

FPGA 经典 100 问 之<HDL 28 问>

FPGA 是什么？

FPGA 即现场可编程逻辑门阵列（英语：Field Programmable Gate Array, FPGA），是一个含有可编辑元件的半导体设备，可供使用者现场程式化的逻辑门阵列元件。它是作为专用集成电路（ASIC）领域中的一种半定制电路而出现的，既解决了定制电路的不足，又克服了原有可编程器件门电路数有限的缺点。

现将历年来众多网友和专家联手打造的 FPGA 经典 100 问奉献给大家,包括<HDL 28 问>、<仿真 20 问>、<设计实现 31 问>、<下载验证 16 问>、<入门与提高 5 问>。**初学者必备!**

1、初学者求助 verilog 问题：按键消抖

[瑞明明](#) 问：

```
always @(negedge S1 or negedge S2 )
begin
if(!S1)
dd = dd -1'b1;
else if(!S2)
dd <= dd - 1'b1;
end
```

S1 和 S2 为两个已经经过消抖的按键，

使用上面语句是把 S1 和 S2 作为敏感信号，

当按下 S2 时寄存器 dd 是减 1 显示，但是按下 S1 时候却不是，是一直在变，求教，此问题困扰多时！谢谢！

[qingxian](#) 答：

首先第一个问题，时序电路中，敏感变量表里面的信号只有一个能综合成时钟信号，另外一个异步载入信号，你可以看一下 RTL 级，就可以看到 S2 是在时钟输入端，而 S1 是在异步载入端口。

[ilove314](#) 答：

你必须一个时钟一次技术，而 S 拉低那要保持好长时间呢，所以你取 S 的上升沿或下降沿作为采集指示就没问题。

[查看更多回答与讨论>>](#)

2、求助特权同学串口通讯问题

[beihang873](#) 问：

深入浅出玩转 FPGA 串口通讯有如下程序：

```
//下面的下降沿检测可以滤掉<20ns-40ns 的毛刺(包括高脉冲和低脉冲毛刺)，
```

```
//这里就是用资源换稳定（前提是我们对时间要求不是那么苛刻，因为输入信号打了好几拍）
```

```
//（当然我们的有效低脉冲信号肯定是远远大于 40ns 的）
```

```
assign neg_rs232_rx = rs232_rx3 & rs232_rx2 & ~rs232_rx1 & ~rs232_rx0; //接收到下降沿后 neg_rs232_rx 置高一个时钟周期
```

求助 这句话是什么意思,怎么能起到滤波的效果呢?

[新人鹰](#)答:

滤波就是相当于延迟 20~40ns,他与前面的非阻塞赋值有一定关系: `rs232_rx0 <= rs232_rx;`
`rs232_rx1 <= rs232_rx0;` `rs232_rx2 <= rs232_rx1;` `rs232_rx3 <= rs232_rx2;` `assign neg_rs232_rx =`
`rs232_rx3 & rs232_rx2 & ~rs232_rx1 & ~rs232_rx0;` 只有当 `rs232_rx` 由 1 变为 0, 且经过
`rs232_rx0 <= rs232_rx;` `rs232_rx1 <= rs232_rx0;` 将 `rs232_rx0` `rs232_rx1` 变为 1 时 `neg_rs232_rx`
才为 1,这个时序在 40ns 以上

[查看更多回答与讨论>>](#)

3、一个月之内做一个 FPGA 项目,很急,请高手指点!

[大白菜 LL](#) 问:

刚开始学 FPGA,对 verilog 语言也不怎么了解,同事让我一个月之内做一个 FPGA 项目,所以请高手们给与支持啊。和大家描述一下项目内容:用 FPGA 芯片 (cyclone IV) 控制 2 片级联的 74HC595 芯片,使得 74HC595 并行输出固定的 16 位数据,而这 16 位数据是由 PC 经过串口发送给 FPGA 的,就是说 FPGA 需要做一个串口和一个 74HC595 的控制程序,并且 FPGA 还要输出一路脉宽可调的 PWM (也要用串口设置)。是用 verilog 写程序做几个模块,还是把 FPGA 做为 NIOS II 控制器用 C 语言写程序。

