Xilinx-FPGA/PLD: Structures

B-509：周五 5-6节

（网通/嵌入...）

沈沛意 pyshen@xidian.edu.cn

叶景涛、杨佳雯

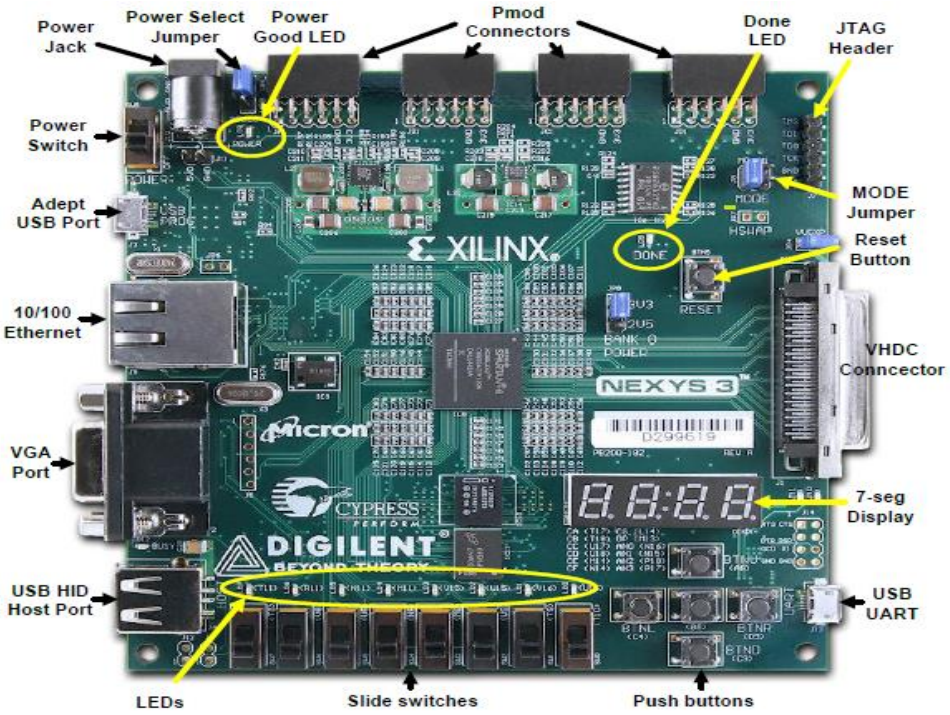
Books and refs:

- XILINX FPGA 设计基础-李云松、宋锐、雷杰、杜建超，西电出版社。
- 基于Xilinx FPGA的多核嵌入式系统设计基础，张亮、沈沛意、刘春红，西电出版社。
- FPGA上的嵌入式系统设计实例-赵峰、马迪铭、孙纬、梁天翼，西电出版社。
- Z. Navabi, *VHDL: Analysis and modeling of digital systems*, New York, McGraw-Hill。
- Internship-Prof. Kelvin Xie （中国XILINX-FPGA的教父）

Refs:

- http://www.altera.com
  https://www.altera.com/support/training/overview.html
- http://www.xilinx.com
- http://opencores.org
- **熟悉可编程逻辑器件设计流程，熟悉集成开发工具如QuartusII、ISE （最新代为QuartusPrime、Vivado）**

XUP Spartan 3E/6 System:



Digilent's Adept software offers a simplified programming interface and many additional features as described below. The Adept USB port is fully compatible with all Xilinx tools, including the iMPACT programming software. The Adept features are always available, regardless of how the FPGA was programmed.

- Key Features：
- On-board USB， Built-in self test Ethernet PHY
- VDEC board interface， Reference designs

XUP is more than donation

IEII(inspiration, enabling, innovation, incubation).

Inspiration:

Workshops, books, etc.

Enabling:

Joint lab, books, boards, etch

Innovation:

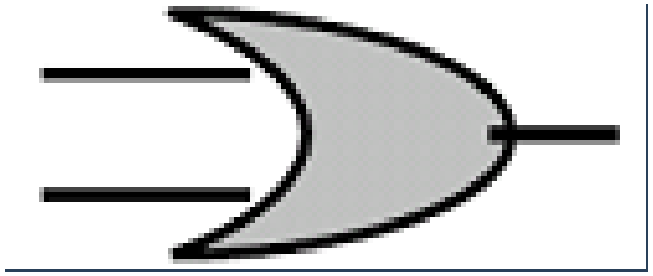
Open source hardware, contest, Open source IP

Incubation:

Joint project, etc.

Xilinx公司有很多集成的开发软件，主要包括以下几部分：

ISE	Xilinx公司集成开发环境
Foundation	Xilinx公司早期的开发工具，已经逐步被ISE取代
ISE Webpack	Xilinx提供的免费开发软件，功能比ISE少一些，可以从Xilinx网站下载
EDK	嵌入式开发套件，用于开发集成PowerPC硬核和Microblaze软核CPU的工具
System Generator	配合Matlab，快速在FPGA中实现数字信号处理功能
Xilinx IPCore	Xilinx公司预编译好的可配置功能模块，性能好，可视化用户界面

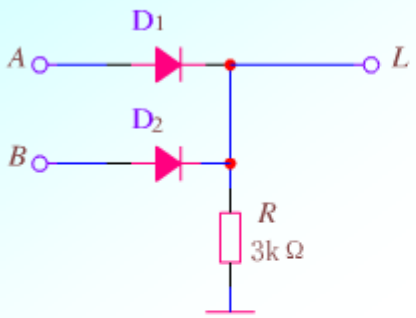


A=b+c

?

?

或门电路



输入、输出电压之间的关系

输入		输出
V_A (V)	V_B (V)	V_L (V)
0V	0V	0V
0V	5V	5V
5V	0V	5V
5V	5V	5V

或逻辑真值表

输入		输出
A	*B*	*L*
0	0	0
0	1	1
1	0	1
1	1	1

Just one statement?

```
Library ieee
Use ieee.std_logic_1164.all;
Use ieee.std_logic_arith.all;
Use ieee.std_logic_unsigned.all;

Entity device is
   port( b: in std_logic; c: in std_logic; a: out std_logic);
End device

Architecture behavioral of device is
A<=b or c;
End behavioral
```

A=b+c

Just one statement?

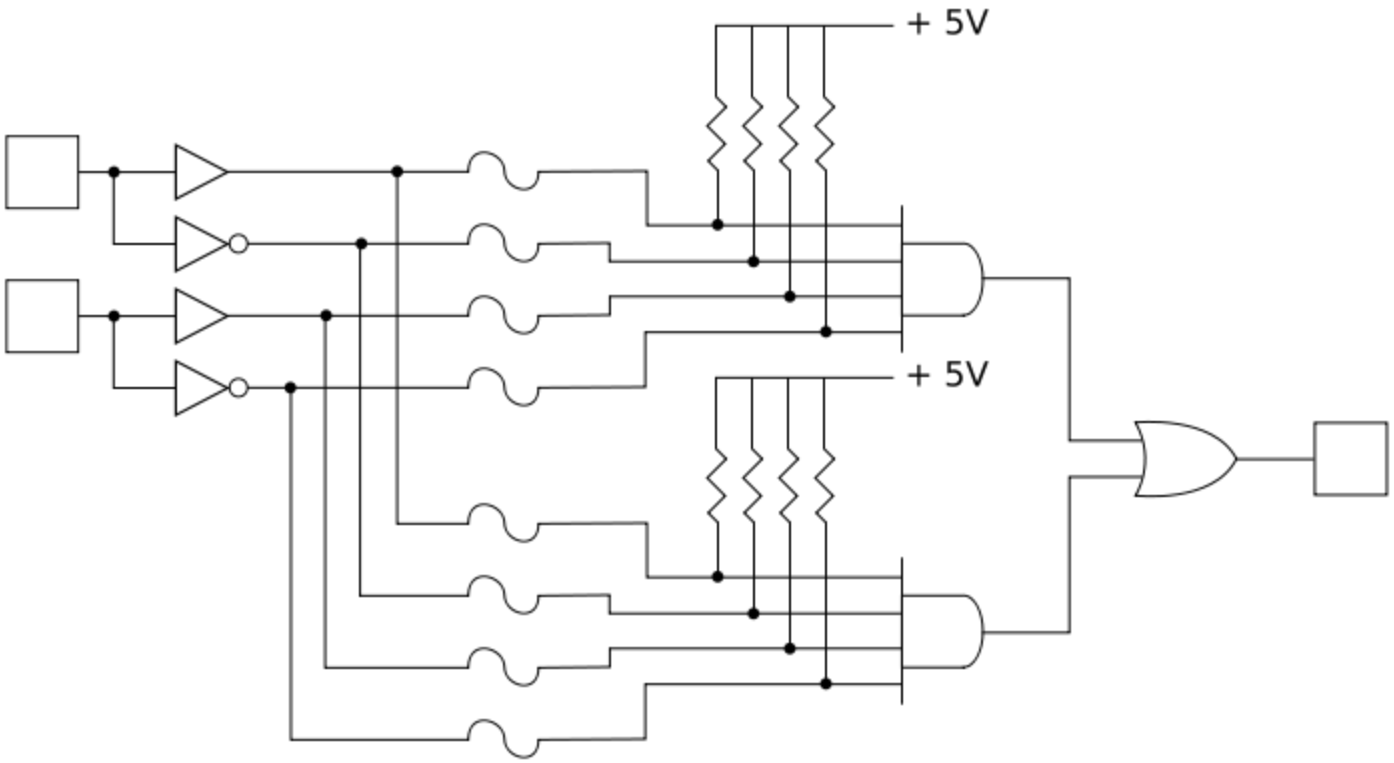
```
Library ieee
Use ieee.std_logic_1164.all;
Use ieee.std_logic_arith.all;
Use ieee.std_logic_unsigned.all;

Entity device is
   port( b: in std_logic; c: in std_logic; a: out std_logic);
End device

Architecture behavioral of device is
b<=1;
c<=0;
A<=b or c;
End behavioral
```

A=b+c

一 可编程功能介绍-FPGA发展历史

- ROM(read only memory)
- PLD(programmable logic device)可编程器件

- 早期的可编程逻辑器件有可编程只读存储器（PROM）、紫外线可擦除只读存储器（EPROM）和电可擦除只读存储器（EEPROM）。这类器件由固定的“与”门阵列和可编程的“或”门阵列构成。PROM采用熔丝工艺编程，电路状态只能改变一次，因此一旦写入内容，就不能擦除和重写。
- 可编程阵列逻辑PAL（Programmable Array Logic）于20世纪70年代出现，它由可编程的“与”门阵列和固定的“或”门阵列构成。PAL采用熔丝编程方式，双极性工艺制造，器件的工作速度很高。PAL是第一个得到普遍应用的可编程逻辑器件。在PAL的基础上，又发展了一种通用阵列逻辑GAL (Generic Array Logic)，如GAL16V8，GAL22V10 等。它采用了EEPROM工艺，实现了电可擦除、电可改写，其输出结构是可编程的逻辑宏单元，因而它的设计具有很强的灵活性，至今仍有许多人使用。



Simplified programmable logic device

A simplified PAL device. The programmable elements (shown as a fuse) connect both the true and complemented inputs to the AND gates. These AND gates, also known as product terms, are ORed together to form a sum-of-products logic array.

Early programmable logic

In 1969, Motorola offered the XC157, a mask-programmed gate array with 12 gates and 30 uncommitted input/output pins.[1]

In 1970, Texas Instruments developed a mask-programmable IC based on the IBM read-only associative memory or ROAM. This device, the TMS2000, had up to 17 inputs and 18 outputs with 8 JK flip flop for memory. TI coined the term Programmable Logic Array for this device.[2]

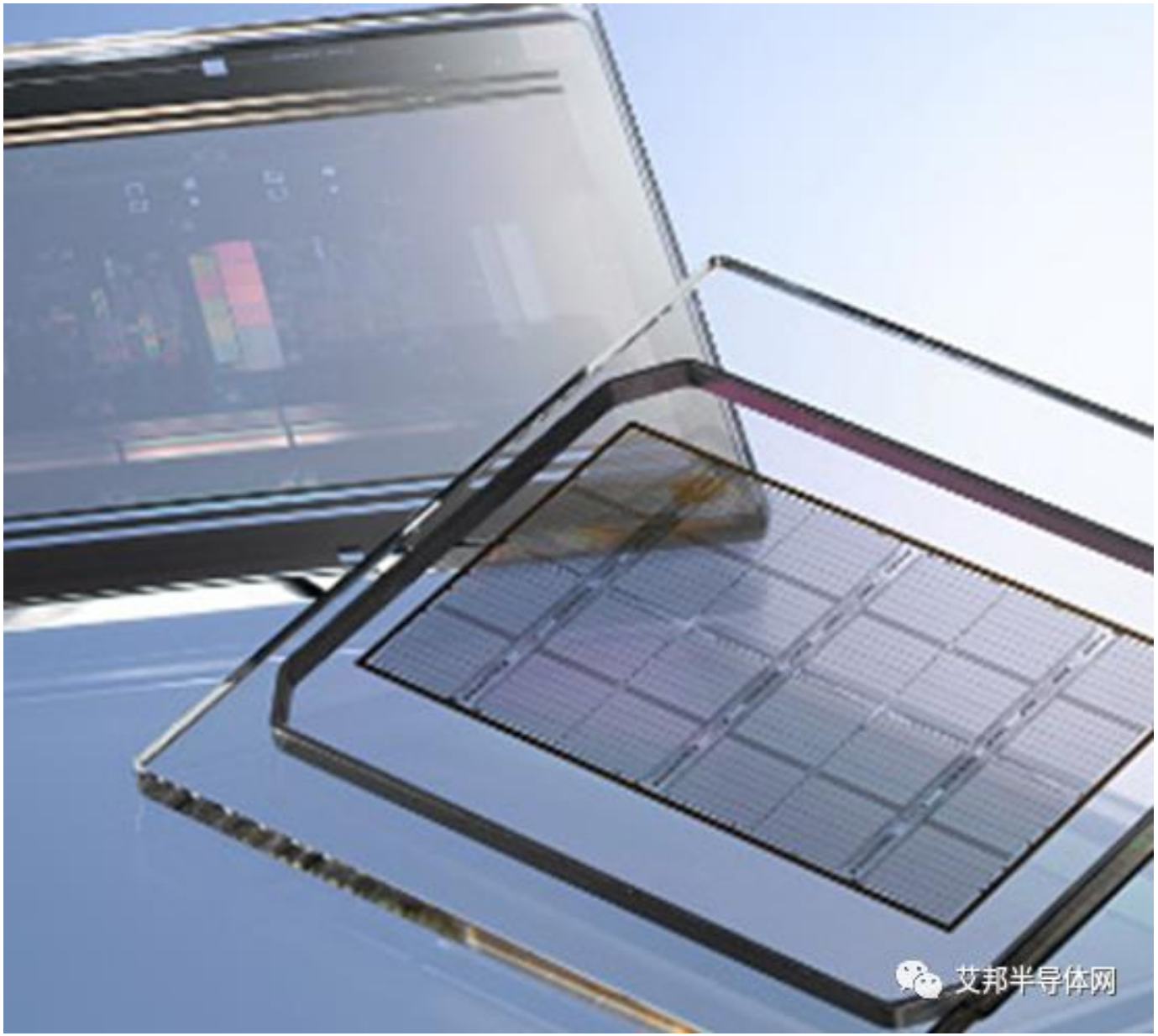
In 1971, General Electric Company (GE) was developing a programmable logic device based on the new PROM technology. Intel had just introduced the floating-gate UV erasable PROM so the researcher at GE incorporated that technology.

