1 **DFT**的描述是正确的是

DFT测试过程中通常会消耗大量动态功耗

DFT主要目的是发现芯片在生产过程出现的缺陷 X

测试覆盖电路时序

寄存器扫描链是一种常用的DFT技术

2 16k\*8位存储器。地址线数据线总和

8+14

3 下面说法正确的是

Source clock latency 也属于FPGA IO接口约束 X

FPGA chip scope设置同样的采样速率，如果想一次观测更长时间的信号波形，可以将采样时钟更换成更高频率的时钟 X

基于SRAM的FPGA器件，每次上电必须重新配置

乘法器在FPGA上必须使用dsp X

4 下面那种逻辑可以实现（A XOR B） OR （C AND D）

NAND

5 **流水线**的理解，错误的是

流水线设计会导致原有通路延时增加

关键路径插流水线，能够提高系统时钟频率

流水线设计思想，是使用面积换速度 X

会消耗较多的组合逻辑资源

6 负反馈电路

电压并联反馈

电压串联

电流并联

电流串联

电容并联 X

7 芯片静态功耗影响最大的是 （ 频率 ）

8 采用（ 补码 ）可以将减法运算转换成加法运算

9 组合电路中的冒险是由（ 多输出 ）引起的

同一信号经不同的路径传输后，到达电路中某一会合点的时间有先有后，这种现象称为逻辑竞争，而因此产生输出干扰脉冲的现象称为冒险

10 **多比特**数据跨时钟域，错误的是

连续变化信号，转化格雷码发送，接受方再转换二进制

发送方数据写入异步FIFO，接收方读出

本地时钟同步两拍 X

发送方给出握手请求，接收方回复，发送方撤销数据

11 指令系统中程序控制类指令功能

实现主存与CPU间数据传送？

实现程序执行顺序改变

算数逻辑运算？

堆栈？

12 在不增加pipeline的情况下，如何解决critical path的setup时序不满足条件？

将部分组合逻辑电路撤移前级path上

降低时钟频率

更先进工艺 X

path插寄存器

13 哪些是非易失性存储器

Flash

Sram X

Dram X

Eprom

14 **异步**处理，哪些正确？

单比特信号打两拍可以避免亚稳态

异步FIFO采用格雷码为了提高速度 X

异步处理需要考虑发送接受间频率 X

静态配置信号不做异步处理

15 16点FFT正确的是 （4级基二蝶形）

每级8个蝶形算法

每个蝶形算法需要1次复数加法 X

每个蝶形算法需要1次复数乘法

共有4级分解

（每一级均包含有N/2次蝶形计算。而每一个蝶形运算包含了1次复数乘法，2次复数加法）

16 速度优化？

流水线

资源共享 X

串行 X

关键路径优化

17 PSK QAM调制

18 c程序中全局数组，编译后可能属于哪个段

Data

bss

19 a为并行所占比例，n为并行处理节点数，并行计算加速比为：

1/（1-a+a/n）

20 FPGA时序检查对于异步复位电路的时序分析叫\_\_\_和\_\_\_

**恢复时间检查**和**移除时间检查**

21 正则匹配

22 write-back cache和 write-through cache区别

Write-through（直写模式）在数据更新时，同时写入缓存Cache和后端存储。此模式的优点是操作简单；缺点是因为数据修改需要同时写入存储，数据写入速度较慢。

Write-back（回写模式）在数据更新时只写入缓存Cache。只在数据被替换出缓存时，被修改的缓存数据才会被写到后端存储。此模式的优点是数据写入速度快，因为不需要写存储；缺点是一旦更新后的数据未被写入存储时出现系统掉电的情况，数据将无法找回。

