

Unit 13

—Programmable Logic Devices

张彦航

School of Computer Science
Zhangyanhang@hit.edu.cn

几大PLD供应商

供应商	特 点	开发软件	
Altera	SOPC倡导者 (可编程芯片系统)	MaxPlusII 和 QuatersII	
Xinlinx	FPGA发明者	ISE 和 Vivado	
Lattace	ISP技术发明者	Diamond	
Actel	保密性好	Libero	

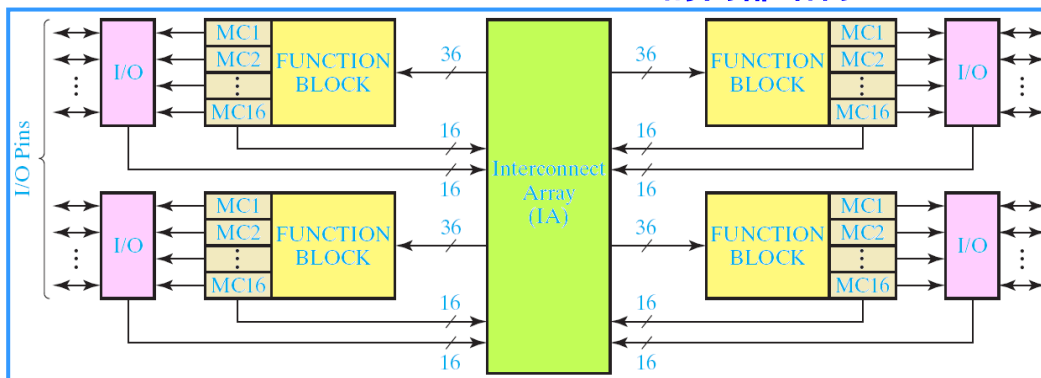
复杂可编程逻辑门阵列

CPLD

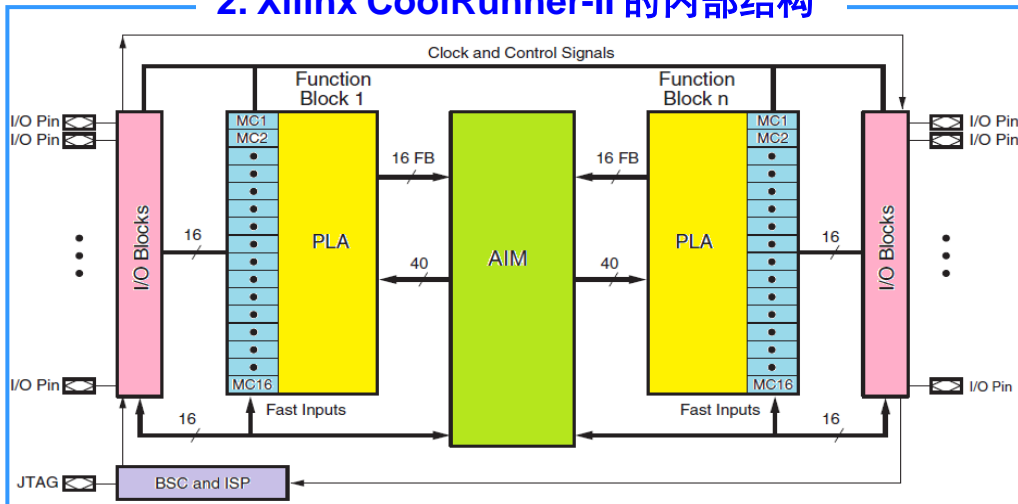
- 可编程的 I/O
- 可编程的逻辑阵列（宏单元）
- 可编程的内部互联资源

- 每块芯片只分成几块内部逻辑块（数十个输入端和一、二十个输出端）
- 乘积项的数量标志了CPLD的容量

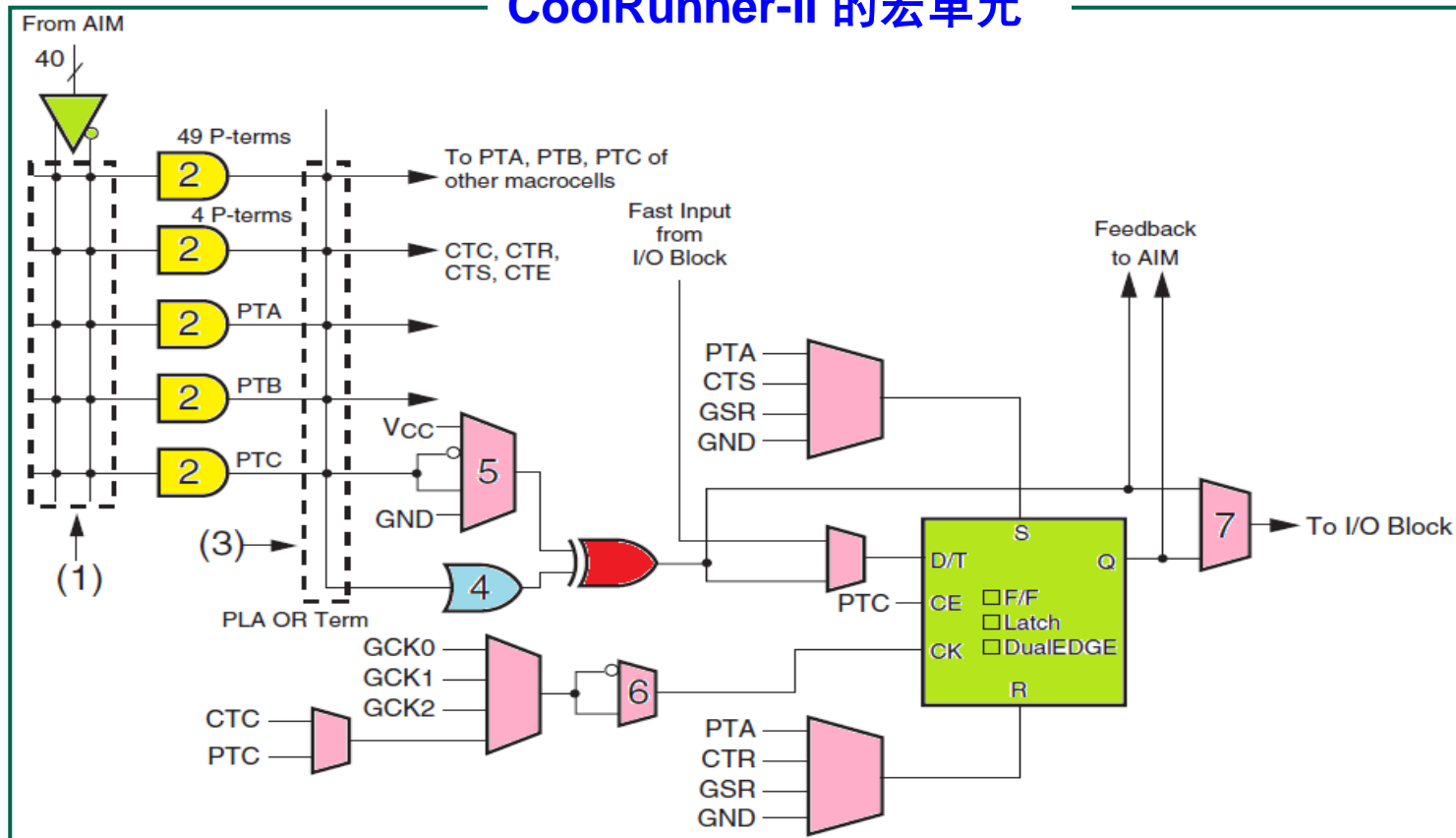
1. Xilinx XCR3064XL CPLD的内部结构



2. Xilinx CoolRunner-II 的内部结构



CoolRunner-II 的宏单元



FPGA（现场可编程逻辑门阵列）

FPGA

- ❖ 可编程的逻辑块CLB
- ❖ 可编程的 I/O模块
- ❖ 可编程的内部互联资源

FPGA分类

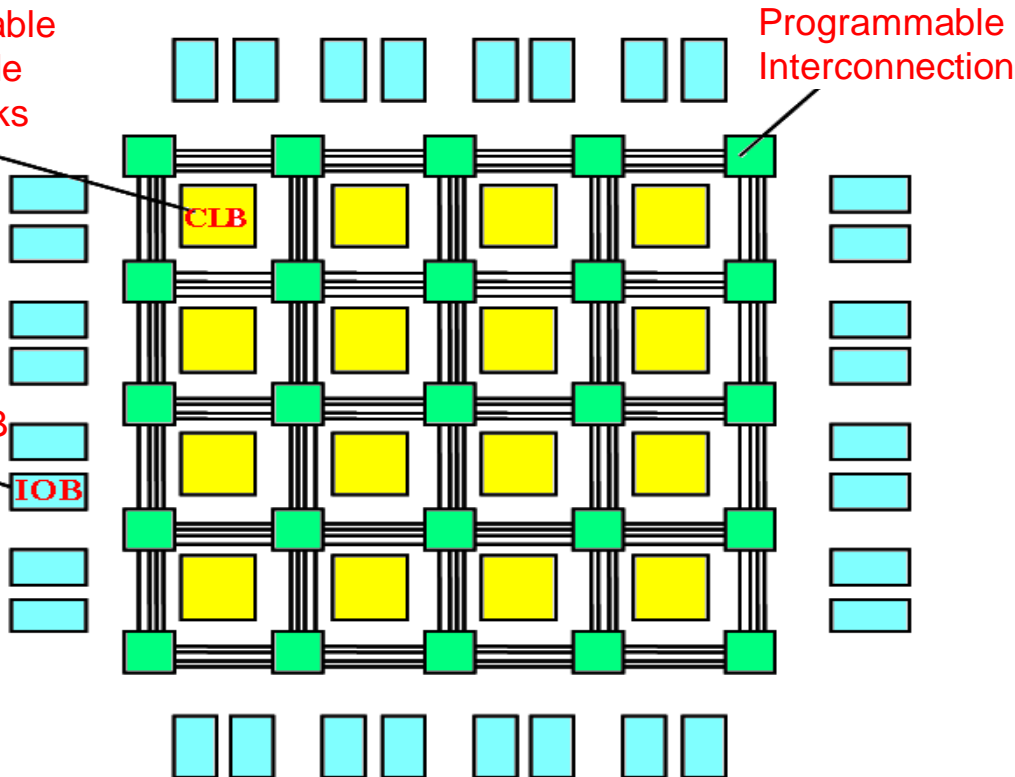
- 基于查找表技术的FPGA（LUT: Look-Up table）
- 基于反熔丝技术的FPGA

