**1.setup和 holdup时间区别 .**

建立时间是指在时钟边沿前，数据信号需要保持不变的时间。

保持时间是指时钟跳变边沿后数据信号需要保持不变的时间。

如果不满足建立和保持时间的话，那么DFF将不能正确地采样到数据，将会出现metastability的情况。如果数据信号在时钟沿触发前后持续的时间均超过建立和保持时间，那么超过量就分别被称为建立时间裕量和保持时间裕量。

**2. 多时域 设计 中 , 如何处理信号跨时域**

情况比较多，如果简单回答的话就是：跨时域的信号要经过同步器同步，防止亚稳态传播。例如：时钟域 1 中的一个信号，要送到时钟域 2 ，那么在这个信号送到时钟域 2 之前，要先经过时钟域 2 的同步器同步后，才能进入时钟域 2 。这个同步器就是两级 d 触发器，其时钟为时钟域 2 的时钟。这样做是怕时钟域 1 中的这个信号，可能不满足时钟域 2 中触发器的建立保持时间，而产生亚稳态，因为它们之间没有必然关系，是异步的。这样做只能防止亚稳态传播，但不能保证采进来的数据的正确性。所以通常只同步很少位数的信号。比如控制信号，或地址。当同步的是地址时，一般该地址应采用格雷码，因为格雷码每次只变一位，相当于每次只有一个同步器在起作用，这样可以降低出错概率，向异步 FIFO 的设计中，比较读写地址的大小时，就是用这种方法。  
     如果两个时钟域之间传送大量的数据，可以用异步 FIFO 来解决问题。

**3.latch 与 register 的区别 , 为什么现在多用 register. 行为级描述中 latch 如何产生的**        区别不多说。为什么避免使用 latch ，因为设计中用 latch 会使设计后期的静态时序分析变的困难（必须用的地方当然另当别论）。  
        行为级描述中 latch 产生的原因：多由于构造组合逻辑 电路 时，使用 if 或 case 语句，没有把所有的条件给足，导致没有提到的条件，其输出未知。或者是每个条件分支中，没有给出所有输出的值，这就会产生 latch 。所以构造组合逻辑电路时，其 always 语句中的敏感信号必须包括所有的输入端，每个条件分支必须把所有的输出端的值都给出来。

**4.BLOCKING NONBLOCKING 赋值的区别**  
Answer:  这个问题可参考的资料很多，讲的都很透彻，可以找一下。基本用法就是常说的 “组合逻辑用BLOCKING，时序逻辑用NONBLOCKING”。

**5.MOORE 与MEELEY 状态机的特征**

Answer:

**6.IC 设计中同步复位与 异步复位的区别**  
Answer:  如果光说概念的话：同步复位在时钟沿采复位信号，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。  
     象芯片的上电复位就是异步复位，因为这时时钟振荡器不一定起振了，可能还没有时钟脉冲。异步复位很容易受到复位端信号毛刺的影响，比如复位端信号由组合逻辑组成，那组合逻辑输出产生的冒险，就会使触发器错误的复位。

**7. 实现 N 位 Johnson Counter,N=**

**8. 用 FSM 实现 101101 的序列检测模块**

**9. 集成电路设计前端流程及工具。**

**10. FPGA 和 ASIC 的概念，他们的区别**

**11. LATCH 和 DFF 的概念和区别**

Answer: LATC 是 H 锁存器， DFF 是触发器，其电路形式完全不同。

**12. 用 DFF 实现二分频。**

Answer:

always @(posedge clk)  
      if (reset) begin  
        sel <= 1;  
        clk1 <= 1;  
        clk2 <= 1;  
      end  
      else begin  
       sel <= ~sel;  
       if (sel)  
          clk1 <= ~clk1;  
       else  
          clk2 <= ~clk2;  
     end