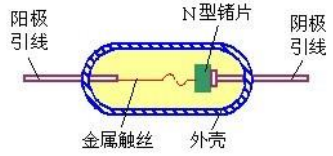
The GE device was the first erasable PLD ever developed, predating the Altera EPLD by over a decade. GE obtained several early patents on programmable logic devices.[3][4][5]

In 1973 National Semiconductor introduced a mask-programmable PLA device (DM7575) with 14 inputs and 8 outputs with no memory registers

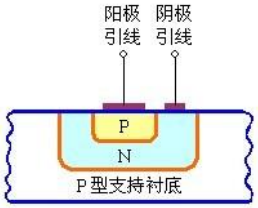
In 1974 GE entered into an agreement with Monolithic Memories to develop a mask- programmable logic device incorporating the GE innovations. The device was named the 'Programmable Associative Logic Array' or PALA. The MMI 5760 was completed in 1976 and could implement multilevel or sequential circuits of over 100 gates. The device was supported by a GE design environment where Boolean equations would be converted to mask patterns for configuring the device. The part was never brought to market.[8]

Deepseek: mask-programmed?

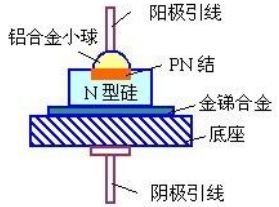

艾邦半导体网



(a)

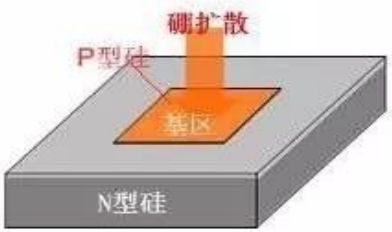


(c)



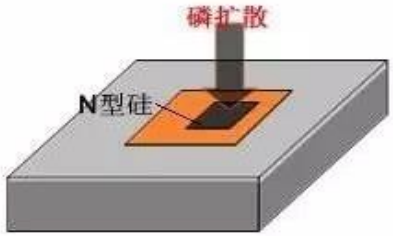
(b)

(d)



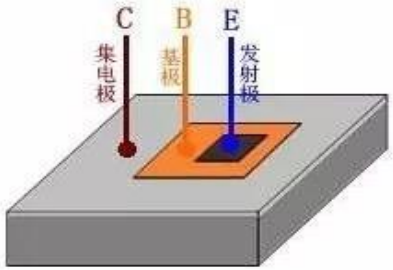
硼扩散制作基区

1100℃氧气氛中硼扩散，同时生成氧化层SiO2



磷扩散制作发射区

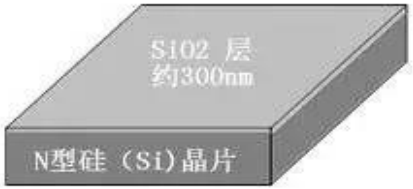
在掺硼基区上进行发射区上通过光刻-曝光-刻蚀-磷扩散(950℃)，制作形成高浓度的N型硅发射区。



引出电极

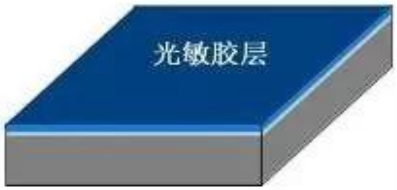
集电区—集电极C
基区—集电极B
发射区—发射极E

NPN硅平面型三极管制造流程



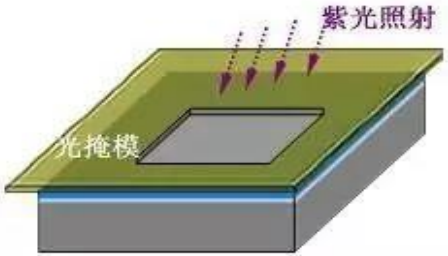
硅基片表面氧化

清洗硅基片，在1200℃氧气氛热处理形成氧化硅层



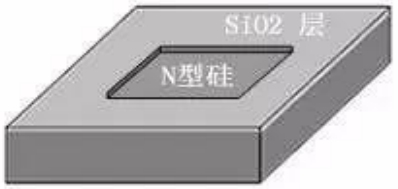
涂布光敏胶

光敏胶涂覆在硅基片表面



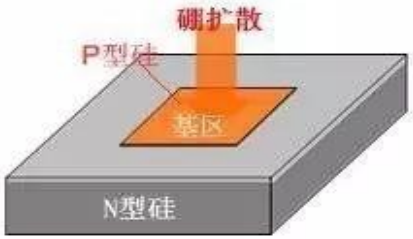
光刻

紫光照射，无光掩模遮挡区域光敏胶曝光后通过显影定影工艺被清除，露出SiO2层。



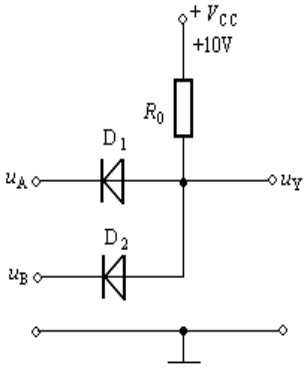
刻蚀出基区窗口

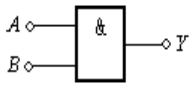
用刻蚀液将无光刻胶膜覆盖的氧化层腐蚀掉，有光刻胶覆盖的区域保留下来。

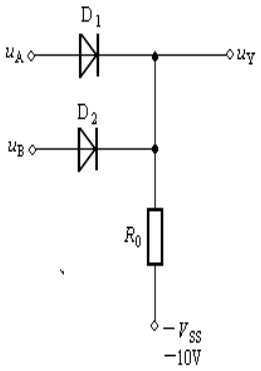


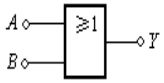
硼扩散制作基区

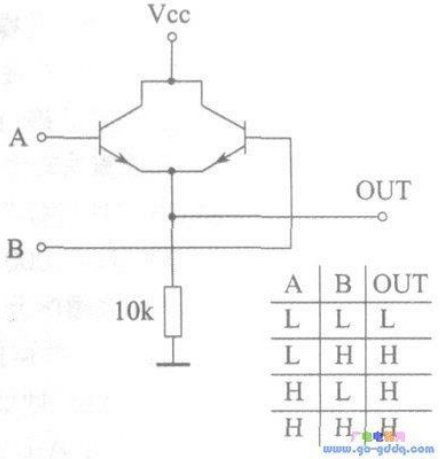
1100℃氧气氛中硼扩散，同时生成氧化层SiO2

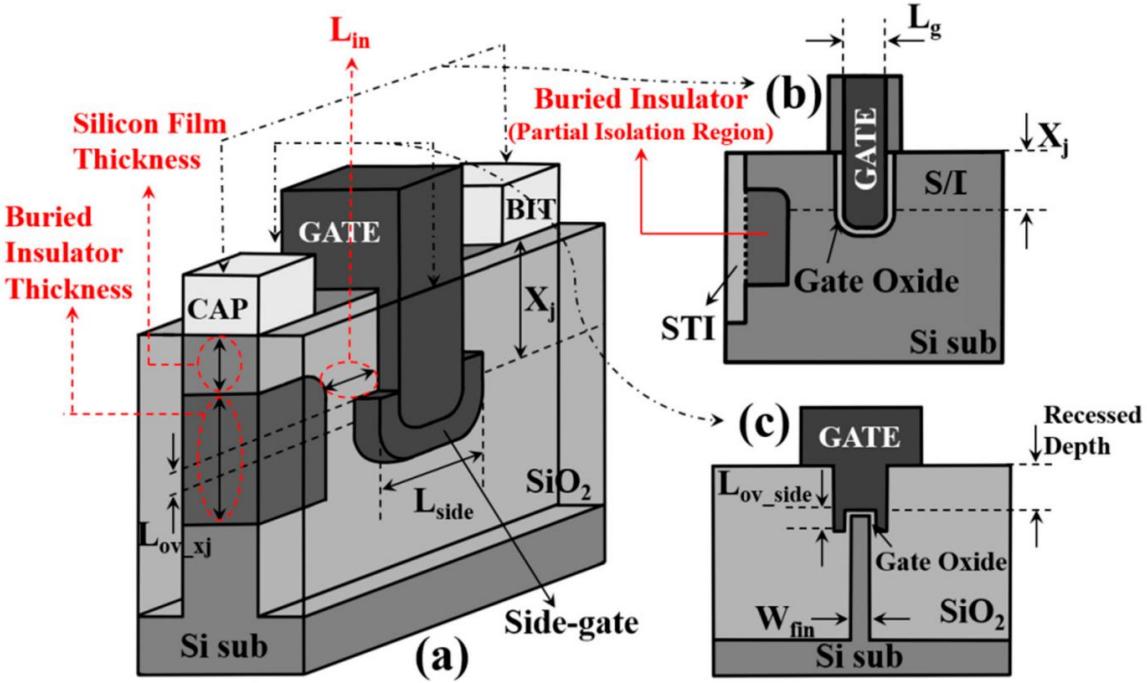


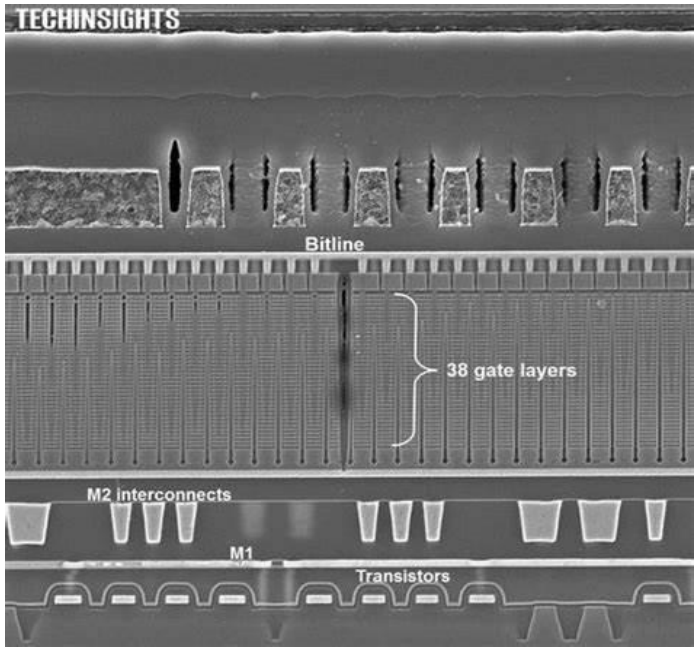






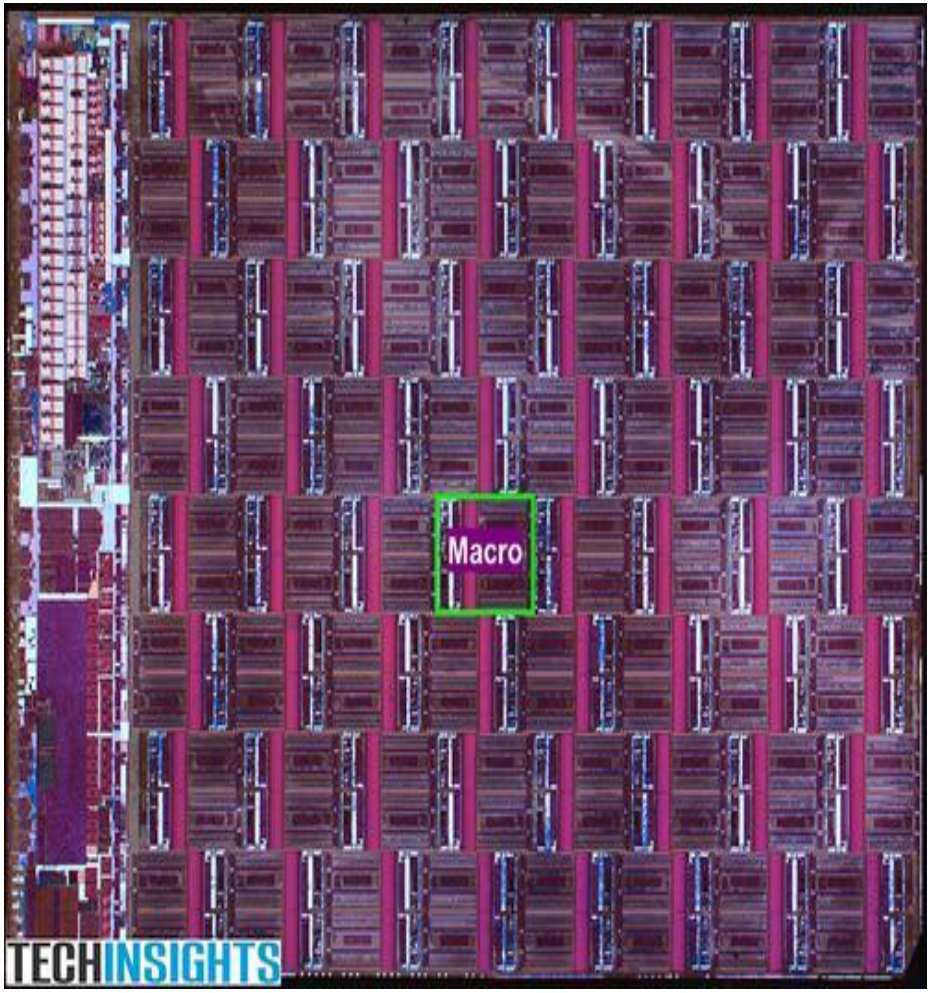




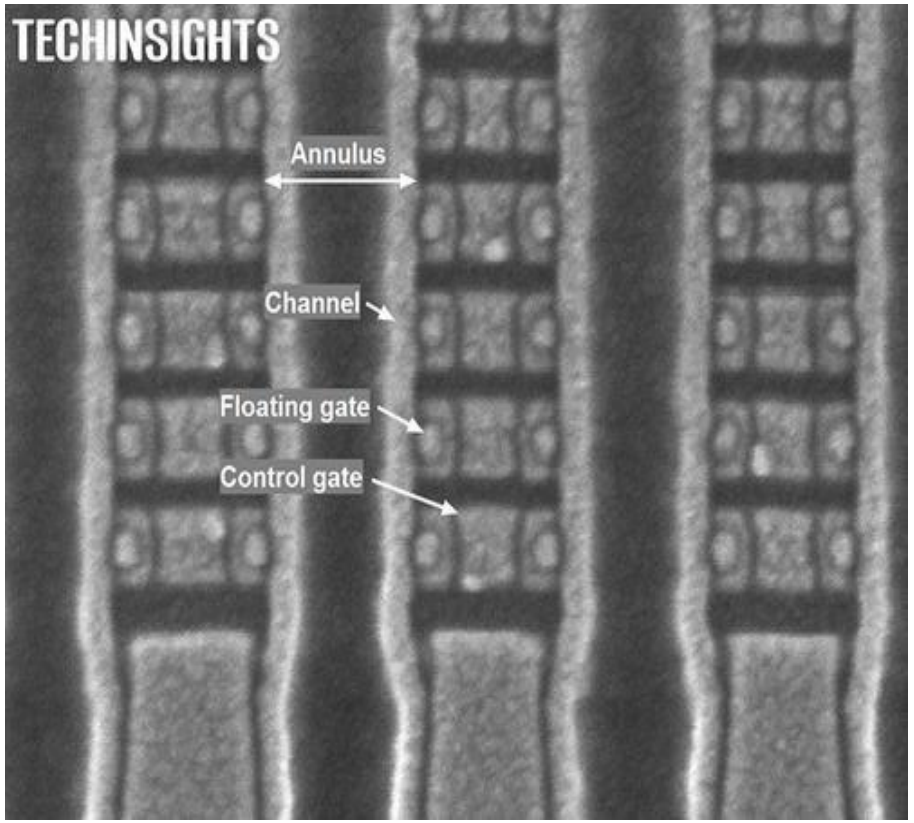


Micron stakes business on 3D NAND with new fab process

https://www.eetindia.co.in/news/article/micron-stakes-business-on-3d-nand-with-new-fab-process