目前应用最多

Programmable
Configurable
Logic Blocks

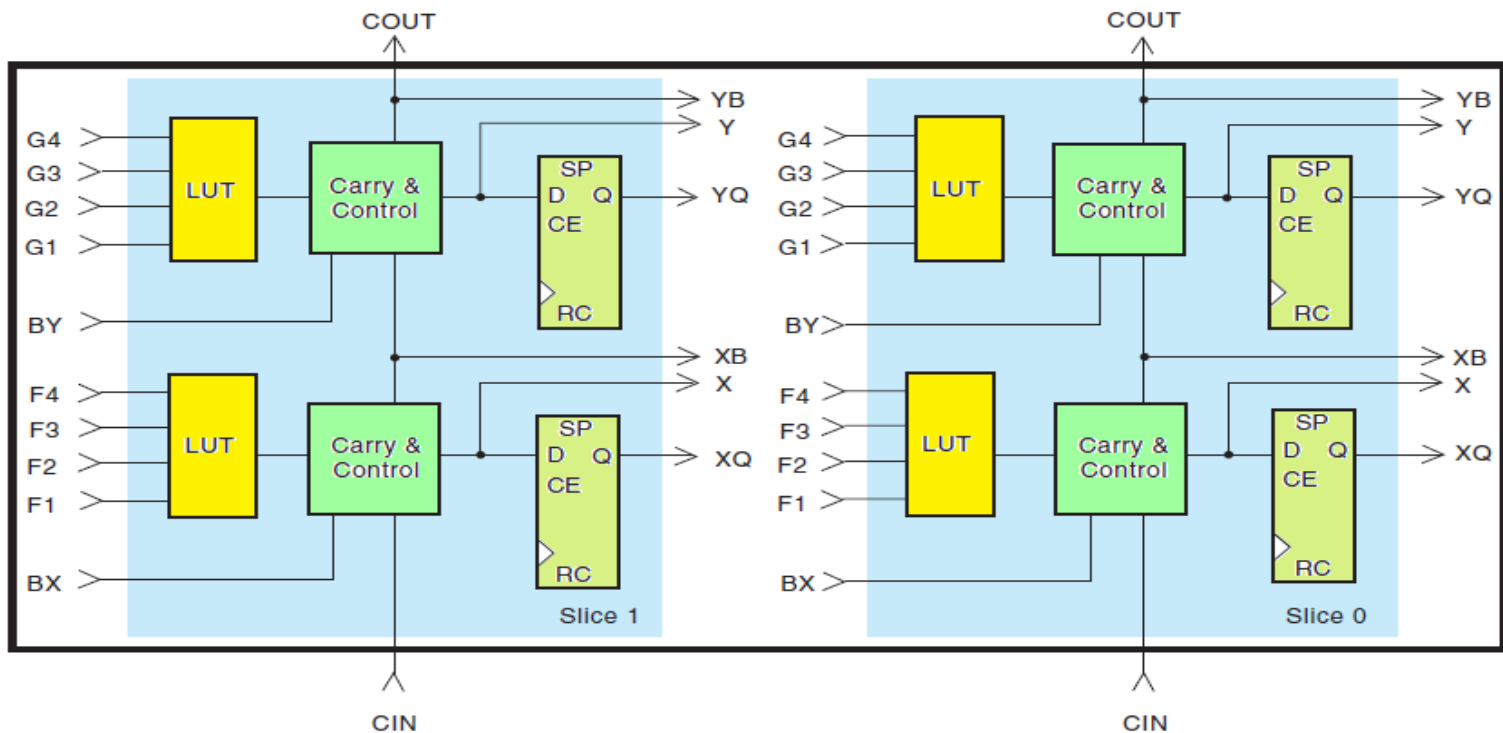
Programmable
Interconnection

Programmable IOB



FPGA（现场可编程逻辑门阵列）

Xilinx Virtex/ Spartan II CLB

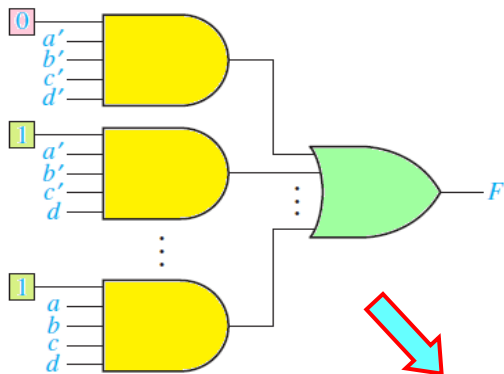


FPGA（现场可编程逻辑门阵列）

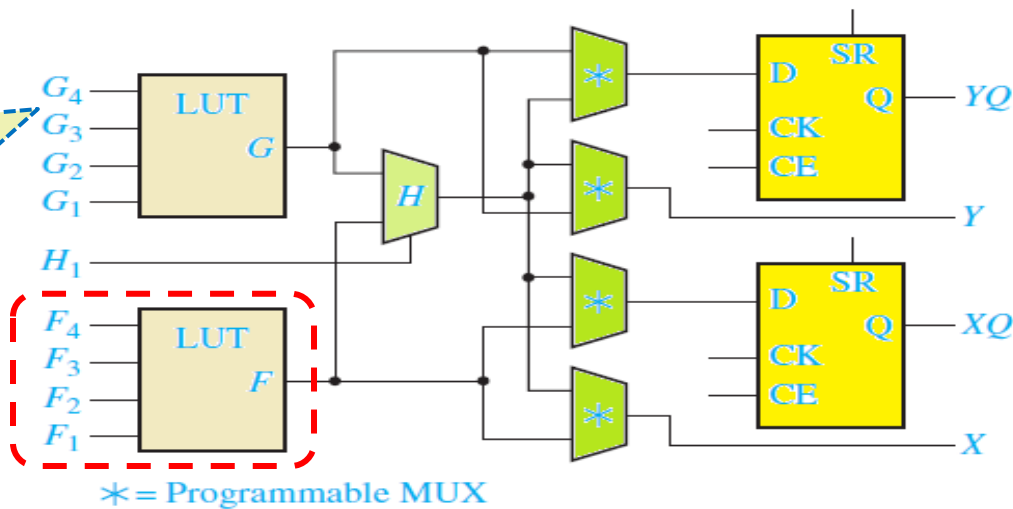
Implementation of LUT

<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>F</i>
0	0	0	0	0
0	0	0	1	1
⋮	⋮	⋮	⋮	⋮
1	1	1	1	1

每个LUT可以实现
一个4输入、1输出
的组合逻辑函数



简化的CLB

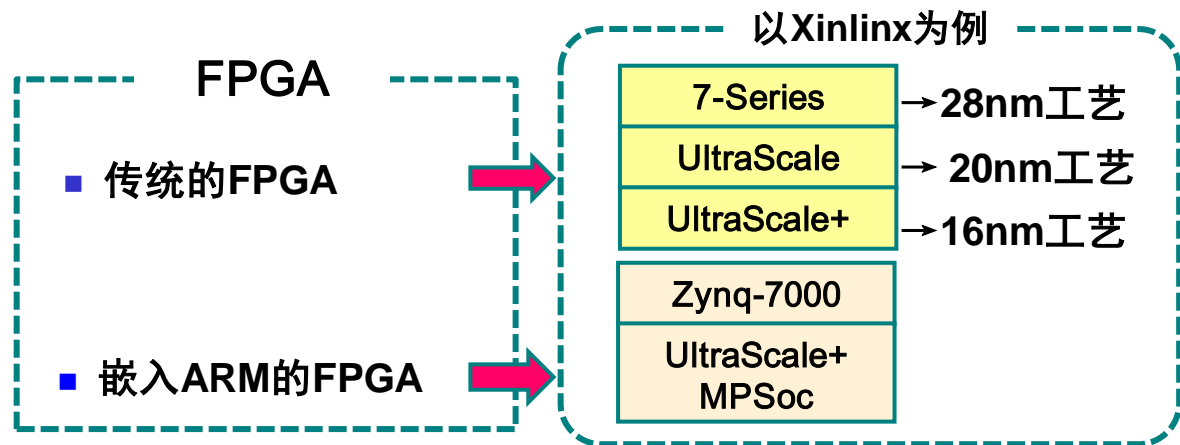


$$F = a'b'c'd + a'b'cd + a'bc'd + a'bcd' + ab'c'd + ab'cd' + abc'd' + abcd$$

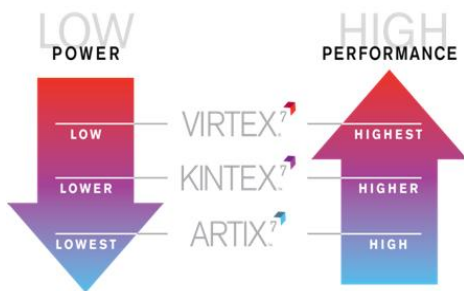
CPLD与FPGA的比较

	CPLD	FPGA
内部逻辑块粒度	粗粒度（数十个输入端和一、二十个输出端, 每块芯片只分成几块）	细粒度（每个LUT输入变量为4~8, 输出为1~2, 每块芯片中有几十到上千个这样的单元.）
资源类型	丰富的与或阵列	丰富的触发器资源