请高手们讲讲方案,给些项目相关的资料。我想对于熟悉 FPGA 的人来说应该很简单,我以前只用过单片机,了解一点 ARM 的东西。单纯的用单片机做一个这样的控制我也可以,之前在学校也都用过。现在是要用一片 FPGA 做 9 路 2 片级联 74HC595 的控制和 9 路脉宽可调的 PWM 输出! (每路 2 片级联的 74HC595 输出的 16 位数据和每路可调脉宽的 PWM 分别送给延时芯片 MC100EP196D,延时芯片锁存这 16 位数据,根据这 16 位数据值将 PWM 经过固定延时后输出)

[ilove314](#) 答:

先去了解你需要实现的芯片的功能,然后把输入输出关系理清楚,再考虑 FPGA 如何设计

[查看更多回答与讨论>>](#)

4、关于双向端口的数据类型

[edelajiang](#) 问:

一直看资料都是说 inout 端口只能是 wire 型,但我在一本书上看到这个例子,把它定义了 reg 型,而且编译的时候还没有报错很是纳闷,谁能给予解释

```
module bit4(ctrl,din,q,dout);
input ctrl;
input[3:0]din;
inout[3:0]q;
output[3:0]dout;
reg[3:0]dout,q;
always@(ctrl,din,q)
if(!ctrl)
dout=q;
```

```
else
begin
q=din; dout=4'hz;
end
endmodule
```

[ilove314](#) 答:

你的代码里面 q 根本不是 reg (例子不恰当), 综合工具把它综合为 wire 型了。如果严格来讲, 如果综合工具认为他是 reg 就会报错的。reg 和 wire 的定义通常是和书本说的一致的, 但是也有例外情况。你写的一段是组合逻辑, 但是你要看看 RTL 视图是否综合工具真的是 wire 型, 没有综合出 reg。

[查看更多回答与讨论>>](#)

5、assign, 连续赋值语句, 你怎么看?

[葱上头](#) 问:

之前看到的连续赋值语句, 括号里都是判断逻辑状态, 再给与赋值, 现遇到一个问题, 想请教各位大侠, assign c= (a>b) - 1'b1:1'b0;这种语句可否? 不管 a,b 的位宽和 a,b 到底是 wire 还是 reg。怎么看?

[ilove314](#) 答:

可以, 你是不是少了一个问号?

[wenzhang](#) 答:

只有被赋值的是 wire 型就可以了

[查看更多回答和讨论>>](#)

6、为什么我定义的数组总是编译错误啊?

[刘毓](#) 问:

请问学习板配套的 quartus ii 软件是什么版本的啊? 支持数组定义吗 为什么我定义的数组总是编译错误啊? 如果要定义常量数组例如 a[i]={FF,FF,05,03,1E,83...}怎么书写啊? i 定义成什么类型啊?

[ilove314](#) 答:

verilog 不是 C 语言, 还分 RTL 级和行为级, 这个概念请先分清楚。如果你发送数据固定, 可以考虑放在 FPGA 内部的 ROM 或 RAM 里, 又或者定义寄存器来存储。

[查看更多回答和讨论>>](#)

7、请教 I2C 通信实验例程问题, 不太明

[lvhao1990](#) 问:

```
STOP1: begin
    if(`SCL_LOW) begin
        sda_link <= 1'b1;//output
        sda_r <= 1'b0;
        cstate <= STOP1;
    end
end
```

```

else if(`SCL_HIG) begin
    sda_r <= 1'b1; //scl 为高时， sda 产生上升沿（结束信号）
    cstate <= STOP2;
end
else cstate <= STOP1;
end
STOP2: begin
    if(`SCL_LOW) sda_r <= 1'b1;
    else if(cnt_20ms==20'hffff0) cstate <= IDLE;
    else cstate <= STOP2;
end

```

上述代码此句是什么意思 else if(cnt_20ms==20'hffff0) cstate <= IDLE;
 还有就是状态机为何要用两个 STOP(STOP1,SSOP2)?
 非常希望高手能指点一下，非常感谢!

