

介绍了。

以上所做的介绍,只是想通过 XC2064 的例子使读者对 FPGA 的数据装载过程有些初步的了解。在选定某种型号 FPGA 器件做设计时,还应仔细阅读所用器件的技术资料。

复习思考题

R8.7.1 FPGA 和 CPLD 在电路结构和工作特性上有何不同?

R8.7.2 为什么用 FPGA 设计的逻辑电路一般还需要配上一个 EPROM?

8.8 在系统可编程通用数字开关 (ispGDS)

在一个由多片 ispPLD 构成的数字系统中,为了改变电路的逻辑功能,有时不仅要重新设置每个 ispPLD 的组态,而且需要改变它们之间的连接以及它们与外围电路(例如负载电路、显示器件等)的连接。为满足这一需要,Lattice 公司生产了在系统可编程通用数字开关(In - System Programmable Generic Digital Switch,简称 ispGDS)。

现以 ispGDS22 为例介绍一下它的结构与工作原理。图 8.8.1 是 ispGDS22 的结构框图,它由可编程的开关矩阵和若干输入/输出单元 IOC 组成。

可编程开关矩阵中的每个交叉点是否接通,由一位编程单元的状态控制。因此,通过编程的方法可将 A 列中的任何一个 IOC 与 B 列中的任何一个 IOC 接通。

IOC 的电路结构如图 8.8.2 所示。当 C_0 被编程为低电平时,电路工作在输出方式,输出端的三态缓冲器为工作状态。这时数据选择器从 4 个输入当中选中一个,经输出三态缓冲器送到输出端。当 $C_2C_1 = 11$ 时,输出的是来自开关矩阵的信号,而 $C_2C_1 = 10$ 时输出的是反相以后的来自开关矩阵的信号。当 C_2C_1 为 01 和 00 时,输出端被分别设置成高电平和低电平输出。