程序存储	基于EEPROM或FLASH, 系统掉电数据不会丢失	SRAM, 外挂EEPROM, 系统掉电易失
保密性	可加密	一般不能加密
设计优势	实现组合逻辑	实现时序逻辑

PROM,PLA,PAL,GAL,大多数CPLD属于阵列型, 基本结构是与或阵列
FPGA属于单元型, 基本结构是可编程的逻辑块



Xilinx 7-Series +SOC FPGAs家族



XILINX UNIFIED FPGA SERIES

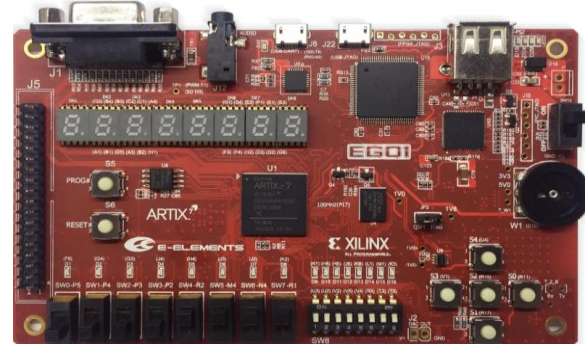
	ARTIX ⁷	KINTEX ⁷	VIRTEX ⁷	ZYNQ ⁷
最大的能力	最低功耗和成本	企业的最佳性价比	企业的最高性能	完全可编程SOC
逻辑单元	33K – 215K	70K – 478K	326K – 1,955K	28K – 444K
RAM块	12 Mb	34 Mb	65 Mb	27 Mb
DSP 片	40 – 700	240 – 1,920	700 – 3,960	80–2,020
峰值DSP性能	504 GMACS	2,450 GMACS	5,053 GMACS	2,662 GMACS
收发器	最多 16	最多32	最多88	最多16
收发器性能	6.6Gbps	12.5Gbps	12.5Gbps, 13.1Gbps 及28Gbps	6.6Gbps, 12.5Gbps
存储器性能	1066Mbps	1866Mbps	1866Mbps	1333Mbps
I/O 引脚	500	500	1,200	400
I/O 电压	3.3V 及以下	3.3V及以下 1.8V及以下	3.3V及以下 1.8V及以下	3.3V及以下 1.8V及以下