1 数字通信特点，错误的是

抗干扰能力强，噪声不积累

易于加密，保密性好

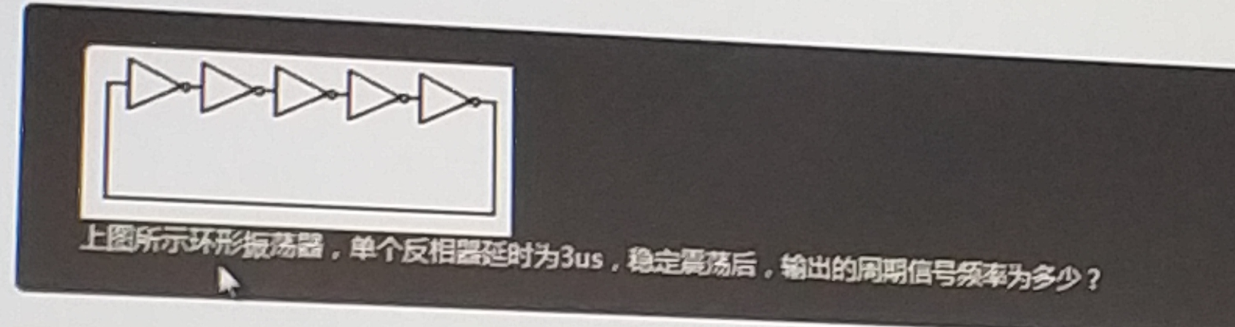
易于集成，使通信设备微型化

比模拟通信占据更窄的系统频带，系统设备简单，对同步要求更低 X

2 perl脚本获取命令行参数的符号

@ARGV

3



2N \* tpd = 5\*3\*2 = 30

4 cache-主存-缓存三级存储系统中各级存储器的作用，速度，容量的描述

cache速度快，容量小，cpu内部，匹配cpu与内存间的速度

5 N=32的基2时间抽取法FFT运算流图中，从x（n）到X（k）需（）级蝶形运算

5

6 奈奎斯特采样定理，为了不失真，采样频率f与信号最高频率fs的关系

f > 2fs

7 **异步复位**哪个正确

寄存器的时钟状态对是否能复位没有影响

寄存器的时钟状态对是否能解复位没有影响 X

是否有毛刺没有影响 X

异步复位不需要同步到对应的时钟域上 X

8 判断电路是否存在竞争冒险的方法

代数法/卡诺图法/实验法

9 latch与Flip-Flop特性描述正确的是？

latch与FF都属于时序逻辑

FF只会在时钟触发沿采样当前输入，产生输出

latch无时钟输入 X

latch可能产生毛刺

latch容易产生毛刺

锁存电平触发，FF边沿触发

latch非同布，FF同步

10 哪些手段可以降低SRAM的动态功耗

不访问SRAM时关闭时钟

不访问SRAM时地址线不翻转

不访问SRAM时写数据不翻转

Power down X

11 **FPGA**资源里的RAM可以配置为哪几种工作模式（）（）（）?

并行主模式为一片FPGA加一片EPROM的方式;

主从模式可以支持一片PROM编程多片FPGA;

串行模式 采用串行PROM编程FPGA;

外设模式可以将FPGA作为微处理器的外设,由微处理器对其编程。

12 已知两级寄存器中含有组合逻辑，逻辑延时为T\_cdelay，寄存器建立时间T\_setup，保持时间T\_hold，传输延时T\_c2q，寄存器时钟周期T\_clk最小为（），逻辑延时最大为（）