在系统可编程逻辑器件和在系统可编程通用数字开关的应用不仅为数字电路的设计提供了很大的方便,而且在很大程度上改变了以往从事数字系统设计、调试、运行的工作方式。

首先,它使硬件的设计工作更加简单、方便了。因为电路的逻辑功能可以由编程数据设定,而且能在线装入和修改,所以硬件的设计和安装完全可以一次完

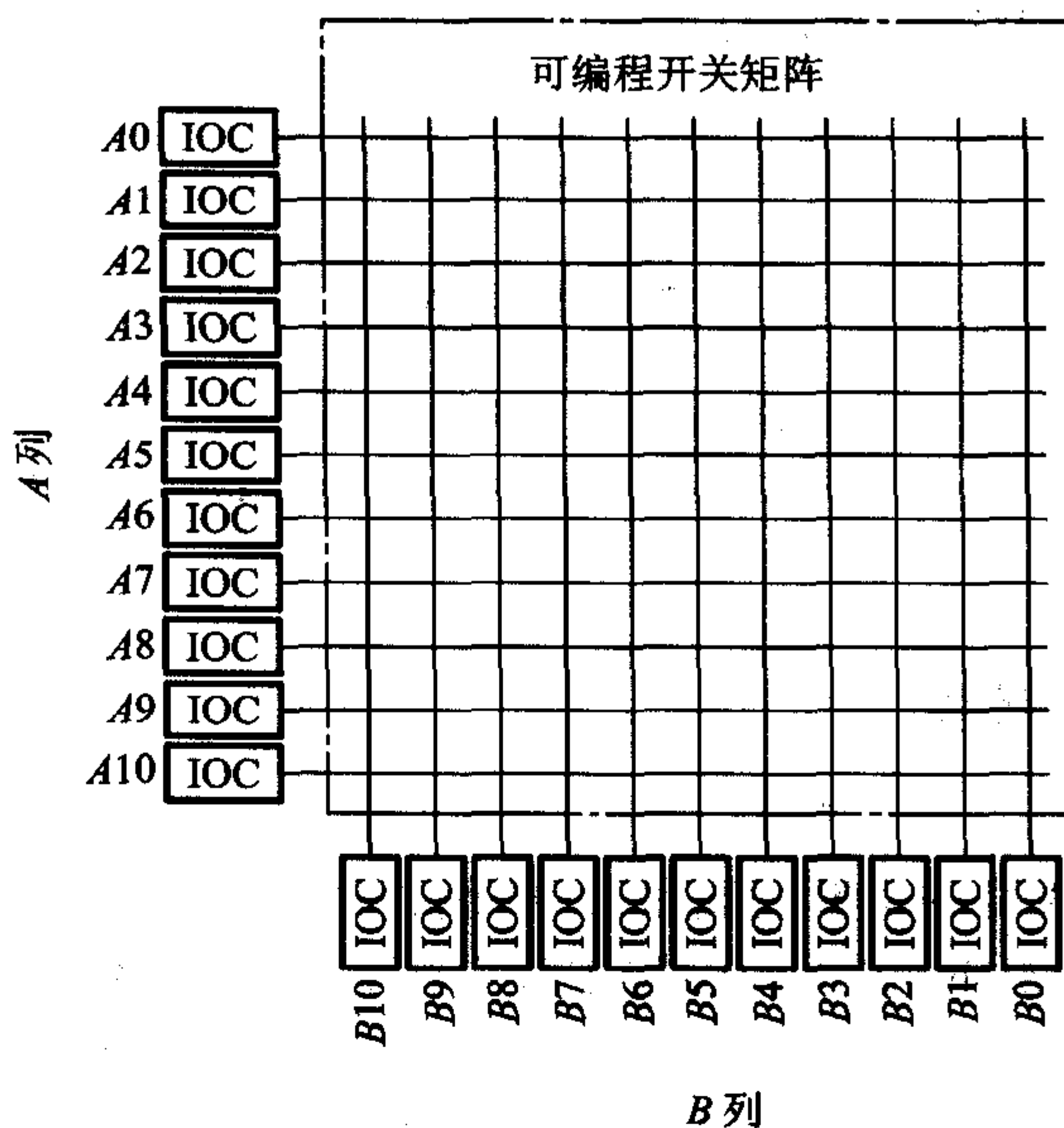


图 8.8.1 ispGDS22 的结构框图

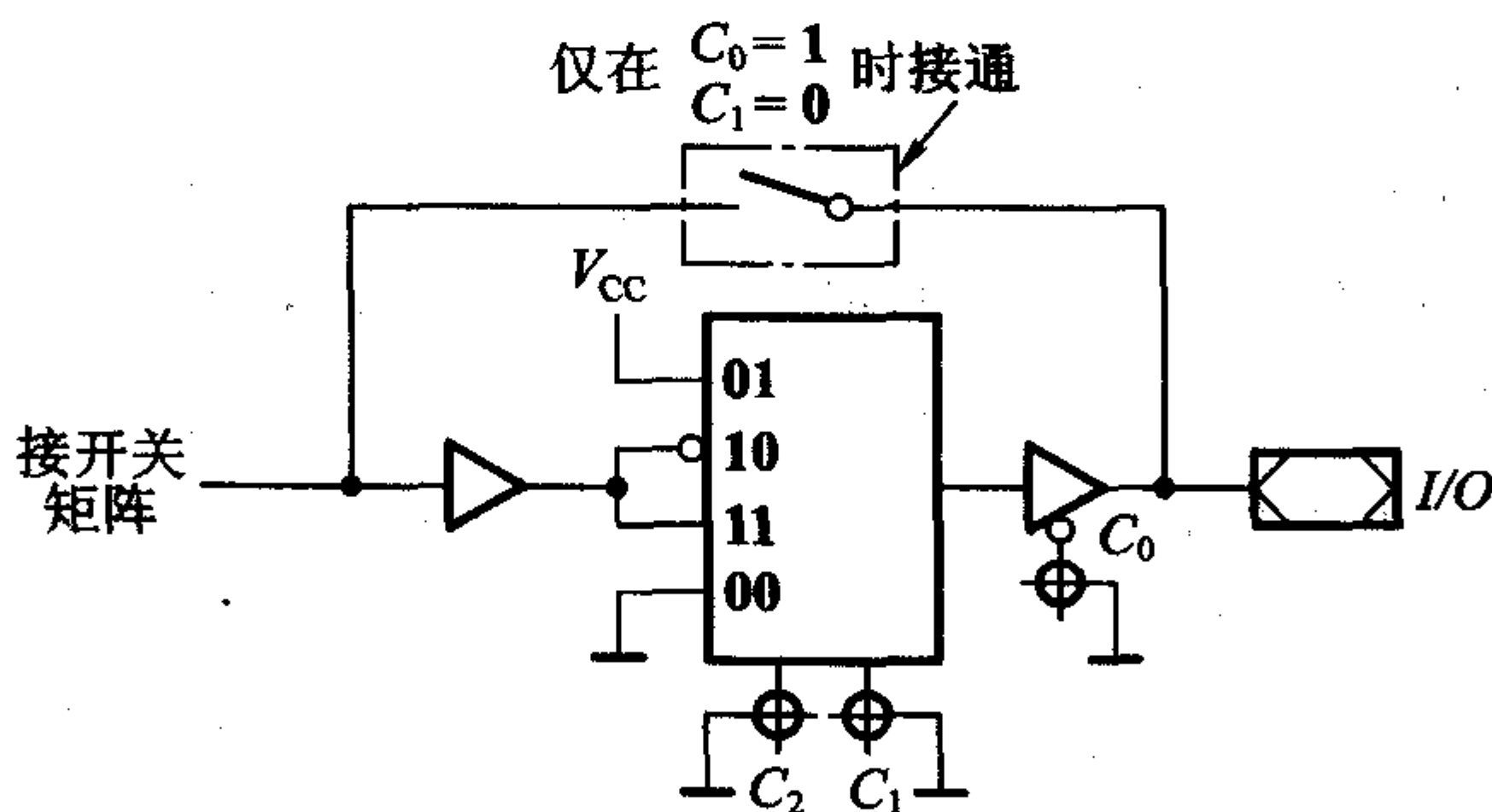


图 8.8.2 ispGDS22 的输入/输出单元(IOC)

成。这样就节省了修改硬件电路耗费的人力和物力。而且有时几种不同功能的逻辑电路可以采用相同的硬件电路,这也减少了许多硬件设计的工作量。

其次,在调试工作中通过写入编程数据很容易将电路设置成各种便于调试的状态,对电路进行测试,这比通过直接设置硬件电路的状态要方便得多。

用 ispPLD 构成的系统在运行操作上也十分方便。在系统工作过程中,随时可以根据需要改变电路的逻辑功能,不必将器件从电路板上取下即可完成。这对于那些需要不断升级换代的数字系统极为有利,因为可以在不改动硬件电路的情况下实现系统的升级换代。利用这种方法还可以通过遥控的方式对那些工

作在恶劣环境中的数字系统进行测试或修改逻辑功能。

复习思考题

R8.8.1 ispGDS 具有什么特别的功能? 它有什么用途?