Artix-7 Family

Device	Logic Cells	Configurable Logic Blocks (CLBs)		DSP48E1 Slices ⁽²⁾	Block RAM Blocks ⁽³⁾			Clock Mgmt Tiles (CMTs) ⁽⁴⁾	PCIe ⁽⁵⁾	GTPs	XADC Blocks ⁽⁶⁾	Total I/O Banks ⁽⁷⁾	Max User I/O ⁽⁸⁾
		Slices ⁽¹⁾	Max Distributed RAM (Kb)		18Kb	36Kb	Max (Kb)						
XC7A20SL	16,000	2,500	208	60	60	30	1,080	3	0	0	1	5	216
XC7A35SL	32,909	5,142	453	120	130	65	2,340	3	0	0	1	5	216
XC7A50SL	52,480	8,200	688	180	190	95	3,420	4	0	0	1	6	300
XC7A75SL	71,642	11,194	974	240	250	125	4,500	4	0	0	1	6	300
XC7A20SLT	16,000	2,500	208	60	60	30	1,080	3	1	4	1	5	216
XC7A35SLT	32,909	5,142	453	120	130	65	2,340	3	1	4	1	5	216
XC7A50SLT	52,480	8,200	688	180	190	95	3,420	4	1	8	1	6	300
XC7A75SLT	71,642	11,194	974	240	250	125	4,500	4	1	8	1	6	300
XC7A100T	101,440	15,850	1,188	240	270	135	4,860	6	1	8	1	6	300
XC7A200T	215,360	33,650	2,888	740	730	365	13,140	10	1	16	1	10	500

FPGA : Xilinx Artix-7 XC7A35T
时钟 : 100MHz
配置方式 : USB-JTAG/SPI Flash
存储器 :
 SRAM : 2Mbit
 SPI Flash : N25Q032A
通用IO :
 Switch : x8
 LED : x16
 Button : x5
 DIP : x8
通用扩展IO : 32pin
音视频/显示 :
 7段数码管 : x8
 VGA视频输出接口
 Audio音频接口
通信接口 :
 UART : USB转UART
 Bluetooth : 蓝牙模块
模拟接口 :
 DAC : 8-bit分辨率
 XADC : 2路12bit 1Msps ADC

	Part Number	XC7A12T	XC7A15T	XC7A25T	XC7A35T
		Logic Cells	12,800	16,640	23,360
Logic Resources	Slices	2,000	2,600	3,650	5,200
	CLB Flip-Flops	16,000	20,800	29,200	41,600
	Maximum Distributed RAM (Kb)	171	200	313	400
Memory Resources	Block RAM/FIFO w/ ECC (36 Kb each)	20	25	45	50
	Total Block RAM (Kb)	720	900	1,620	1,800
	CMTs (1 MMCM + 1 PLL)	3	5	3	5
Clock Resources	Maximum Single-Ended I/O	150	250	150	250
I/O Resources	Maximum Differential I/O Pairs	72	120	72	120
	DSP Slices	40	45	80	90
	PCIe® Gen2 ⁽¹⁾	1	1	1	1
Embedded Hard IP Resources	Analog Mixed Signal (AMS) / XADC	1	1	1	1
	Configuration AES / HMAC Blocks	1	1	1	1
	GTP Transceivers (6.6 Gb/s Max Rate) ⁽²⁾	2	4	4	4
	Commercial	-1, -2	-1, -2	-1, -2	-1, -2
Speed Grades	Extended	-2L, -3	-2L, -3	-2L, -3	-2L, -3
	Industrial	-1, -2, -1L	-1, -2, -1L	-1, -2, -1L	-1, -2, -1L

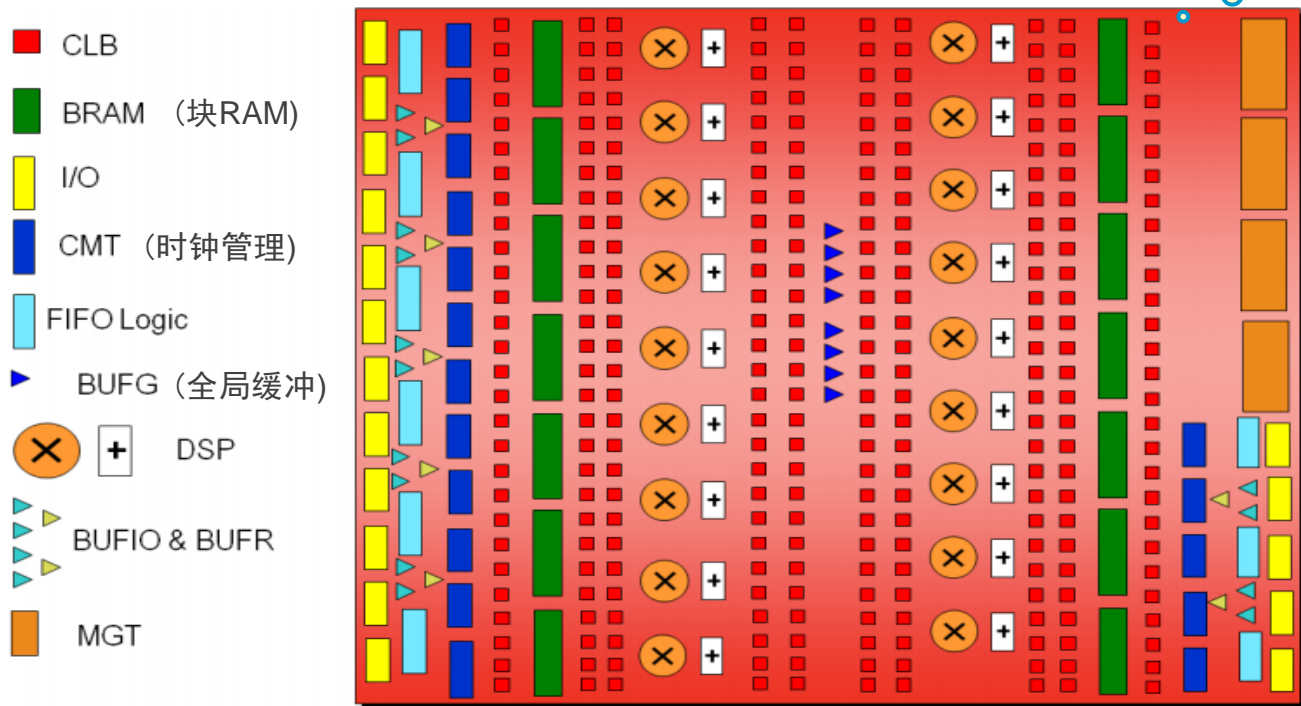


例：Xinlinx Artix-7 Xc7A35T
 ——口袋板

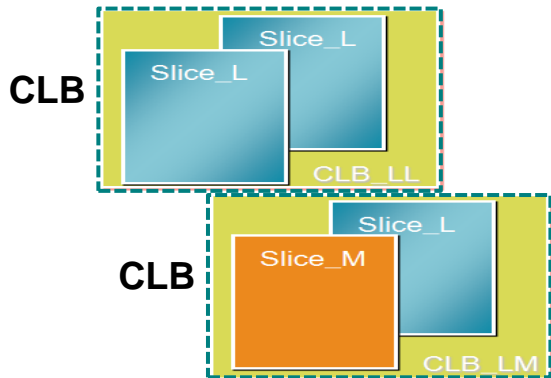
FPGA（现场可编程逻辑门阵列）

CLB是FPGA的主要逻辑单元

Artix-7 FPGA 架构



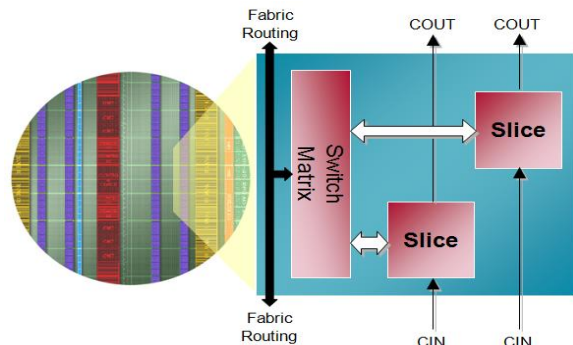
Xinlinx 7- Series FPGA



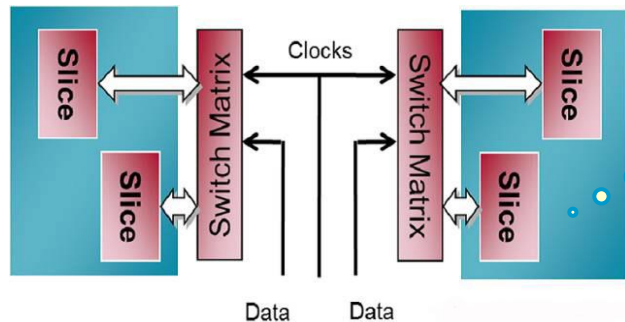
■ 每个CLB包含2个Slice，Slice分为2种：Slice_L和Slice_M。