Tdata\_path\_min - Thold >= Tclk\_MAX

Tdata\_path\_MAX + Tsetup <= Tclk\_min + Tperiod

T\_clk = T\_c2q + T\_cdelay + Tsu

Tcomb + Tco > Thold

13 c语言实现整数到ASCII码字符串转换

14 SoC系统某层AXI总线上同时具有数据位宽为32bit的AXI，APB，AHB slave接口。系统要求总线要挂512KB的SRAM，目前有两块256KB，16bit位宽的SRAM

1)结合性能，总线和SRAM特性等因素，对wrapper的选型进行评估

2)负责SRAM wrapper的单元验证，阐述验证覆盖点

3)画出testbench框架，阐述如何数据比较

4)单元验证完成后，SoC上SRAM集成验证，如何评价验证已经充分

256KB，16bit，需要地址线17根，

有效位，非法位，边沿

15 FPGA相对于ASIC，优势是灵活可编程，不足是可实现的最高频率有限，请介绍FPGA开发中是典型时序优化方法

对时序关键路径实施面积（资源占用率）换时间（路径延时大小）

对非关键路径实施时延换面积

手段是重定时、折叠、展开、脉动

插入寄存器，并行结构，逻辑展开，寄存器平衡，路径重组

16 100MHz时钟的单时钟脉冲信号，如何同步到时钟域300M中

17 A[7：0]\*11111011，请尽量用最少的资源来实现，写出对应RTL码

18 moore型状态机实现序列1101从右到左的不重叠检测。

1）请画出状态转移图

2）针对设计如何衡量验证的完备性

1 电路**亚稳态**描述

亚稳态是一种介于逻辑1和逻辑0之间的状态,可能引起电路解析的歧义

2 处理器在**中断处理**过程中，硬件来实现的功能是

中断请求 中断判别都可实现，中断响应、中断返回由硬件实现，中断处理由软件实现。

3 FPGA **IO接口**时序约束

Input delay output delay source clock latency？ transaction

4 同步**FIFO**，每100 cycle可以写80个数据，10 cycle可以读8个，深度多少？

16？ 8+8，同一时间要有相同读写

5 下列FPGA说法正确的是

Altera MAX7000系类属于CPLD，

FPGA是现场可编程逻辑阵列

FPGA更适合于触发器丰富的结构，而CPLD更适合于触发器有限而乘积项丰富的结构

SRAM的FPGA，每次上电后必须重新配置

6 对信号做降采样前添加滤波器，需要

低通，抗混叠滤波器

7 一个cycle只能完成一个（8bits\*8bits）或一个（17+17+17）操作，那么设计16bits\*16bits最少可以多少cycle完成。

3cycle（8+8）\*（8+8）=4\*（8\*8）=3\*（8\*8）+8\*8

8 12.918做无损定点化，需要最小位宽是多少位。位宽选择11位时的量化误差是多少？

12位，整数部分需要四位，小数部分需要8位

3位整数位宽最大只能表示7 所以整数12需要4位位宽

小数部分0.918 判断:0.918/0.000244140625 = 3760.128 12位能

位宽11位时，0.918\*2^7=117.504, 117/2^7-0.918=0.0039

例:3.1415926进行定点化

3.1415926/(8/32768) ~=12868

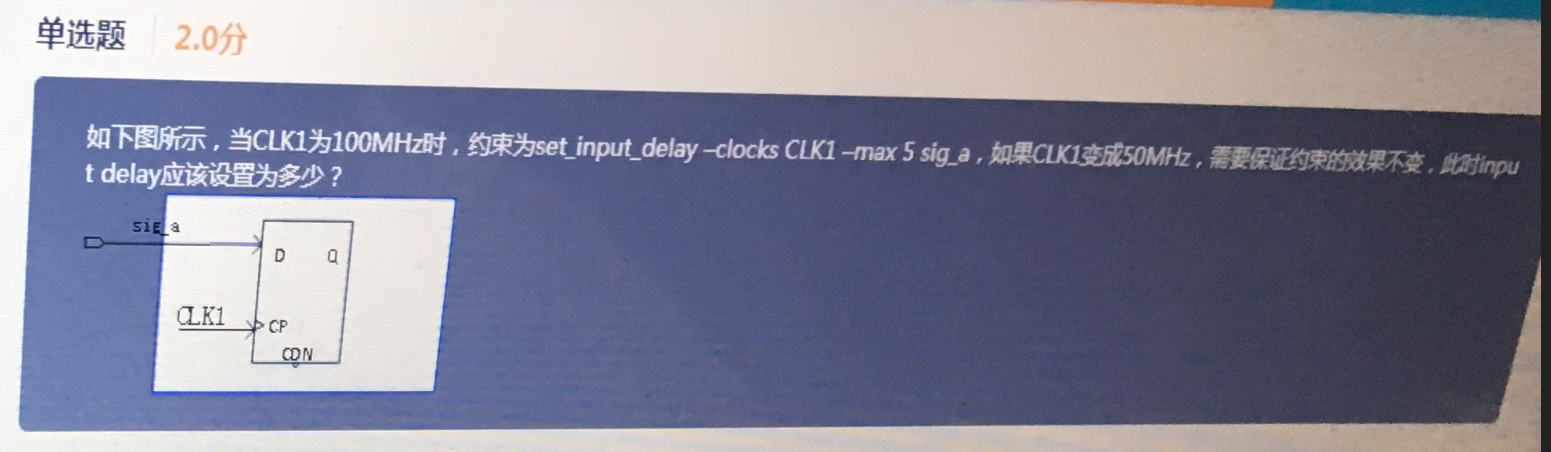
12868转化为浮点数: 16’d12868 = 16’b0011 0010 0100 0100