**13. 用 VERILOG 或 VHDL 写一段代码，实现消除一个 glitch**

Answer: glitch 主要发生在组合逻辑电路输出，可以加 double DFF 输出稳定信号。

**14. 给一个表达式 f=xxxx+xxxx+xxxxx+xxxx 用最少数量的与非门实现**

**15. 用 VERILOG 或 VHDL 写一段代码，实现 10 进制计数器。**

**16. A,B,C,D,E 进行投票，多数服从少数，输出是 F （也就是如果 A,B,C,D,E 中 1 的个数比 0 多，那么 F 输出为 1 ，否则 F 为 0 ），用与非门实现，输入数目没有限制**  
  
**17. 负数与正数相乘的问题：  1010 （ -6 ） \*0010 （ 2 ）**  
Answer ：用补码相乘时应该进行相应的符号扩展，比如上面是 4bit 相乘，结果应该为 8bit 。这样符号扩展后分别为 11111010 和 00000010 ，然后再用这两个数直接相乘，结果为 111110100 ，取其低 8 位 11110100 ，作为 -6\*2 的结果。这也是个补码形式，再判断一下高位恢复为原码，得到结果。

二. 数字电路设计

1. 画出 fsm （有限状态机）  
2. 用 verilog 编程 ，语法要符合 fpga 设计的要求  
系统方面：如果简历上还说做过 cpu 之类，就会问到诸如 cpu 如何 工作，流水线之类的问题  
3. 单片机、 DSP 、 FPGA 、 嵌入式 方面（从没碰过，就大概知道几个名字胡扯几句，欢迎拍砖，也欢迎牛人帮忙补充）  
如 单片机 中断几个 / 类型，编中断程序注意什么问题  
DSP 的结构（冯 . 诺伊曼结构吗？）  
嵌入式处理器类型 ( 如 ARM ) ，操作系统种类（ Vxworks,ucos,winCE,linux ），操作系统方  
面偏 CS 方向了，在 CS 篇里面讲了  
4. 信号系统基础  
拉氏变换与 Z 变换公式等类似东西，随便翻翻书把  
如 .h(n)=-a\*h(n-1)+b\*δ(n)    a. 求 h(n) 的 z 变换     b. 问该系统是否为稳定系统      
c. 写出 F IR 数字滤波器的差分方程  
以往各种笔试题举例  
利用 4 选 1 实现 F(x,y,z)=xz+yz'  
用 mos 管搭出一个二输入与非门。 用传输门和倒向器搭一个边沿触发器  
用运算放大器组成一个 10 倍的放大器  
微波电路的匹配电阻。  
名词解释，无聊的外文缩写罢了，比如 PCI 、 ECC 、 DDR 、 interrupt 、 pipeline  
IRQ,BIOS,USB,VHDL,VLSI VCO( 压控振荡器 ) RAM ( 动态随机存储器 ) ， FIR IIR DFT( 离散傅立叶变换）或者是中文的，比如 a 量化误差     b. 直方图     c. 白平衡

2什么是竞争与冒险现象？怎样判断？如何消除？

在组合逻辑中，由于门的输入信号通路中经过了不同的延时，导致到达该门的时间不一致叫竞争。产生毛刺叫冒险。如果布尔式中有相反的信号则可能产生竞争和冒险现象。解决方法：一是添加布尔式的消去项，二是在芯片外部加电容。

3 用D触发器实现2倍分频的逻辑电路？

Verilog描述：

module divide2( clk , clk\_o, reset);

input clk , reset;

output clk\_o; wire in;

reg out ;

always @ ( posedge clk or posedge reset)

if ( reset) out <= 0;

else out <= in;

assign in = ~out;

assign clk\_o = out;

endmodule

**4 什么是"线与"逻辑，要实现它，在硬件特性上有什么具体要求？**

线与逻辑是两个输出信号相连可以实现与的功能。在硬件上，要用 oc门来实现，由于不用oc门可能使灌电流过大，而烧坏逻辑门。 同时在输出端口应加一个上拉电阻。

**5 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。

6 请画出微机接口电路中，典型的输入设备与微机接口逻辑示意图（数据接口、控制接口、所存器/缓冲器）。

**7 你知道那些常用逻辑电平？TTL与COMS电平可以直接互连吗？**

12，5，3.3 TTL和CMOS不可以直接互连，由于TTL是在0.3-3.6V之间，而CMOS则是有在12V的有在5V的。CMOS输出接到TTL是可以直接互连。TTL接到CMOS需要在输出端口加一上拉电阻接到5V或者12V。

