

8. 大规模可编程器件

8.1 复杂可编程逻辑器件(CPLD)简介

8.2 现场可编程门阵列(FPGA)

8.3 可编程逻辑器件开发过程简介

8.1 复杂可编程逻辑器件(CPLD)简介

1. 逻辑块

2. 可编程内部连线

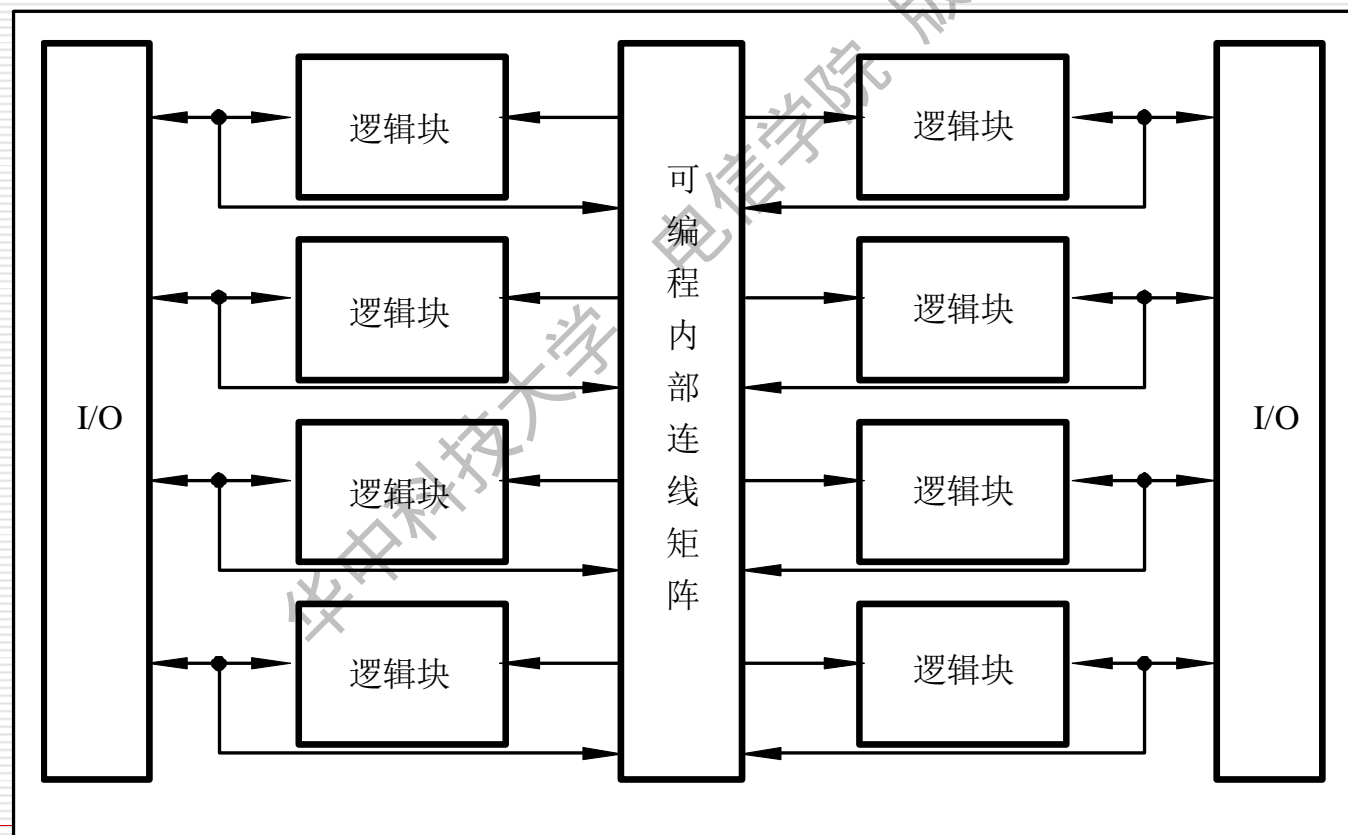
3. I/O单元

8.1 复杂可编程逻辑器件(CPLD)简介

- 与PAL、GAL相比，CPLD的集成度更高，有更多的输入端、乘积项和更多的宏单元；
- CPLD器件内部含有多个逻辑块，每个逻辑块都相当于一个GAL器件；
- 每个块之间可以使用可编程内部连线(或者称为可编程的开关矩阵)实现相互连接。

