

《芯动力——硬件加速设计方法》作业答案解析

- “中国大学 MOOC”平台选课主页

<https://www.icourse163.org/course/SWJTU-1207492806>

- 课程相关资料下载网址 www.dizhixiong.cn



选课链接

<https://www.icourse163.org/course/SWJTU-1207492806>

课件全部共享下载 (www.dizhixiong.cn)

习题答案可发邮件申请 zxdi@home.swjtu.edu.cn



选课二维码

- ✓ 课程实验案例源自多项科创竞赛国家级获奖作品，源代码全部开源
- ✓ “基于RV处理器玄铁E902的SoC设计”入选**2021年教育部产学合作协同育人优秀项目案例**
- ✓ 选课人数 **1.8万** 人 ≈ 全国28所国家级示范性微电子学院 **3年** 毕业学生（本硕博）**总数**
ps:2019年全国 28 所示范性微电子学的毕业人数为 6092 人(中国电子信息产业发展研究院《中国集成电路产业人才白皮书》)
- ✓ 以MOOC为核心的教学论文发表于ACM GLSVLSI '21(CFF-C)
Zhixiong Di(*), et al. ASIC Design Principle Course with Combination of Online-MOOC and Offline-Inexpensive FPGA Board. In Proceedings of the 2021 on Great Lakes Symposium on VLSI (GLSVLSI '21). New York, NY, USA, 431–436.
- ✓ 教学成果获“电子学报”、“中国电子学会”报道

- 若 VerilogHDL 基础较弱，可在 B 站学习《从电路设计的角度入门 Verilog HDL》：

https://www.bilibili.com/video/BV1PS4y1s7XW/?spm_id_from=333.999.0.0&vd_source=11e1c053bd139ab5a2798c3f31d057df

- 如果期望继续深入静态时序分析，可在 B 站学习《数字集成电路静态时序分析》

https://www.bilibili.com/video/BV1if4y1p7Dq/?spm_id_from=333.999.0.0&vd_source=11e1c053bd139ab5a2798c3f31d057df

——By 西南交通大学信息科学与技术学院 邸志雄老师 (zxdi@home.swjtu.edu.cn)



第二章 VerilogHDL 作业题:

1. 是否会产生 latch

哪种写法会产生 latch?

```
a) always@(b or d)
begin
case(d) //synopsys full_case
2'b00: a=b>>1;
2'b11: c=b>>1;
endcase
end
```

```
c) always @(b or d)
case(d)
2'b00: a=b>>1;
2'b11: c=b>>1;
default:
begin
a=b;
c=b;
end
endcase
```

```
b) always @(b or d)
begin
a=b;
c=b;
case(d)
2'b00: a=b>>1;
2'b11: c=b>>1;
endcase
end
```

答: a,c 会产生 latch

写法 a,c 中当变量 d 处于某一种 case 时变量 a 或 c 的状态不能同时更新, 总有一个变量会保持前一个状态, 从而产生 latch

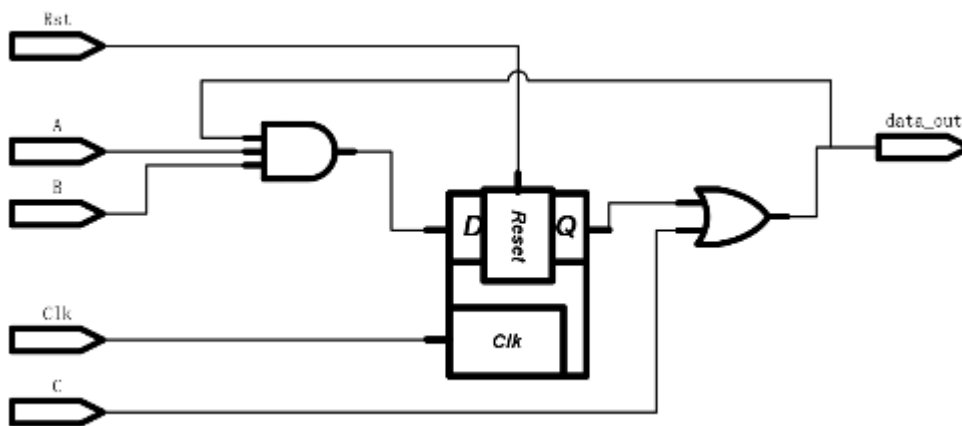
2. 锁存器比寄存器省面积, 但为什么在数字芯片设计中通常使用寄存器?