[紫菜蛋花汤](#) 答:

防止同一次按键再次被响应

[查看更多回答与讨论>>](#)

8、求大神：关于 verilog 中语句时序问题

[J1](#) 问:

在 always 语句下有如下情况:

...

```

if(clk_bps)

begin
    num <= num+1'b1;
    case (num)
        4'd0: rs232_tx_r <= 1'b0; //发送起始位
        4'd1: rs232_tx_r <= tx_data[0]; //发送 bit0
        4'd2: rs232_tx_r <= tx_data[1]; //发送 bit1
        4'd3: rs232_tx_r <= tx_data[2]; //发送 bit2
        4'd4: rs232_tx_r <= tx_data[3]; //发送 bit3
        4'd5: rs232_tx_r <= tx_data[4]; //发送 bit4
        4'd6: rs232_tx_r <= tx_data[5]; //发送 bit5
        4'd7: rs232_tx_r <= tx_data[6]; //发送 bit6
        4'd8: rs232_tx_r <= tx_data[7]; //发送 bit7
        4'd9: rs232_tx_r <= 1'b1; //发送结束位
        default: rs232_tx_r <= 1'b1;
    endcase
end
else if(num==4'd11) num <= 4'd0;

```

第一个问题：如果上述语句中 num 初始值为 0，当 clk_bps 为真时。那 case 中为 0 的语句还

执行吗？

第二个问题:本语句来自特权同学的串口发送语句,想问 num 在定义为 reg 类型初始值为 x,为什么可以不初始化为 0 便能正确执行发送 8 位数据? (补充:在整个串口发送程序中如果 clk_bps 不为真时,也没有复位信号时,num 一直为 x,为什么程序能正常运行?)

[ilove314](#) 答:

1. 并行处理,这个不是软件。 2. 在实际器件当中,默认初始值为 0。

[查看更多回答与讨论>>](#)

9、请帮忙看看这个按键消抖的程序。

[june guo](#) 问:

目前正在看特权的 I2C 程序,发现按键部分没有消抖,于是就将按键部分改了下,在捕捉按键信号,打 3 拍调试程序发现按键没有反应,于是改为打 2 拍,调试就 ok 了。3 拍比 2 拍就多了 20ns,而按键消抖要延迟 20ms,差那么多,为何多了一拍就不行了呢?

高手路过一定要告诉我为什么,我是个喜欢钻牛角尖的女生,找不到答案的话,恐怕再也出不来了。

```
// 按键检测
reg sw1_r0,sw1_r1,sw1_r;
reg sw2_r0,sw2_r1,sw2_r;
reg[19:0] cnt_20ms;
always @ (posedge clk or negedge rst_n)
if(!rst_n) begin
sw1_r0 <= 1'b1;
sw1_r1 <= 1'b1;
//sw1_r2 <= 1'b1;
//为何打三拍,调试时就不能实现,三拍就用了 60ns,为何就不能实现按键的捕捉?
sw2_r0 <= 1'b1;
sw2_r1 <= 1'b1;
//sw2_r2 <= 1'b1;
end
else begin
sw1_r0 <= sw1;
sw1_r1 <= sw1_r0;
//sw1_r2 <= sw1_r2;
sw2_r0 <= sw2;
sw2_r1 <= sw2_r0;
//sw2_r2 <= sw2_r2;
end
end
wire sw1_rst = sw1_r0 || sw1_r1;
wire sw2_rst = sw2_r0 || sw2_r1;//打了两拍,若为 0,表示有按下
```

```
always @ (posedge clk or negedge rst_n)
if(!rst_n) cnt_20ms <= 0;
else if(sw1_rst && sw2_rst) cnt_20ms <= 0;
else cnt_20ms <= cnt_20ms + 1'b1;
```

```
always @ (posedge clk or negedge rst_n)
if(!rst_n) begin
sw1_r <= 1'b1; //键值寄存器复位，没有键盘按下时键值都为 1
sw2_r <= 1'b1;
end
else if(cnt_20ms == 20'hffff) begin
sw1_r <= sw1; //按键 1 值锁存
sw2_r <= sw2;
end
end
```

[ilove314](#) 答:

//sw2_r2 <= sw2_r2; 当然出不来了; 应该是//sw2_r2 <= sw2_r1;吧
[查看更多回答与讨论>>](#)

10、 Verilog HDL 程序问题

[wenjiaopu](#) 问:

```
always @(negedge clk) begin
di<=din;
end
always @(posedge clk) begin
dout<=d1;
end
end
```

程序目的是在 clk 的上升与下降沿分别进行赋值。

请问：在 Verilog HDL 中能否写两个 always 语句？？

[jlx_cuc](#) 答:

不太清楚你的意思？ 一个.v 文件里面可不可以包含两个 always 语句？当然可以啊，这个没有限制……但是就上面那个例子，还是建议不要这么写，在一个模块里（也就是一个.v 文件里），既用到时钟的上升沿又用到时钟的下降沿，这样会让编译器把时钟当做普通逻辑信号来处理，那整个电路的时序就慢下来了

[查看更多回答与讨论>>](#)

11、如何编写出综合工具可以识别的程序，请高手指导…

[低调的激情](#) 问:

本人刚学习 verilog，经常编写出综合工具不能识别的程序，对此非常苦恼，希望高手谈谈你们写程序的经验方法，谢谢！

[jlx_cuc](#) 答:

“不能识别”是指不能综合么？还是别的…… 如果不能综合的话，楼主不妨看看 Altera 提供的 Coding Style，参考参考几个例子以后就大概能了解什么样的代码风格可综合了~祝你好远~

[查看更多回答与讨论>>](#)

12、关于 test bench 如何编写，求推荐？

[milpla](#) 问：

本人学习 FPGA 已经一段时间了，可以编写一定的代码，但是测试验证的时候，总是不知道激励模块应该怎样写才能更好的测试代码，求各位高手推荐学习方法，和相关学习书籍！

[jlx_cuc](#) 答：

要说理论就上升到验证方法学了，testbench 的编写可以找找牛人们的博客看看是个不错的选择，设计的方法慢慢就熟悉了。

[查看更多回答与讨论>>](#)

13、求助啊！新手不会 Verilog HDL

[wenjiaopu](#) 问：

```
module dual_trig
(in1,in2,out1);
input in1, in2;
output out1;
always @(posedge in1 or negedge in2)
begin
if(in1)
begin
out1=1'b1;
end
else if (~in2)
begin
out1=1'b0;
end
end
endmodule
```

出错了：

Error (10137): Verilog HDL Procedural Assignment error at dual_trig.v(25): object "out1" on left-hand side of assignment must have a variable data type

Error (10137): Verilog HDL Procedural Assignment error at dual_trig.v(29): object "out1" on left-hand side of assignment must have a variable data type

哪位高手帮忙解决一下啊！？

[jlx_cuc](#) 答：

这个应该是 out1 作为在 always 里面赋值，需要给它定义成一个 reg 型，你可以把 output out1 改成 output reg out1

http://bbs.ednchina.com/FORUM_POST_23_515822_0.HTM 我在这里传了本 pdf 的书啊, 可以推荐给你看看, 上面给了很多参考设计。至于你说的这个问题为什么——always 语句你可以理解成是被动触发的事件, 就是当 in1 或者 in2 发生变化的时候, out1 执行相应的变化处理, 那就需要一个“东西”来保存 out1 前后的两种状态, 也就是寄存器了。还有另一种语法 assign, 它可以说只是起到一个连接的作用, 所以它的左边变量一般要求是 wire“线”型, 也就是把****样的逻辑结果“连接”到某个“线”上去。不知道这么讲会不会更明白一点
[查看更多回答与讨论>>](#)

14、FPGA 产生波形信号（正弦波、方波）

[gxunddy](#) 问:

想用 FPGA (verilog hdl 语言编写) 做个波形发生器, 其中产生正弦波、方波, 但不知从何下手。求助 ING!!
菜鸟向老鸟们致敬了!

[paradoxfx](#) 答:

写出表达式来, 然后编程序就可以了啊, 正弦波可以查表, 方波只要达到周期值取反就行了
[smarteebit](#) 答:

简要提示:

正弦波可以用 DDS 算法, 或者 CORDIC 算法产生, 输出接 DAC 实现数字到模拟的变换;
方波的话, FPGA 本身输出不就是方波吗? 不过 FPGA 输出的方波本身的时间分辨率是和 FPGA 工作时钟相关的, 最小也就是做到一个时钟周期, 所以如果需要方波时间分辨率很小的话, 可以将上面产生的正弦波经过模拟比较器后获得。

[查看更多回答与讨论>>](#)

15、用 verilog 编写一句汉语显示在 12864 液晶显示上, 怎么放置要显示的字?

[xiangzi090716](#) 问:

各位大神大家好, 现在遇到个问题: 我想用 verilog 语言编写一个在 12864 液晶显示上显示一句汉语, 但是不知道怎么放置要显示的汉字, 所以请大神们指导下, 或者给我一个简单的实例, 谢谢啊!