其中: 整数部分表示 011 = 3 小数部分表示 0010 0100 0100 = 580

(10进制数580需要转换成小数:580/4096=0.1416015625)

量化误差:0.0000089625

9



5/10/15/20

10？

10 复杂的FPGA包含资源类型

BlockRam/Register/LUT

11 线与 逻辑

OC门/OD门/加上拉电阻

12 将两输入与非门当非门使用时，另一端的接法应该是？

高电平

13 逻辑分析仪测试以下哪一种异常？

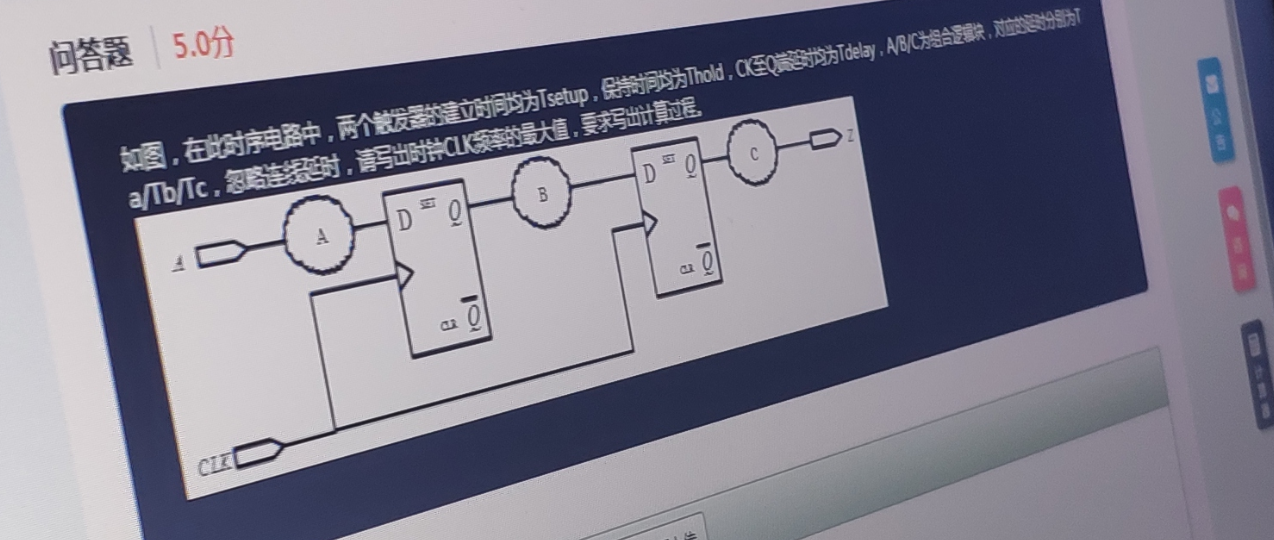
6个信号的异常组合

14 SRAM和DRAM

DRAM需要定时刷新，一个电容一个晶体管

SRAM需要6个晶体管

15 Ta/Tb/Tc



Tdata\_path\_min - Thold >= Tclk\_MAX

Tclk = Tsu + Tb + Tdelay

16 在评估某个低速IP时发现此IP需要使用一种同步的双端口SRAM（一个clk,两套独立地址线，一个读端口，一个写端口），但选定的工艺条件下只有普通的同步sram（一个clk，一个地址线，一个读端口，一个写端口）。已知此低速IP和双端口SRAM工作在20MHZ，是系统PLL的4分频时钟，你认为该如何解决，请简要阐述思路。

