数字IC设计经典笔试题之【FPGA基础】

**FPGA**

# 1：系统最高速度计算（最快时钟频率）和流水线设计思想：

同步电路的速度是指同步系统时钟的速度，同步时钟愈快，电路处理数据的时间间隔越短，电路在单位时间内处理的数据量就愈大。假设Tco是触发器的输入数据被时钟打入到触发器到数据到达触发器输出端的延时时间(Tco=Tsetpup+Thold)；Tdelay是组合逻辑的延时；Tsetup是Ｄ触发器的建立时间。假设数据已被时钟打入D触发器，那么数据到达第一个触发器的Ｑ输出端需要的延时时间是Tco，经过组合逻辑的延时时间为Tdelay，然后到达第二个触发器的Ｄ端，要希望时钟能在第二个触发器再次被稳定地打入触发器，则时钟的延迟必须大于Tco＋Tdelay＋Tsetup，也就是说最小的时钟周期Tmin=Tco＋Tdelay＋Tsetup，即最快的时钟频率Fmax=1/Tmin。FPGA开发软件也是通过这种方法来计算系统最高运行速度Fmax。因为Tco和Tsetup是由具体的器件工艺决定的，故设计电路时只能改变组合逻辑的延迟时间Tdelay，所以说缩短触发器间组合逻辑的延时时间是提高同步电路速度的关键所在。

由于一般同步电路都大于一级锁存，而要使电路稳定工作，时钟周期必须满足最大延时要求。故只有缩短最长延时路径，才能提高电路的工作频率。可以将较大的组合逻辑分解为较小的N块，通过适当的方法平均分配组合逻辑，然后在中间插入触发器，并和原触发器使用相同的时钟，就可以避免在两个触发器之间出现过大的延时，消除速度瓶颈，这样可以提高电路的工作频率。这就是所谓"流水线"技术的基本设计思想，即原设计速度受限部分用一个时钟周期实现，采用流水线技术插入触发器后，可用N个时钟周期实现，因此系统的工作速度可以加快，吞吐量加大。注意，流水线设计会在原数据通路上加入延时，另外硬件面积也会稍有增加。

# 2：时序约束的概念和基本策略？

时序约束主要包括周期约束，偏移约束，静态时序路径约束三种。通过附加时序约束可以综合布线工具调整映射和布局布线，使设计达到时序要求。

附加时序约束的一般策略是先附加全局约束，然后对快速和慢速例外路径附加专门约束。附加全局约束时，首先定义设计的所有时钟，对各时钟域内的同步元件进行分组，对分组附加周期约束，然后对FPGA/CPLD输入输出PAD附加偏移约束、对全组合逻辑的PADTOPAD路径附加约束。附加专门约束时，首先约束分组之间的路径，然后约束快、慢速例外路径和多周期路径，以及其他特殊路径。

# 3：附加约束的作用？

1：提高设计的工作频率（减少了逻辑和布线延时）；

2：获得正确的时序分析报告；（静态时序分析工具以约束作为判断时序是否满足设计要求的标准，因此要求设计者正确输入约束，以便静态时序分析工具可以正确的输出时序报告）；

3：指定FPGA/CPLD的电气标准和引脚位置。

# 4：FPGA设计工程师努力的方向：

SOPC，高速串行I/O，低功耗，可靠性，可测试性和设计验证流程的优化等方面。

随着芯片工艺的提高，芯片容量、集成度都在增加，FPGA设计也朝着高速、高度集成、低功耗、高可靠性、高可测、可验证性发展。芯片可测、可验证，正在成为复杂设计所必备的条件，尽量在上板之前查出bug，将发现bug的时间提前，这也是一些公司花大力气设计仿真平台的原因。另外随着单板功能的提高、成本的压力，低功耗也逐渐进入FPGA设计者的考虑范围，完成相同的功能下，考虑如何能够使芯片的功耗最低，据说Altera、Xilinx都在根据自己的芯片特点整理如何降低功耗的文档。高速串行IO的应用，也丰富了FPGA的应用范围，像Xilinx的v2pro中的高速链路也逐渐被应用。

# 5：FPGA芯片内有哪两种存储器资源？

FPGA芯片内有两种存储器资源：一种叫BLOCKRAM,另一种是由LUT配置成的内部存储器（也就是分布式RAM）。BLOCKRAM由一定数量固定大小的存储块构成的，使用BLOCKRAM资源不占用额外的逻辑资源，并且速度快。但是使用的时候消耗的BLOCKRAM资源是其块大小的整数倍。

# 6：FPGA设计中对时钟的使用？（例如分频等）

FPGA芯片有固定的时钟路由，这些路由能有减少时钟抖动和偏差。需要对时钟进行相位移动或变频的时候，一般不允许对时钟进行逻辑操作，这样不仅会增加时钟的偏差和抖动，还会使时钟带上毛刺。一般的处理方法是采用FPGA芯片自带的时钟管理器如PLL,DLL或DCM，或者把逻辑转换到触发器的D输入（这些也是对时钟逻辑操作的替代方案）。

# 7：FPGA设计中如何实现同步时序电路的延时？

首先说说异步电路的延时实现：异步电路一般是通过加buffer、两级与非门等来实现延时（我还没用过所以也不是很清楚），但这是不适合同步电路实现延时的。

在同步电路中，对于比较大的和特殊要求的延时，一般通过高速时钟产生计数器，通过计数器来控制延时；对于比较小的延时，可以通过触发器打一拍，不过这样只能延迟一个时钟周期。

# 8：FPGA中可以综合实现为RAM/ROM/CAM的三种资源及其注意事项？

三种资源：BLOCKRAM，触发器（FF），查找表（LUT）；

注意事项：

a：在生成RAM等存储单元时，应该首选BLOCKRAM资源；其原因有二：第一：使用BLOCKRAM等资源，可以节约更多的FF和4-LUT等底层可编程单元。使用BLOCKRAM可以说是“不用白不用”，是最大程度发挥器件效能，节约成本的一种体现；第二：BLOCKRAM是一种可以配置的硬件结构，其可靠性和速度与用LUT和REGISTER构建的存储器更有优势。

b：弄清FPGA的硬件结构，合理使用BLOCKRAM资源；

c：分析BLOCKRAM容量，高效使用BLOCKRAM资源；

d：分布式RAM资源（DISTRIBUTERAM）

# 9：查找表的原理与结构？

查找表（look-up-table）简称为LUT，LUT本质上就是一个RAM。目前FPGA中多使用4输入的LUT，所以每一个LUT可以看成一个有4位地址线的16x1的RAM。当用户通过原理图或HDL语言描述了一个逻辑电路以后，PLD/FPGA开发软件会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。

# 10：IC设计前端到后端的流程和EDA工具？

设计前端也称逻辑设计，后端设计也称物理设计，两者并没有严格的界限，一般涉及到与工艺有关的设计就是后端设计。

a：规格制定：客户向芯片设计公司提出设计要求。

b：详细设计：芯片设计公司（Fabless）根据客户提出的规格要求，拿出设计解决方案和具体实现架构，划分模块功能。目前架构的验证一般基于systemC语言，对价后模型的仿真可以使用systemC的仿真工具。例如：CoCentric和VisualElite等。

c：HDL编码：设计输入工具：ultra，visualVHDL等

d：仿真验证：modelsim

e：逻辑综合：synplify

f：静态时序分析：synopsys的PrimeTime

g：形式验证：Synopsys的Formality.

**-END-**