答: 锁存器不利于静态时序分析, 且会导致毛刺传播

```
3 (20分) module DDFornet(rst,clk,a,b,c,data_out);  
    input rst,clk;  
    input a,b,c;  
    output data_out;  
    reg data_out;  
    reg data_temp;  
    always @(posedge clk or negedge rst)  
    if(!rst)  
        data_temp<=1'b0;  
    else  
        data_temp<=a&b&data_out;  
    always @(*)  
        data_out=data_temp|c;  
endmodule
```

问题：请画出代码对应的电路结构

答：



这个题目考察的要点是，要理解 DFF 的电路结构，理解时序逻辑和组合逻辑的描述方法，理解 DFF 存在反馈路径

第三章 电路结构设计作业题

1. 异步 FIFO，若深度为 8 时，如何设置指针来保证在跨时钟域传播时，尽可能地消除亚稳态，请写出具体的指针数值；若深度不是 2 次幂时，则采用何种方法如何设置其读写指针。

答：0_000, 0_001, 0_011, 0_010, 0_110, 0_111, 0_101, 0_100

1_100,1_101,1_111,1_110,1_010,1_011,1_001,1_000

对于任意深度为偶数 N 的 FIFO，我们可以在对称轴（在除去 MSB 位的情况下，011 和 111 是对称的，010 和 110 是对称的）上下取 N/2 个格雷码

比如：深度为 6 的 FIFO，读写指针就可设置为 0_000, 0_001, 0_011,（去掉 0_010, 0_110）0_111, 0_101, 0_100

也可以回答如下文字：

在非 2 次幂深度情况下，格雷码已经不再适用，此时的解决方法通常有：

- 1) 若深度为偶数，可采用最接近的 2 次幂的格雷码编码，在此基础上修改；
- 2) 深度为一般数值时，可自行设计一种逻辑电路，或者查找表，实现指针每次只跳变一次的功能；
- 3) 以上方法通常在设计层面较为复杂，若无特定需求，可将 FIFO 深度设置为 2 次幂，浪费一些存储空间，来化简控制电路的复杂度。

2. 系统时钟 12M(source clock), 分频出来 8K 和 1M 的时钟(generated clock)。

请问这种情况下，数据从 12M 到 8K 传输和 8K 到 12M 间传输是跨时钟域吗？为什么？

答：不是，他们来自同一个时钟源，相位和倍数是可控的，所以不属于跨时钟域。

跨时钟域是指时钟来自两个不同的时钟源，相互之间的相位和倍数关系彼此不可控制

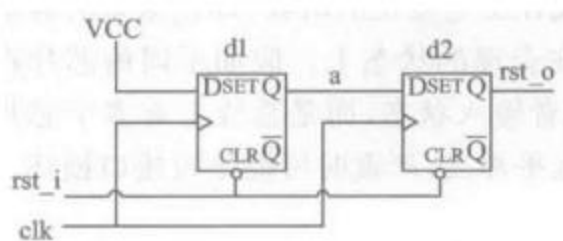
3. 若写时钟 50MHz，读时钟 40MHz，如果不丢失地将 10 万个数据送入读时钟域下游节点，则 FIFO 深度应设置为多少？

答：将 10 万个数据送入 fifo 需要的时间为 $10 \text{ 万} / 50 \text{ Mhz} = 2 \text{ ms}$

在极端的背靠背情况下，两个突发传输持续发生，中间的盈余数据需要 fifo 缓存，这就是其最小深度在 2ms 内送出数据 $40 \text{ Mhz} * 2 \text{ ms} = 8 \text{ 万}$ ，所以在读写都是相同位宽的情况下，fifo 的最小深度为 2 万