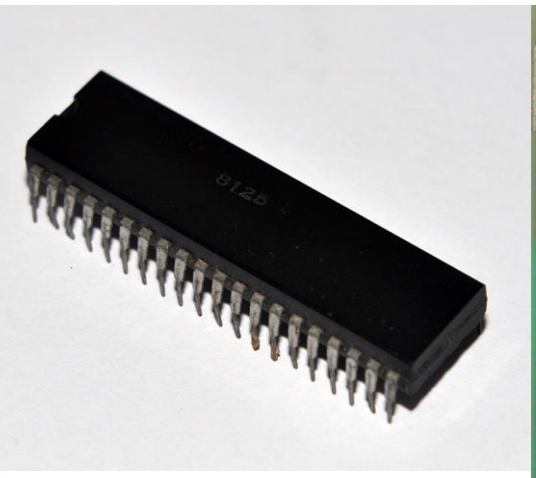


3D NAND diffusion level die photograph

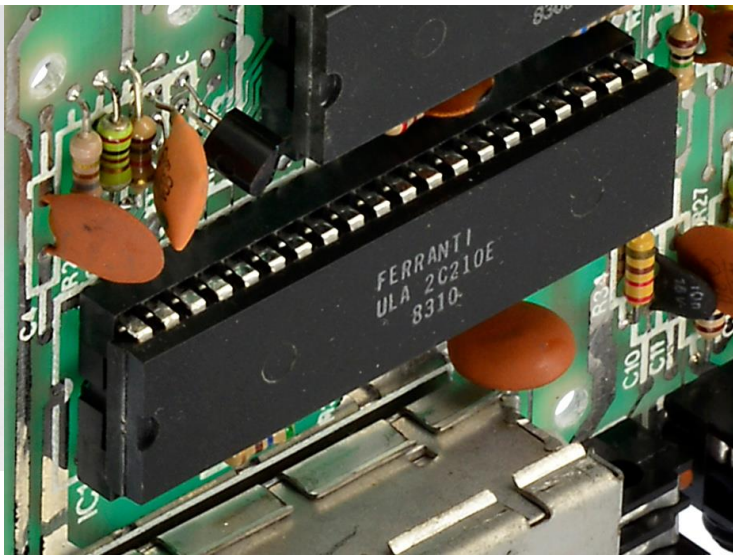


cross section of Micron's NAND cells

Gate Array?



Sinclair ZX81 ULA



Ferranti ULA 2C210E on a Timex Sinclair 1000 motherboard

- 开始时...TTL逻辑设计
- 用分立的芯片实现基本逻辑功能
  - NAND、OR、复用器、触发器等
- 著名的德州仪器74系列
- 通常由成本和可用的器件来决定设计选择

HomeCoputers:
Units sold： More than 1.5 million

摩尔定律：价格不变，性能提升一倍，密度增加一倍（18-24个月）？

Home computers?

Children playing *Paperboy* on an Amstrad CPC 464 in 1988

Most home computers, such as this Tandy Color Computer 3, featured a version of the BASIC programming language. The sometimes-sprawling nature of the well-outfitted home computer system is very much in evidence.

Computer at home, USA 1965

A gate array is an approach to the design and manufacture of application-specific integrated circuits (ASICs) using a prefabricated chip with components that are later interconnected into logic devices (e.g. NAND gates, flip-flops,etc.) according to a custom order by adding metal interconnect layers in the factory.

Similar technologies have also been employed to design and manufacture analog, analog-digital, and structured arrays, but, in general, these are not called gate arrays.
Gate arrays have also been known as Uncommitted Logic Arrays (ULAs) and semi-custom chips.

By the early 1980s gate arrays were starting to move out of their niche applications to the general market.

Home computers?

Children playing *Paperboy* on an Amstrad CPC 464 in 1988

Most home computers, such as this Tandy Color Computer 3, featured a version of the BASIC programming language. The sometimes-sprawling nature of the well-outfitted home computer system is very much in evidence.

Computer at home, USA 1965

A gate array is an approach to the design and manufacture of application-specific integrated circuits (ASICs) using a prefabricated chip with components that are later interconnected into logic devices (e.g. NAND gates, flip-flops,etc.) according to a custom order by adding metal interconnect layers in the factory.

Indirect competition arose with the development of the field-programmable gate array (FPGA). Xilinx was founded in 1984 and its first products were much like early gate arrays, slow and expensive, fit only for some niche markets. However, Moore's Law quickly made them a force and by the early 90's were seriously disrupting the gate array market. WHY?

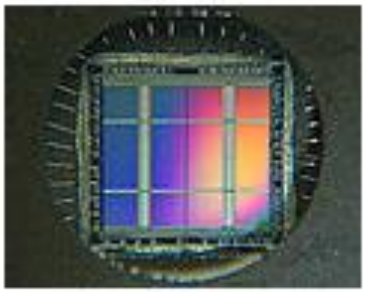
By the early 1980s gate arrays were starting to move out of their niche applications to the general market.

二、PROM特性-FPGA发展历史

- 早期的可编程器件，包括：
  - 可编程只读存储器PROM（Programmable Read Only Memory）；特点在于：
    - 一是只能进行一次编程
    - 二是空白PROM价格低廉，高成本ROM制作之前的数据原型测试的最佳选择
  - 紫外线可擦除只读存储器EPROM（Erasable Programmable Read Only Memory）；特点在于：
    - 可进行多次编程
    - 只读，EPROM芯片，命名时一般以27开头，前面添加生产厂商，如TMS27C512-60

二、PROM特性-FPGA发展历史

- 电可擦除只读存储器EEPROM（Electrically Erasable Programmable Read Only Memory）；特点在于：
  - 一是可多次编程、二是可在线编程，常用操作接口方式I2C,SPI接口



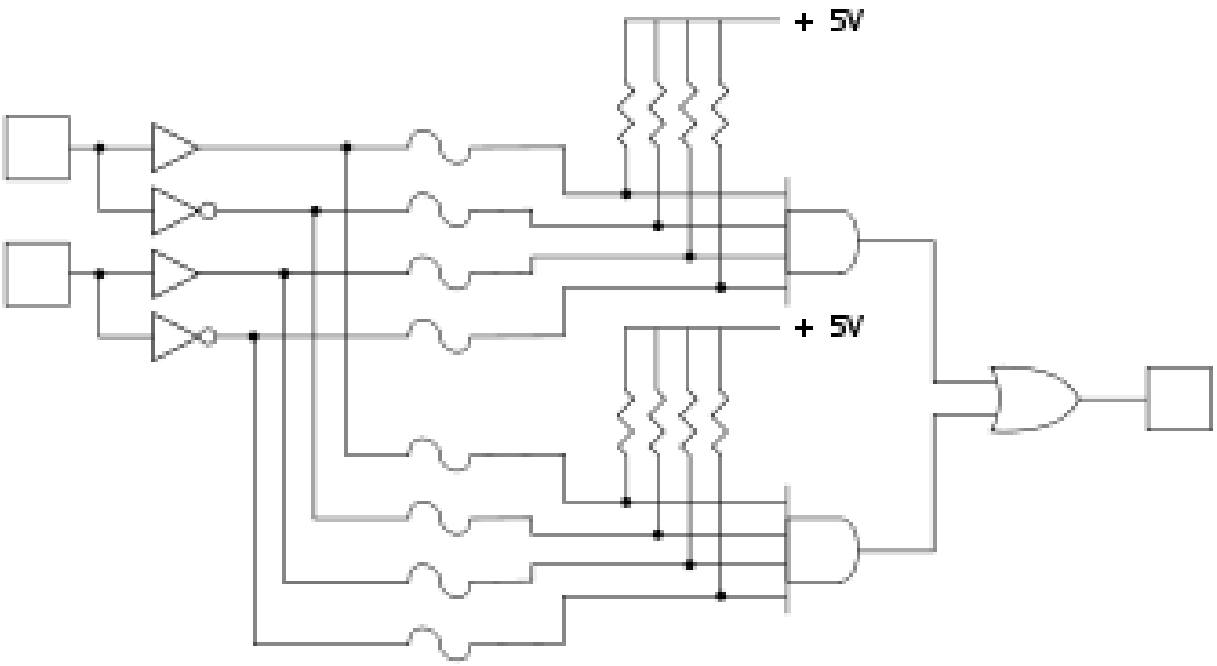
由擦除窗口看 EPROM 晶粒

32KB (256Kbit) EPROM 27C256

EPROM 晶粒的 60倍近照

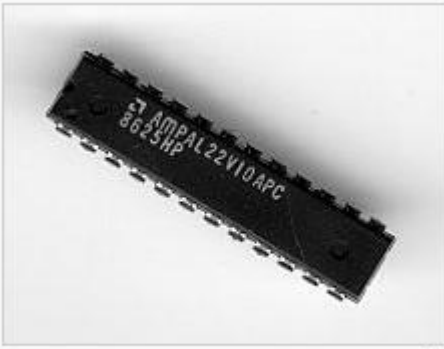
以 EPROM 来存储程序的 8749 单片机

PAL



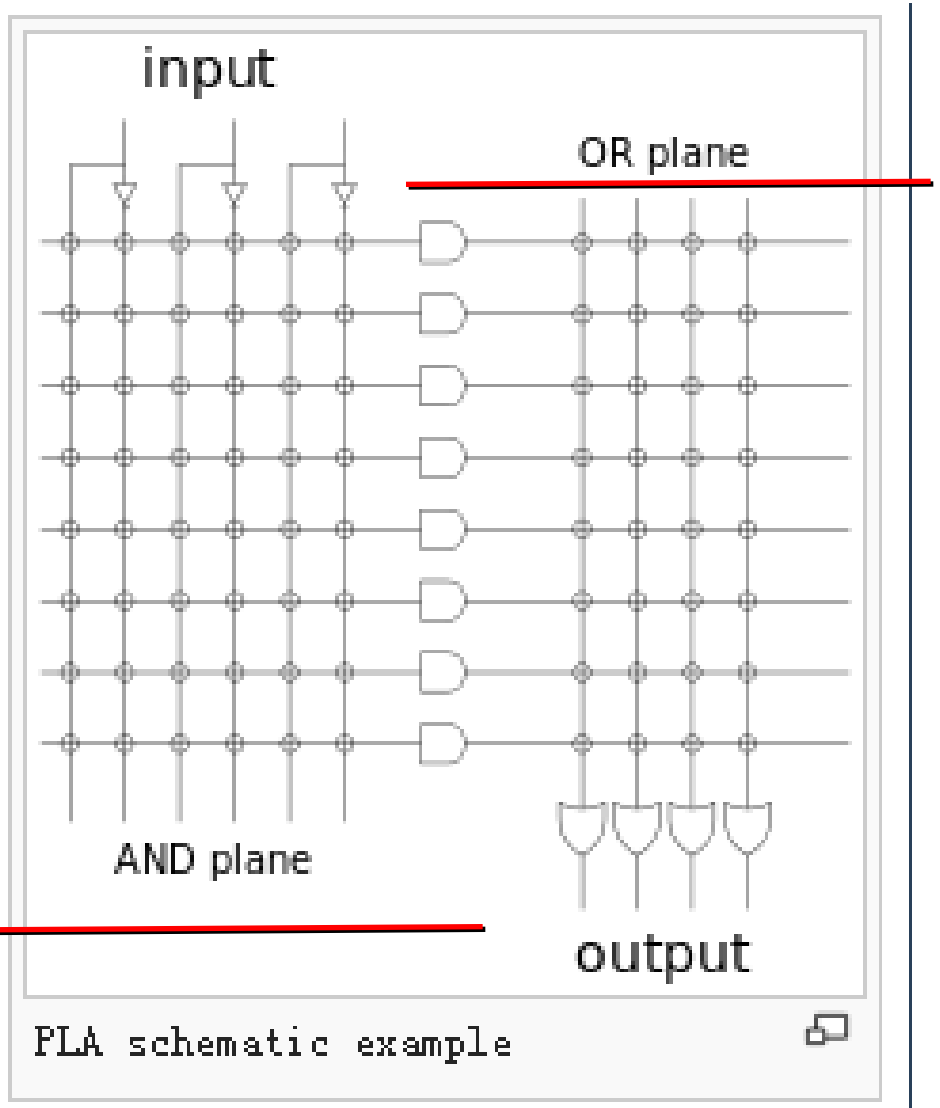
Simplified programmable logic device

The programmable elements (shown as a fuse) connect both the true and complemented inputs to the AND gates. These AND gates, also known as *product terms*, are ORed together to form a *sum-of-products* logic array.

