给出,是可编程的。因此,这种类型的 EPLD 不仅可以用于同步时序逻辑电路的设计,而且可以用于异步时序逻辑电路的设计。

此外,由于 AT750 的 OLMC 中含有两个触发器,而且每个触发器的状态都直接反馈到了与逻辑阵列,这就为时序逻辑电路的设计提供了更多的方便和更大的灵活性。

8.6 复杂的可编程逻辑器件 (CPLD)

8.6.1 CPLD 的总体结构

近年来迅速发展起来的复杂的可编程逻辑器件 CPLD,可以认为是从 EPLD 演变而来的。为了提高集成度,同时又保持 EPLD 传输时间可预测的优点,将若干个类似于 GAL 的功能模块和实现互连的开关矩阵集成于同一芯片上,就形成了所谓的 CPLD。CPLD 多采用 E^2CMOS 工艺制作。

同时,为了使用方便,越来越多的 CPLD 都做成了在系统可编程器件 isp-PLD。在 ispPLD 电路中除了原有的可编程逻辑电路以外,还集成了编程所需的高压脉冲产生电路以及编程控制电路。因此,编程时不需要使用另外的编程器,也无需将 ispPLD 从系统中拨出,在正常的工作电压下即可完成对器件的编程(写入编程数据或擦除)。

CPLD 产品的种类和型号繁多,目前各大半导体器件生产厂商仍在不断推出 CPLD 新产品。虽然它们的具体结构形式各不相同,但基本上都由若干个可编程的逻辑模块、输入/输出模块和一些可编程的内部连线阵列组成。在 ispPLD 中都包含有编程电路部分,不过通常在为用户提供的结构框图中都没有画出。

下面我们以 Lattice 公司生产的在系统可编程器件 ispLSI1032 为例,来了解一下 CPLD 的具体结构。

图 8.6.1 是 ispLSI1032 的电路结构框图,它由 32 个通用逻辑模块(Generic Logic Block,简称 GLB)、64 个输入/输出单元(I/O Cell,简称 IOC)、可编程的内