设计改动后验证时需要考虑哪些corner情况，功能覆盖率上有哪些点？

针对上述改动，如何保证的完备性。

提高频率。

速率换面积，快时钟域到慢时钟域的处理，多bit信号，两个端口优先级的处理，保证读写同一地址时，符合预期。

增加一倍频率，前一拍总是判断写，后一拍判断读，防止冲突。

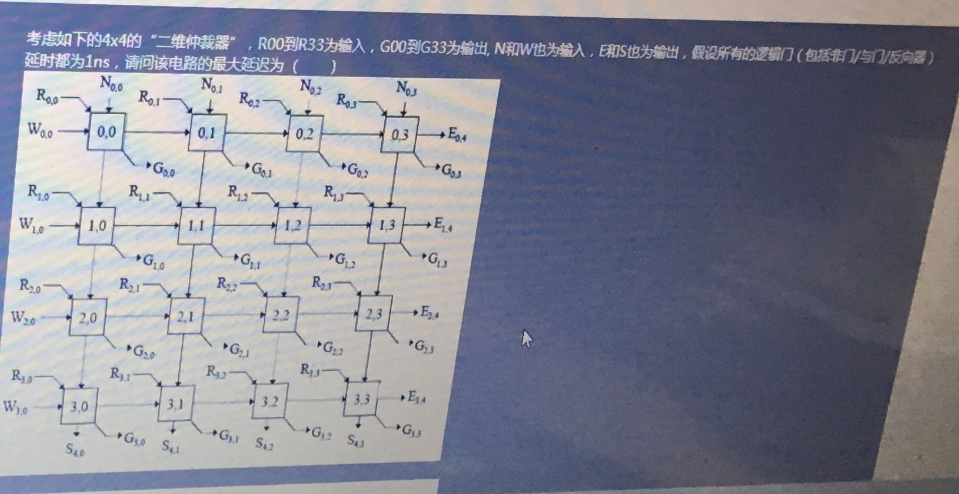
17 a=1'b1;b=2'b001,则c={a,b}=\_\_\_ , &c=\_\_\_

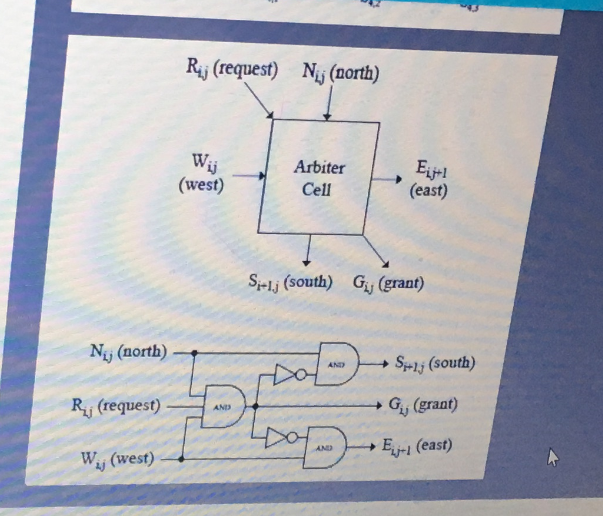
1001,0

18 同步有限状态机的描述

merly型moore型

19





21

1 跨时钟域设计的方法，请解释格雷码在跨时钟域设计中有什么作用？

打两拍；两级寄存是一级寄存的平方，两级并不能完全消除亚稳态危害，但是提高了可靠性减少其发生概率。总的来讲，就是一级概率很大，三级改善不大。

异步双口RAM

格雷码转换：转换成单bit跨时钟域处理

2 历史项目中验证平台的框图，并按照数据流说明如何完成数据比对。

3 总线/接口信号及其时序示意图，并解释协议。

4 UVM中phase

没有start\_phase

5 UVM agent包含以下哪些component( ABC )

1. monitor B. driver C. sequencer D. sequence

6 测试点分解一般采用哪些方法( )

等价类划分，边界值验证，错误推测，场景分析方法

7 代码覆盖率低，功能覆盖率高的原因

测试点分解有遗漏，RTL代码大量冗余

8 相比于FPGA验证，不是EDA验证的强项( AD )