- Slice_L: 具备基本功能，包括逻辑和算术
- Slice_M: 在基本逻辑功能的基础上可以扩展为分布式RAM或移位寄存器

■ 一个CLB可以有2个Slice_L，或者1个Slice_L+1个Slice_M。



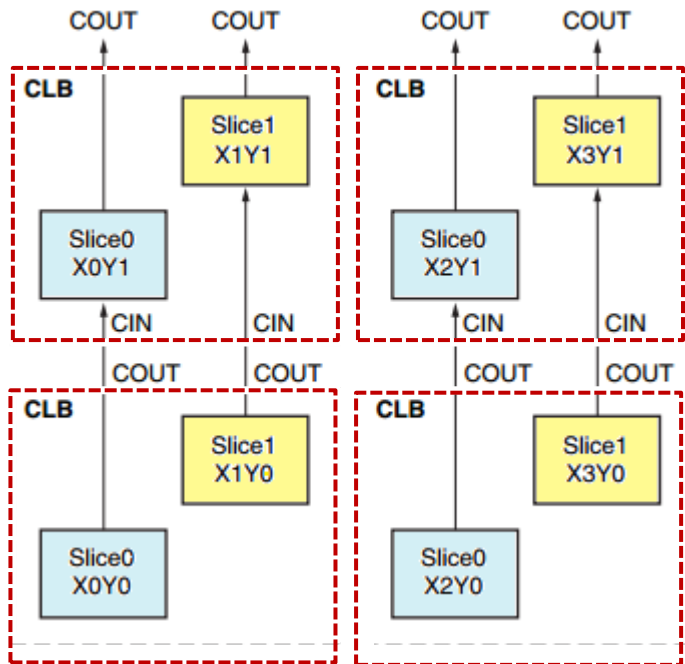
■ Slice 通过开关矩阵、进位链互连



可以共享时钟

■ CLB是成对、对称的连接在一起

Xilinx 7-Series FPGAs



CLB中的slice

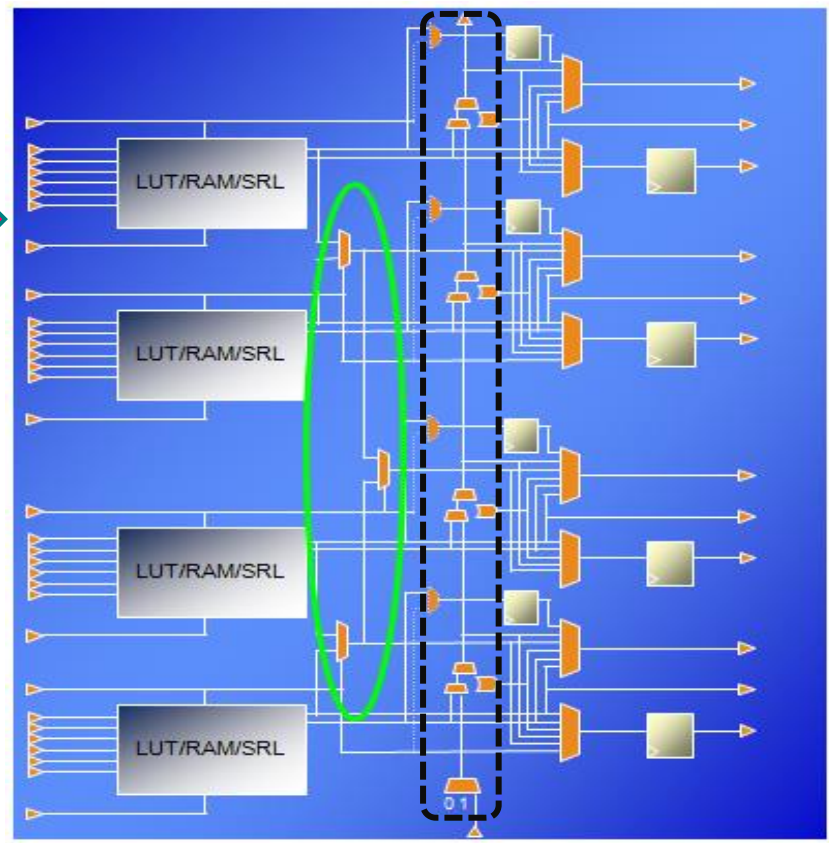
- ❑ CLB是成对、对称的连接在一起
- ❑ 同一CLB中的两片slices没有直接的线路连接，它们分属于两个不同的列。
- ❑ 每列拥有独立的快速进位链资源。

Xilinx 7-Series FPGAs

Slice 构成

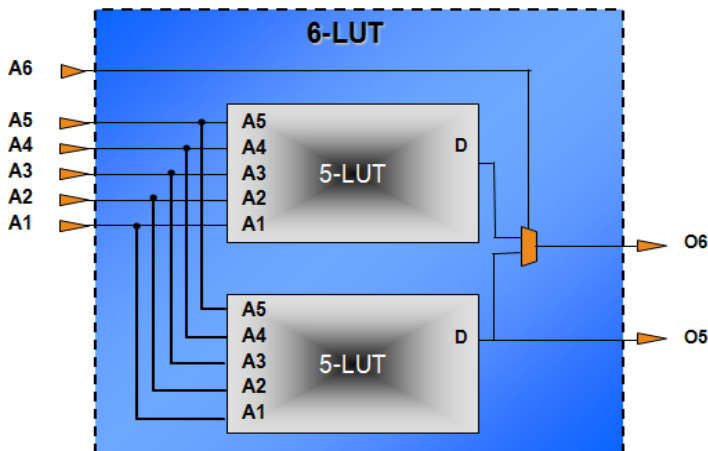
Slice

- ❑ 每个Slice有4个6输入-input 的LUT
- ❑ 数据选择器 (Multiplexers)
- ❑ 进位链 (Carry chains)
- ❑ 8个寄存器 (即: 触发器/锁存器)



Xilinx 7-Series FPGAs

6-Input LUT



- 6-input LUT内部由2个5-input LUT组成，有两个输出分别对应O6和O5。
- 可以表示任意一个6变量的逻辑函数，或者两个5变量的逻辑函数（前提是5个输入是必须相同的）
- 两个5-LUTs 的输出可以经过2选1数据选择器输出，或单独输出