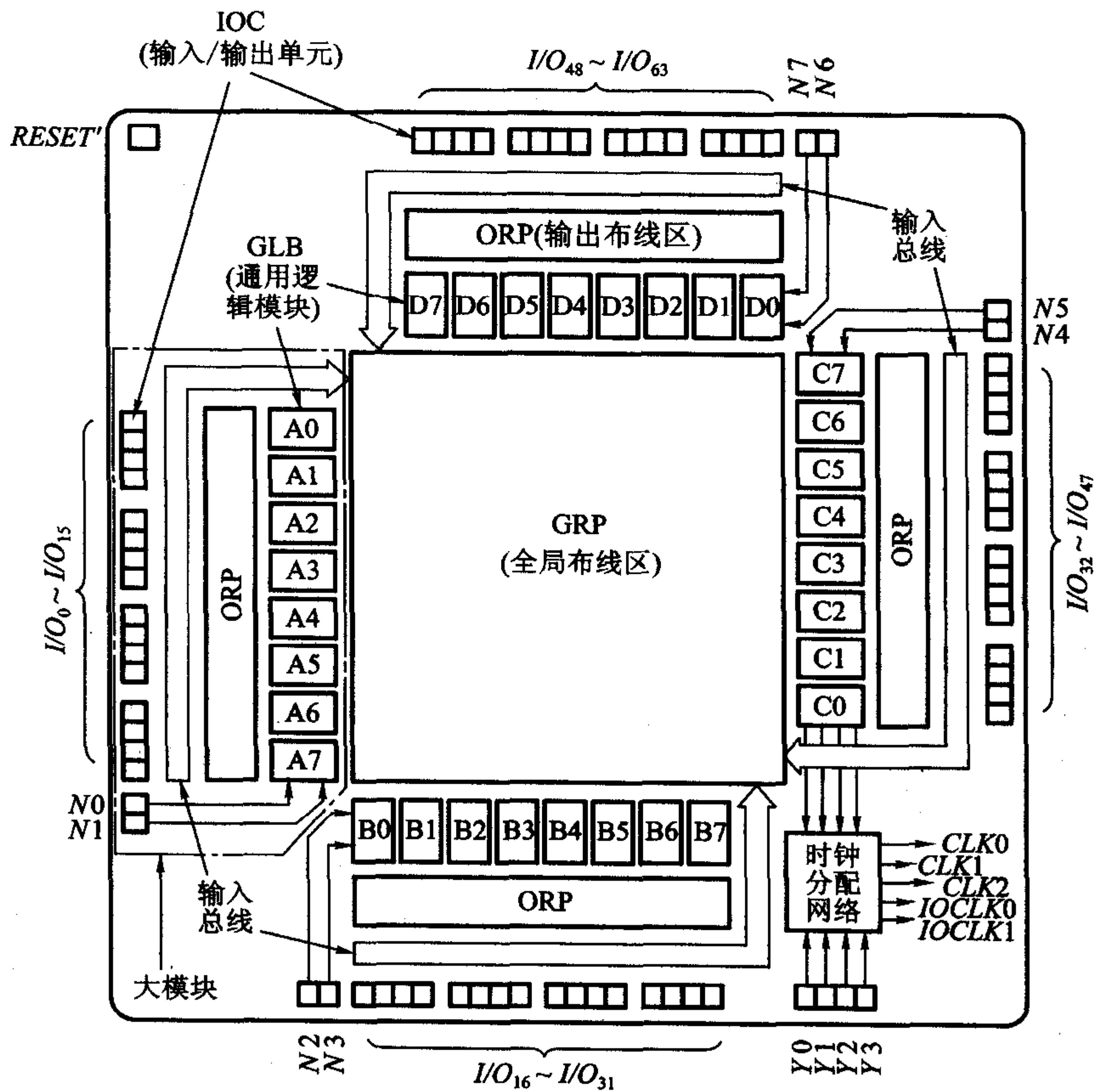


图 8.6.1 ispLSI1032 的电路结构框图

部连线区和编程控制电路组成。在全局布线区的四周,形成了4个结构相同的大模块。图中没有画出编程控制电路这部分。各部分之间的关系和实现的功能如图8.6.2所示。

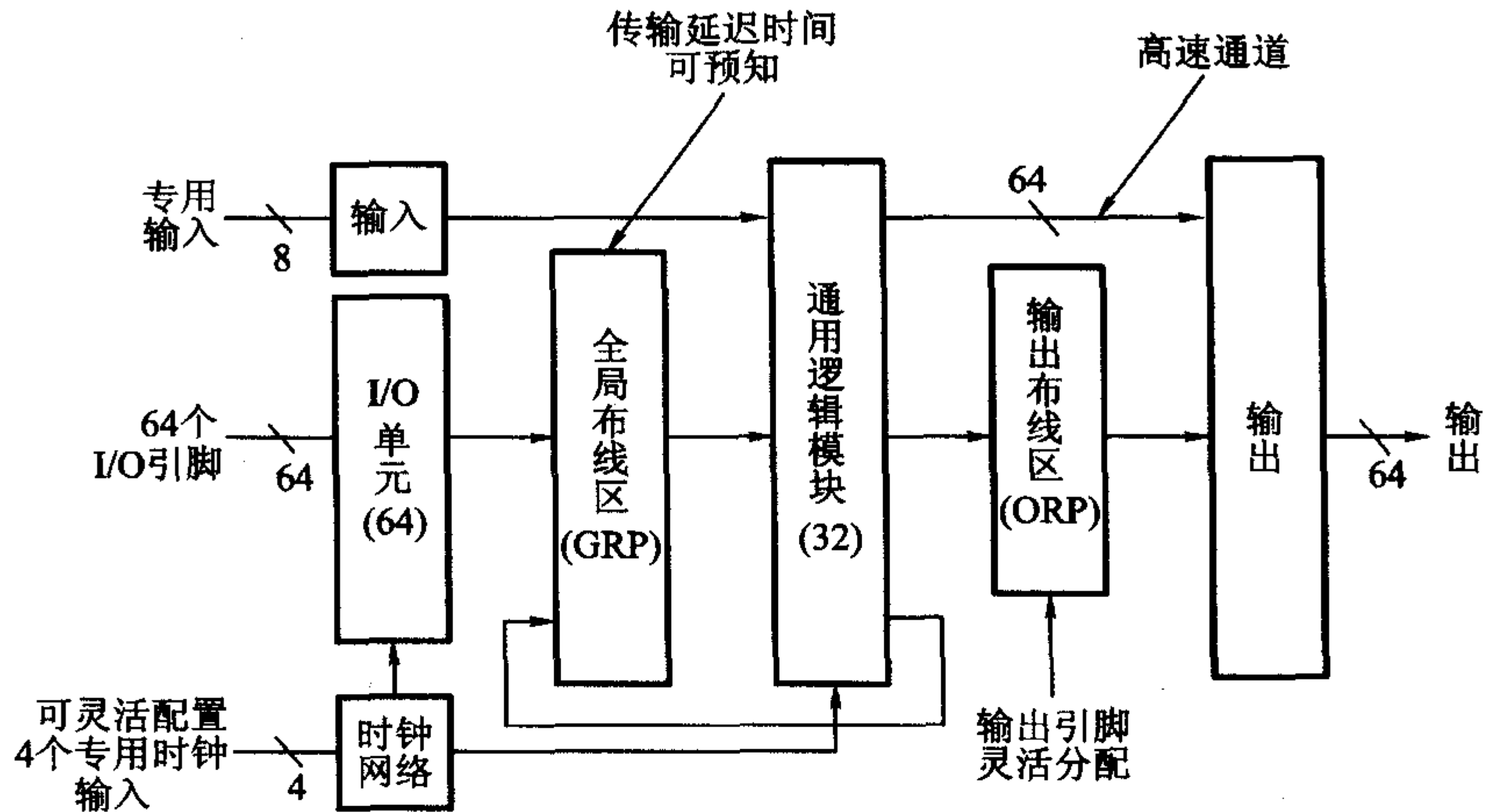


图 8.6.2 ispLSI1032 的逻辑功能划分框图

* 8.6.2 CPLD 的通用逻辑模块 (GLB)