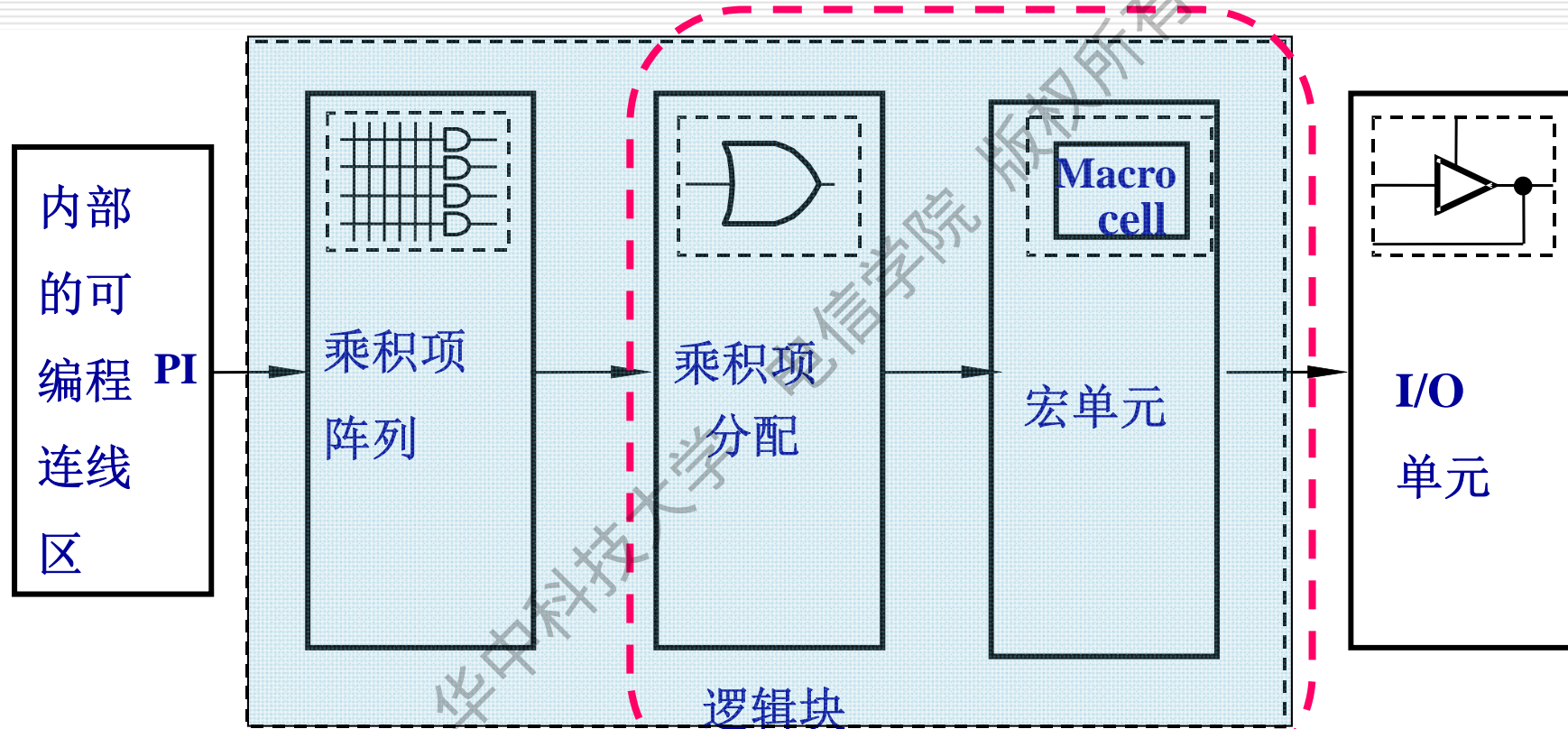
1. 逻辑块

逻辑块是CPLD实现逻辑功能的核心模块。



(1) 可编程乘积项阵列

通用的CPLD器件逻辑块的结构



Xilinx XG500: 90个36变量的乘积项,宏单元36个

Altera MAX7000: 80个36变量的乘积项,宏单元16个

(2) 乘积项分配和宏单元

GAL中的乘积项是固定的，对应一个宏单元。但逻辑块中的乘积项可以编程，分配到不同的宏单元。灵活性大大提高。

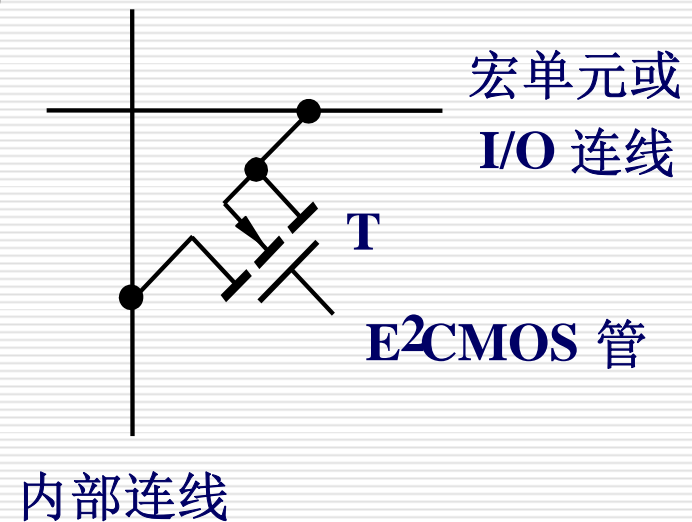
CPLD中的宏单元与**GAL**中的类似。

2. 可编程内部连线

可编程内部连线的作用是实现逻辑块与逻辑块之间、逻辑块与I/O块之间以及全局信号到逻辑块和I/O块之间的连接。

连线区的可编程连接一般由E²CMOS管实现。