1. 设计方案正确性 B. 多电源的设计 C. 多时钟域的设计 D. 接口IP设计的可靠性

 1 D触发器和latch的区别。**同步复位**和**异步复位**的区别及优缺点。

同步复位信号只有在时钟上升沿到来时才能有效；异步复位无论时钟沿是否到来，只要复位信号有效

|  |  |
| --- | --- |
| 同步 | 异步 |
| always@(posedge CLK) | always@(posedge CLK , negedge Rst\_n) |

异步设计简单，节省资源。但是易受毛刺影响，复位信号容易出现亚稳态。

同步利于仿真器仿真，可以滤除高于时钟频率的毛刺，有利于时序分析。但是复位信号有效时长必须大于时钟周期，才能完成复位

推荐异步复位，同步释放

2 画出状态机检测串行比特流中的‘10011’

3 现有3个输入信号a，b，c，并且前级DFF clock到abc的延时Ta，Tb，Tc（A>B>C），请使用两个二输入选择器对a,b,c实现三输入选择输出，不考虑clock skew，画出电路图并解释

4 （验证）定义函数，传递三个int类型参数A,B,C，函数返回，A的值不变，B的值变成A+B，C的值变成A-C；写出函数原型定义与函数体的实现

**1：什么是同步逻辑和异步逻辑？**

同步逻辑是时钟之间有固定的因果关系。异步逻辑是各时钟之间没有固定的因果关系。

同步时序逻辑电路的特点：各触发器的时钟端全部连接在一起，并接在系统时钟端，只有当时钟脉冲到来时，电路的状态才能改变。改变后的状态将一直保持到下一个时钟脉冲的到来，此时无论外部输入 x 有无变化，状态表中的每个状态都是稳定的。

异步时序逻辑电路的特点：电路中除可以使用带时钟的触发器外，还可以使用不带时钟的触发器和延迟元件作为存储元件，电路中没有统一的时钟，电路状态的改变由外部输入的变化直接引起。

**2：同步电路和异步电路的区别：**

同步电路：存储电路中所有触发器的时钟输入端都接同一个时钟脉冲源，因而所有触发器的状态的变化都与所加的时钟脉冲信号同步。

异步电路：电路没有统一的时钟，有些触发器的时钟输入端与时钟脉冲源相连，只有这些触发器的状态变化与时钟脉冲同步，而其他的触发器的状态变化不与时钟脉冲同步。

**3：时序设计的实质：**

时序设计的实质就是满足每一个触发器的建立/保持时间的要求。

**4：建立时间与保持时间的概念？**

建立时间：触发器在时钟上升沿到来之前，其数据输入端的数据必须保持不变的最小时间。

保持时间：触发器在时钟上升沿到来之后，其数据输入端的数据必须保持不变的最小时间。

**5：为什么触发器要满足建立时间和保持时间？**

因为触发器内部数据的形成是需要一定的时间的，如果不满足建立和保持时间，触发器将进入亚稳态，进入亚稳态后触发器的输出将不稳定，在0和1之间变化，这时需要经过一个恢复时间，其输出才能稳定，但稳定后的值并不一定是你的输入值。这就是为什么要用两级触发器来同步异步输入信号。这样做可以防止由于异步输入信号对于本级时钟可能不满足建立保持时间而使本级触发器产生的亚稳态传播到后面逻辑中，导致亚稳态的传播。

**6：什么是亚稳态？为什么两级触发器可以防止亚稳态传播？**

这也是一个异步电路同步化的问题。亚稳态是指触发器无法在某个规定的时间段内到达一个可以确认的状态。使用两级触发器来使异步电路同步化的电路其实叫做“一位同步器”，他只能用来对一位异步信号进行同步。两级触发器可防止亚稳态传播的原理：假设第一级触发器的输入不满足其建立保持时间，它在第一个脉冲沿到来后输出的数据就为亚稳态，那么在下一个脉冲沿到来之前，其输出的亚稳态数据在一段恢复时间后必须稳定下来，而且稳定的数据必须满足第二级触发器的建立时间，如果都满足了，在下一个脉冲沿到来时，第二级触发器将不会出现亚稳态，因为其输入端的数据满足其建立保持时间。**同步器有效的条件：第一级触发器进入亚稳态后的恢复时间** **+** **第二级触发器的建立时间** **< =** **时钟周期**。