图 8.6.3 是通用逻辑模块的电路结构图。由图可见,它由可编程的与逻辑阵列、乘积项共享的或逻辑阵列和输出逻辑宏单元 (OLMC) 三部分构成。这种结构形式与 GAL 类似,但又在 GAL 的基础上做了若干改进,在组态时有更大的灵活性。

首先,它的或逻辑阵列采取了乘积项共享的结构形式。它的输入和输出关系是可编程的,4 个输入 $F_0 \sim F_3$ 中任何一个都可以送到 4 个 D 触发器当中任何一个的输入端,每个输入又可以同时送给几个触发器,4 个输入还可以再组合成更大规模的与或逻辑函数送到任何一个触发器的输入端。

此外,除了图 8.6.3 所示的标准配置模式以外,通过编程还可以将 GLB 设置成其他 4 种连接模式,即高速旁路模式、异或逻辑模式、单乘积项模式和多重模式。

在高速旁路模式中,为了减少传输延迟时间,越过了乘积项共享的编程阵列,将与-或阵列的输出 $F_0 \sim F_3$ 直接与 OLMC 相接,如图 8.6.4(a)所示。

在异或逻辑模式中,乘积项共享或阵列的输出与 OLMC 之间又串进了异或门,如图 8.6.4(b)所示。异或门的一个输入来自 $F_0 \sim F_3$,另一个是来自与逻辑阵列的乘积项。