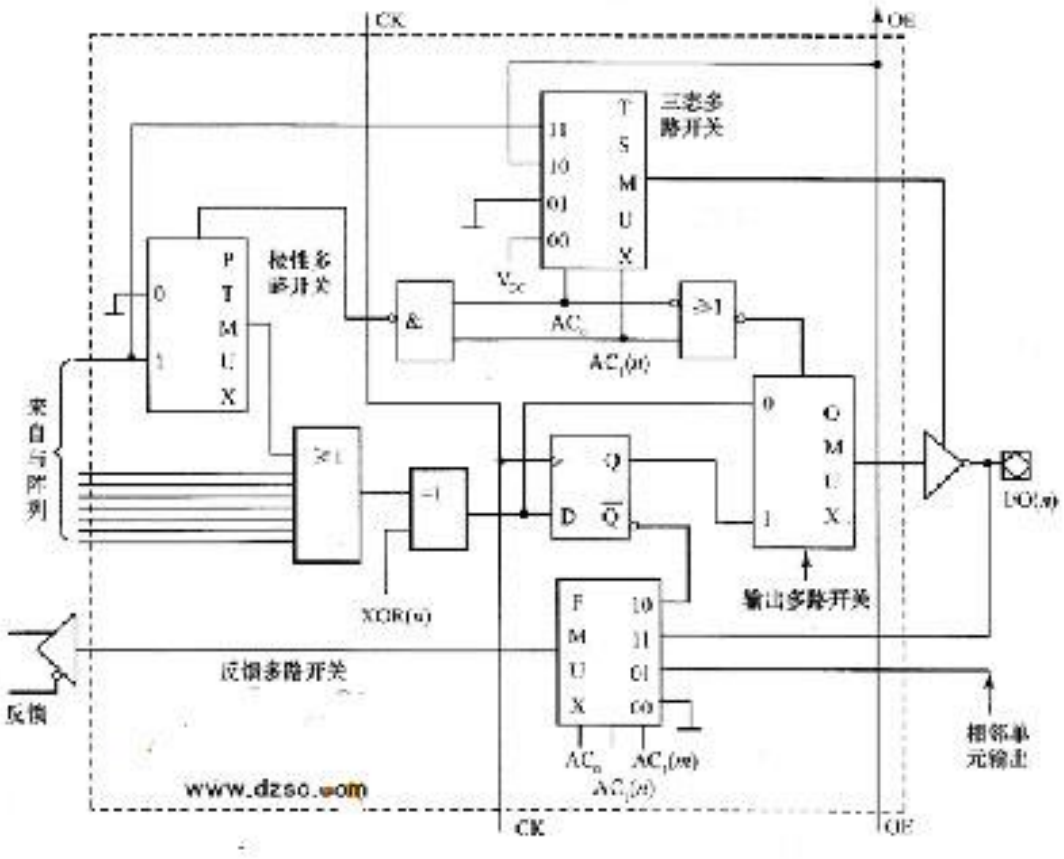


AMD 22V10 in 24-pin DIP

PLA



GAL



Output Logic Micro Cell

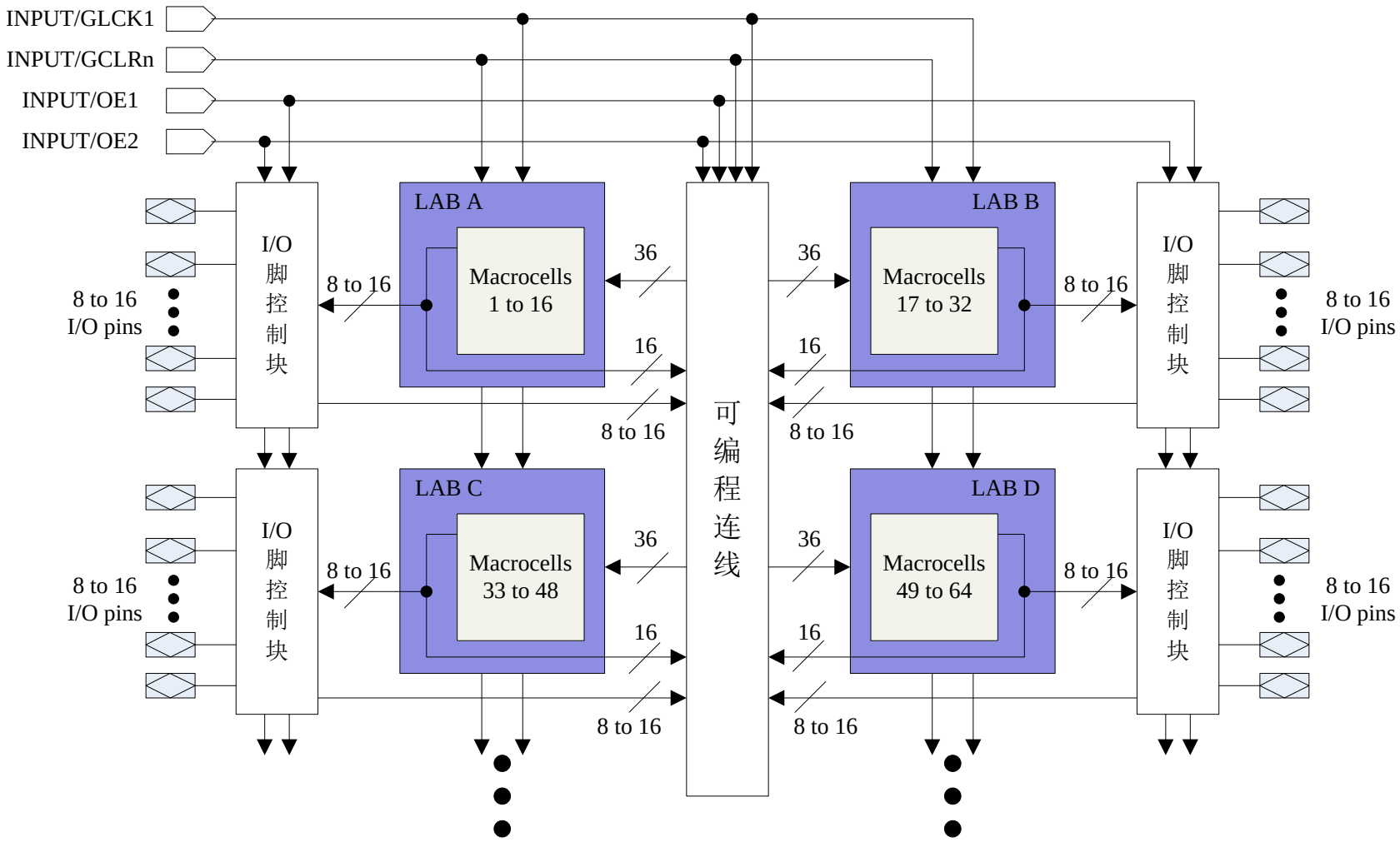
Lattice Semiconductor

Type	Public (NASDAQ: LSCC)
Industry	Integrated Circuits
Founded	1983, public since 1989
Headquarters	Hillsboro, Oregon, United States 45.527216° N 122.926626° W
Key people	Darin Billerbeck, CEO
Products	FPGAs, CPLDs
Revenue	$332.5 million (2013)[1]
Net income	▲ $22.3 million (2013)[1]
Number of employees	700 (2011)[2]
Website	www.latticesemi.com

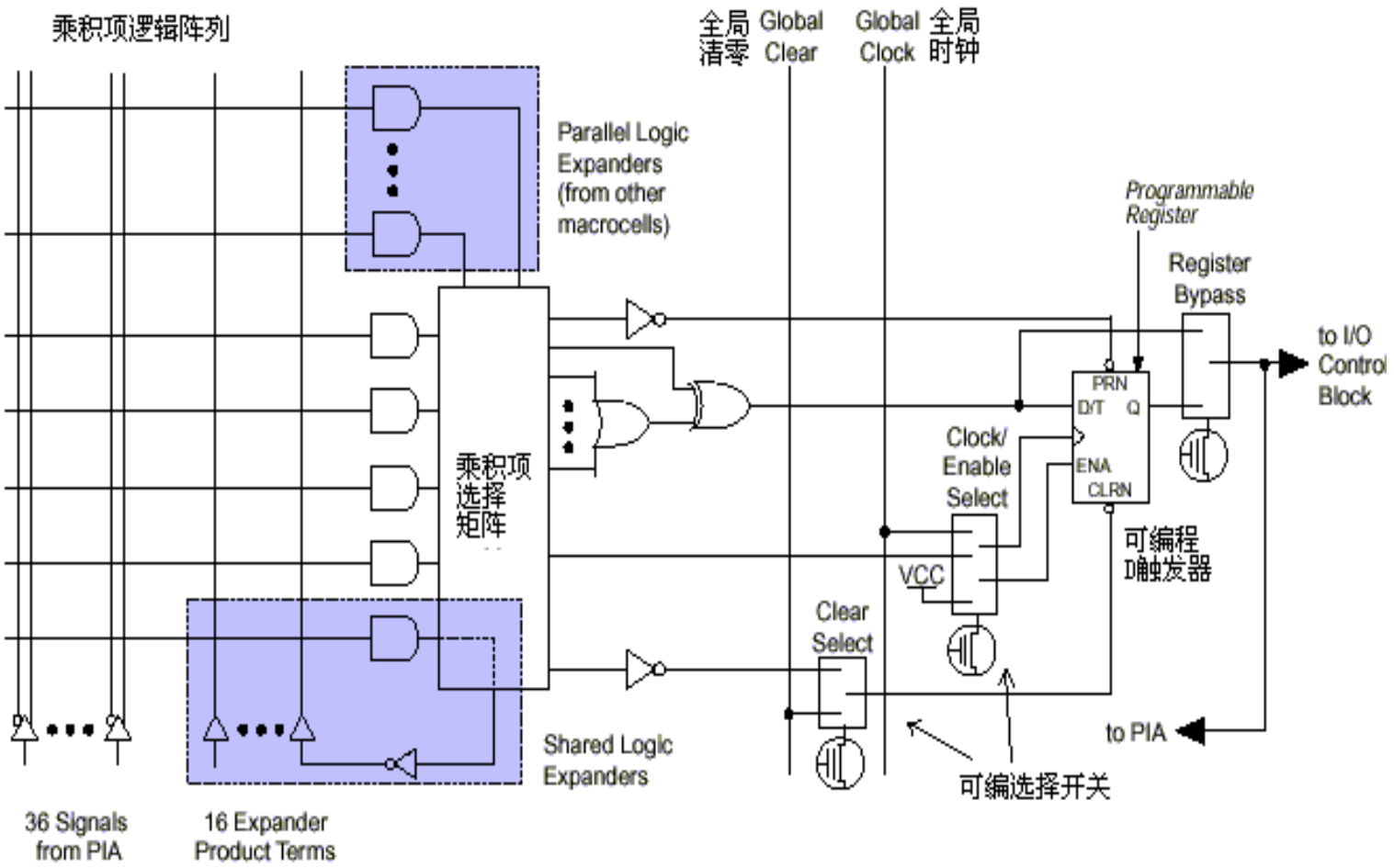
Why CPLD ?

- A **complex programmabl**
  is a programmable logic de
  between that of PALs and
  architectural features of bo
  a CPLD is the macrocell, w
  implementing disjunctive normal form expressions
  and more specialized logic operations.

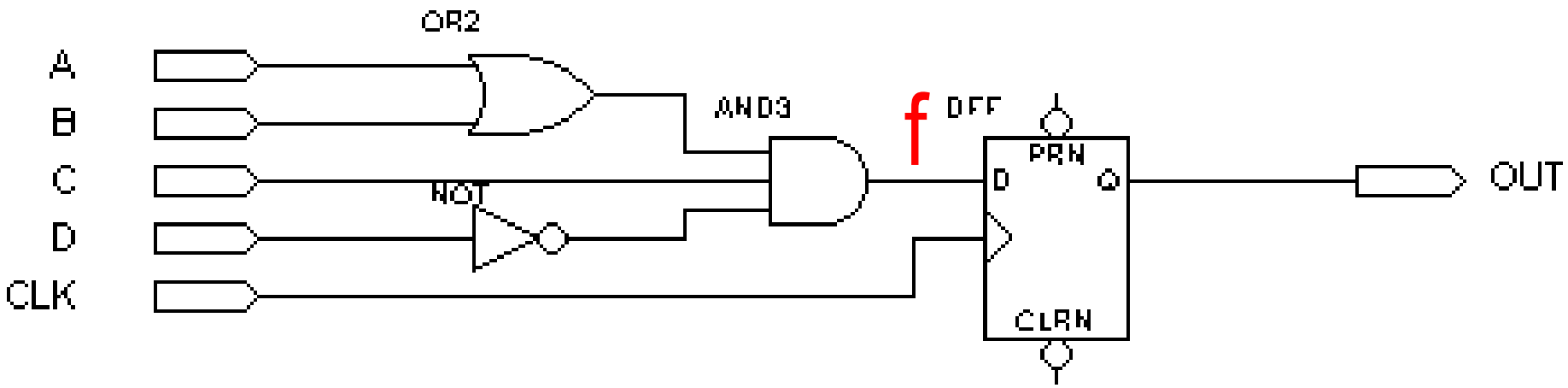
三、CPLD的基本结构


INPUT/GLCK1
INPUT/GCLRn
INPUT/OE1
INPUT/OE2
LAB A
LAB B
LAB C
LAB D
Macrocells 1 to 16
Macrocells 17 to 32
Macrocells 33 to 48
Macrocells 49 to 64
I/O 脚控制块
可编程连线
36
16
8 to 16
8 to 16 I/O pins

宏单元结构

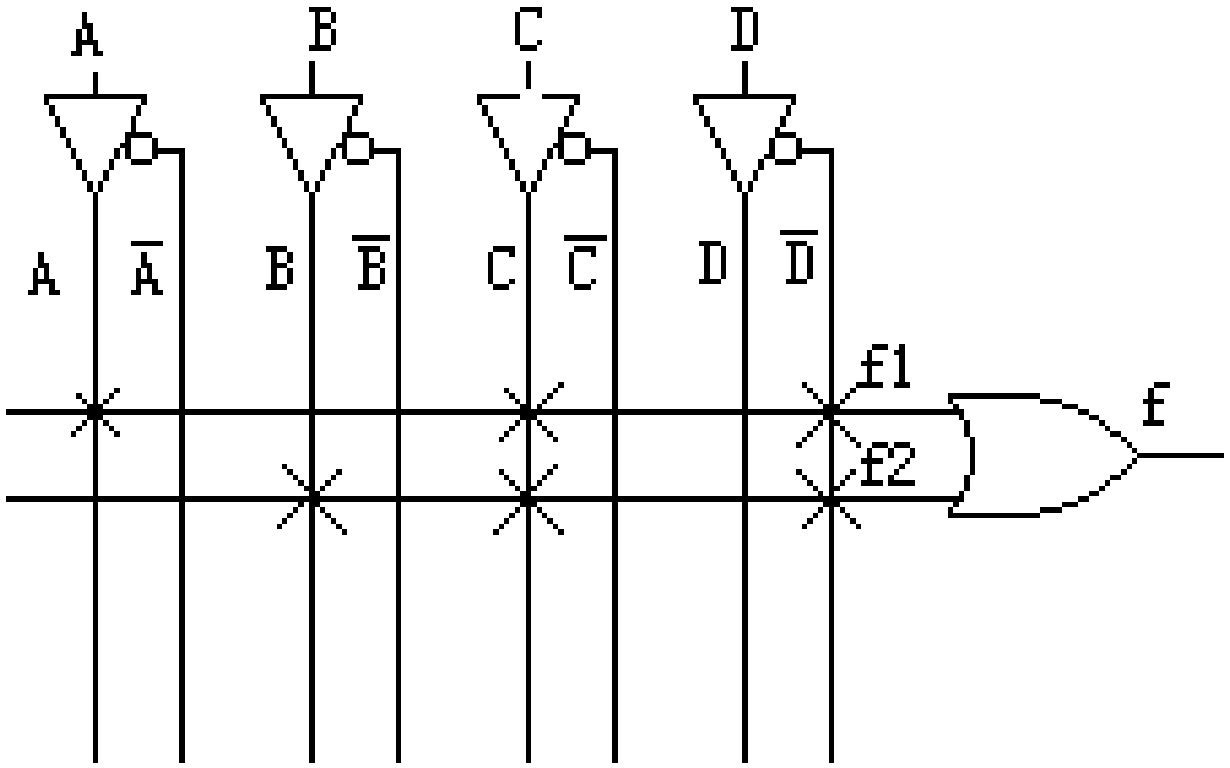

乘积项逻辑阵列
全局 Global 清零 Clear
Global Clock 全局 时钟
Parallel Logic Expanders (from other macrocells)
Programmable Register
Register Bypass
to I/O Control Block
乘积项选择矩阵
PRN
D/T Q
ENA
CLRN
Clock/ Enable Select
可编程D触发器
VCC
Clear Select
to PIA
Shared Logic Expanders
可编选择开关
36 Signals from PIA
16 Expander Product Terms