更确切地说，输入脉冲宽度必须大于同步时钟周期与第一级触发器所需的保持时间之和。最保险的脉冲宽度是两倍同步时钟周期。 所以，这样的同步电路对于从较慢的时钟域来的异步信号进入较快的时钟域比较有效，对于进入一个较慢的时钟域，则没有作用 。

**7：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。

**8：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PAD TO PAD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。

**9：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。

**10：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，象xilinx的v2pro中的高速链路也逐渐被应用。

**11：对于多位的异步信号如何进行同步？**

对以一位的异步信号可以使用“一位同步器进行同步”（使用两级触发器），而对于多位的异步信号，可以采用如下方法：1：可以采用保持寄存器加握手信号的方法（多数据，控制，地址）；2：特殊的具体应用电路结构,根据应用的不同而不同；3：异步FIFO。（最常用的缓存单元是DPRAM）

**12：FPGA和CPLD的区别？**

|  |  |  |
| --- | --- | --- |
|  | CPLD | FPGA |
| 内部结构 | Product term（基于乘积项） | Look up Table（基于查找表） |
| 程序存储 | 内部EEPROM/FLASH | SRAM，外挂EEPROM |
| 资源类型 | 组合逻辑资源丰富 | 时序逻辑资源丰富 |
| 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 | 慢 | 快  ？？ |
| 其他资源 | － | PLL、RAM和乘法器等 |
| 保密性 | 可加密 | 一般不能保密 |

**13：锁存器（latch）和触发器（flip-flop）区别？**

电平敏感的存储器件称为锁存器。可分为高电平锁存器和低电平锁存器，用于不同时钟之间的信号同步。

有交叉耦合的门构成的双稳态的存储原件称为触发器。分为上升沿触发和下降沿触发。可以认为是两个不同电平敏感的锁存器串连而成。前一个锁存器决定了触发器的建立时间，后一个锁存器则决定了保持时间。

**14：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BLOCK RAM,另一种是由LUT配置成的内部存储器（也就是分布式RAM）。BLOCK RAM由一定数量固定大小的存储块构成的，使用BLOCK RAM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BLOCK RAM资源是其块大小的整数倍。

**15：什么是时钟抖动？**

时钟抖动是指芯片的某一个给定点上时钟周期发生暂时性变化，也就是说时钟周期在不同的周期上可能加长或缩短。它是一个平均值为0的平均变量。

**16：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）。

**17：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：异步电路一半是通过加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，一半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。

**18：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：BLOCK RAM，触发器（FF），查找表（LUT）；

注意事项：

1：在生成RAM等存储单元时，应该首选BLOCK RAM 资源；其原因有二：第一：使用BLOCK RAM等资源，可以节约更多的FF和4-LUT等底层可编程单元。使用BLOCK RAM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：BLOCK RAM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。

2：弄清FPGA的硬件结构，合理使用BLOCK RAM资源；

3：分析BLOCK RAM容量，高效使用BLOCK RAM资源；

4：分布式RAM资源（DISTRIBUTE RAM）

**19：Xilinx中与全局时钟资源和DLL相关的硬件原语：**

常用的与全局时钟资源相关的Xilinx器件原语包括：IBUFG,IBUFGDS,BUFG,BUFGP,BUFGCE,BUFGMUX,BUFGDLL,DCM等。

**20：HDL语言的层次概念？**

HDL语言是分层次的、类型的，最常用的层次概念有系统与标准级、功能模块级，行为级，寄存器传输级和门级。系统级，算法级，RTL级(行为级)，门级，开关级

**21：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

**22：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。

1：规格制定：客户向芯片设计公司提出设计要求。

2：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Visual Elite等。

3：HDL编码：设计输入工具：ultra ，visual VHDL等

4：仿真验证：modelsim

5：逻辑综合：synplify

6：静态时序分析：synopsys的Prime Time

7：形式验证：Synopsys的Formality.