当E²CMOS管被编程为导通时，纵线和横线连通；未被编程为截止时，两线则不通。

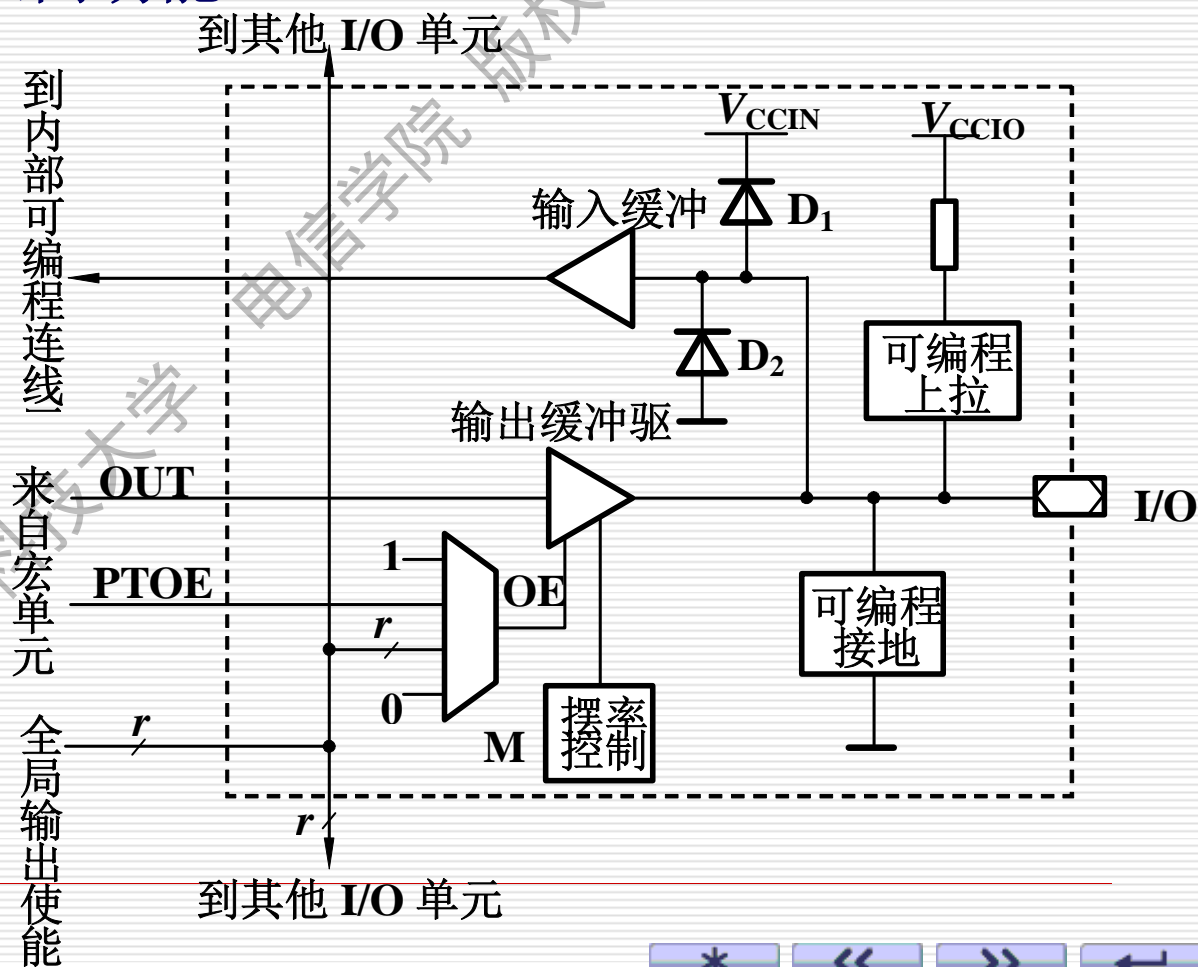


可编程连接原理图

3. I/O单元

I/O单元是CPLD外部封装引脚和内部逻辑间的接口。每个I/O单元对应一个封装引脚，对I/O单元编程，可将引脚定义为输入、输出和双向功能。

数据选择器
提供OE号。
OE=1, I/O引
脚为输出



8.2 现场可编程门阵列(FPGA)

8.2.1 FPGA实现逻辑功能的基本原理

8.2.2 FPGA结构简介

8.2 现场可编程门阵列(FPGA)

- **CPLD**用可编程“与-或”阵列实现逻辑函数。编程基于**E²PROM**或快闪存储器。
- **FPGA**是用查找表(**LUT**)实现逻辑函数。复杂函数使用众多的**LUT**和触发器实现。编程基于**SRAM**。