8.9 PLD 的编程

随着 PLD 集成度的不断提高, PLD 的编程也日益复杂, 设计的工作量也越来越大。在这种情况下, PLD 的编程工作必须在开发系统的支持下才能完成。为此, 一些 PLD 的生产厂商和软件公司相继研制成了各种功能完善、高效率的 PLD 开发系统。其中一些系统还具有较强的通用性, 可以支持不同厂家生产的、各种型号的 PAL、GAL、EPLD、CPLD 以及 FPGA 产品的开发。

PLD 开发系统包括软件和硬件两部分。

开发系统软件是指 PLD 专用的编程语言和相应的汇编程序或编译程序。

早期使用的多为一些汇编型软件。这类软件要求以化简后的与或逻辑式输入, 不具备自动化简功能, 而且对不同类型 PLD 的兼容性较差。例如, 由 MMI 公司研制的 PALASM 以及随后出现的 FM (Fast - Map) 等就属于这一类。

进入 20 世纪 80 年代以后, 功能更强、效率更高、兼容性更好的编译型开发系统软件很快地得到了推广应用。其中比较流行的有 Data I/O 公司研制的 ABEL 和 Logical Device 公司的 CUPL。这类软件输入的源程序采用专用的高级编程语言 (也称为硬件描述语言 HDL) 编写, 有自动化简和优化设计功能。除了能自动完成设计以外, 还有电路模拟和自动测试等附加功能。

20 世纪 80 年代后期又出现了功能更强的开发系统软件。这种软件不仅可以用高级编程语言输入, 而且可以用电路原理图输入。这对于想把已有的电路 (例如用中、小规模集成器件组成的一个数字系统) 写入 PLD 的人来说, 提供了最便捷的设计手段。例如, Data I/O 公司的 Synario 就属于这样的软件。

20 世纪 90 年代以来, PLD 开发系统软件开始向集成化方向发展。为了给用户提供更加方便的设计手段, 一些生产 PLD 产品的主要公司都推出了自己的集成化开发系统软件 (软件包)。这些集成化开发系统软件通过一个设计程序管理软件将一些已经广为应用的优秀 PLD 开发软件集成为一个大的软件系统, 在设计时技术人员可以灵活地调用这些资源完成设计工作。属于这种集成化的

软件系统有 Altera 公司的 MAX + plus II、Lattice 公司的 ISP Synario、Xilinx 公司的 Foundation 等。

所有这些 PLD 开发系统软件都可以在 PC 机或工作站上运行。虽然它们对计算机内存容量的要求不同,但都没有超过目前 PC 机一般的内存容量。

开发系统的硬件部分包括计算机和编程器。编程器是对 PLD 进行写入和擦除的专用装置,能提供写入或擦除操作所需要的电源电压和控制信号,并通过串行接口从计算机接收编程数据,最终写进 PLD 中。早期生产的编程器往往只适用于一种或少数几种类型的 PLD 产品,而目前生产的编程器都有较强的通用性。

PLD 的编程工作大体上可按如下步骤进行。

第一步,进行逻辑抽象。首先要把需要实现的逻辑功能表示为逻辑函数的形式——逻辑方程、真值表、状态转换表(图)。

第二步,选定 PLD 的类型和型号。选择时应考虑到是否需要擦除改写;是否要求能在系统编程;是组合逻辑电路还是时序逻辑电路;电路的规模和特点(有多少输入端和输出端,多少个触发器,与或函数中乘积项的最大数目,是否要求对输出进行三态控制等);对工作速度、功耗的要求;是否需要加密等。

第三步,选定开发系统。选用的开发系统必须能支持选定器件的开发工作。与 PLD 器件相比,开发系统的价格要昂贵得多。因此,应该充分利用现有的开发系统,在系统所能支持的 PLD 种类和型号中选择适用的器件。

第四步,以开发系统软件能接受的逻辑功能描述方式(例如逻辑图、硬件描述语言、波形图等)编写计算机输入文件。

第五步,上机运行。将源程序输入计算机,运行相应的编译程序或汇编程序,产生 JEDEC 下载文件和其他程序说明文件。进行仿真分析,检查设计结果是否符合要求,并做必要的修改。

所谓 JEDEC 文件是一种由电子器件工程联合会制定的记录 PLD 编程数据的标准文件格式。一般的编程器都要求以这种文件格式输入编程数据。

第六步,下载。所谓下载,就是将 JEDEC 文件由计算机送给编程器,再由编程器将编程数据写入 PLD 中。

第七步,测试。将写好数据的 PLD 从编程器上取下,用实验方法测试它的逻辑功能,检查它是否达到了设计要求。

如果选用的是在系统可编程逻辑器件 ispPLD,则向 ispPLD 中下载数据不需要使用编程器。计算机根据用户编写的源程序运行开发系统软件,产生相应的编程数据和编程命令以后,直接通过五线编程接口将编程数据下载到 ispPLD 中,如图 8.9.1 所示。其中 $ispEN'$ 是编程使能信号, $ispEN' = 1$ 时 ispPLD 器件为正常工作状态; $ispEN' = 0$ 时所有 IOC 的输出三态缓冲器均被置成高阻态,并允

许器件进入编程工作状态。*MODE* 是模式控制信号。*SCLK* 是串行时钟输入,它为片内接受输入数据的移位寄存器以及控制编程操作的时序逻辑电路提供时钟信号。*SDI* 是串行数据和命令输入端,*SDO* 是串行数据输出端。

ispPLD 器件内部设有控制编程操作的时序逻辑电路,它的状态转换受 *MODE* 和 *SDI* 信号控制。计算机运行结果得到的编程数据和命令以串行方式从 *SDI* 送入 *ispPLD*。在写入数据的同时,又以串行方式将写入的数据从 *SDO* 读出并送回计算机,以便进行校验和发出下面的数据和命令。