**8 可编程逻辑器件在现代电子设计中越来越重要，请问：你所知道的可编程逻辑器件有哪些？**

PAL，PLD，CPLD，FPGA。

**9 试用VHDL或VERILOG、ABLE描述8位D触发器逻辑。**

module dff8(clk , reset, d, q);

input clk; input reset;

input [7:0] d;

output [7:0] q;

reg [7:0] q;

always @ (posedge clk or posedge reset)

if(reset) q <= 0;

else q <= d;

endmodule

10 设想你将设计完成一个电子电路方案。请简述用EDA软件（如PROTEL）进行设计（包 括原理图和PCB图）到调试出样机的整个过程。在各环节应注意哪些问题？ 电源的稳定上，电容的选取上，以及布局的大小。

11 用逻辑门和cmos电路实现ab+cd

12 用一个二选一mux和一个inv实现异或

13 给了reg的setup,hold时间，求中间组合逻辑的delay范围。

Delay < period - setup - hold

**14 如何解决亚稳态**

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。

15 用verilog/vhdl写一个fifo控制器 包括空，满，半满信号。

16 用verilog/vddl检测stream中的特定字符串分状态用状态机写。

17 用mos管搭出一个二输入与非门。

18 集成电路前段设计流程，写出相关的工具。

19 名词IRQ,BIOS,USB,VHDL,SDR IRQ: Interrupt ReQuest BIOS: Basic Input Output System USB: Universal Serial Bus VHDL: VHIC Hardware Description Language SDR: Single Data Rate

20 unix 命令cp -r, rm,uname

21 用波形表示D触发器的功能

22 写异步D触发器的

verilog module module dff8(clk , reset, d, q);

input clk;

input reset;

input d;

output q;

reg q;

always @ (posedge clk or posedge reset)

if(reset) q <= 0;

else q <= d;

endmodule

**23 What is PC Chipset?**

芯片组（ Chipset）是主板的核心组成部分，按照在主板上的排列位置的不同，通常分为北桥芯片和南桥芯片。北桥芯片提供对CPU的类型和主频、内存的类型和最大容量、ISA/PCI/AGP插槽、ECC纠错等支持。南桥芯片则提供对KBC（键盘控制器）、RTC（实时时钟控制器）、USB（通用串行总线）、Ultra DMA/33(66)EIDE数据传输方式和ACPI（高级能源管理）等的支持。其中北桥芯片起着主导性的作用，也称为主桥（Host Bridge）。　　除了最通用的南北桥结构外，目前芯片组正向更高级的加速集线架 构发展， Intel的8xx系列芯片组就是这类芯片组的代表，它将一些子系统如IDE接口、音效、MODEM和USB直接接入主芯片，能够提供比PCI总线宽一倍的带宽，达到了266MB/s。