乘积项技术和查找表技术

❑ 乘积项技术 Product-Term

- 即采用与门—或门来实现逻辑函数，可对与门、或门实现编程连接。由传统PLD发展而来。

❑ 查找表技术 Look-Up-Table

- 自动计算逻辑电路的所有可能的结果，并把结果事先写入RAM, 这样，每输入一个信号进行逻辑运算就等于输入一个地址进行查表，找出地址对应的内容，然后输出即可。

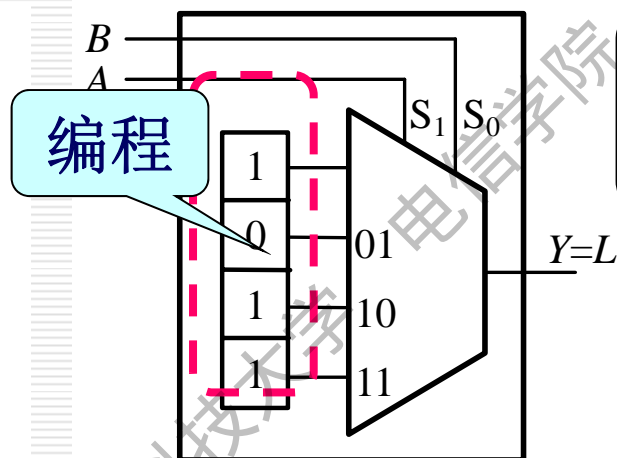
- ❑ 一般的，采用PT技术的称CPLD。采用LUT技术的称为FPGA。
-

8.2.1 FPGA实现逻辑功能的基本原理

LUT是FPGA实现逻辑函数的基本单元。**2输入LUT**可实现任意**2**变量组合逻辑函数。

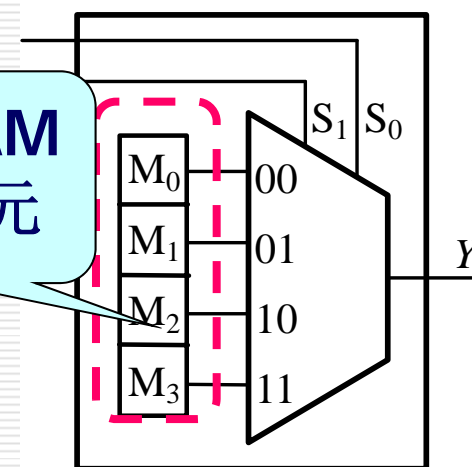
某函数 L 的真值表

A	B	L
0	0	1
0	1	0
1	0	1
1	1	1



实现 L 的 LUT

**4个SRAM
存储单元**



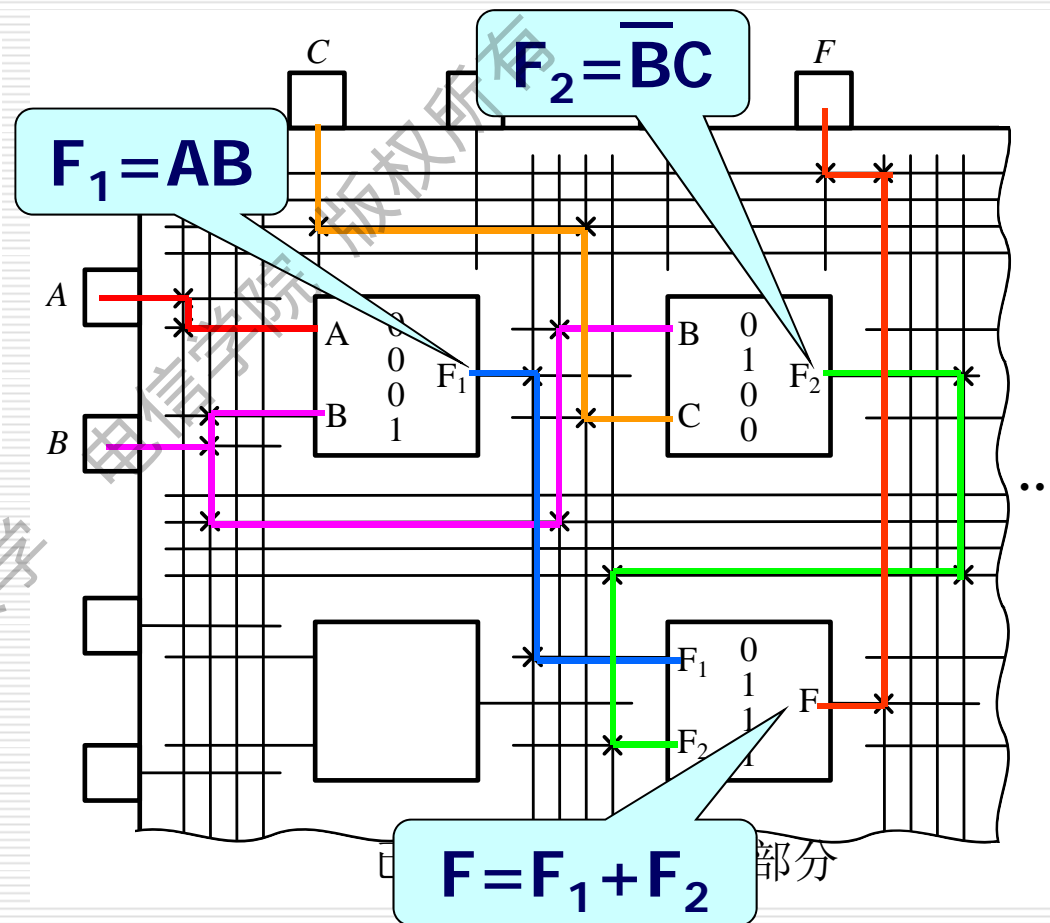
目前FPGA中的LUT大多是4~5个输入，1个输出。当变量数超过一个LUT的输入数时，需要将多个LUT扩展使用。