在单乘积项模式中,每个 OLMC 的输入取自与逻辑阵列一个单乘积项的输出,如图 8.6.4(c)所示。这种结构模式可以获得最快的信号传输速度。

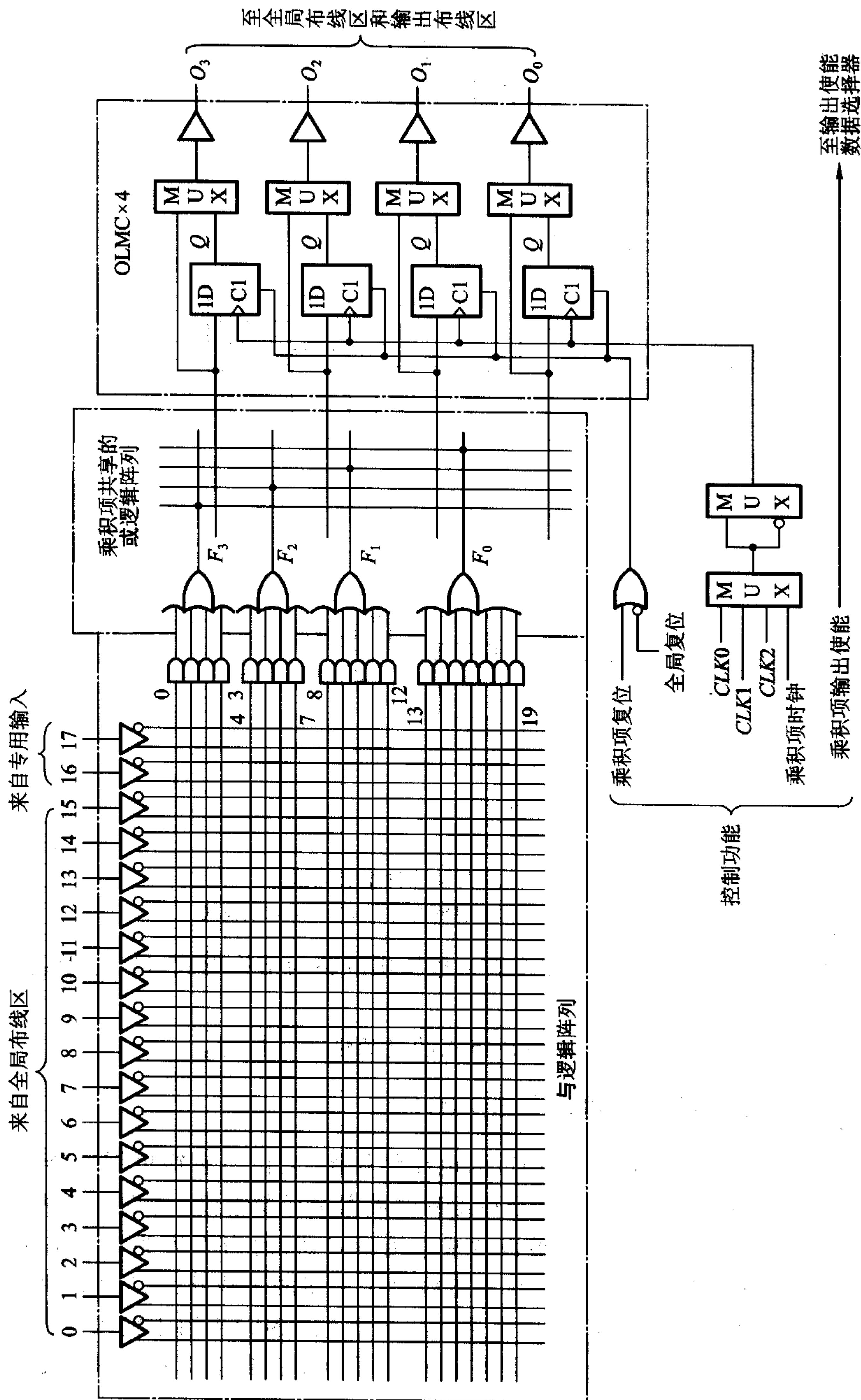


图 8.6.3 通用逻辑模块 (GLB) 的电路结构

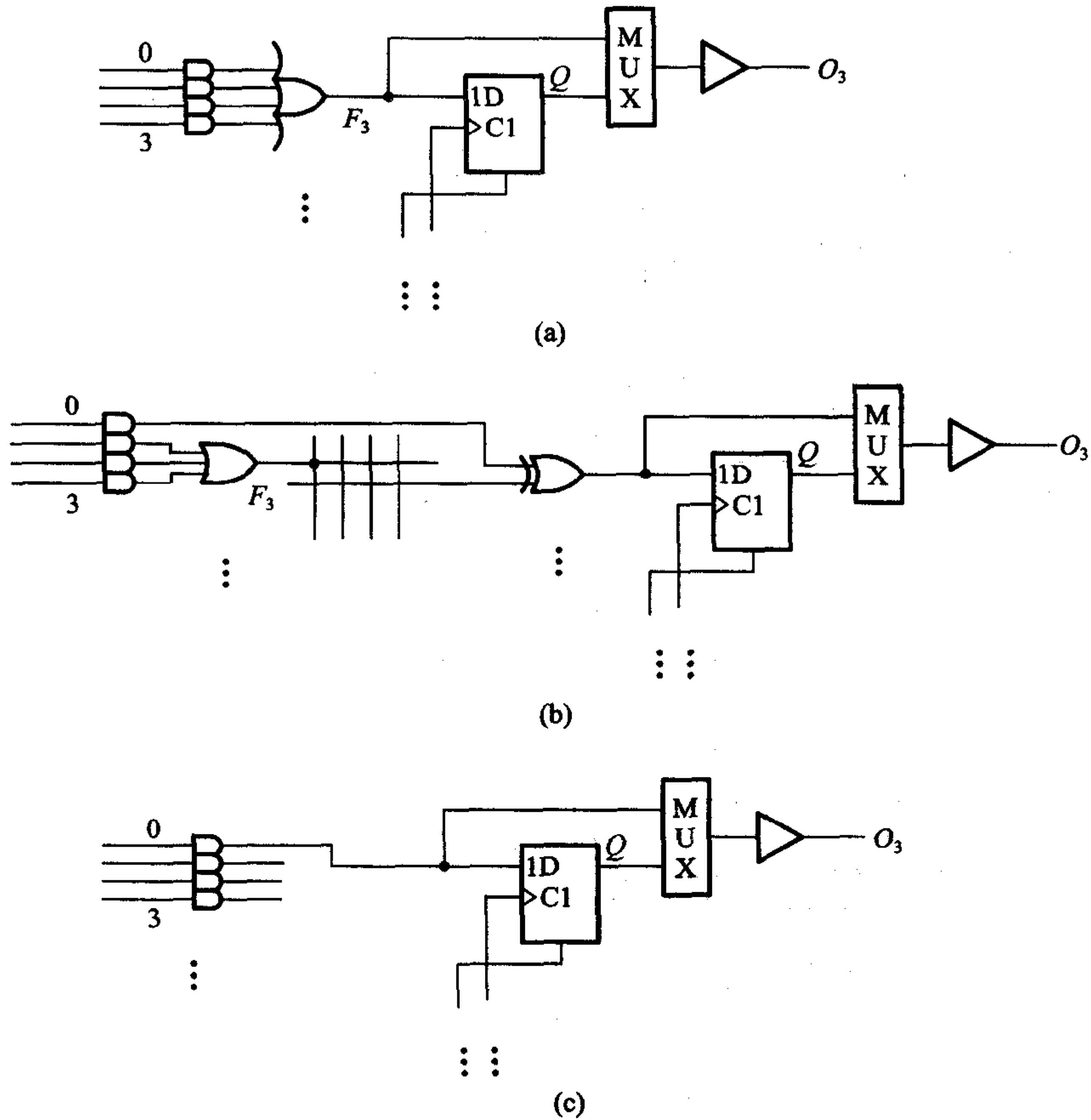


图 8.6.4 GLB 的其他几种组态模式

(a) 高速旁路模式 (b) 异或逻辑模式 (c) 单乘积项模式

在多重模式中,每个 GLB 中 4 个与 - 或输出电路的结构形式可以分别组成上述几种不同模式。

这些不同的组态模式增加了 GLB 组态的灵活性和多样性。

*8.6.3 CPLD 的输入/输出单元(IOC)