4.请画出异步复位、同步释放的电路结构

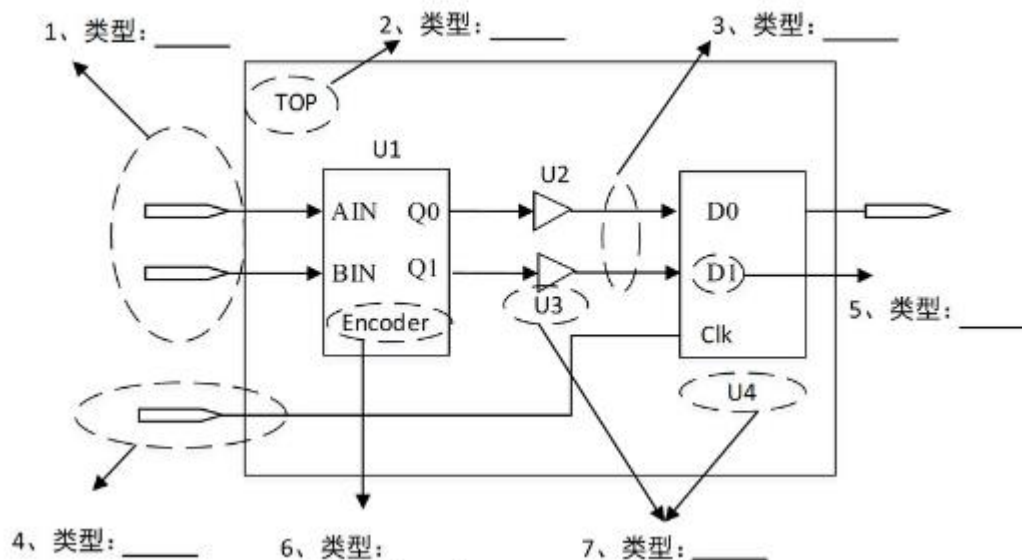
答：



第四章 逻辑综合作业题

1.

请根据下图将design、cell、net、pin、port、reference、clock等填入图1所示的电路框图中。

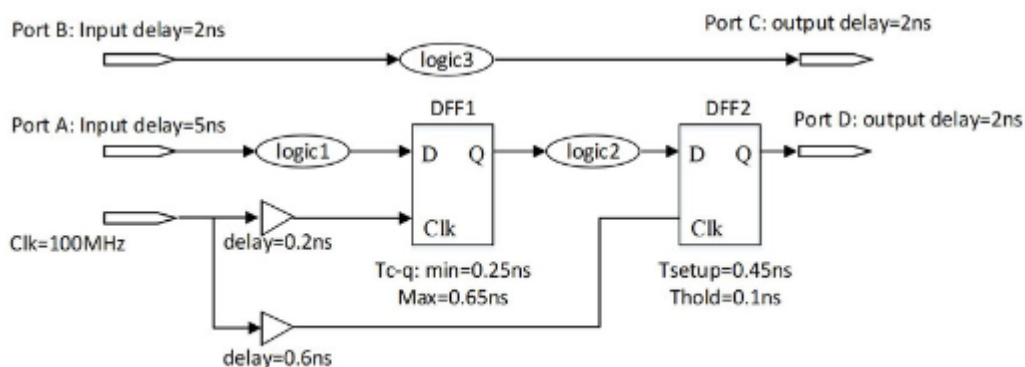


答：1.port 2.design 3.net 4.clk 5.pin 6.reference 7.cell

2.

请根据下图回答：

- (1) 在逻辑综合中，加哪几条关于延时或时间的约束，能使DC对logic1进行逻辑综合？
- (2) 在逻辑综合中，加哪几条关于延时或时间的约束，能使DC对logic2进行逻辑综合？
- (3) 在逻辑综合中，加哪几条关于延时或时间的约束，能使DC对logic3进行逻辑综合？