LUT扩展--用2输入LUT实现函数 $F = AB + \overline{B}C = F_1 + F_2$

函数 F 的真值表

$A B$	F_1	$B C$	F_2
0 0	0	0 0	0
0 1	0	0 1	1
1 0	0	1 0	0
1 1	1	1 1	0

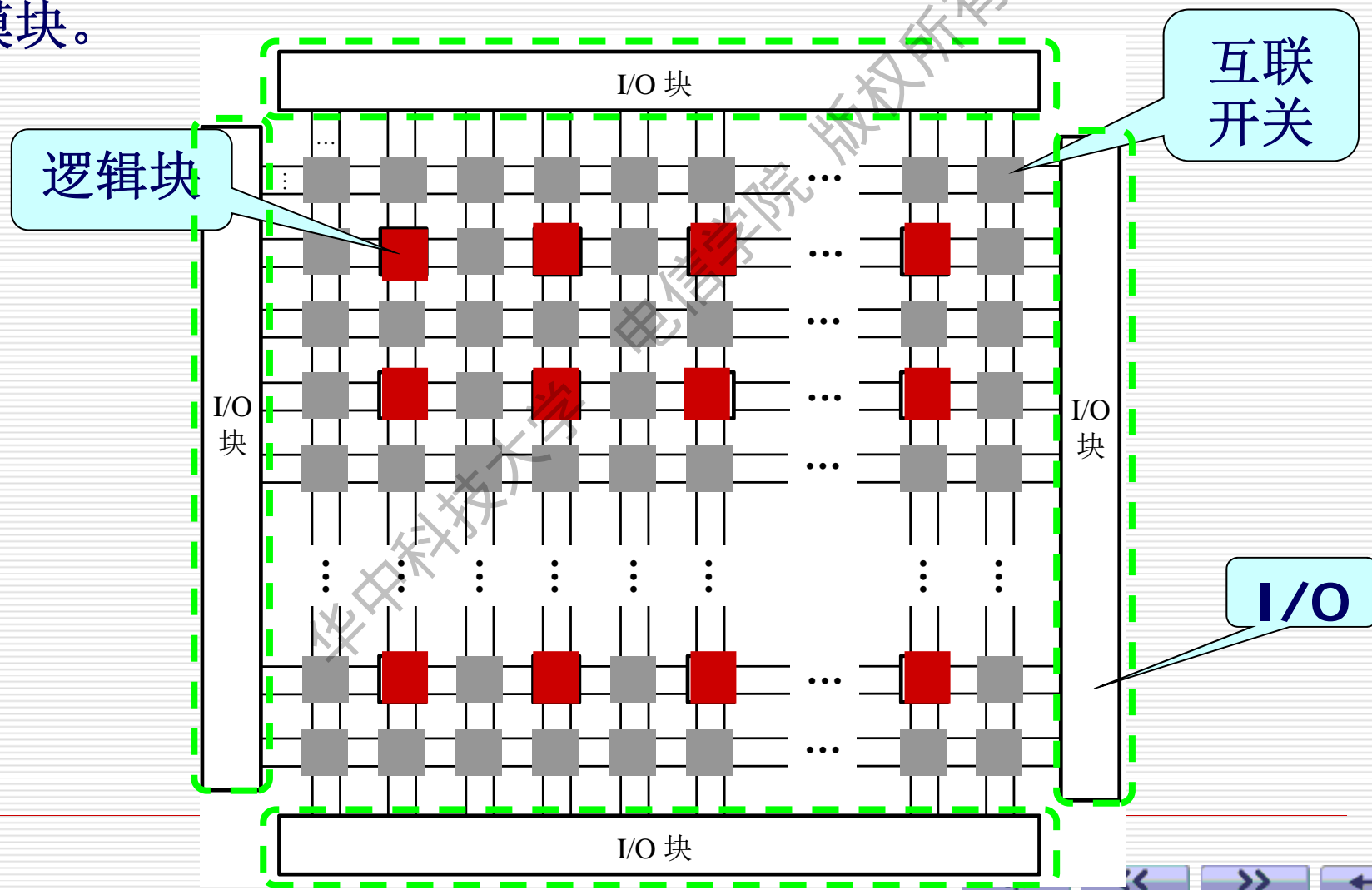
$F_1 F_2$	F
0 0	0
0 1	1
1 0	1
1 1	1



在LUT的基础上增加触发器便可实现时序电路。

8.2.2 FPGA结构简介

FPGA包括：可编程逻辑块、可编程互联开关、可编程I/O模块。

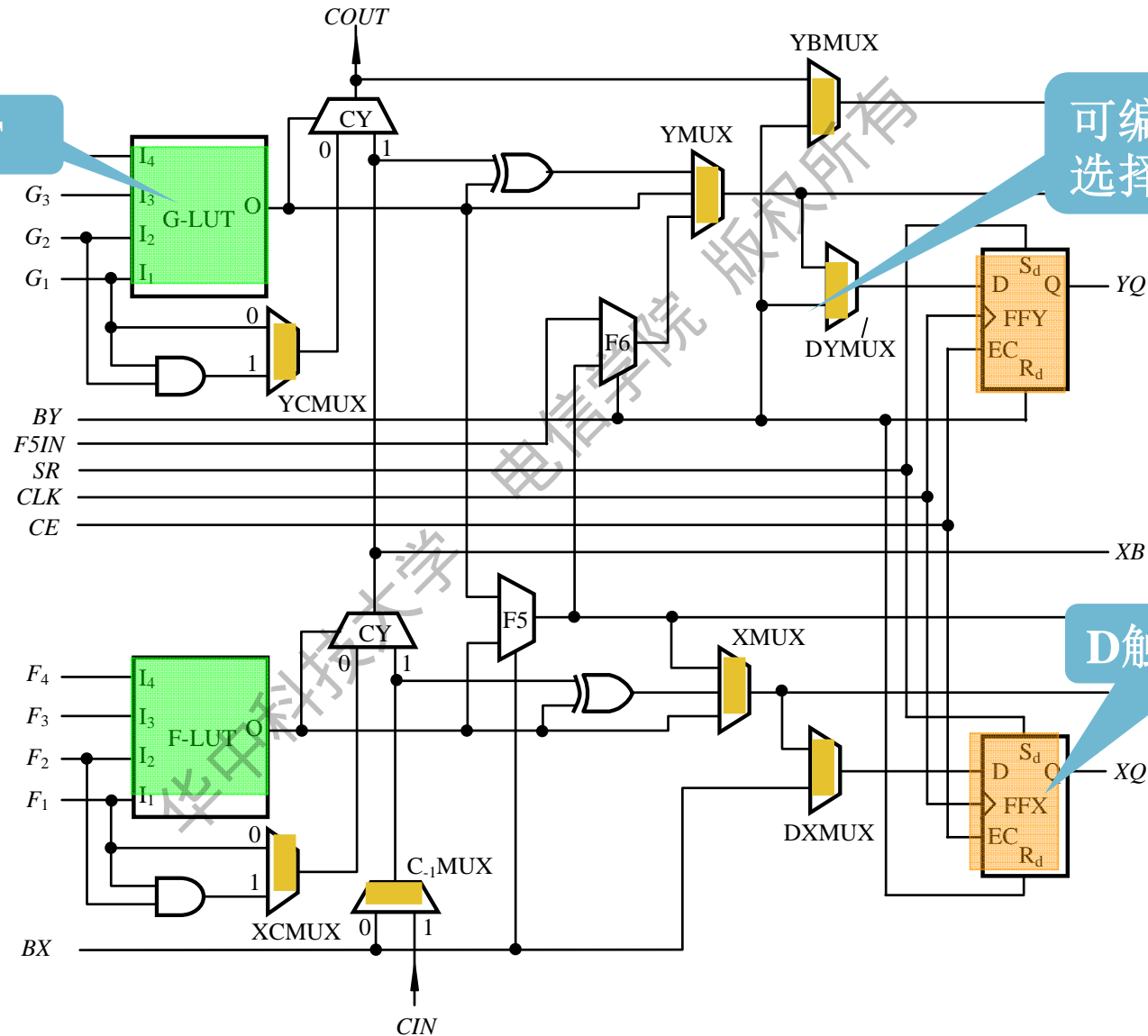


1. 可编程逻辑块

4输入LUT

可编程数据选择器

D触发器



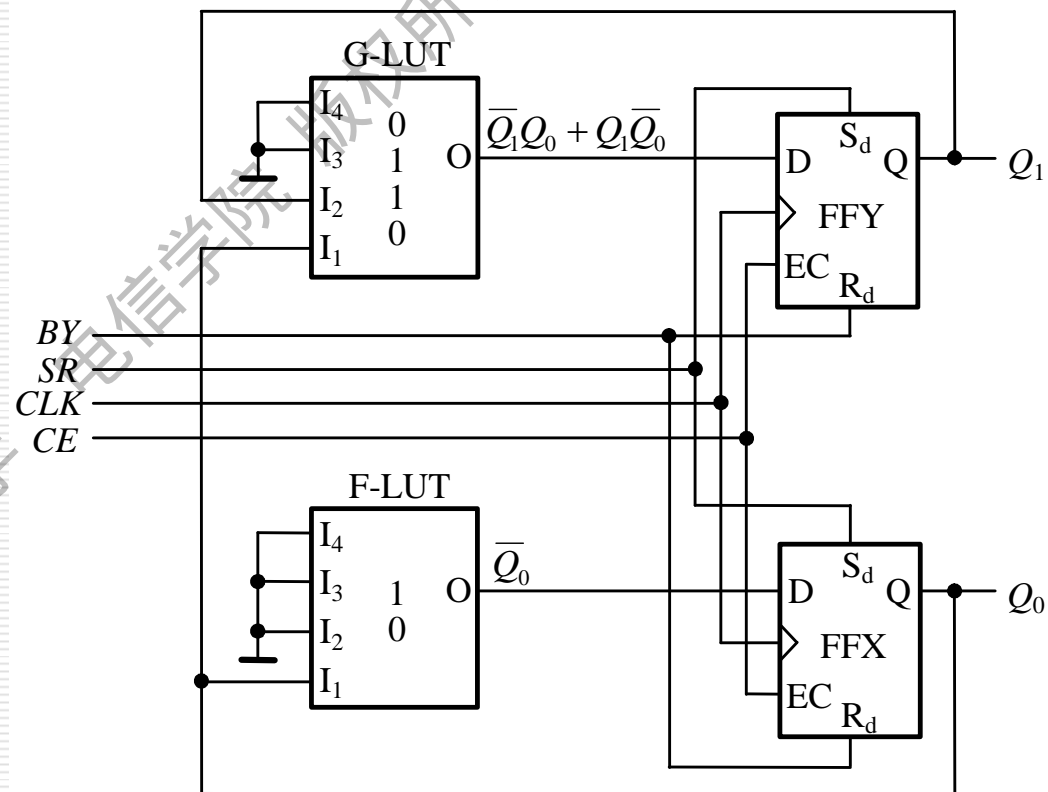