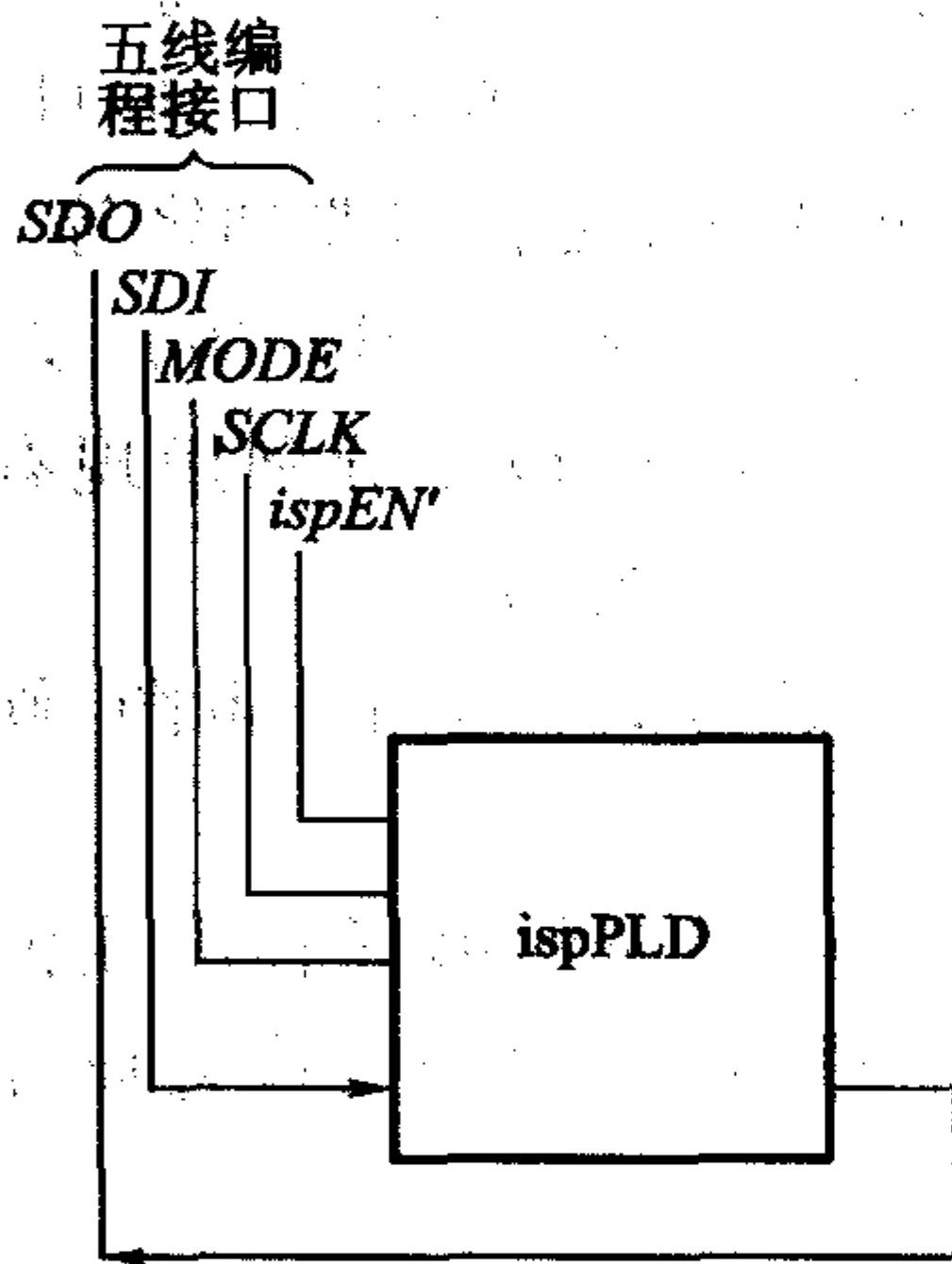


图 8.9.1 *ispPLD* 器件的编程接口

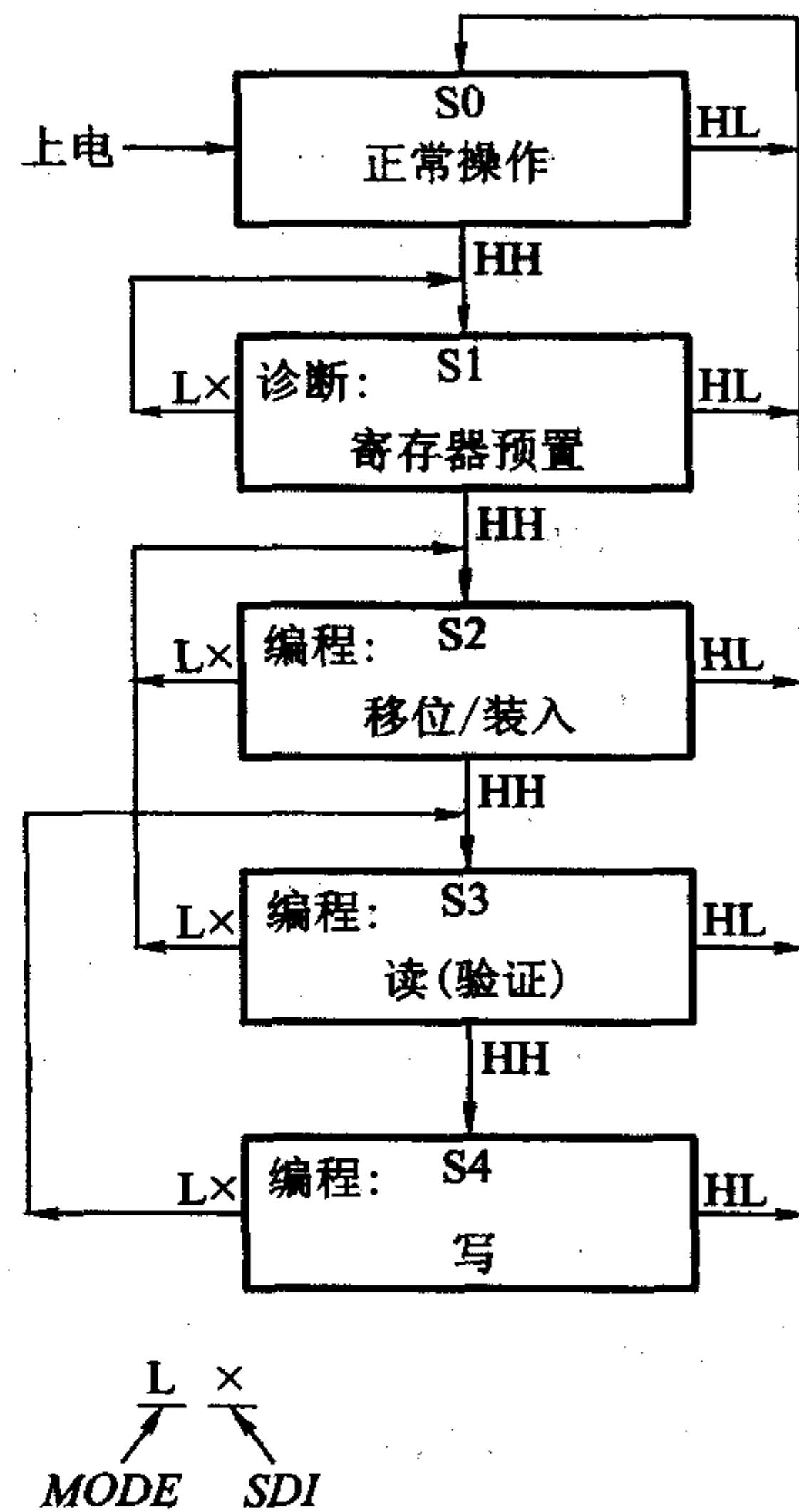


图 8.9.2 *ispPLD* 的编程操作流程

在编程过程中除 *MODE*、*SDI*、*SDO*、*SCLK* 以外的所有引脚均被置成高阻态,与外接电路隔离。工作状态的转换在内部的程序控制逻辑电路的指挥下自动完成。图 8.9.2 中给出了工作状态转换的流程图。

计算机的并行口和 *ispPLD* 之间的连接除了上述的 5 条信号线以外,还需要一条地线和一条对 *ispPLD* 所在系统电源电压的监测线,所以实际需要用 7 根连接线。

本章小结

PLD 是 20 世纪 80 年代以后迅速发展起来的一种新型半导体数字集成电路,它的最大特点是可以通过编程的方法设置其逻辑功能。本章的重点在于介绍各种 PLD 在电路结构和性能上的特点,以及它们都能用来实现哪些逻辑功能,适用在哪些场合。

到目前为止,已经开发出的 PLD 有 FPLA、PAL、GAL、EPLD、CPLD 以及 FPGA 等几种类型,并且广泛地采用了在系统可编程技术。

FPLA 和 PAL 是较早应用的两种 PLD。这两种器件多采用双极型、熔丝工艺或 UVC MOS 工艺制作,电路的基本结构是与 - 或逻辑阵列型。因为 FPLA 需要设置比 PAL 更多的熔丝,占用更大的硅片面积,所以目前已基本上被 PAL 所取代。虽然采用熔丝工艺的器件不能改写,采用 UVC MOS 工艺的擦除和改写也不甚方便,但由于采用这两种工艺制作的 PAL 可靠性好,成本也较低,所以在一些定型产品中仍然有使用价值。

GAL 是继 PAL 之后出现的一种 PLD,它采用 E^2 CMOS 工艺生产,可以用电信号擦除和改写。电路的基本结构形式仍为与 - 或阵列形式,但由于输出电路做成了可编程的 OLMC 结构,能设置成不同的输出电路结构,所以有较强的通用性。而且,用电信号擦除比用紫外线擦除要方便得多。