答案：

考察要点：理解 STA 关于 timing path 的划分要求，理解 STA 对时序路径的约

束。不一定把指令写出来（不需要背指令），大概意思对就可以了

1、input delay、时钟周期（虚拟时钟）

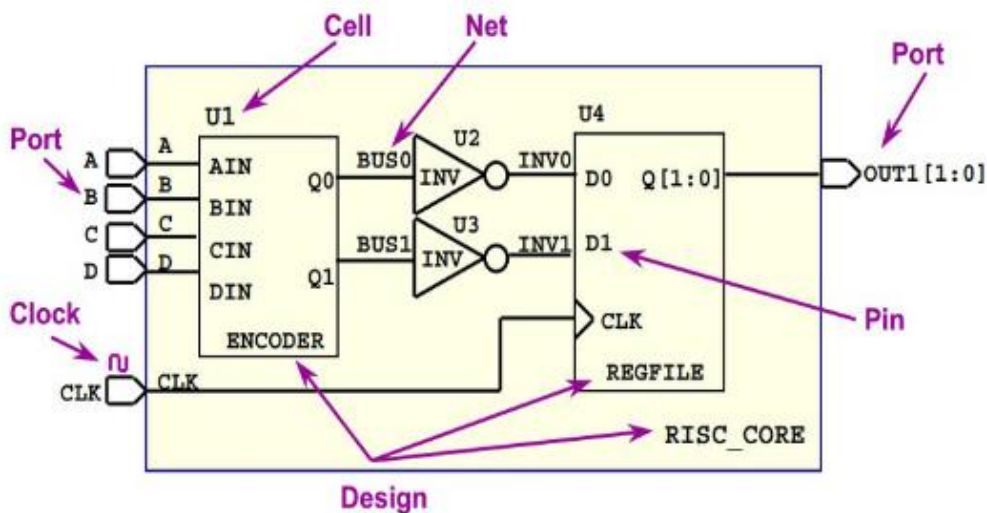
2、时钟周期

3、（1）input delay、output delay、时钟周期（虚拟时钟），输出 port 的 load

或者（2）input delay、output delay，加上 set_max_delay 约束组合逻辑，以及输出 port 的 load

出 port 的 load

请根据下图回答问题



- (1) 如何书写TCL指令，查看design当中有没有一个port叫做CLK?
- (2) 如何书写TCL指令，查看design当中所有的port
- (3) 如何书写TCL指令，得到所有方向是input的port
- (4) 请写出图中reference name?
- (5) 请写出图中instance name?

答：1.get_ports CLK

2.get_ports *

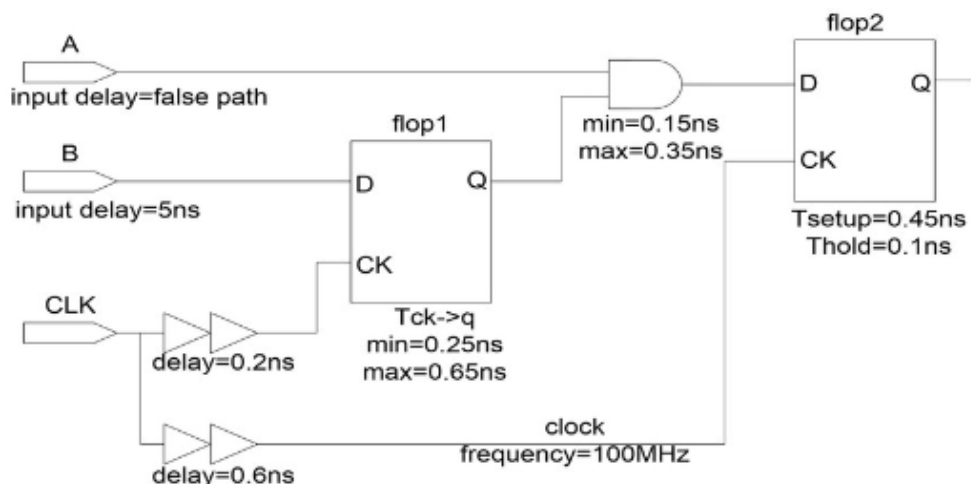
3.get_ports * -f "direction=in"

4.encoder inv regfile

5.U1 U2 U3 U4

第五章 静态时序分析作业题

1.



(1)[10分]请问对于flop2, 其建立时间裕量为多少?

(2)[10分]请问对于flop2, 其保持时间裕量为多少?

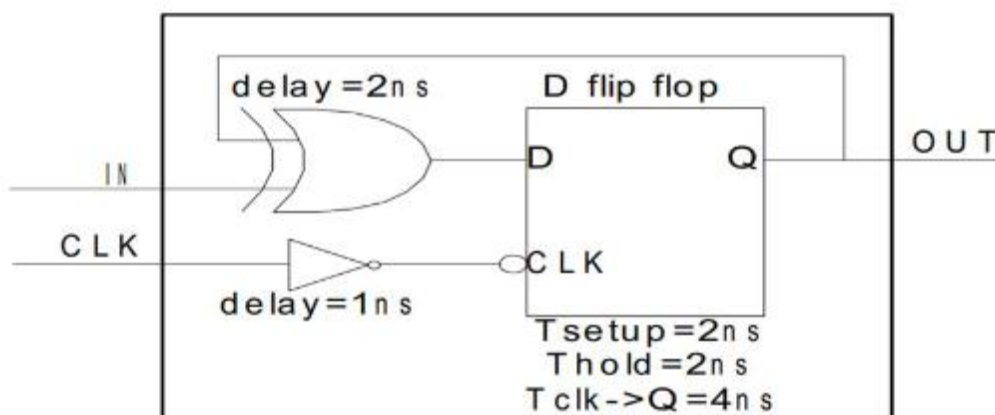
1. $T = 10$ $T_{capture} = 0.6$, $T_{launch} = 0.2$, $T_{ck2q} = 0.65$, $T_{dp} = 0.35$

$T_{setup} < 0.6 + 10 - 0.2 - 0.65 - 0.35 = 9.4$ 余量: $9.4 - 0.45 = 8.95$

2. $T_{capture} = 0.6$, $T_{launch} = 0.2$, $T_{ck2q} = 0.25$, $T_{dp} = 0.15$

$T_{hold} < 0.2 + 0.25 + 0.15 - 0.6 = 0$ 余量: $0 - 0.1 = -0.1$

2.