用可编程逻辑块实现2位二进制计数器。

2位二进制状态转换表

$Q_1^n Q_0^n$	$Q_1^{n+1}(D_1) Q_0^{n+1}(D_0)$
00	01
01	10
10	11
11	00

$$\text{得 } D_1 = \overline{Q_1}Q_0 + Q_1\overline{Q_0}$$

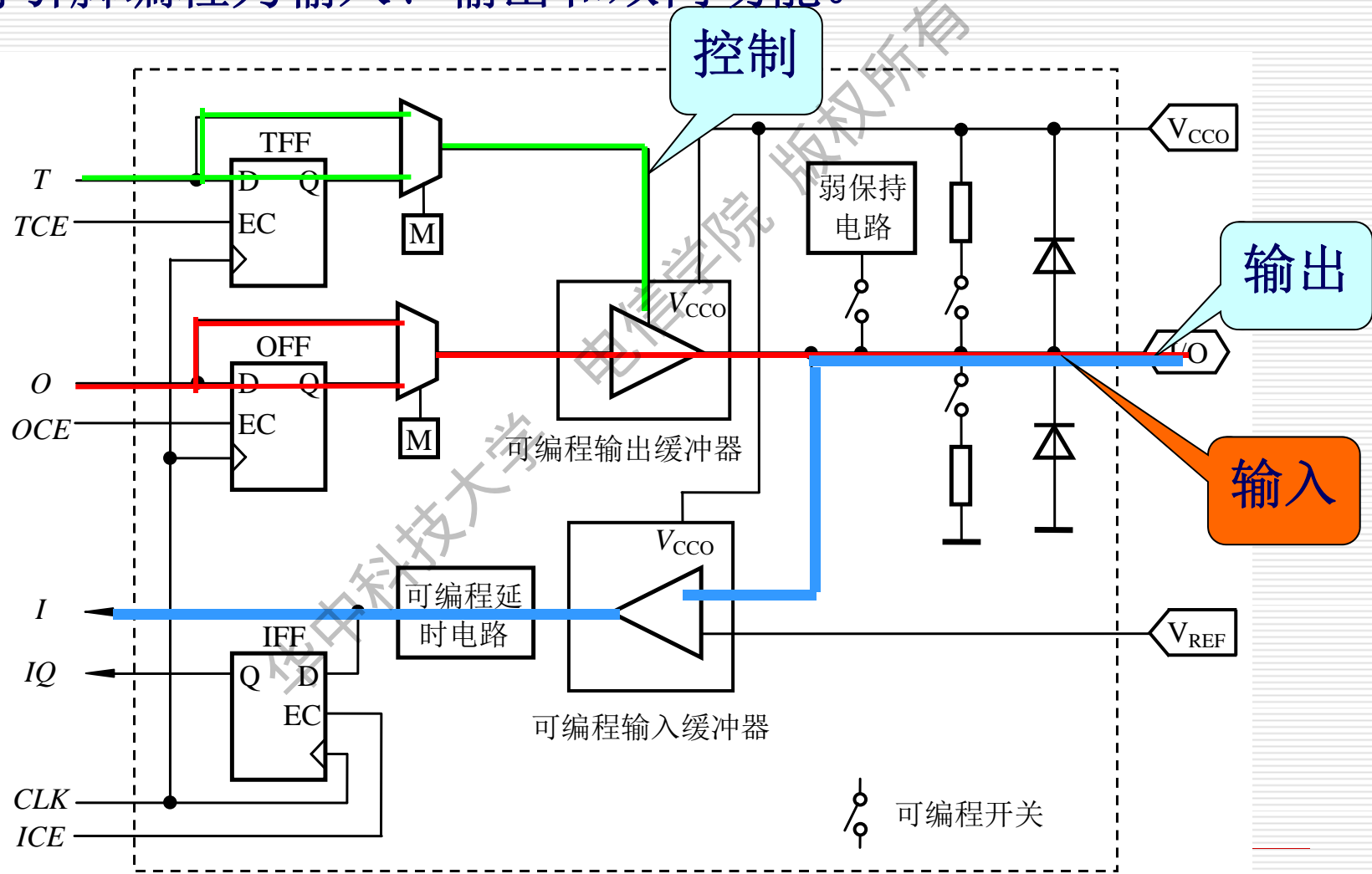
$$D_0 = \overline{Q_0}$$



逻辑块编程实现 2 位二进制计数器

2. I/O块

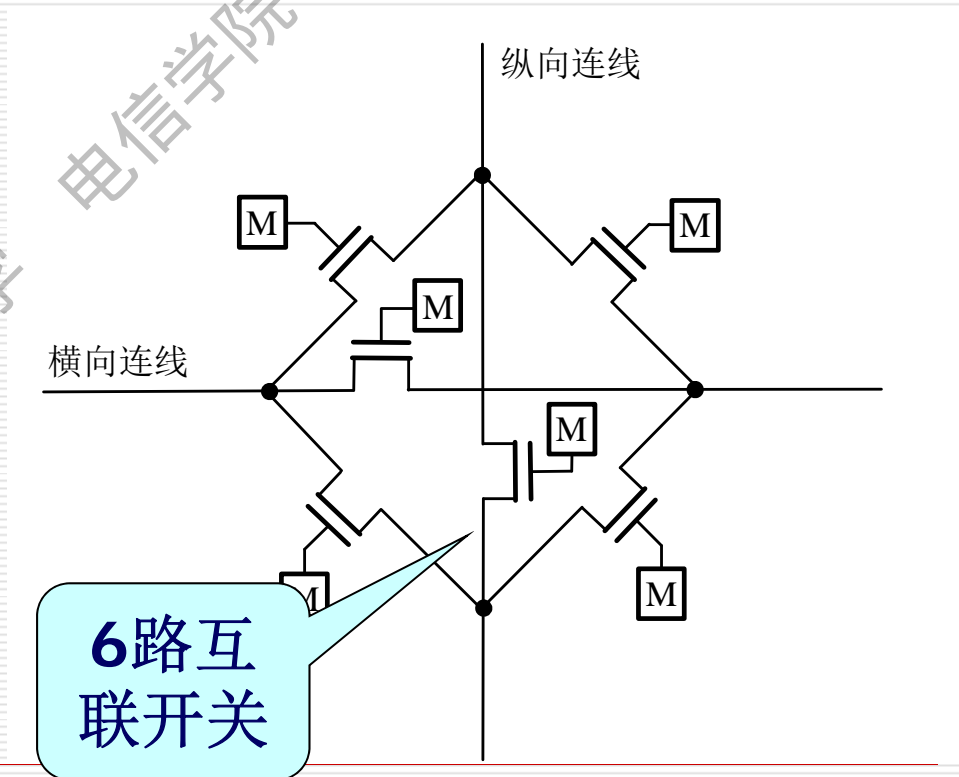
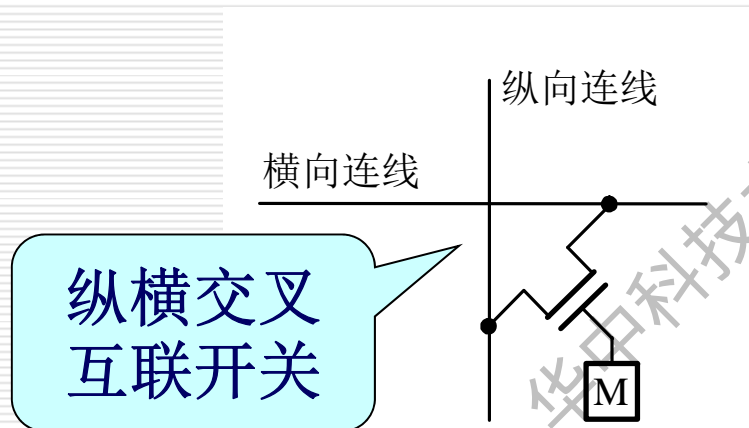
将引脚编程为输入、输出和双向功能。



3. 可编程连线资源

可编程开关实现逻辑块与逻辑块之间，逻辑块与连线之间，逻辑块与I/O之间等的连接。

两种典型的互联开关结构如图。



CPLD与FPGA的区别

	CPLD	FPGA
内部结构	Product-term	Look-up Table
程序存储	内部EEPROM	SRAM, 外挂EEPROM
资源类型	组合电路资源丰富	触发器资源丰富
集成度	低	高
使用场合	完成控制逻辑	能完成比较复杂的算法
速度	快	慢
其他资源	—	EAB, 锁相环
保密性	可加密	一般不能保密

FPGA与CPLD的区别

- FPGA采用SRAM进行功能配置，可重复编程，但系统掉电后，SRAM中的数据丢失。因此，需在FPGA外加EPROM，将配置数据写入其中，系统每次上电自动将数据引入SRAM中。CPLD器件一般采用EEPROM存储技术，可重复编程，并且系统掉电后，EEPROM中的数据不会丢失，适于数据的保密。