FPLA、PAL 和 GAL 的集成度都比较低,一般在千门以下,因此又将它们统称为低密度 PLD。集成度更高的所谓高密度 PLD 大体上可以分为 EPLD、CPLD 和 FPGA 三种类型。

EPLD 是采用 UVC MOS 工艺制作的高密度 PLD,集成度可达数千门。它的电路结构形式类似于 GAL,由若干个与 - 或阵列模块和一些 OLMC 组成,可以构成较大的数字系统。这种结构的优点是信号传输时间较短,而且是可预知的。

CPLD 和 FPGA 是集成度更高的两种可编程逻辑器件,两者在电路结构形式和工作方式上有所不同。CPLD 由若干个大的可编程逻辑模块、输入/输出模块和可编程的连线阵列组成。每个可编程逻辑模块类似于一个 PAL 或 GAL,相互间的连接比 EPLD 更灵活,而且传输延迟时间是确定的。此外,为方便用户使用,越来越多地都做成了在系统可编程器件。

FPGA 采用 CMOS - SRAM 工艺制作,电路结构为逻辑单元阵列形式。每个逻辑单元是可编程的,可以组成规模不大的组合或时序电路。单元之间可以灵活地互相连接,没有与 - 或阵列结构的局限性。但由于编程数据是存放在器件内部的静态随机存储器中的,一旦停电后这些编程数据便会丢失,所以每次开始工作时需要重新装载编程数据。此外,在将逻辑单元连接成复杂的系统时,不同

的信号传输途径传输延迟时间也不同,这也是设计时必须考虑的一个因素。

各种 PLD 的编程工作都需要在开发系统的支持下进行。开发系统的硬件部分由计算机(一般的 PC 机就可以)和编程器组成,软件部分是专用的编程语言和相应的编程软件。开发系统的种类很多,性能差别很大,各有一定的适用范围。因此,在选择 PLD 的具体型号时必须同时考虑到使用的开发系统能否支持这种型号 PLD 的编程工作。

对于在系统可编程逻辑器件(ispPLD),由于将编程控制电路和高压脉冲发生电路集成于 ispPLD 内部,所以编程时不需要使用编程器,并且可以在系统内完成,不用将器件从电路板上取下。ispPLD 的应用进一步提高了数字系统设计自动化的水平,同时也为系统的安装、调试、修改提供了更大的方便和灵活性。