图 8.6.5 是输入/输出单元(IOC)的电路结构图,它由三态输出缓冲器、输入缓冲器、输入寄存器/锁存器和几个可编程的数据选择器组成。触发器有两种工作方式:当 R/L 为高电平时,它被设置成边沿触发器;而当 R/L 为低电平时,它被设置成锁存器。MUX1 用于控制三态输出缓冲器的工作状态, MUX2 用于选择输出信号的传送通道, MUX3 用来选择输出极性。MUX4 用于输入方式的选择:在异步输入方式下,输入信号直接经输入缓冲器送到全局布线区的输入

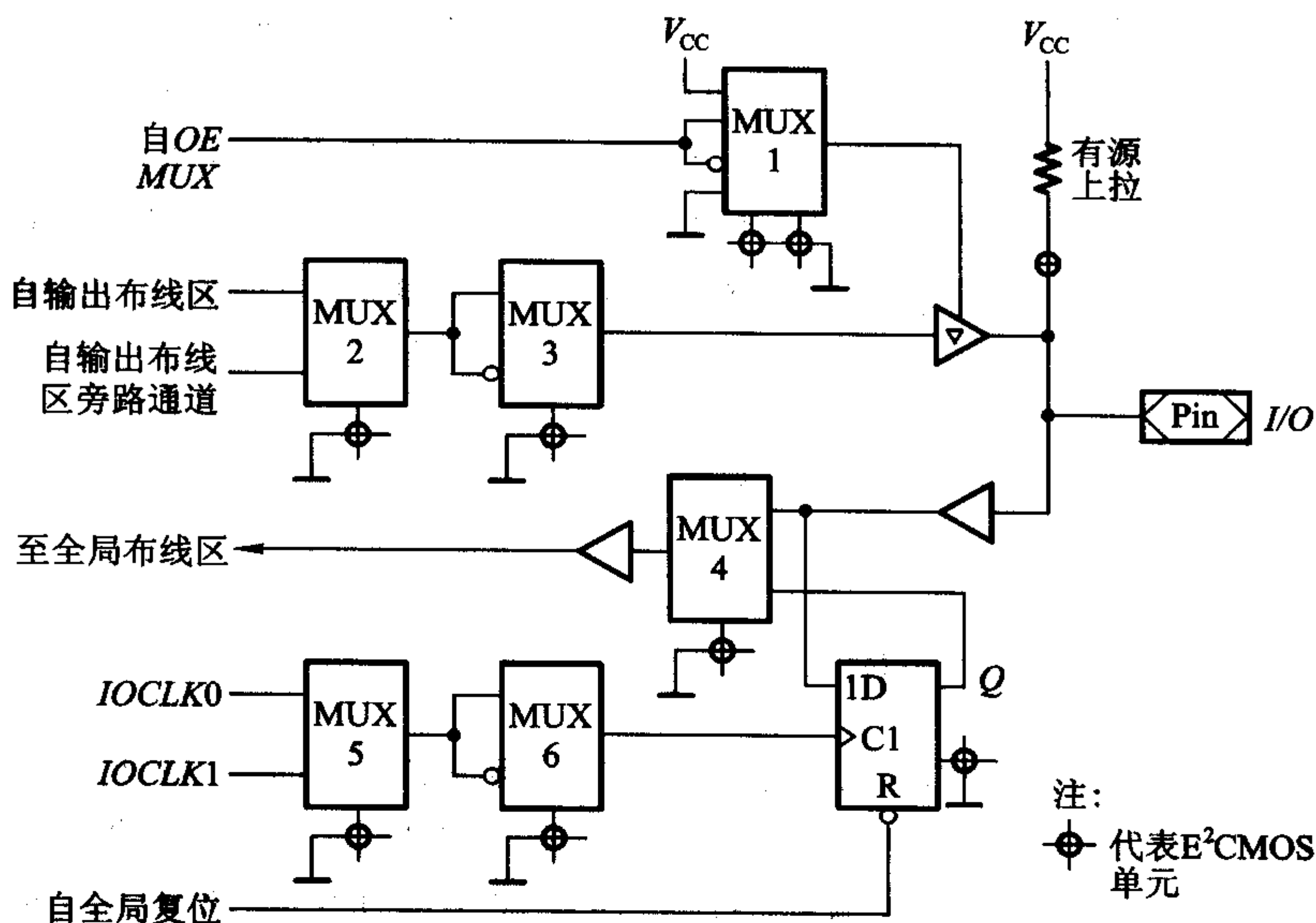


图 8.6.5 输入/输出单元(IOC)的电路结构

端；在同步输入方式下，输入信号加到触发器的输入端，必须等时钟信号 *IOCLK* 到达后才能被存入触发器，并经过输入缓冲器加到全局布线区。MUX5 和 MUX6 用于时钟信号的来源和极性的选择。根据这些数据选择器编程状态的组合，得到各种可能的 IOC 组态，如图 8.6.6 所示。