如果把上述电路整体看为一个触发器, 请回答如下问题

《芯动力——硬件加速设计方法》作业答案解析

(1) [20分]该电路的有效建立时间和保持时间是多少?请写出计算过程。

- A. $T_{\text{setup}} = 4 \text{ ns}$, $T_{\text{hold}} = 1 \text{ ns}$
- B. $T_{\text{setup}} = 3 \text{ ns}$, $T_{\text{hold}} = 0 \text{ ns}$
- C. $T_{\text{setup}} = 3 \text{ ns}$, $T_{\text{hold}} = 1 \text{ ns}$
- D. $T_{\text{setup}} = 2 \text{ ns}$, $T_{\text{hold}} = 0 \text{ ns}$

(2) [20分]该电路的最高时钟频率为多少? 请写出计算过程。

- A. 250 MHz
- B. 80 MHz
- C. 125 MHz
- D. 166.7 MHz

(3) [10分]该电路的功能与下列哪个触发器相似?[B]

- A. D flip flop with enable
- B. T flip flop
- C. JK flip flop
- D. SR flip flop

(4) [10分]对于一个同步电路, 以下哪个公式可以用于计算最高工作频率?

- A. $\text{Max Freq} = 1/(T_{\text{prop_delay}} + T_{\text{su}} + T_{\text{hold}})$
- B. $\text{Max Freq} = 1/(T_{\text{prop_delay}} + T_{\text{su}} + T_{\text{co}} + T_{\text{hold}})$
- C. $\text{Max Freq} = 1/(T_{\text{su}} + T_{\text{co}} + T_{\text{hold}} + T_{\text{clock_skew}})$
- D. $\text{Max Freq} = 1/(T_{\text{prop_delay}} + T_{\text{su}} + T_{\text{co}} + T_{\text{clock_skew}})$
- E. $\text{Max Freq} = 1/(T_{\text{prop_delay}} + T_{\text{su}} + T_{\text{hold}} + T_{\text{co}} + T_{\text{clock_skew}})$

1. 答: $\text{setup}=2+2-1=3\text{ns}$; $\text{hold}=2-2+1=1\text{ns}$

2. $T>2+4+2=8\text{ns}$, $f=125\text{M}$

3. B

4. D

3.

《芯动力——硬件加速设计方法》作业答案解析

Please reference the following for the next group of questions

Startpoint: fifo_rd_pd_reg_16_

(rising edge-triggered flip-flop clocked by my_clock)

Endpoint: we_bank_0_reg_2_

(rising edge-triggered flip-flop clocked by my_clock)

Path Group: my_group

Path Type: max

Point	Cap	Trans	Incr	Path
clock my_clock (rise edge)			0.0000	0.0000
clock network delay (ideal)			0.0000	0.0000
fifo_rd_pd_reg_16_/CK (p_SDFFHX4)			0.0000	0.0000 0.0000 r
fifo_rd_pd_reg_16_/Q (p_SDFFHX4)		0.0226	0.0542	0.1166 0.1166 f
obuf_U1904/Y (NAND3BX4)		0.0095	0.0643	0.0915 0.2082 f
obuf_U1903/Y (INVX8)	0.0188	0.0450	0.0385	0.2467 r
obuf_buf_1_add_524_U14/Y (NAND2X4)		0.0088	0.0431	0.0374 0.2841 f
U7015/Y (OAI21X4)	0.0111	0.1008	0.0781	0.3622 r
U22745/Y (AOI21X4)	0.0054	0.0481	0.0294	0.3916 f
obuf_U9743/Y (OAI21X4)	0.0067	0.0772	0.0638	0.4554 r
obuf_buf_1_add_524_U57/Y (XNOR2X4)		0.0098	0.1071	0.0769 0.5323 r
DP_OP_248_5346_8_U51/CO0 (AFCSHCINX2)		0.0037	0.0970	0.1378 0.6702 r
obuf_U9662/Y (MX2X4)	0.0077	0.0382	0.0825	0.7527 r
DP_OP_248_5346_8_U44/S (AFCSHCINX4)		0.0075	0.0462	0.1200 0.8726 f
obuf_U9746/Y (INVX6)	0.0174	0.0463	0.0394	0.9120 r
obuf_U1772/Y (NOR2X4)	0.0113	0.0554	0.0294	0.9414 f
U17999/Y (OA22X4)	0.0058	0.0398	0.1128	1.0542 f
obuf_U1852/Y (NOR2X4)	0.0116	0.0909	0.0686	1.1228 r
U26782/Y (INVX10)	0.0549	0.0558	0.0538	1.1765 f
U7011/Y (OA22X4)	0.0092	0.0465	0.0982	1.2747 f