习 题

[题 8.1] 试分析图 P8.1 的与-或逻辑阵列,写出 Y_1 、 Y_2 、 Y_3 与 A 、 B 、 C 、 D 之间的逻辑函数式。

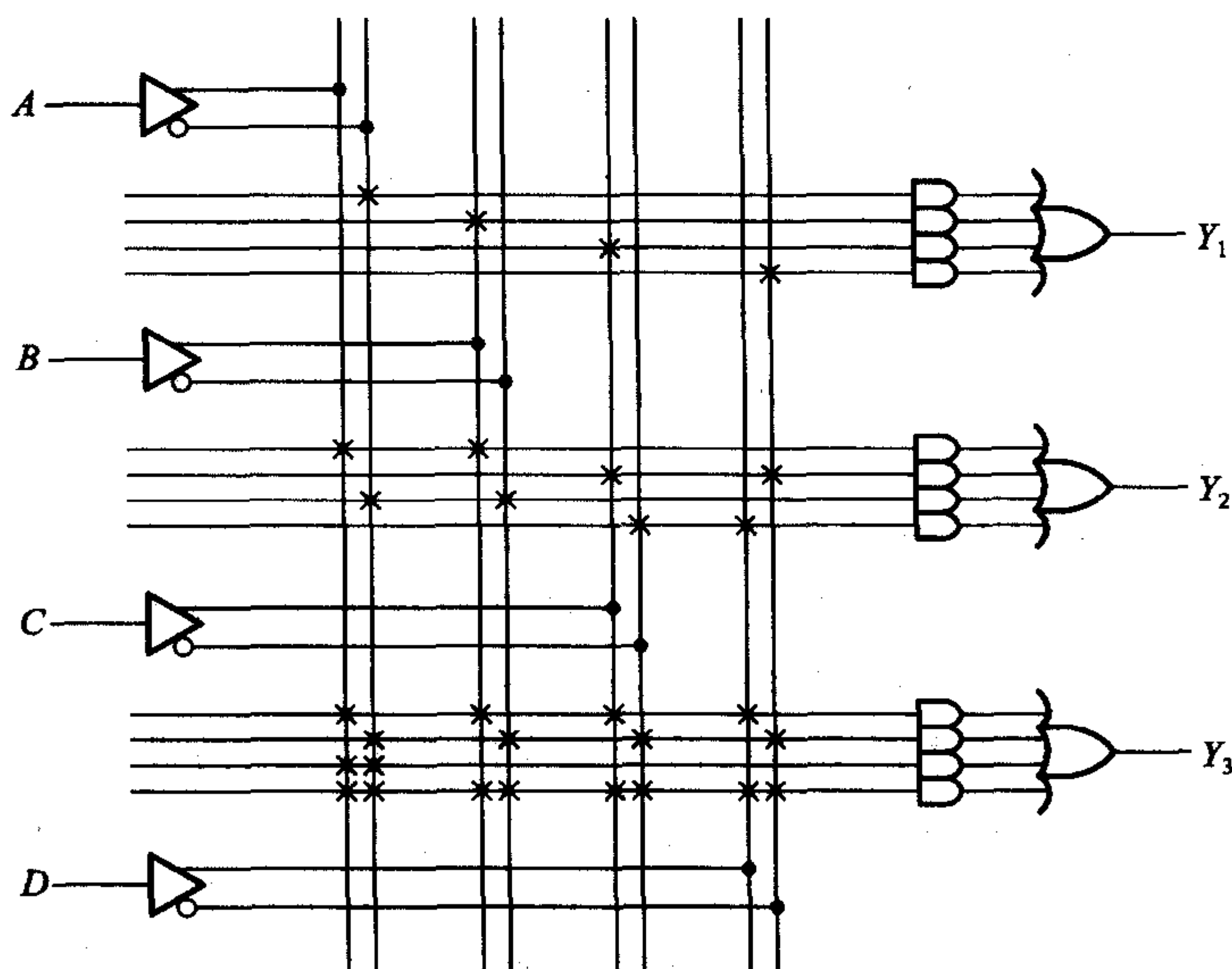


图 P8.1

[题 8.2] 试分析图 P8.2 的与-或逻辑阵列,写出 Y_1 、 Y_2 与 A 、 B 、 C 、 D 之间的逻辑关系式。

[题 8.3] 试分析图 P8.3 中由 PAL16L8 构成的逻辑电路,写出 Y_1 、 Y_2 、 Y_3 与 A 、 B 、 C 、 D 、 E 之间的逻辑关系式。