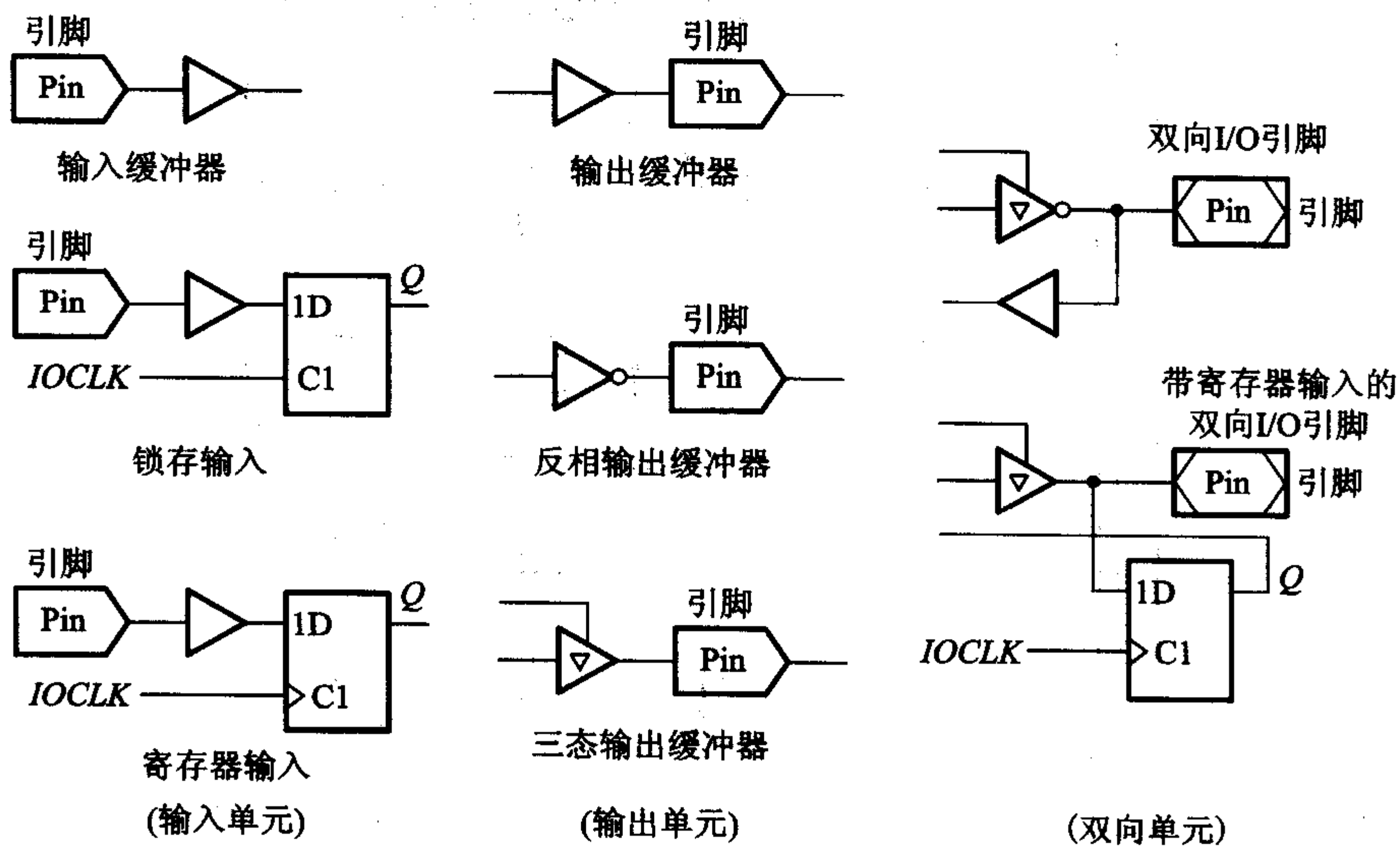


图 8.6.6 IOC 的各种组态

ispLSI1032 中有一个全局布线区(Global Routing Pool,简称 GRP)和 4 个输出布线区(Output Routing Pool,简称 ORP)。这些布线区都是可编程的矩阵网络,每条纵线和每条横线的交叉点接通与否受一位编程单元状态的控制。通过对 GRP 的编程,可以实现 32 个 GLB 间的互相连接以及 IOC 与 GRP 的连接。通过对 ORP 的编程,可以使每个大模块中任何一个 GLB 能与任何一个 IOC 相连。

8.7 现场可编程门阵列(FPGA)