**23：寄生效应在IC设计中怎样加以克服和利用？**

所谓寄生效应就是那些溜进你的PCB并在电路中大施破坏、令人头痛、原因不明的小故障。它们就是渗入高速电路中隐藏的寄生电容和寄生电感。其中包括由封装引脚和印制线过长形成的寄生电感；焊盘到地、焊盘到电源平面和焊盘到印制线之间形成的寄生电容；通孔之间的相互影响，以及许多其它可能的寄生效应。

理想状态下，导线是没有电阻，电容和电感的。而在实际中，导线用到了金属铜，它有一定的电阻率，如果导线足够长，积累的电阻也相当可观。两条平行的导线，如果互相之间有电压差异，就相当于形成了一个平行板电容器（你想象一下）。通电的导线周围会形成磁场（特别是电流变化时），磁场会产生感生电场，会对电子的移动产生影响，可以说每条实际的导线包括元器件的管脚都会产生感生电动势，这也就是寄生电感。

在直流或者低频情况下，这种寄生效应看不太出来。而在交流特别是高频交流条件下，影响就非常巨大了。根据复阻抗公式，电容、电感会在交流情况下会对电流的移动产生巨大阻碍，也就可以折算成阻抗。这种寄生效应很难克服，也难摸到。只能通过优化线路，尽量使用管脚短的SMT元器件来减少其影响，要完全消除是不可能的。

**24：用flip-flop和logic-gate设计一个1位加法器，输入carryin和current-stage，输出carryout和next-stage？**

carryout=carryin\*current-stage；与门

next-stage=carryin’\*current-stage+carryin\*current-stage’; 与门，非门，或门（或者异或门）

module(clk,current-stage,carryin,next-stage,carryout);

input clk, current-stage,carryin;

output next-stage,carryout;

always@(posedge clk)

carryout<=carryin&current-stage;

nextstage<=

**25：IC设计中同步复位与异步复位的区别？**

同步复位在时钟沿变化时，完成复位动作。异步复位不管时钟，只要复位信号满足条件，就完成复位动作。异步复位对复位信号要求比较高，不能有毛刺，如果其与时钟关系不确定，也可能出现亚稳态。

**26：MOORE 与 MEELEY状态机的特征？**

 Moore 状态机的输出仅与当前状态值有关, 且只在时钟边沿到来时才会有状态变化。

 Mealy 状态机的输出不仅与当前状态值有关, 而且与当前输入值有关。

**27：多时域设计中,如何处理信号跨时域？**

不同的时钟域之间信号通信时需要进行**同步处理**，这样可以防止新时钟域中第一级触发器的亚稳态信号对下级逻辑造成影响。

信号跨时钟域同步：当单个信号跨时钟域时，可以采用两级触发器来同步；数据或地址总线跨时钟域时可以采用异步FIFO来实现时钟同步；第三种方法就是采用握手信号。

**28：说说静态、动态时序模拟的优缺点？**

*静态时序分析*是采用穷尽分析方法来提取出整个电路存在的所有时序路径，计算信号在这些路径上的传播延时，检查信号的建立和保持时间是否满足时序要求，通过对最大路径延时和最小路径延时的分析，找出违背时序约束的错误。它不需要输入向量就能穷尽所有的路径，且运行速度很快、占用内存较少，不仅可以对芯片设计进行全面的时序功能检查，而且还可利用时序分析的结果来优化设计，因此静态时序分析已经越来越多地被用到数字集成电路设计的验证中。

*动态时序模拟*就是通常的仿真，因为不可能产生完备的测试向量，覆盖门级网表中的每一条路径。因此在动态时序分析中，无法暴露一些路径上可能存在的时序问题；

**29：如何防止亚稳态？**

亚稳态是指触发器无法在某个规定时间段内达到一个可确认的状态。当一个触发器进入亚稳态时，既无法预测该单元的输出电平，也无法预测何时输出才能稳定在某个正确的电平上。在这个稳定期间，触发器输出一些中间级电平，或者可能处于振荡状态，并且这种无用的输出电平可以沿信号通道上的各个触发器级联式传播下去。

解决方法：

1 降低系统时钟频率

2 用反应更快的FF

3 引入同步机制，防止亚稳态传播（可以采用前面说的加两级触发器）。

4 改善时钟质量，用边沿变化快速的时钟信号