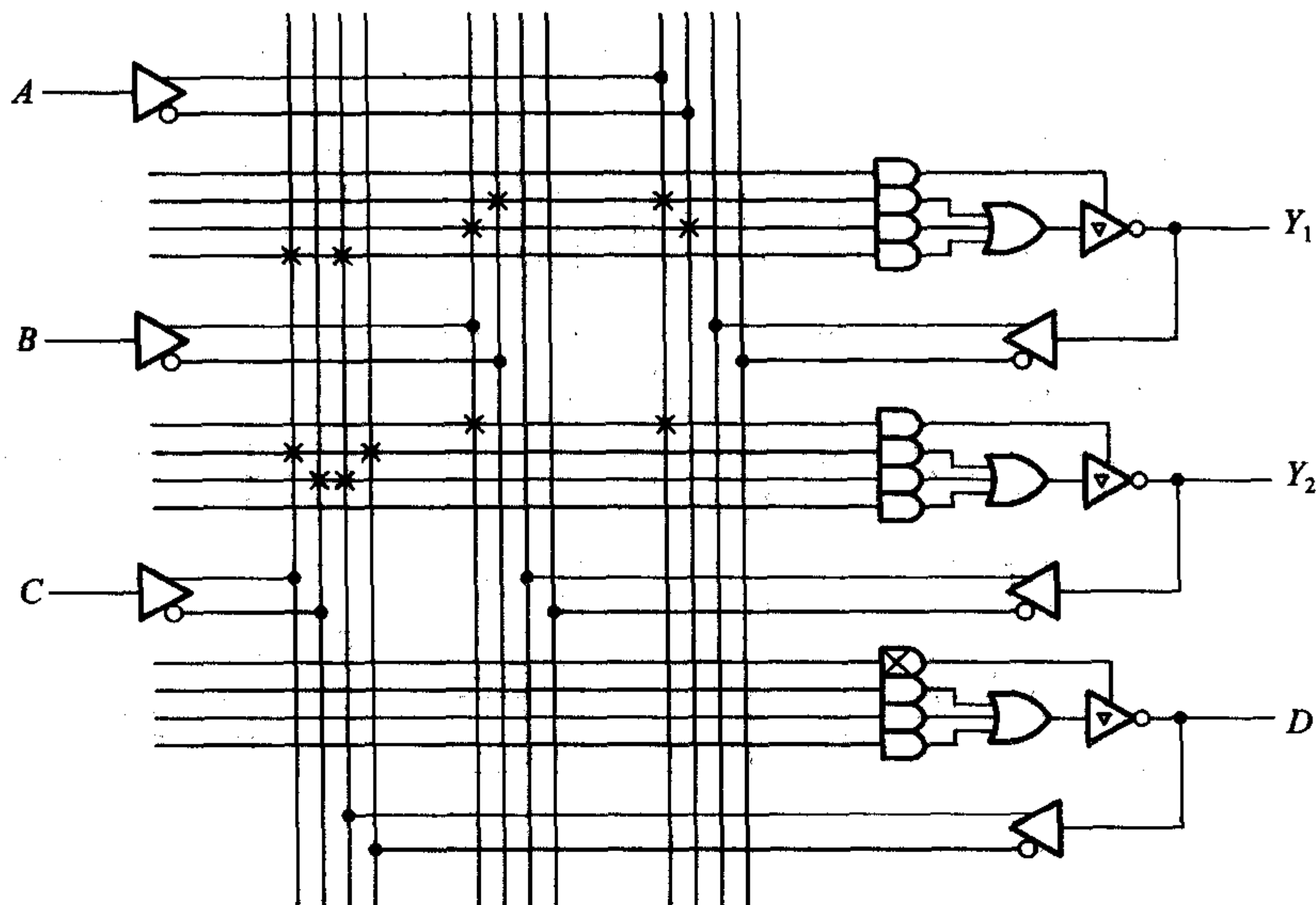


图 P8.2

[题 8.4] 用 PAL16L8 产生如下一组组合逻辑函数。

$$\begin{cases} Y_1 = (A'B'C'D' + A'B'CD + A'BC'D + A'BCD' + ABC'D' + ABCD + AB'C'D + AB'CD')' \\ Y_2 = A'BC'D' + A'BCD + ABC'D + ABCD' + AB'C'D' + AB'CD \\ Y_3 = (A'B'C'D' + ABCD)' \\ Y_4 = AB + AC \end{cases}$$

画出与-或逻辑阵列编程后的电路图。PAL16L8 的电路图见图 P8.3。

[题 8.5] 试分析图 P8.5 给出的用 PAL16R4 构成的时序逻辑电路,写出电路的驱动方程、状态方程、输出方程,画出电路的状态转换图。工作时,11 脚接低电平。

[题 8.6] 用 PAL16R4 设计一个 4 位二进制可控计数器。要求在控制信号 $M_1M_0 = 11$ 时做加法计数;在 $M_1M_0 = 10$ 时为预置数状态(时钟信号到达时将输入数据 $D_3D_2D_1D_0$ 并行置入 4 个触发器中); $M_1M_0 = 01$ 时为保持状态(时钟信号到达时所有的触发器保持状态不变); $M_1M_0 = 00$ 时为复位状态(时钟信号到达时所有的触发器同时被置 1)。此外,还应给出进位输出信号。PAL16R4 的电路图见图 P8.5。

[题 8.7] 试说明在下列应用场合下选用哪种类型的 PLD 最为合适。

1. 小批量定型产品中的中规模逻辑电路。
2. 产品研制过程中需要不断修改的中、小规模逻辑电路。
3. 少量的定型产品中需要的规模较大的逻辑电路。
4. 需要经常改变其逻辑功能的规模较大的逻辑电路。
5. 要求能以遥控方式改变其逻辑功能的逻辑电路。