8.7.1 FPGA 的基本结构

在前面所讲的几种 PLD 电路中,都采用了与-或逻辑阵列加上输出逻辑单元的结构形式。而 FPGA 的电路结构形式则完全不同,它由若干独立的可编程逻辑模块组成,用户可以通过编程将这些模块连接成所需要的数字系统。因为这些模块的排列形式和门阵列(GA)中单元的排列形式相似,所以沿用了门阵列这个名称。FPGA 属于高密度 PLD,其集成度可达百万门/片以上。

图 8.7.1 是 FPGA 基本结构形式的示意图,它由三种可编程单元和一个用于存放编程数据的静态存储器组成。这三种可编程的单元是输入/输出模块 IOB(I/O Block)、可编程逻辑模块 CLB(Configurable Logic Block)和互连资源 IR

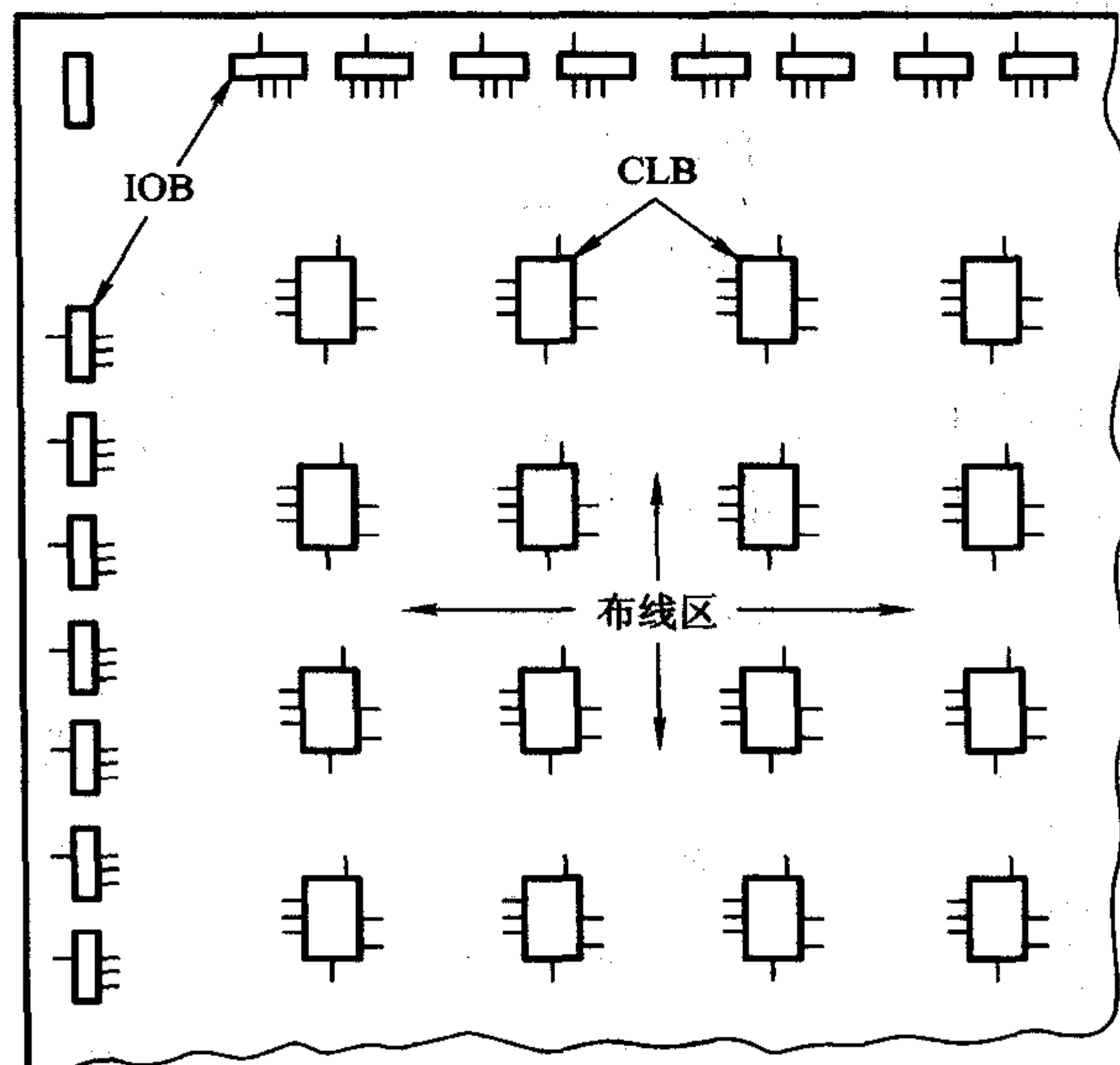


图 8.7.1 FPGA 的基本结构框图

(Interconnect Resource)。它们的工作状态全都由编程数据存储器中的数据设定。

FPGA 中除了个别的几个引脚以外,大部分引脚都与可编程的 IOB 相连,均可根据需要设置成输入端或输出端。因此,FPGA 器件最大可能的输入端数和输出端数要比同等规模的 EPLD 多。