利用LUT生成不同功能

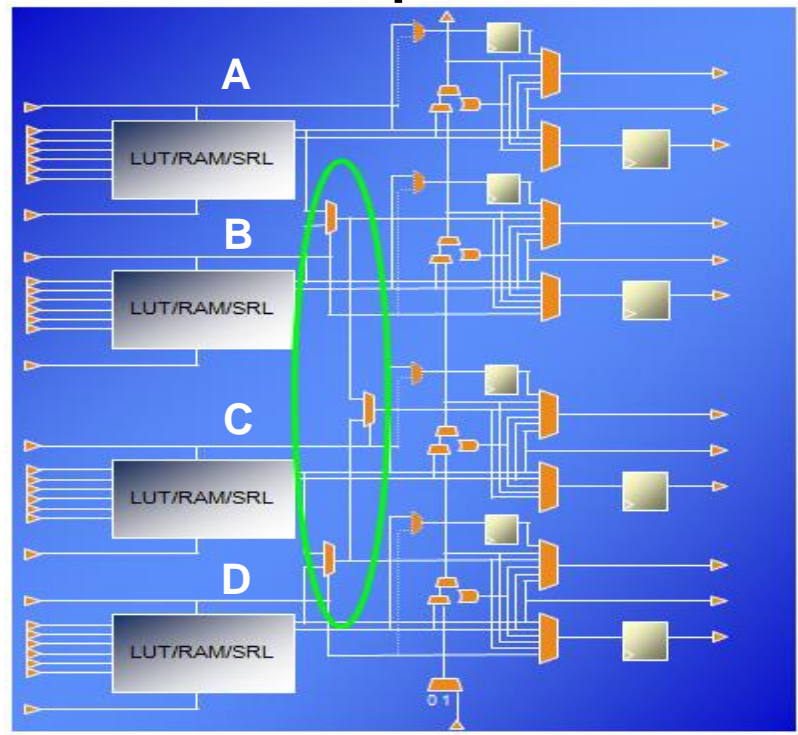
- 一个LUT能够实现1个4选1数据选择器，一个Slice能够实现4个4选1数据选择器。
- 两个LUT能够实现1个8选1数据选择器，一个Slice能够实现2个8选1数据选择器。
- 四个LUT能够实现1个16选1数据选择器，一个Slice能够实现1个16选1数据选择器。
- 每个LUT可配置成32-bit的移位寄存器。切记：直接使用LUT作为移位寄存器，一定不能有复位端口！

Xilinx 7-Series FPGAs

数据选择器

- 一个Slice中包含有3个多路复用器：**F7MUXA、F7MUXB和F8MUX**。
- 两个LUT的输出可以经过一个数据选择器进行控制。
- **F7MUXA**组合LUT A和LUT B成为7输入LUT，**F7MUXB**组合LUT C和LUT D成为7输入LUT，而**F8MUX**组合1个Slice中的4个LUT成为8输入LUT。
- 通过**F7MUXA、F7MUXB和F8MUX**的搭配，1个Slice可以实现4:1、8:1和16:1多路复用器
- 数据选择器的输出可以驱动组合逻辑，也可以驱动触发器（时序逻辑）

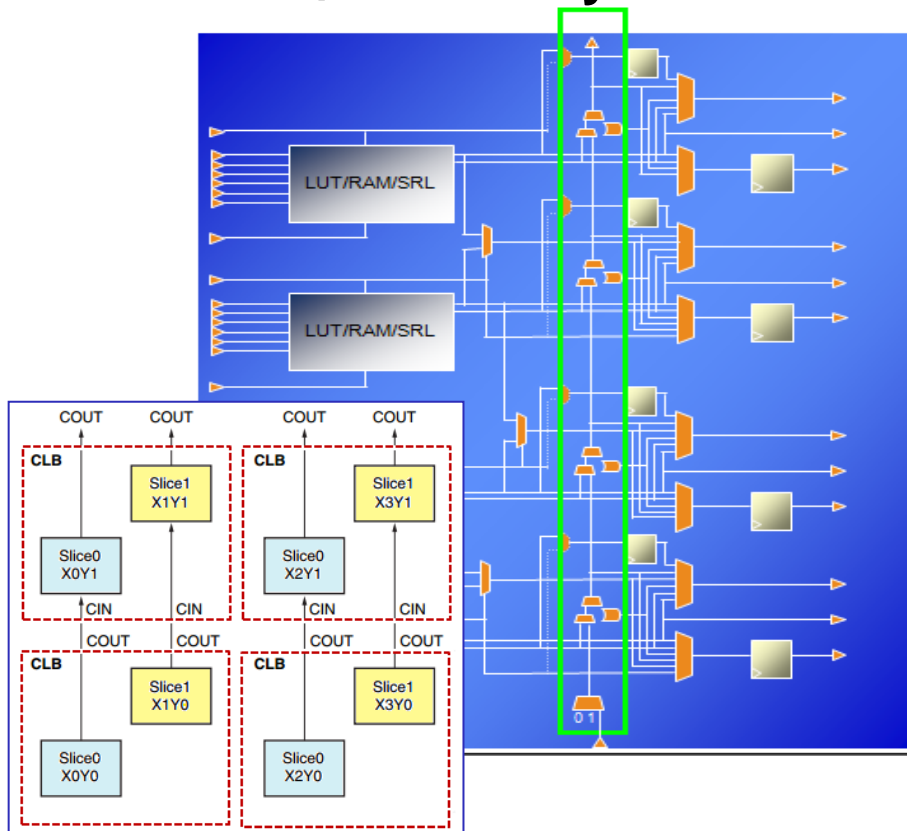
数据选择器 (multiplexer)



Xilinx 7-Series FPGAs

- 进位链是一种可以提高算术功能（如加法器，累加器，减法器、比较器）的专用硬件资源，在两个slice间实现快速进位。
- 同一CLB中的两片slices没有直接的线路连接，它们分属于两个不同的列。每列拥有独立的快速进位链资源。
- 每个SLICE有4bit的进位链。每bit都由一个进位MUX和一个异或门组成，可在实现加法/减法器时生成进位逻辑。
- 进位链逻辑走向垂直向上，设计时，最起始的bit应该被放置在进位链的底部，从而节省很多的资源。

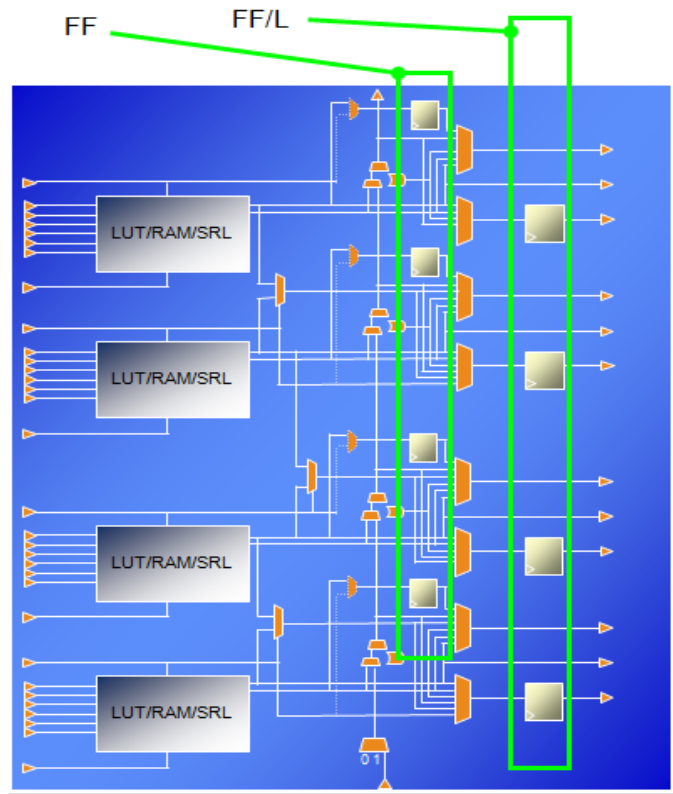
进位链（Carry Chain）



Xilinx 7-Series FPGAs

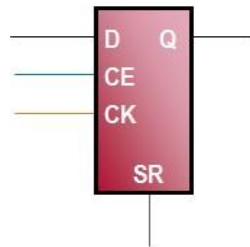
- 每个slice 有4个flip-flop/latches (FF/L)
- 第1列的register: 只能作为Flip-Flop使用, 第2列的register: 既能作为Flip-Flop也能作为Latch使用。
- Flip-flops和latches共享相同的控制信号: CLK, SR, CE等。
 - 意味着: 如果一个组中的一个flip-flop用了时钟使能信号, 那么所有其他的Flip-Flop必须使用相同的时钟使能或者不用时钟使能。如果一个组中的一个flip-flop用了置位或者复位信号, 那么所有其他的Flip-Flop必须使用相同的置位或者复位信号。
 - 同一个Slice中的Flip-Flops会在同一时间进入复位或者置位的状态。
 - 写代码时, 复位/置位选择**同步高电平**有效。