24 用传输门和反向器搭一个边沿触发器

25 画状态机，接受1，2，5分钱的卖报机，每份报纸5分钱

1. setup time 和 hold time 不满足情况下应该如何解决？

2. 什么叫做亚稳态，如何解决？

3. Verilog中 => 和 = 有什么区别？

4. 画一个D触发器的原理图（门级），并且用verilog gate level表示出来；

5. 用最少的Mos管画出一个与非门；

6. 写一段finite state machine（主要考察coding style）；

答：如果触发器的 setup time/hold time不满足,这个数据就不能被这一时钟打入触发器，只有在下一个时钟上升沿到来时，数据才能被打入触发器。在同步系统中，如果触发器的setup time/hold time不满足，就可能产生亚稳态（Metastability），导致采样错误。此时触发器输出端Q在有效时钟沿之后比较长的一段时间处于不确定的状态，在这段时间里Q端毛刺、振荡、固定的某一电压值，而不是等于数据输入端D的值。这段之间成为决断时间（resolution time）。经过resolution time之后Q端将稳定到0或1上，但是究竟是0还是1，这是随机的，与输入没有必然的关系。 只要系统中有异步元件，亚稳态就是无法避免的，因此设计的电路首先要减少亚稳态导致错误的发生，其次要使系统对产生的错误不敏感。前者需要同步来实现，而后者根据不同的设计应用有不同的处理办法。 在IC设计中，如果setup time没有满足，只有重新综合，重新约束计，如果hold time不满足，那么可以在post layout时候fix，也可以在综合时候使用set\_fix-\_hold命令来修正 建立时间和保持时间要看在什么阶 段出现问题了，如果在仿真阶段则必须重新改写代码，在综合阶段则需要通过标准单元的选择调整，如果综合中没有负的时隙，而在后端设计中出现问题，也可以通过调整布局与布线达到优化设计，并非象楼上说得，一定要从头综合。

**同步复位与异步复位**

异步复位是不受时钟影响的，在一个芯片系统初始化（或者说上电）的时候需要这么一个全局的信号来对整个芯片进行整体的复位，到一个初始的确定状态。而同步复位需要在时钟沿来临的时候才会对整个系统进行复位。请注意，如果芯片是有多个时钟的系统，那么如何保证不同时钟域的电路能够“同时”复位将会是一个重要的问题，此外，如果你的时钟是一个低频时钟，那么在这个系统（包括其他芯片）上电时如何保证能和其他芯片同时复位？硬件全局异步复位是必要的，请注意这里加上了“全局”，这是因为异步复位既然要对整个芯片“同时”复位，那么布线延时绝不 能不考虑，使用 FPGA设计时芯片的异步复位必须要走全局网络。再提醒一点，芯片中最好不要出现多个异步复位。一个关键原因是对于FPGA而言，多个异步复位信号难以实现前面要求的“全局网络”。

异步复位最大的优点是, 数据通路就可以不依赖于时钟而确保清洁可控。然而, 异步复位也有其固有的缺陷。异步复位设计的DFT (可测性设计) 与STA (静态时序分析) 的复杂性要高于同步复位设计; 但异步复位中最严重的问题是, 如果异步复位信号在触发器时钟有效沿附近“释放”(复位信号从有效变为无效) 的话, 可能会导致触发器输出的亚稳态。

IR压降是指出现在集成电路中电源和地网络上电压下降或升高的一种现象。随着半导体工艺的演进金属互连线的宽度越来越窄，导致它的电阻值上升，所以在整个芯片范围内将存在一定的IR压降。IR压降的大小决定于从电源PAD到所计算的逻辑门单元之间的等效电阻的大小SoC设计中的每一个逻辑门单元的电流都会对设计中的其它逻辑门单元造成不同程度的IR压降。如果连接到金属连线上的逻辑门单元同时有翻转动作，那么因此而导致的IR压降将会很大。然而，设计中的某些部分的同时翻转又是非常重要的，例如时钟网络和它所驱动的寄存器，在一个同步设计 中它们必须同时翻转。因此，一定程度的 IR压降是不可避免的。

IR压降可能是局部或全局性的。当相邻位置一定数量的逻辑门单元同时有逻辑翻转动作时，就引起局部IR压降现象，而电源网格某一特定部分的电阻值特别高时，例如R14远远超出预计时，也会导致局部IR压降；当芯片某一区域内的逻辑动作导致其它区域的IR压降时，称之为全局现象。