乘积项结构CPLD的逻辑实现



f ? 假设组合逻辑的输出(AND3的输出)为f，
则 f=(A+B)*C*(!D)=A*C*!D + B*C*!D
(以!D表示D的“非”）。

f: 最小项的与非表达式



百度：“CPLD原理” 李荣

南 京 理 工 大 学

作　　者：李荣　　学　号：0901170129

学院(系)：机械工程学院

专　　业：测控技术与仪器

题　　目：CPLD 概述

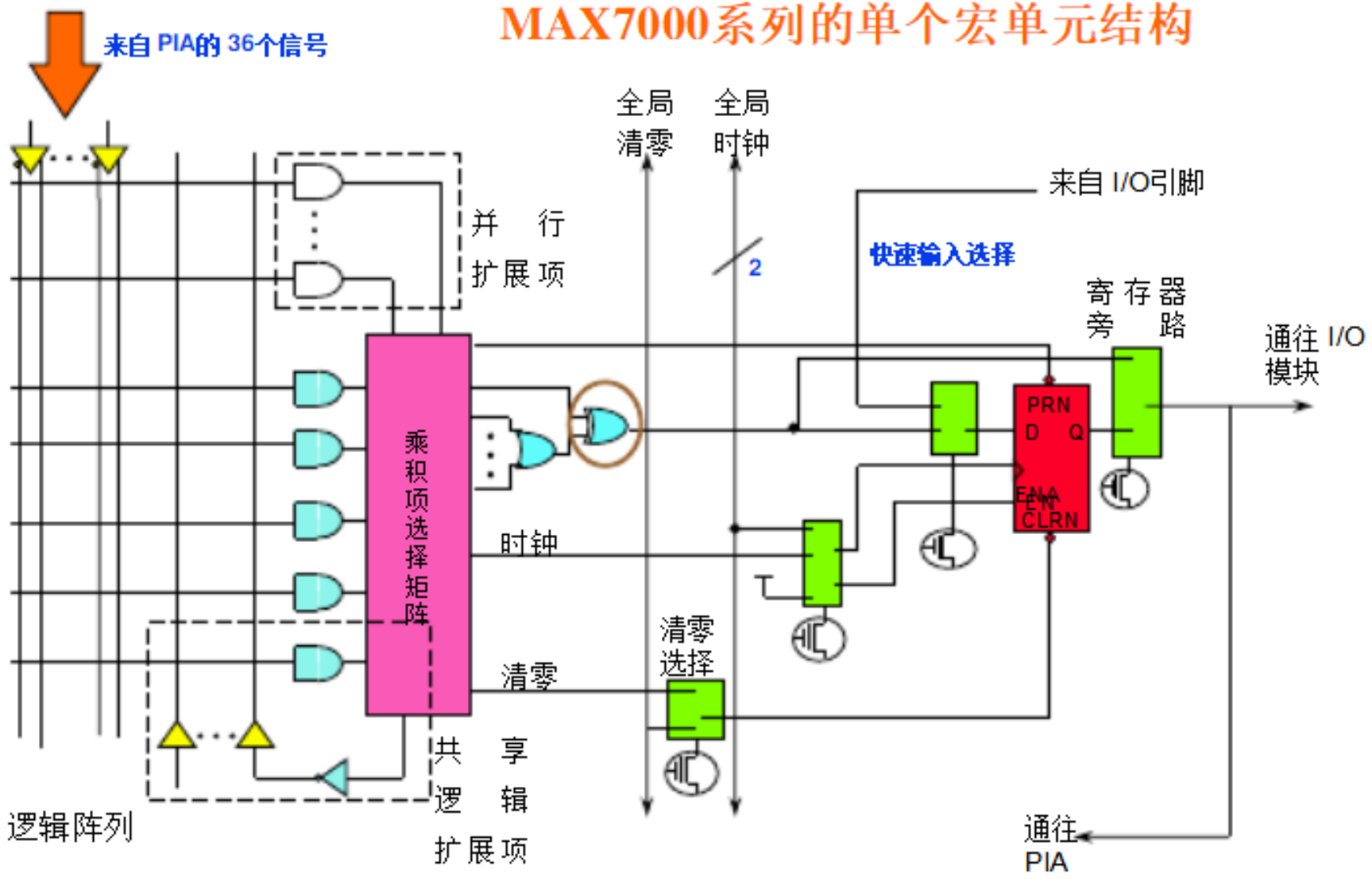
指导者：牛国柱
(姓 名)　　(专业技术职务)

评阅者：
(姓 名)　　(专业技术职务)

2012 年 3 月 10

XILINX®

CPLD结构与工作原理



2．宏单元

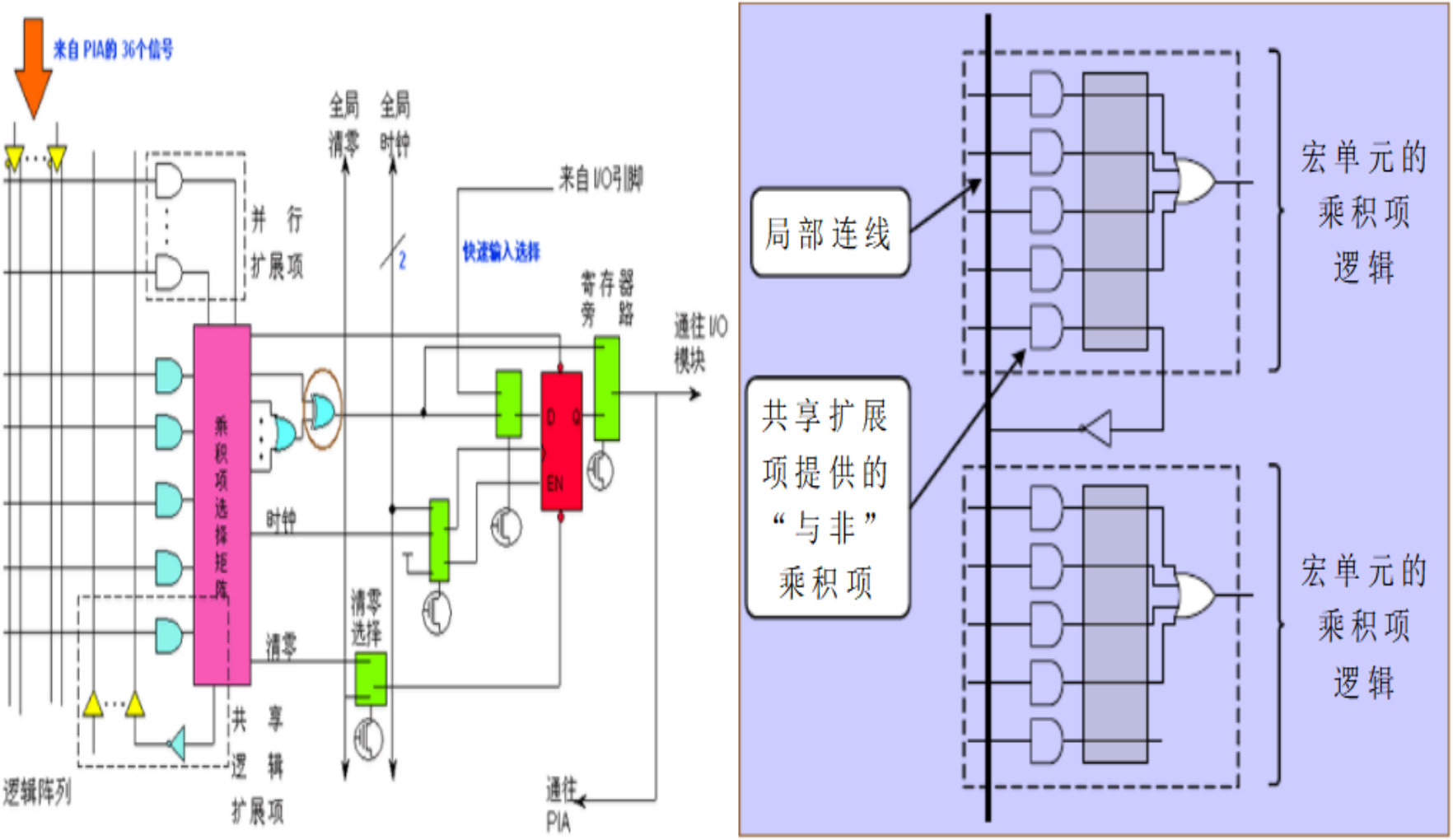
MAX7000 系列中的宏单元由 3 个功能块组成：逻辑阵列、乘积项选择矩阵和可编程寄存器。各部分可以被独自配置为时序逻辑和组合逻辑工作方式。其中逻辑阵列实现组合逻辑，可以为每个宏单元提供 5 个乘积项。乘积项选择矩阵分配这些乘积项作为到“或门”和“异或门”的主要逻辑输入，以实现组合逻辑函数，或者把这些乘积项作为宏单元中寄存器的辅助输入：如清零、置位、时钟和时钟使能控制。每个宏单元中的触发器可以单独地编程为具有可编程时钟控制的 D、T、JK 或 RS 触发器的工作方式。触发器的时钟、清零输入可以通过编程选择使用专用的全局清零和全局时钟，或使用内部逻辑（乘积项逻辑阵列）产生的时钟和清零。触发器也支持异步清零和异步置位功能，乘积项选择矩阵分配乘积项来控制这些操作。如果不需要触发器，也可以将此触发器旁路，信号直接输给 PIA 或输出到 I/O 引脚，以实现组合逻辑工作方式。

3．扩展乘积项

每个宏单元的一个乘积项可以反相回送到逻辑阵列。这个“可共享”的乘积项能够连到同一个 LAB 中的任何其它乘积项上。尽管大多数逻辑函数能够用每个宏单元中的 5 个乘积项实现，但在某些复杂的逻辑函数中需要附加乘积项。为提供所需的逻辑资源，可以利用另一个宏单元，MAX70000 结构也允许利用共享和并联扩展乘积项，这两种扩展项可作为附加的乘积项直接送到本 LAB 的任意宏单元中。利用扩展项可保证在实现逻辑综合时，用尽可能少的逻辑资源实现尽可能快的工作速度。

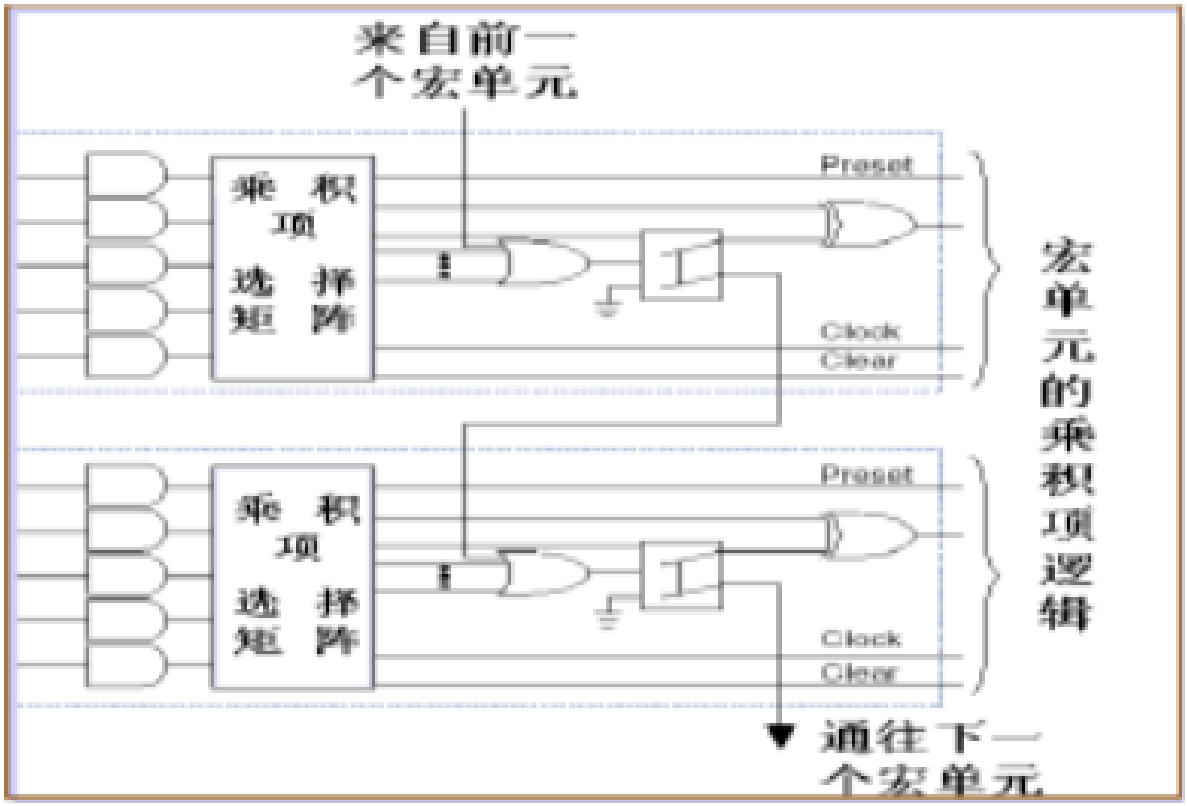
1）共享扩展项

每个 LAB 有多达 16 个共享扩展项。共享扩展项就是由每个宏单元提供一个未投入使用的乘积项，并将它们反相后反馈到逻辑阵列，便于集中使用。每个共享扩展项可被 LAB 内任何（或全部）宏单元使用和共享，以实现复杂的逻辑函数。图 5.3 给出了共享扩展项是如何馈送到多个宏单元的。