每个 CLB 中都包含组合逻辑电路和存储电路(触发器)两部分,可以设置成规模不大的组合逻辑电路或时序逻辑电路。

为了能将这些 CLB 灵活地连接成各种应用电路,在 CLB 之间的布线区内配备了丰富的连线资源。这些互连资源包括不同类型的金属线、可编程的开关矩阵和可编程的连接点。

静态存储器的存储单元由两个 CMOS 反相器和一个控制管 T 组成,如图 8.7.2 所示。由于采用了独特的工艺设计,这种存储单元有很强的抗干扰能力和很高的可靠性。但停电以后存储器中的数据不能保存,因而每次接通电源以后必须重新给存储器“装载”编程数据。装载的过程是在 FPGA 内部的一个时序电路的控制下自动进行的。这些数据通常都需要存放在一片 EPROM 当中。

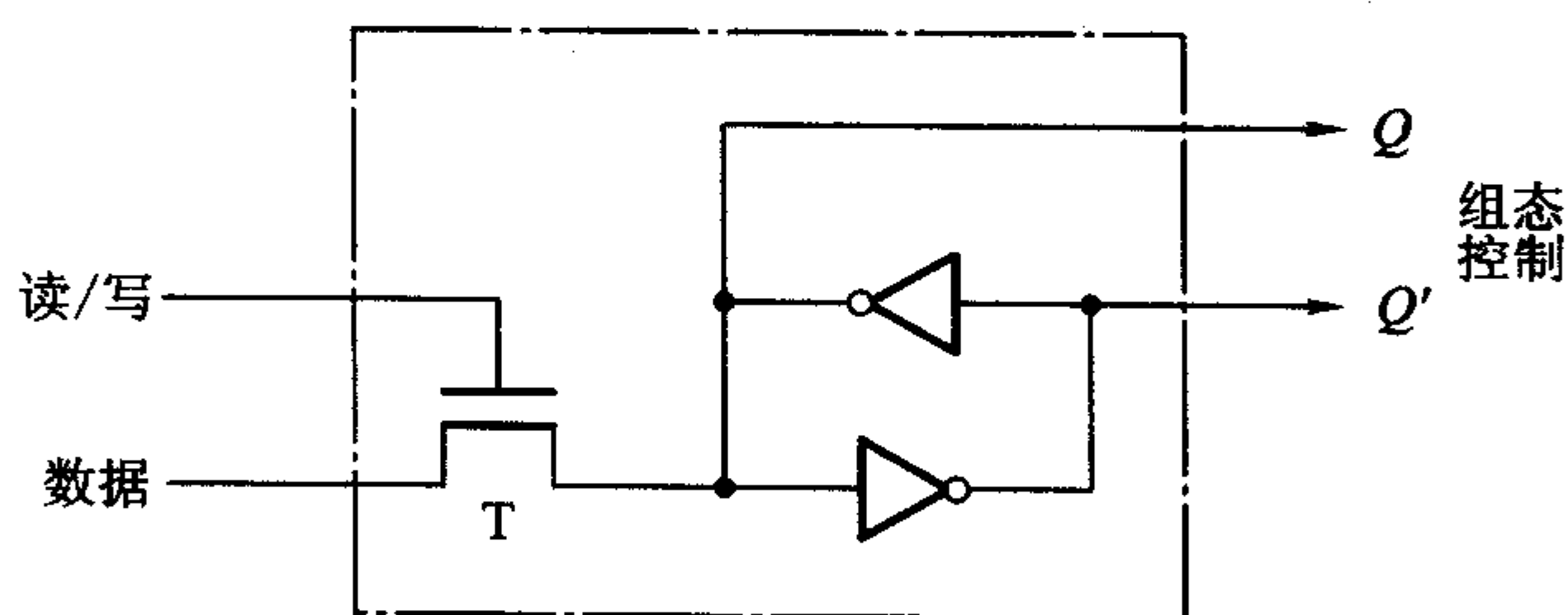


图 8.7.2 FPGA 内静态存储器的存储单元

FPGA 的这种 CLB 阵列结构形式克服了 PAL 等 PLD 中那种固定的与 - 或逻辑阵列结构的局限性,在组成一些复杂的、特殊的数字系统时显得更加灵活。同时,由于加大了可编程 I/O 端的数目,也使得各引脚信号的安排更加方便和合理。

但 FPGA 本身也存在着一些明显的缺点。首先,它的信号传输延迟时间不是确定的。在构成复杂的数字系统时一般总要将若干个 CLB 组合起来才能实现。而由于每个信号的传输途径各异,所以传输延迟时间也就不可能相等。这不仅会给设计工作带来麻烦,而且也限制了器件的工作速度。在 CPLD 中就不存在这个问题。

其次,由于 FPGA 中的编程数据存储器是一个静态随机存储器结构,所以断