华中科技大学

FPGA与CPLD的区别

- FPGA为细粒度结构，CPLD为粗粒度结构。FPGA内部有丰富连线资源，CLB分块较小，芯片的利用率较高。CPLD的宏单元的与或阵列较大，通常不能完全被应用，且宏单元之间主要通过高速数据通道连接，其容量有限，限制了器件的灵活布线，因此CPLD利用率较FPGA器件低。

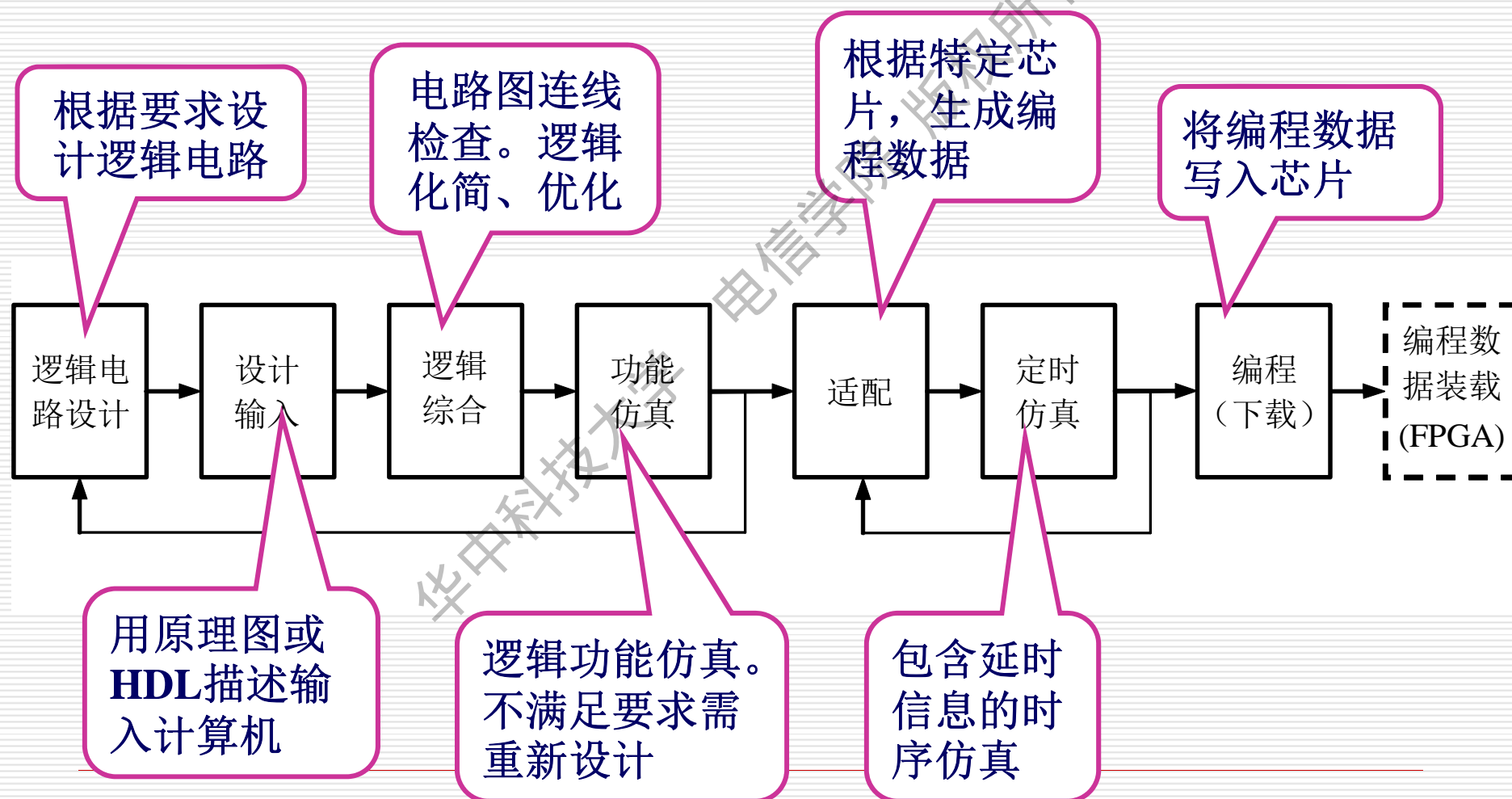
华中科技大学

FPGA与CPLD的区别

- FPGA为非连续式布线，CPLD为连续式布线。
FPGA器件在每次编程时实现的逻辑功能一样，但走的路线不同，因此延时不易控制，要求开发软件允许工程师对关键的路线给予限制。
CPLD每次布线路径一样，CPLD的连续式互连结构利用具有同样长度的一些金属线实现逻辑单元之间的互连。连续式互连结构消除了分段式互连结构在定时上的差异，并在逻辑单元之间提供快速且具有固定延时的通路。CPLD的延时较小。

8.3 可编程逻辑器件开发过程简介

可编程器件的一般开发过程



8.3 可编程逻辑器件开发过程简介

为什么**FPGA**需要编程数据装载？

- **CPLD**采用**CMOS E²PROM**工艺制造，编程后，即使切断电源，其逻辑也不会消失，且可以在系统编程（**ISP**特性）。
- **FPGA**的**LUT**由数据选择器和**SRAM**构成，切断电源后，其逻辑会消失。所以**FPGA**需要外部的**PROM**保存编程数据。每次通电，自动将**PROM**中的编程数据装载到**FPGA**中。

编程条件

(1) 微机； (2) CPLD编程软件； (3) 专用编程电缆。

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，通过五线编程电缆接口与芯片连接。

将电缆接到计算机的并行口，通过编程软件发出编程命令，将编程数据文件(*JED)中的数据转换成串行数据送入芯片。