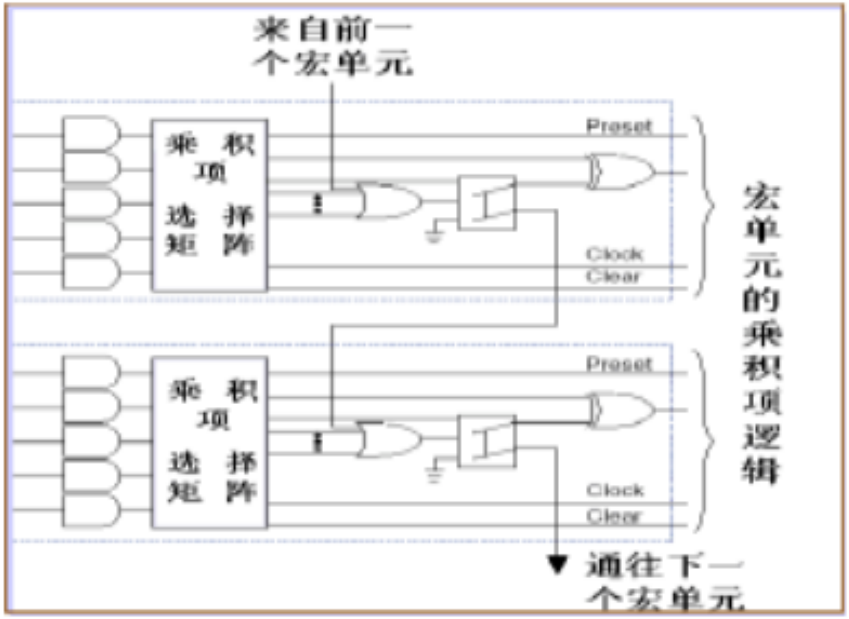

来自 PIA的 36个信号
全局 清零
全局 时钟
来自 I/O引脚
并 行 扩展项
快速输入选择
寄存器 旁 路
通往 I/O 模块
乘积项选择矩阵
D
Q
EN
时钟
清零
清零 选择
共 享 逻 辑 扩展项
逻辑阵列
通往 PIA
局部连线
共享扩展项提供的“与非”乘积项
宏单元的乘积项逻辑
宏单元的乘积项逻辑

2) 并联扩展项

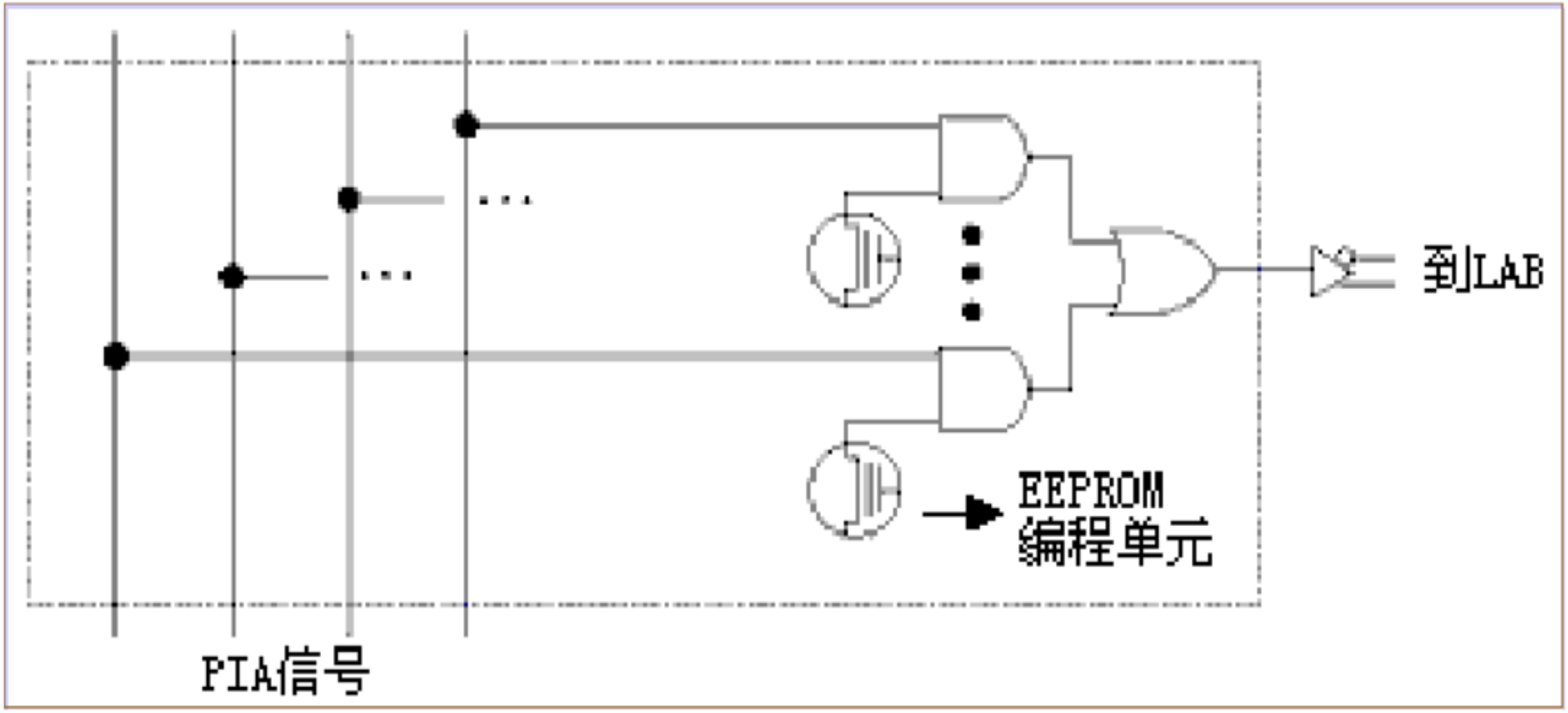
并联扩展项是一些宏观单元中没有使用的乘积项，并且这些乘积项可分配到邻近的宏单元去实现快速复杂的逻辑函数。并联扩展项允许多达 20 个乘积项直接馈送到宏单元的“或”逻辑，其中 5 个乘积项是由宏单元本身提供的，15 个并联扩展项是由 LAB 中邻近宏单元提供的。



每个 LAB 有两组宏单元，每组含有 8 个宏单元（例如，一组为 1～8，另一组为 9～16）。在 LAB 中形成 2 个出借或借用并联扩展项的链。一个宏单元可以从较小编号的宏单元中借用并联扩展项。例如，宏单元 8 能够从宏单元 7，或从宏单元 7 和 6，或从宏单元 7、6 和 5 中借用并联扩展项。在有 8 个宏单元的每个组中，最小编号的宏单元仅能出借并联扩展项；而最大编号的宏单元仅能借用并联扩展项。如图 5.4 给出了并联扩展项是如何从邻近的宏单元中借用的。宏单元中不用的乘积项可分配给邻近的宏单元。

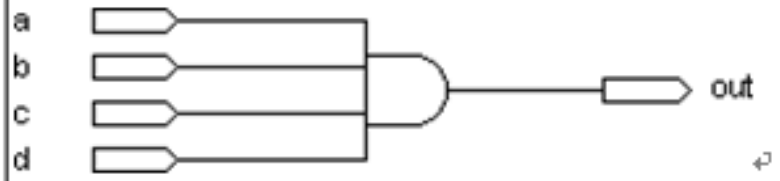
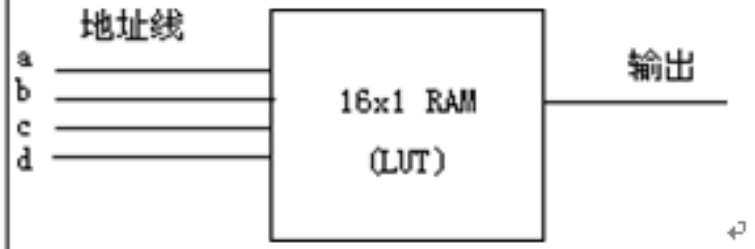


通过可编程连线阵列可将各 LAB 相互连接构成所需的逻辑。这个全局总线是可编程的通道，它能把器件中任何信号源连到其目的地。所有 MAX7000 系列器件的专用输入、I/O 引脚和宏单元输出均馈送到 PIA，PIA 可把这些信号送到整个器件内的各个地方。只有每个 LAB 所需的信号才真正给它布置从 PIA 到该 LAB 的连线，如图 5.5 是 PIA 信号布线到 LAB 的方式。



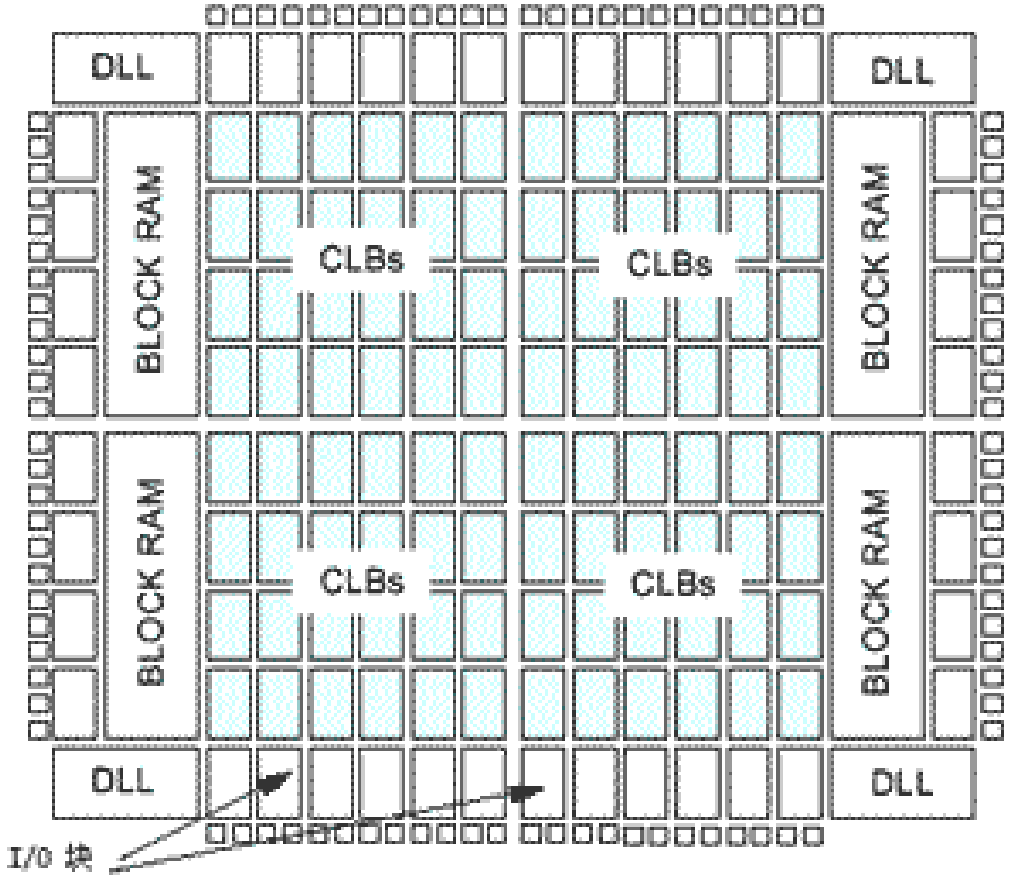
四、FPGA结构

- 查找表LUT vs. PLD的乘积项（线与、熔丝）
- 基于查找表技术、SRAM工艺、要外挂配置EEPROM的PLD叫做FPGA。

实际逻辑电路		LUT 的实现方式	
			
a，b，c，d 输入	逻辑输出	地址	RAM 中存储的内容
0000	0	0000	0
0001	0	0001	0
....	0	...	0
1111	1	1111	1