IR压降问题的表现常常类似一些时序甚至可能是信号的完整性问题。如果芯片的全局IR压降过高，则逻辑门就有功能故障，使芯片彻底失效，尽管逻辑仿真显示设计是正确的。而局部IR压降比较敏感，它只在一些特定的条件下才可能发生，例如所有的总线数据同步进行翻转，因此芯片会间歇性的表现出一些功能故障。而IR压降比较普遍的影响就是降低了芯片的速度。试验表明，逻辑门单元上5%的IR压降将使正常的门速度降低15%

总的说来，是电压降的意思。规模越大的芯片的电流就越大，在电源上产生的压降也越大。面积增大的芯片也会恶化电压。芯片的速度又与电压成相关性。几个因数导致时序，抗噪等问题。

危害：

1。性能（performance）

由管子的 Tdelay＝c/u可知，电压降低，门的开关速度越慢，性能越差。

2。功能(function）

实际上在极端的情况下甚至功能也会受影响的。在深亚微米下，如果 Power/Ground network做的也很差，然后碰上了很不好的case，IR drop会很大，如果用的是high Vt的process，则DC noise margin就比较小了。这样就有可能功能错误。

3。功耗（power）

如果没有做详细的 IR drop分析，又想功能正确，那就只有留很大的margin了，本来1.2v可以跑的，也只能用1.5v了。但是这样功耗也就上去了。

4。面积(area)

如果要在一定程度上限制 IR drop，就要在chip里面加上很多的decoupling capacitance.占用了很多面积。

5。成本(cost)

功耗上去了，响应的散热，封装都成了问题，需要额外花费啦。而且面积变大，也是钱啊～～

所以， IR drop还是一个比较讨厌的问题，需要小心对待。

异步电路和同步时序电路：

同步电路的时序电路共享同一个时钟 CLK，而所有的状态变化都是在时钟的上升沿(或下降沿)完成的。

同步设计中最长的组合逻辑延迟，即关键路径 (Critical Path)将影响整个系统的运行速度。

异步电路没有统一的时钟，状态变化的时刻是不稳定的，通常输入信号只在电路处于稳定状态时才发

生变化。

**异步电路：**

电路核心逻辑有用组合电路实现

异步时序电路的最大缺点是容易产生毛刺

不利于器件移植

不利于静态时序分析（ STA）、验证设计时序性能

**同步时序电路：**

电路核心逻辑是用各种触发器实现

电路主要信号、输出信号等都是在某个时钟沿驱动触发器产生的

同步时序电路可以很好的避免毛刺

利于器件移植

利于静态时序分析（ STA）、验证设计时序性能。

同步设计中，稳定可靠的数据采样必须遵从以下两个基本原则：

1)在有效时钟沿到达前，数据输入至少已经稳定了采样寄存器的Setup时间之久，这条原则简称满足Setup时间原则；

2)在有效时钟沿到达后，数据输入至少还将稳定保持采样寄存器的Hold时钟之旧，这条原则简称满足Hold时间原则。

用 verilog描述时序电路

相关知识

1.所有的时序逻辑电路都必须包含存储电路，因而记忆特性时时序逻辑电路的基本特征。

2.存储电路时通过各种存储单元来实现的，而存储单元按照触发方式可以分为锁存器和触发器。以电平触发方式触发

的为锁存器，以边沿触发方式触发的为触发器。

3.在设计过程中，设计人员可以选择使用锁存器和触发器，但一般情况应尽量使用触发器，特别是在使用FPGA器件时。

4.触发器在使用时要满足两个重要的参数：建立时间（Setup time）和保持时间。

描述时序电路

1.在设计时一般将组合逻辑电路和存储电路分开描述，尽量不要写到一个always块中。

2.描述存储电路时，存储单元可以通过寄存器型变量来建模，但只能使用过程赋值（即always赋值）。对于电平触发的

存储单元，应使用阻塞赋值（ =）；

对于边沿触发方式的存储单元，应使用非阻塞赋值（ <=）。

3.若将组合逻辑和存储电路写到同一个always块时，应该使用非阻塞赋值。