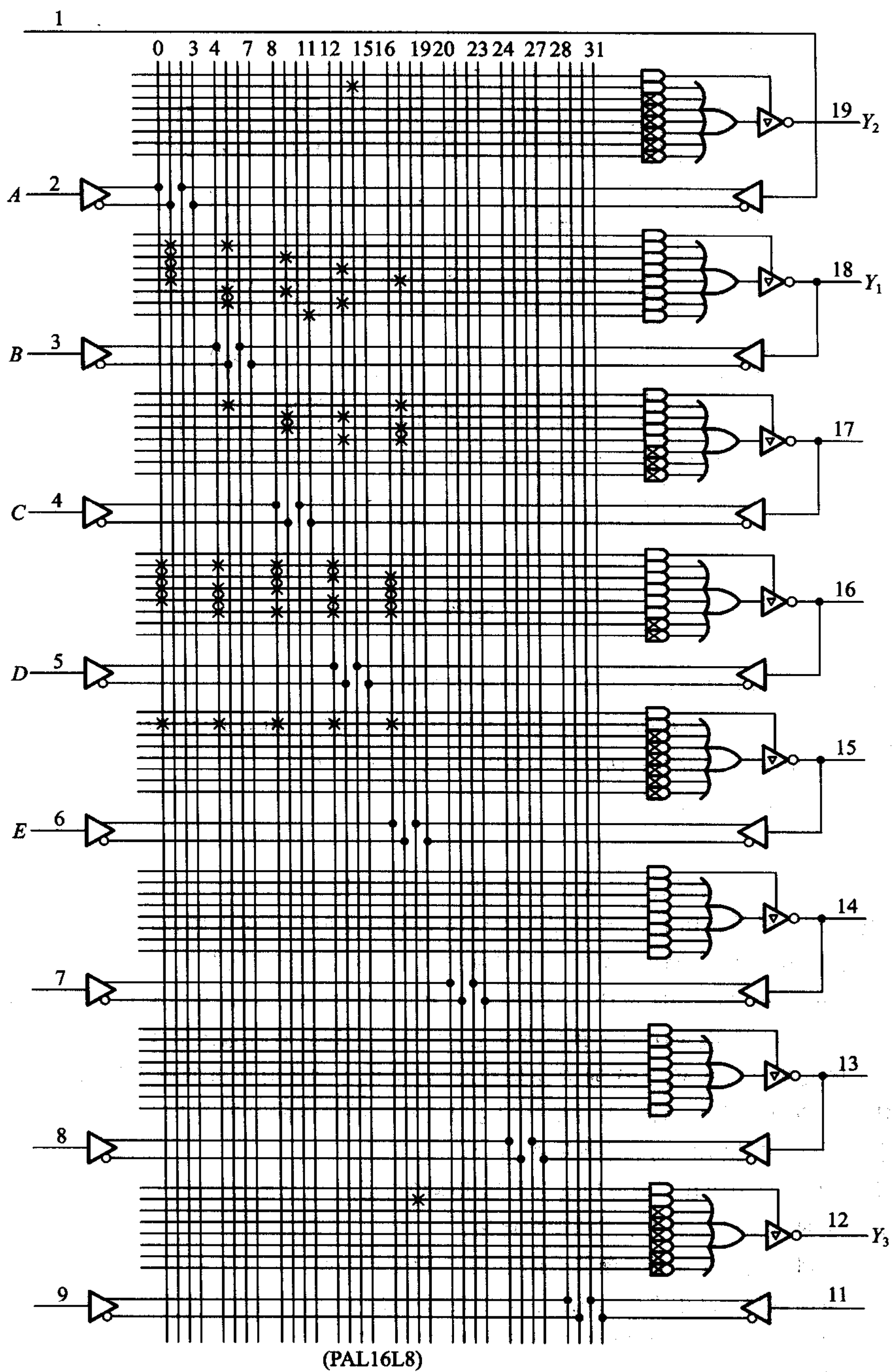


图 P8.3

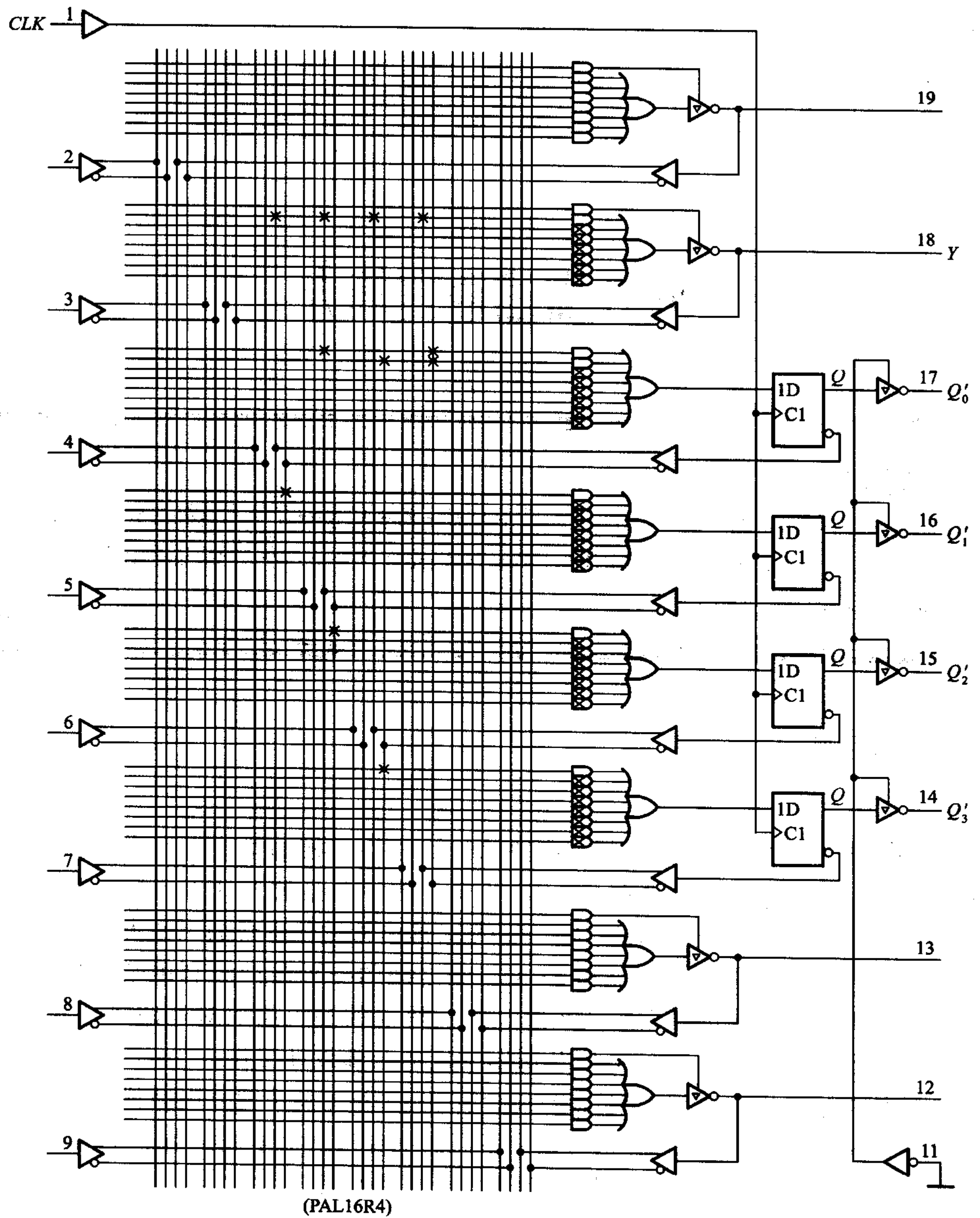


图 P8.5