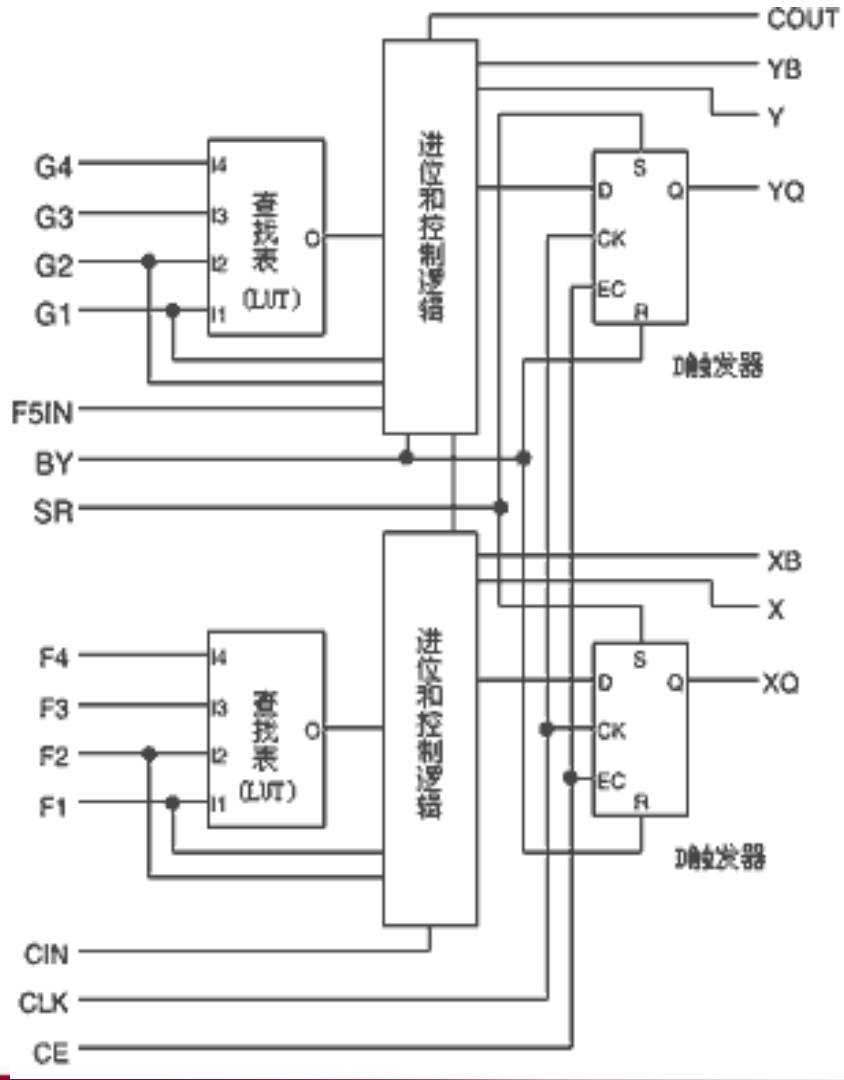
FPGA内部结构

- Spartan-II 内部结构



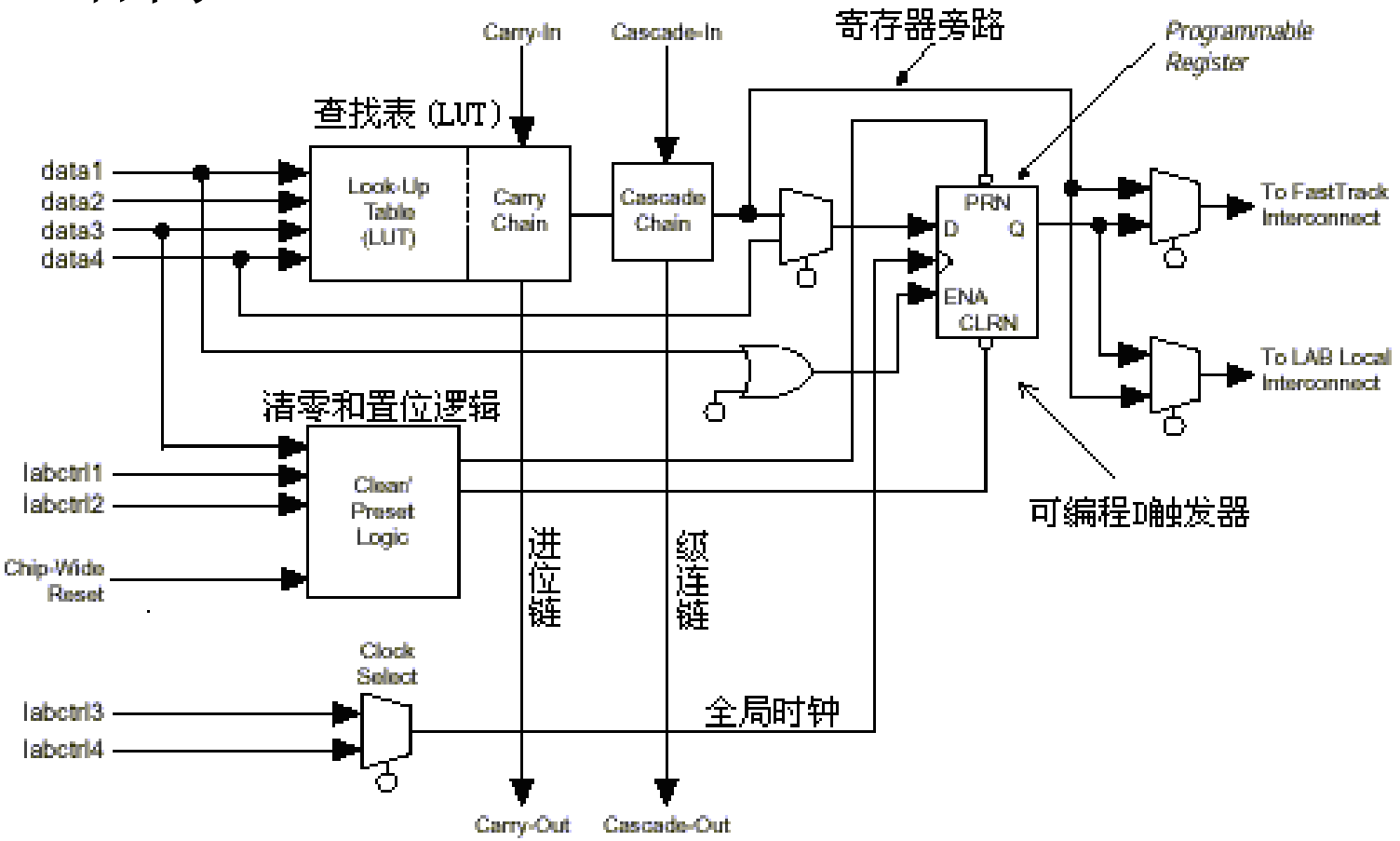
FPGA内部结构

- slices 结构

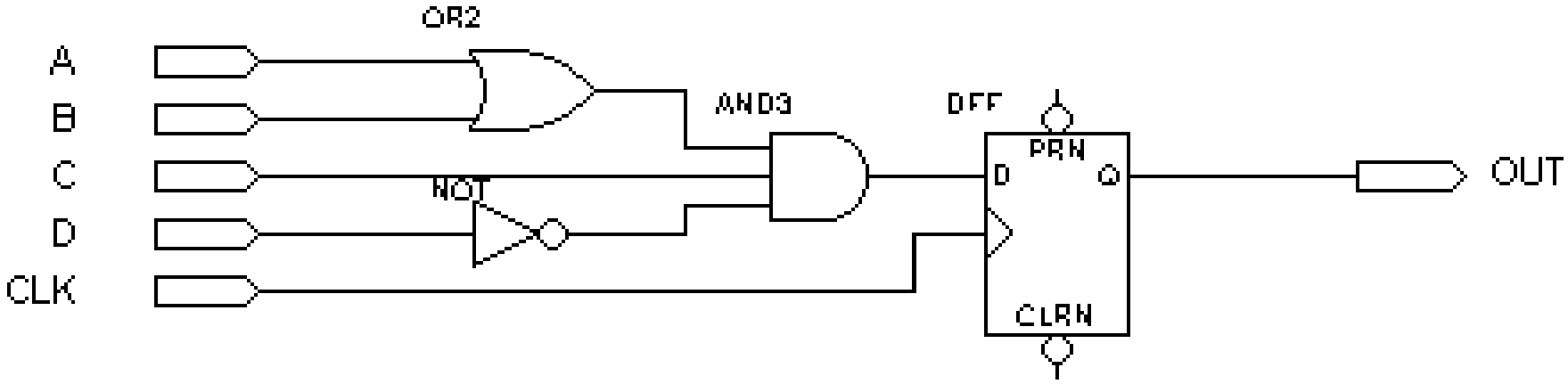


逻辑单元内部结构

- LE 结构

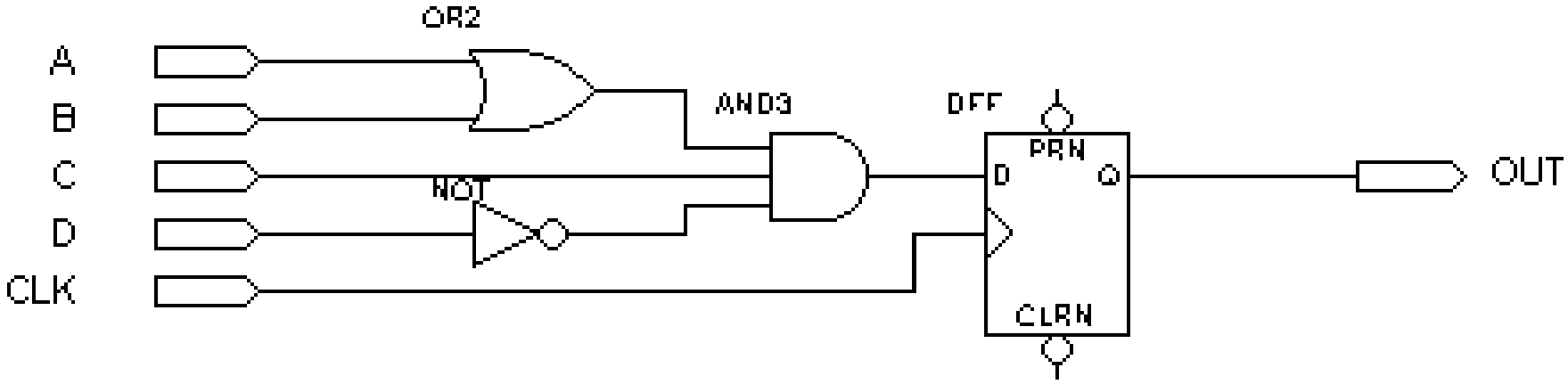


查找表结构的实现原理



- A，B，C，D由FPGA芯片的管脚输入后进入可编程连线，然后作为地址线连到到LUT，LUT中已经事先写入了所有可能的逻辑结果，通过地址查找到相应的数据然后输出，这样组合逻辑（f）就实现了。
  f=(A+B)*C*(!D)=A*C*!D + B*C*!D
- 该电路中D触发器是直接利用LUT后面D触发器来实现。时钟信号CLK由I/O脚输入后进入芯片内部的时钟专用通道，直接连接到触发器的时钟端。触发器的输出与I/O脚相连，把结果输出到芯片管脚。这样PLD就完成了图3所示电路的功能。（以上这些步骤都是由软件自动完成的，不需要人为干预）。

查找表结构的实现原理2



- 对于一个LUT无法完成的的电路，就需要通过进位逻辑将多个单元相连，这样FPGA就可以实现复杂的逻辑。
- 由于LUT主要适合SRAM工艺生产，所以目前大部分FPGA都是基于SRAM工艺的，而SRAM工艺的芯片在掉电后信息就会丢失，一定需要外加一片专用配置芯片，在上电的时候，由这个专用配置芯片把数据加载到FPGA中，然后FPGA就可以正常工作，由于配置时间很短，不会影响系统正常工作。 也有少数FPGA采用反熔丝或Flash工艺，对这种FPGA，就不需要外加专用的配置芯片。

? LUT or Interconnect
US Patent 4,870,302

- Ross Freeman是Xilinx创始人之一，发明了“现场可编程门阵列”（FPGA）这种新型可编程逻辑。

- 1）CPLD适合触发器有限而乘积项丰富的结构，适合完
FPGA适合触发器丰富的结构，适合完成时序逻辑；
- 2）CPLD的运行速度一般优于FPGA，这是因为CPLD是
辑块互连是集总式的，CPLD 通过修改具有固定内连电
而FPGA 是门级编程，且LAB 之间是采用分布式互连，
部连线的布线来编程。
- 3）CPLD 的逻辑块互连是集总式的,其特点是等延时,任
等的,这种结构给设计者带来很大方便; FPGA的互连则
统的布局有关；
- 4）CPLD主要是基于EEPROM或者Flash存储器编程，编程次数有[illegible]，且在系统断电后，信息不会丢失。CPLD不需要外部存储器，使用较为简单。FPGA主要基于SRAM，优点是可进行任意次数编程，并可在工作中快速编程，实现系统级的动态配置，因此可成为可重配置硬件。但是，由于其信息存放在外部存储器上，系统断电时信息会丢失，每次上电时，需从器件的外部存储器或计算机中将编程数据写入SRAM中，使用较为复杂。

Loring Wirbel, EDN, "Remembering Ross Freeman. February 27, 2009. Retrieved on March 16, 2009"

Company Release, "Xilinx Co-Founder Ross Freeman Honored as 2009 National Inventors Hall of Fame Inductee for Invention of FPGA". February 11, 2009. Retrieved February 13, 2009.

Frederick E. Allen, Forbes, "The Greatest Inventors You've Never Heard Of".February 12, 2009. Retrieved on February 13, 2009.

Innovation: interconnect ?

United States Patent [19]

Freeman

[11] **Patent Number: 4,870,302**

[45] **Date of Patent: Sep. 26, 1989**

[54] **CONFIGURABLE ELECTRICAL CIRCUIT HAVING CONFIGURABLE LOGIC ELEMENTS AND CONFIGURABLE INTERCONNECTS**

[75] Inventor: **Ross H. Freeman**, San Jose, Calif.

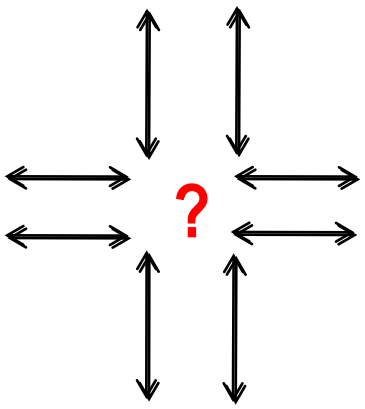
[73] Assignee: **Xilinx, Inc.**, San Jose, Calif.

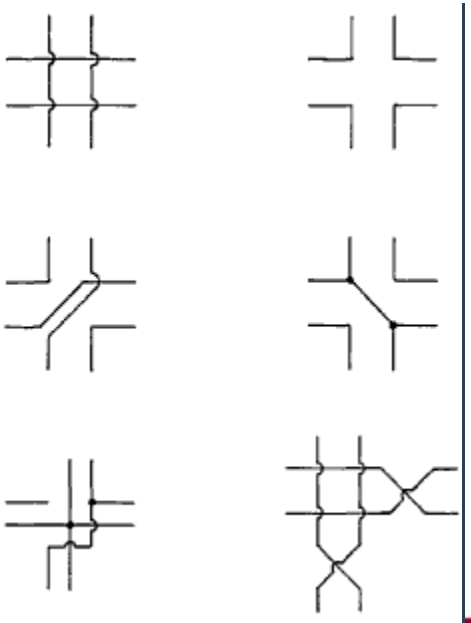
[21] Appl. No.: **158,011**

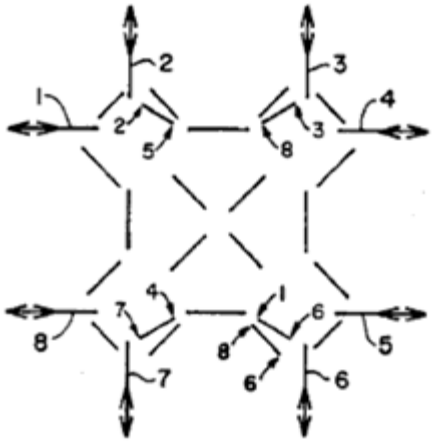
[22] Filed: **Feb. 19, 1988**

FOREIGN PATENT DOCUMENTS

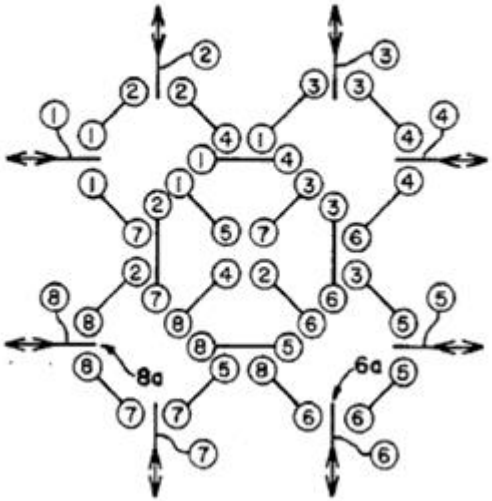
0094234	11/1983	European Pat. Off. .
0031431	3/1984	European Pat. Off. .
3202498A1	8/1983	Fed. Rep. of Germany .
2160969	7/1973	France .
137616	1/1978	Japan .
141836	11/1980	Japan .
191535	11/1983	Japan .
1090520	11/1967	United Kingdom .
1101851	1/1968	United Kingdom .
1516817	7/1978	United Kingdom .