Flip-Flops and Flip-Flop/Latches



FPGA（现场可编程逻辑门阵列）

- 所有触发器都是D触发器
- 所有的触发器都有：
 - 一个时钟输入（CLK）
 - 一个使能信号（CE）
 - 一个高有效的复位信号（SR），可以基于用户的编码风格被配置成同步置位/复位、异步置位或者复位、清零端口。
- XILINX器件大部分都是**高有效**。意味着：若使用低有效，需要额外占用LUT构成反转逻辑。



```
always @ (posedge CLK or posedge RST )  
begin  
    if (RST)  
        Q <= 1'b0;  
    else  
        Q <= D;  
End
```

异步复位
(高电平)

```
always @ (posedge CLK)  
begin  
    if (RST)  
        Q <= 1'b0;  
    else  
        Q <= D;  
end
```

同步复位
(高电平)：
复位信号不能
在敏感列表里

FPGA设计流程

1. 原理框图的设计

2. 设计输入

3. 逻辑综合

4. 仿真

5. 下载到PLD芯片，调试验证

6. 发现错误，在计算机上修改设计，重复上述过程。

□ 按功能模块划分的系统框图

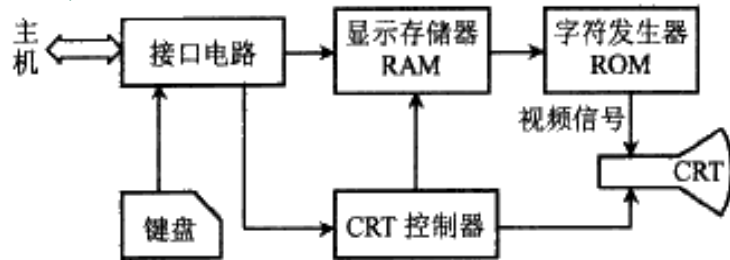
□ 不涉及具体实现

两种方式：

□ 逻辑图输入方式

□ HDL输入方式

原理框图设计



数字系统设计趋势

EDA软件



计算机

编程



空白PLD

数字系统

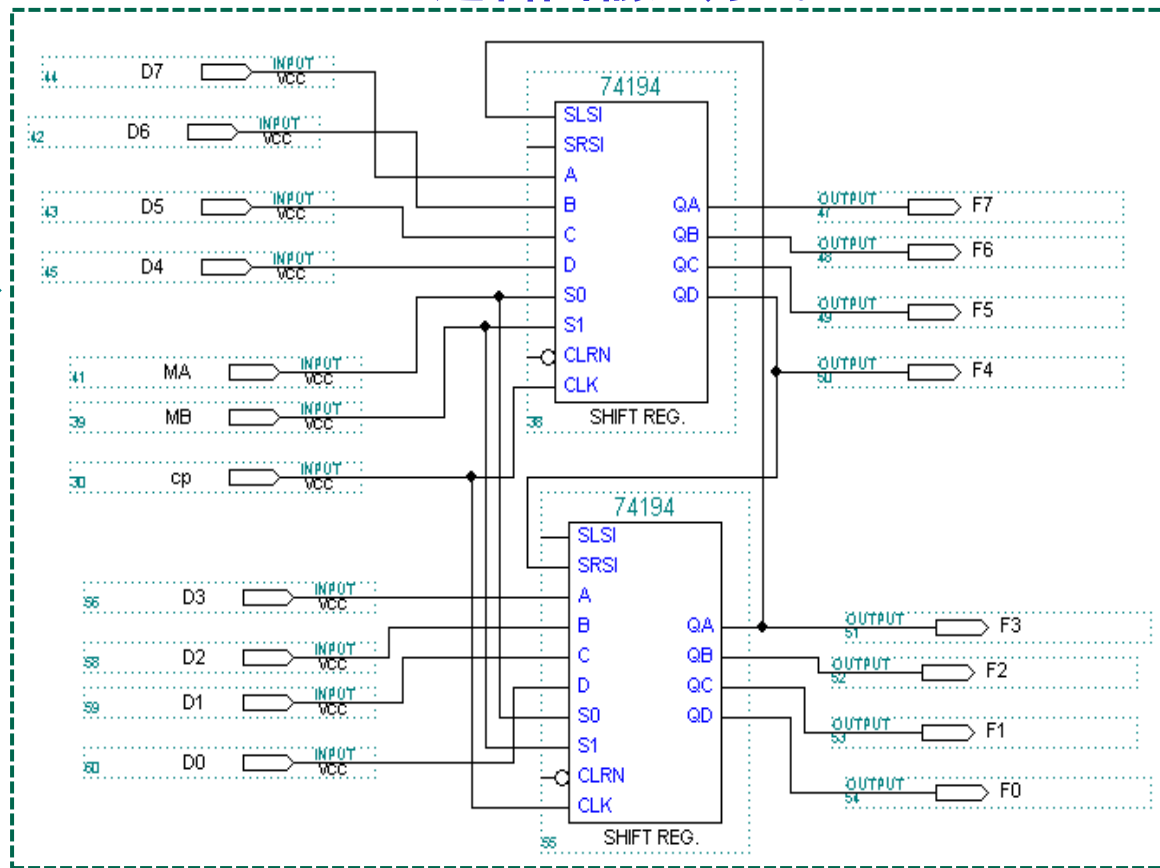
明确各个模块之间的关系，确定输入和输出。

FPGA设计流程

逻辑图输入方式

FPGA设计流程

1. 原理框图的设计
2. 设计输入
3. 逻辑综合
4. 仿真
5. 下载到PLD芯片，调试验证
6. 发现错误，在计算机上修改设计，重复上述过程。



□ HDL输入方式

```
ENTITY incoder IS
PORT ( G, a,b : IN STD_LOGIC ;
      Y : OUT std_logic_vector(3 downto 0));
END incoder;

ARCHITECTURE bhv1 OF incoder IS
  signal inab: std_logic_vector(1 downto 0);
  BEGIN
    inab<=b & a;
    PROCESS (inab,G)
    BEGIN
      IF G <='0' THEN
        CASE inab IS
          WHEN "00" => Y<="1110";
          WHEN "01" => Y<="1101";
          WHEN "10" => Y<="1011";
          WHEN "11" => Y<="0111";
          WHEN OTHERS => NULL;
        END CASE;
      ELSE
        Y<="1111";
      END IF;
    END PROCESS ;
  END bhv1;
```

VHDL
(2-4 译码器)

```
module full_add1
(
  input a, b, cin,
  output sum, cout
);
  assign sum= a^b^cin;
  assign cout= ((a^b)&cin) | (a&b);
endmodule
```

Verilog (全加器)