《芯动力——硬件加速设计方法》作业答案解析

U27252/Y (CLKNAND2X2)	0.0057	0.0484	0.0402	1.3149 r
U24494/Y (XOR2X3)	0.0058	0.0531	0.0386	1.3536 f
obuf_U3105/Y (NOR2X4)	0.0056	0.0583	0.0497	1.4033 r
obuf_U3097/Y (NAND2X4)	0.0056	0.0341	0.0329	1.4362 f
obuf_U3094/Y (NOR2X4)	0.0053	0.0534	0.0421	1.4784 r
obuf_U1886/Y (AOI21X4)	0.0157	0.0607	0.0580	1.5364 f
obuf_U1861/Y (NOR3X4)	0.0101	0.1561	0.1036	1.6400 r
U24496/Y (IN VX4)	0.0073	0.0442	0.0351	1.6751 f
obuf_U1760/Y (OR2X4)	0.0135	0.0378	0.0731	1.7481 f
obuf_U9462/Y (IN VX12)	0.0325	0.0428	0.0358	1.7839 r
U17857/Y (OAI211X4)	0.0062	0.0992	0.0506	1.8345 f
U24495/Y (CLKIN VX6)	0.0036	0.0334	0.0188	1.8533 r
we_bank_0_reg_2_/D (p_SDFFRHQX4)		0.0334	0.0000	1.8533 r
data arrival time			1.8533	
clock my_clock (rise edge)			1.8000	1.8000
clock network delay (ideal)			0.0000	1.8000
we_bank_0_reg_2_/CK (p_SDFFRHQX4)			0.0000	1.8000 r
library setup time			-0.0701	1.7299
data required time				1.7299

data required time				1.7299
data arrival time				-1.8533

slack (VIOLATED)				-0.1234

分]根据以下静态时序分析报告回答问题。

- (1) 这个报告是哪个EDA工具产生的? [2分]
- (2) 报告最右侧一列中的“f”和“r”表示什么含义? [2分]
- (3) 该报告描述的是建立时间还是保持时间?
- (4) 该电路能跑的最高时钟频率为多少? [4分]
- (5) 该报告是否能够求出保持时间? [4分]
- (6) 该报告的数据与芯片布局布线之后的数据是否相同? 芯片布局布线阶段与DC综合阶段, 在处理时钟网络时有何不同? [4分]

1.design compiler 或者 PT

2.f 表示数据下降翻转， r 表示数据上升翻转

3.建立时间

4. $t=1.8533+0.0701=1.9234\text{ns}$ $f=1/t=519.9\text{MHZ}$

5.不能

6.不同，DC 时钟网络是理想的， 布局布线后时钟网络有延迟

第六章 作业题

1 (100分) 参考ASIC的设计流程，判断如下FPGA硬件开发的陈述是否正确。

1. FPGA的开发代码不需要可综合。
2. FPGA不需要进行静态时序分析。
3. FPGA不需要做时序约束。
4. FPGA不需要检查建立时间和保持时间。
5. FPGA通常也会存在亚稳态的同步问题。
6. FPGA硬件资源无限大。
7. FPGA通常有很对IP核可以调用。
8. PYNQ开发板中，Python程序运行在FPGA的PL部分。
9. FPGA的bit文件可以在不同FPGA开发板之间随意迁移。
10. FPGA完成bit文件下载后，需要使用片内逻辑分析仪来完成debug。

1.错 2.错 3.错 4.错 5.对 6.错 7.对 8.错(PS) 9.错(芯片不同，架构不同，引脚不同) 10.对；片内主要是虚拟逻辑分析仪，片外是真实的逻辑分析仪。