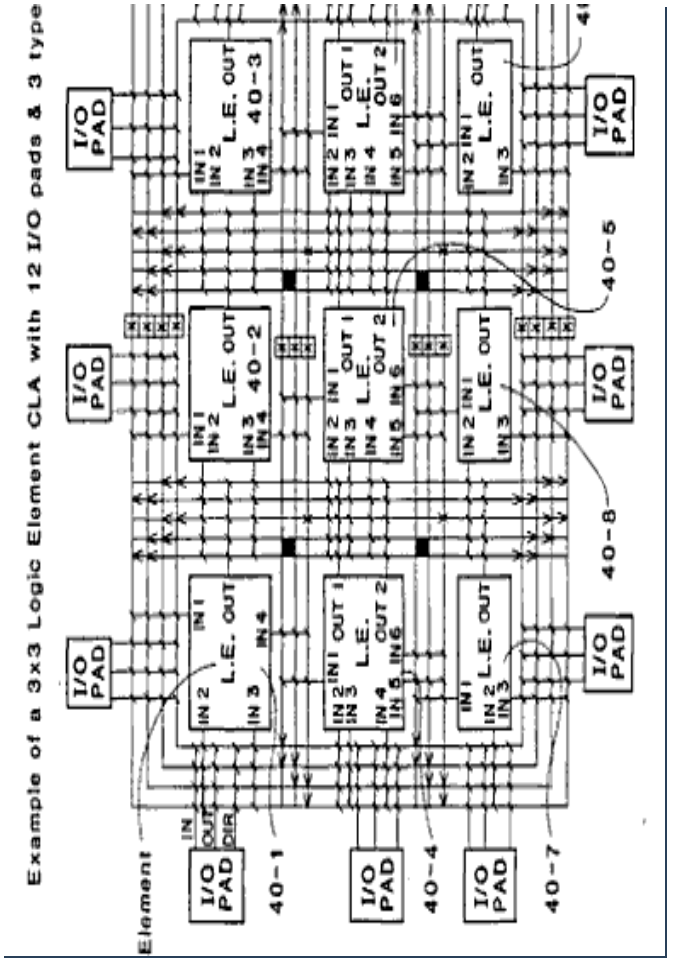
Circled numbers on ends of pass devices indicate which paths are directly connected.



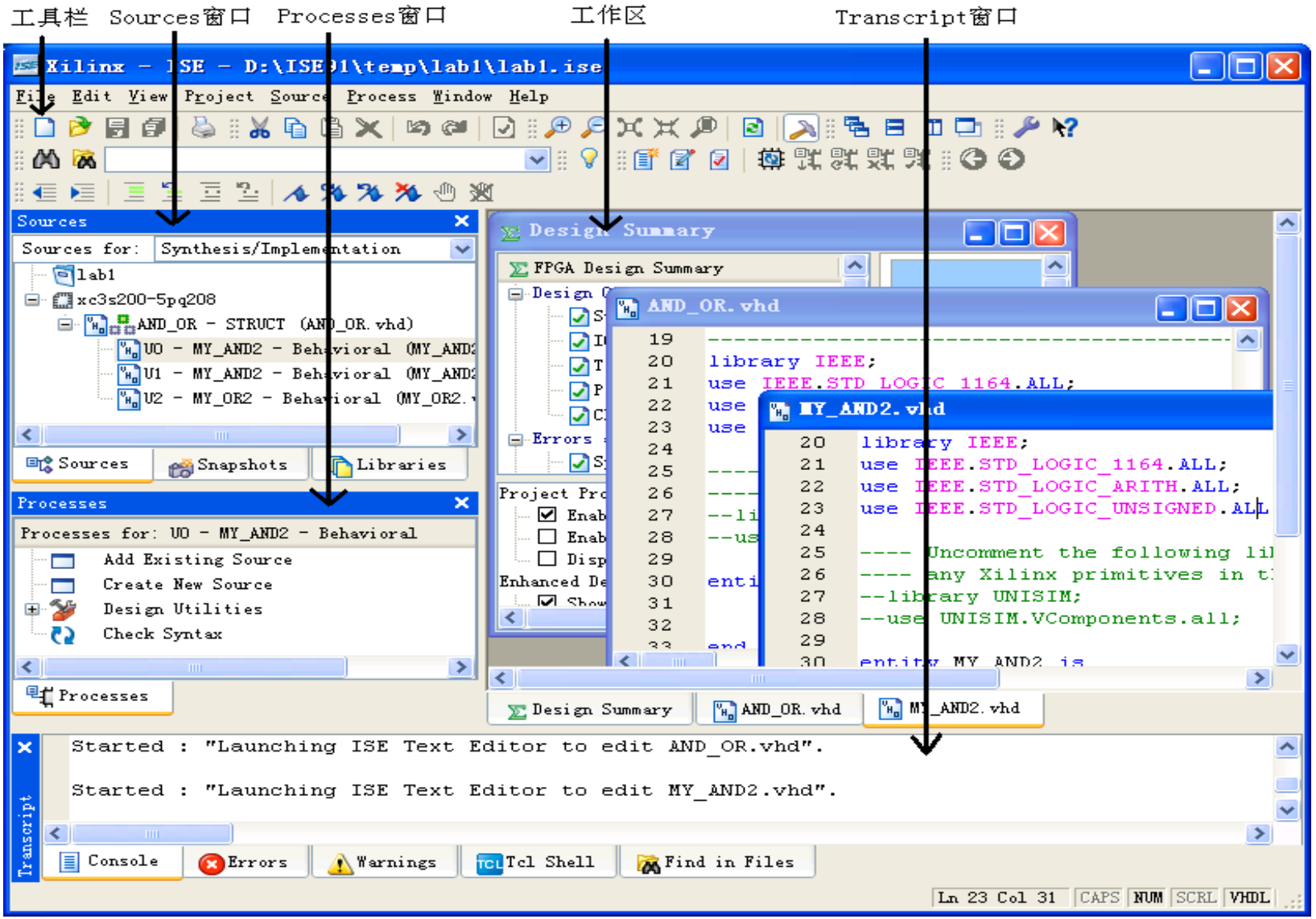
XILINX®

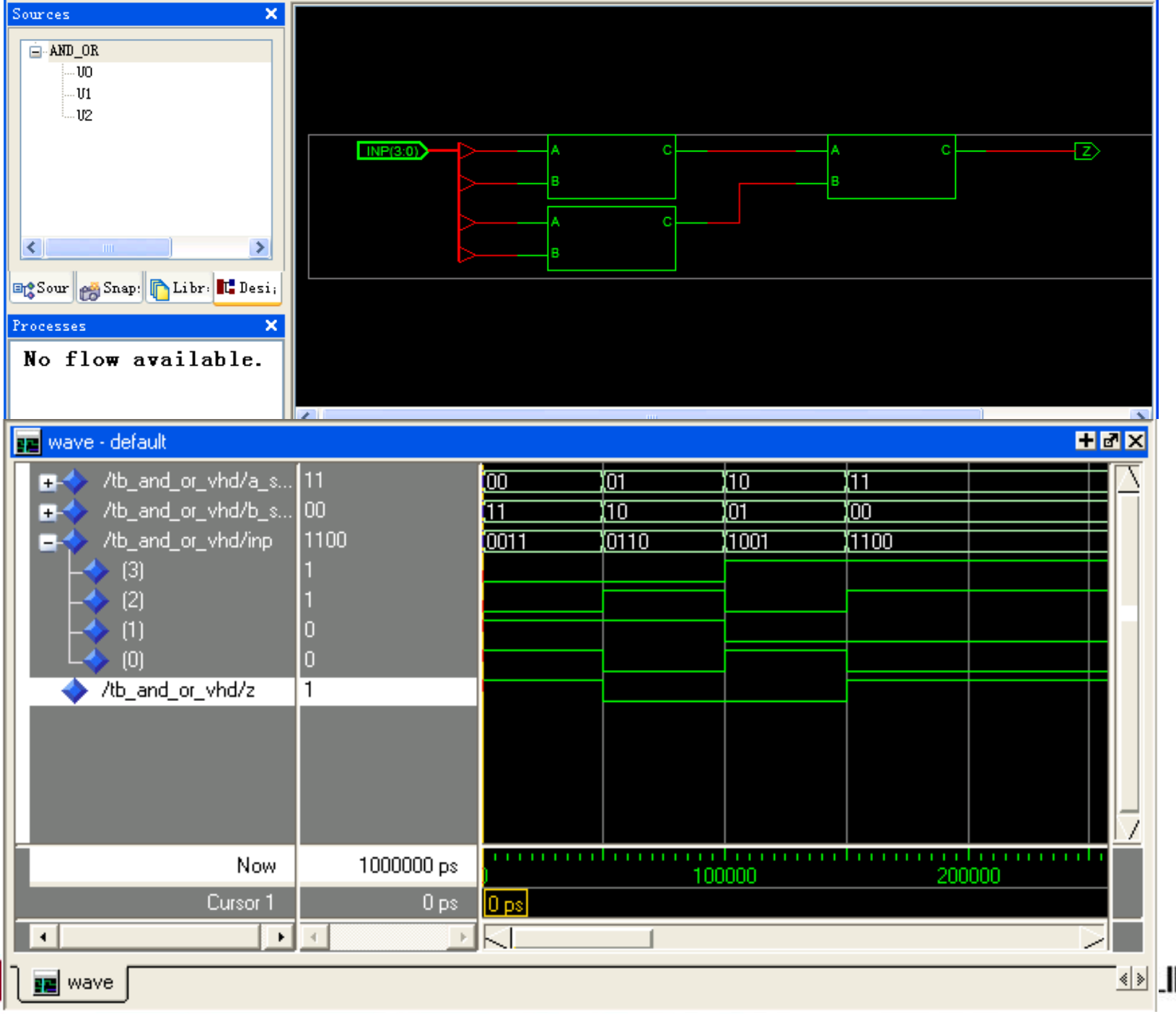
US Patent 4,870,302

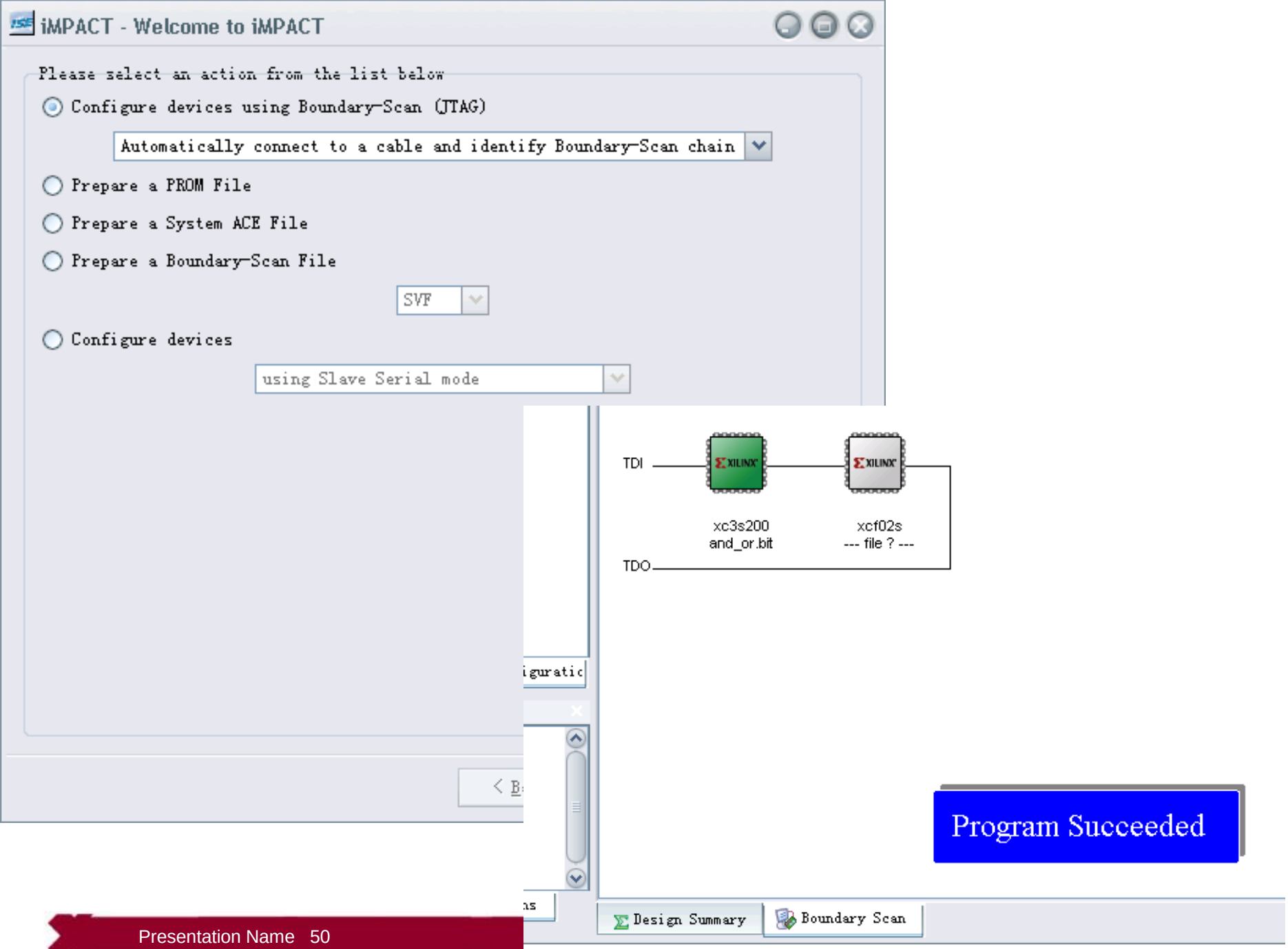
- A configurable logic array comprises a plurality of configurable logic elements variably interconnected in response to control signals to perform a selected logic function. Each configurable logic element in the array is in itself capable of performing any one of a plurality of logic functions depending upon the control information placed in the configurable logic element. Each configurable logic element can have its function varied even after it is installed in a system by changing the control information placed in that element. Structure is provided for storing control information and providing access to the stored control information to allow each configurable logic element to be properly configured prior to the initiation of operation of the system of which the array is a part. **Novel interconnection structures are provided to facilitate the configuring of each logic element.**



五、ISE（Integrated Software Environment）软件




Sources
AND_OR
U0
U1
U2
Sour
Snap
Libr
Desi
Processes
No flow available.
INP(3:0)
A
B
C
Z
wave - default
/tb_and_or_vhd/a_s...
/tb_and_or_vhd/b_s...
/tb_and_or_vhd/inp
(3)
(2)
(1)
(0)
/tb_and_or_vhd/z
11
00
1100
1
1
0
0
1
00
01
10
11
11
10
01
00
0011
0110
1001
1100
Now
1000000 ps
Cursor 1
0 ps
100000
200000
0 ps
wave


iMPACT - Welcome to iMPACT
Please select an action from the list below
Configure devices using Boundary-Scan (JTAG)
Automatically connect to a cable and identify Boundary-Scan chain
Prepare a PROM File
Prepare a System ACE File
Prepare a Boundary-Scan File
SVF
Configure devices
using Slave Serial mode
TDI
TDO
xc3s200
and_or.bit
xcf02s
--- file ? ---
Program Succeeded
Design Summary
Boundary Scan
register values:

Summary

- Why PLD
- From PLD TO FPGA
- 继FPGA之后的新技术是什么？
- XILINX FPGA 设计基础 第1-2 章，
- Lab: 第六章实验一。
- Homework.

作业

- 尝试安装XILINX 的ISE (WEBPACK, license)开发平台
- 任选一个题目(基于FPGA的逻辑设计、基于FPGA的嵌入式系统设计、基于FPGA的DSP系统设计)，写一个关于此方向的综述（3-5页）
  - 逻辑设计：设计方法，设计工具
  - 嵌入式系统设计：发展现状
  - FPGA-DSP系统设计：发展现状
- 了解Spartan3E开发板和Spartan6开发板资源

Thank you