FPGA设计流程

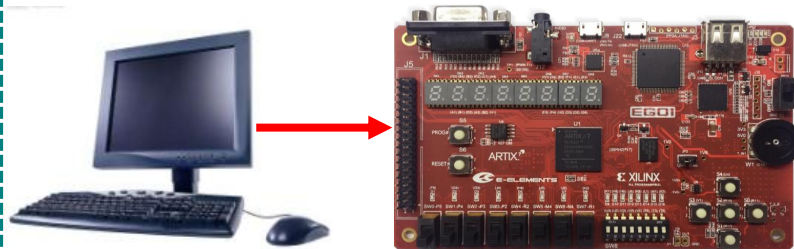
1. 原理框图的设计
2. 设计输入
3. 逻辑综合
4. 仿真
5. 下载到PLD芯片, 调试验证
6. 发现错误, 在计算机上修改设计, 重复上述过程。

FPGA设计流程

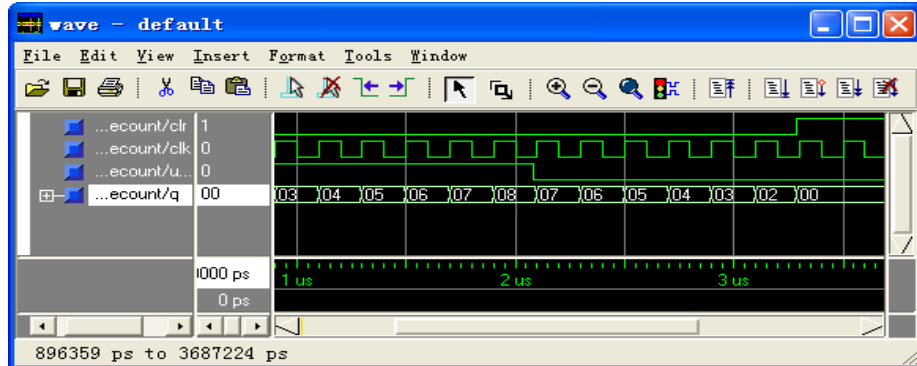
FPGA设计流程

1. 原理框图的设计
 2. 设计输入
 3. 逻辑综合
 4. 仿真
 5. 下载到PLD芯片，调试验证
 6. 发现错误，在计算机上修改设计，重复上述过程。
- 可综合的HDL代码
 - RTL代码实现是否合理
 - 对FPGA器件特点的理解

□ 下载调试



□ 仿真



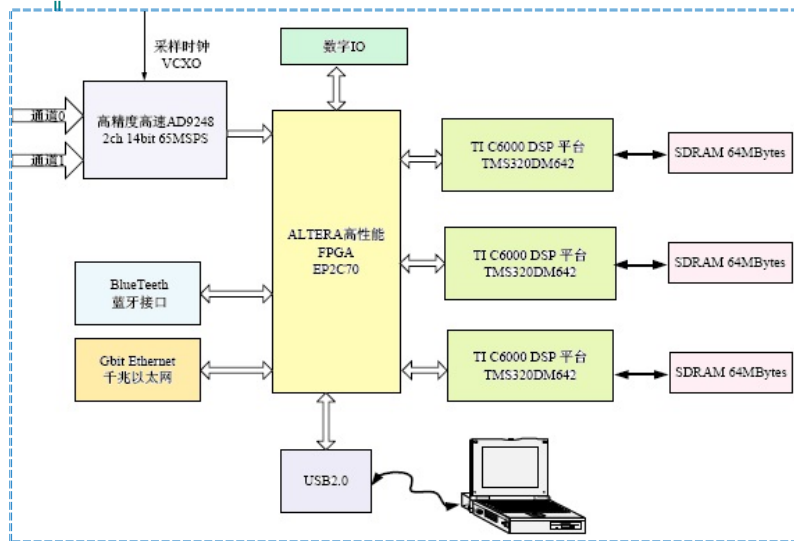
FPGA的典型应用

□ 典型应用1——高速接口电路

(1) 粘合功能：高速的A/D和D/A转换芯片（如视频Decoder或者Encoder，不能与通用的MCU或者DSP直接接口。利用FPGA可以完成粘合逻辑功能）

(2) 简化外围电路设计：设计中如果需要的接口比较多，就需要较多的外围芯片，体积、功耗都比较大。使用FPGA，接口逻辑可以在FPGA内部实现，大大简化了外围电路的设计

(3) 实现各种存储接口的控制器：利用FPGA可以实现各种存储接口的控制器（如SDRAM、SRAM、Flash等）



FPGA的典型应用

□ 典型应用2——高速数学运算与数据处理

(1) 替代传统的多片DSP并联的多处理器系统:

多处理器系统的问题：设计复杂度及系统功耗高，系统稳定性受到影响。FPGA支持并行计算（例如，高清视频编码算法，采用的DSP芯片需要4片，FPGA只需要1片）

(2) 新兴应用：人工智能、大数据处理、云计算加速

视频会议处理



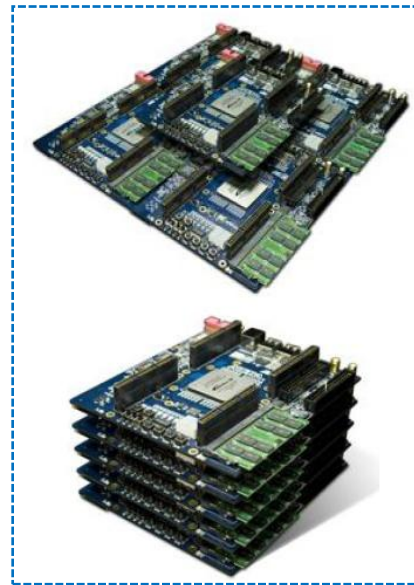
FPGA的典型应用

▣ 典型应用3——ASIC/SoC的原型验证

ASIC/SoC的原型验证：一种在FPGA上搭建ASIC/SoC设计原型的方法,可以方便的进行硬件验证和早期软件开发。

(1) 缩短验证时间，降低开发成本：FPGA具有可编程的特点，可以反复修改逻辑设计，直至达到满意的结果，加速开发速度，降低开发成本。

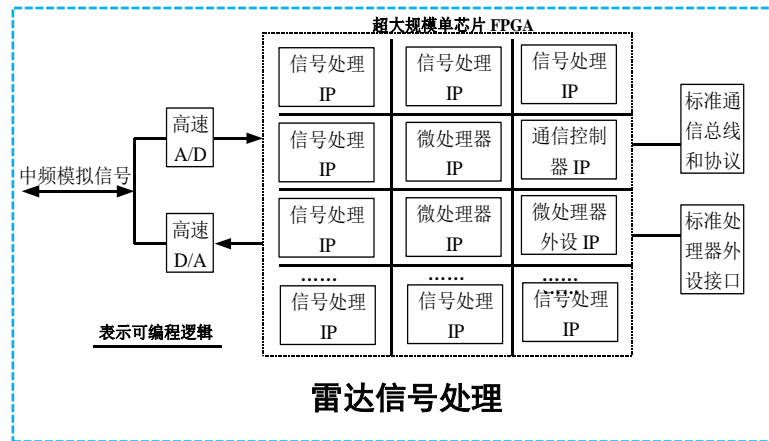
(2) 基于FPGA的原型验证可以比软件仿真速度高出4~6个数量级，而且还可以提高流片成功率，并为软件开发提供了硬件平台，加速了软件的开发速度。



FPGA的典型应用

□ 其它领域应用

- (1) 汽车电子领域：如网关控制器/车用PC机、远程信息处理系统。
- (2) 军事领域：如安全通信、雷达和声纳、电子战。
- (3) 测试和测量领域：如通信测试和监测、半导体自动测试设备、通用仪表
- (4) 消费产品领域：如显示器、投影仪、数字电视和机顶盒、家庭网络。
- (5) 医疗领域



通用母板设计



交通信息处理