[ppc68](#) 答:

找个字库软件, 把汉字转成数据再对应坐标显示

[查看更多回答与讨论>>](#)

16、这个 VHDL 的信号赋初值语句怎么理解?

[andyandy](#) 问:

```
signal abc: std_logic_vector(5 downto 0):=(others=>'0');
```

困扰很久, 一直不明白。哪位大侠能给详解一下? 谢谢!

[guangbaohu](#) 答:

不要使用它来赋初值, 一般上电后寄存器会有一个初值, 但不会是一个固定的数, 要么是 0, 要么是 1, 还有未知的状态, 如果你用它来赋初值, 在功能仿真里没问题, 但时序仿真里会出现乱七八糟的错误结果 (可以用一个简单的计数器为例, 试试你就明白了), 除非你有一个复位信号给它打一下初值, 你完全可以通过更改你的设计来避开它赋初值。

[查看更多回答与讨论>>](#)

17、新手提问：寄存器都为 1 了 neg_rs232_rx 才为真啊？

[517650971](#) 问：

```
reg rs232_rx0,rs232_rx1,rs232_rx2,rs232_rx3;
```

```
//接收数据寄存器，滤波用
```

```
wire neg_rs232_rx;
```

```
//表示数据线接收到下降沿
```

```
always @ (posedge clk or negedge rst_n) begin
```

```
if(!rst_n) begin
```

```
rs232_rx0 <= 1'b0;
```

```
rs232_rx1 <= 1'b0;
```

```
rs232_rx2 <= 1'b0;
```

```
rs232_rx3 <= 1'b0;
```

```
end
```

```
else begin
```

```
rs232_rx0 <= rs232_rx;
```

```
rs232_rx1 <= rs232_rx0;
```

```
rs232_rx2 <= rs232_rx1;
```

```
rs232_rx3 <= rs232_rx2;
```

```
end
```

```
end
```

```
//下面的下降沿检测可以滤掉<20ns-40ns 的毛刺(包括高脉冲和低脉冲毛刺)，
```

```
//这里就是用资源换稳定(前提是我们对时间要求不是那么苛刻，因为输入信号打了好几拍)
```

```
// (当然我们的有效低脉冲信号肯定是远远大于 40ns 的)
```

```
assign neg_rs232_rx = rs232_rx3 & rs232_rx2 & ~rs232_rx1 & ~rs232_rx0;
```

```
//接收到下降沿后 neg_rs232_rx 置高一个时钟周期
```

```
assign neg_rs232_rx = rs232_rx3 & rs232_rx2 & ~rs232_rx1 & ~rs232_rx0;这句话是什么意思啊？
```

```
是不是 rs232_rx0,rs232_rx1,rs232_rx2,rs232_rx3;这些寄存器都为 1 了 neg_rs232_rx 才为真啊？
```

[wsgcn](#) 答：

一般用于滤波\去抖，

如 rs232_rx 空闲为 1,则 neg_rs232_rx =0(~rs232_rx0 =0,~rs232_rx1 =0,rs232_rx2 =1,rs232_rx3

=1);当 rs232_rx 为 0,产生下降沿,则在第一个 CLK,rs232_rx0 =0,此时 neg_rs232_rx 仍然为 0;下一个 CLK,rs232_rx0 =0,rs232_rx1 =0,此时 neg_rs232_rx =1;再下一个 CLK,neg_rs232_rx =0 可以看出,neg_rs232_rx 产生了一个 CLK 的高电平.可以用来确认事件的开始.

[参与更多回答与讨论>>](#)

18、求助：流水灯实验中的一个问题

[weihei](#) 问：

用此种写法编译不通过。经分析为画红线部分的问题

```
always @(posedge clk or negedge rst_n)
] begin
  if(!rst_n)
    count1 <= 21'b0;
  else if(count1 > 21'd2_000)
    count1 <= 21'd0;
  else
    count1 <= count1 + 1'b1;
  end

always @(posedge clk or negedge rst_n)
] begin
  if(!rst_n)
    count1 <= 21'b0;
  else if(count1 > 21'd500_000 && count1 <
    led <= 1'b0;
  else
    led <= 1'b1;
  end
```

改动后如下。此时编译通过。。但是原因未知。跪求高手解释

```
always @(posedge clk or negedge rst_n)
] begin
  if(!rst_n)
    count1 <= 21'b0;
  else if(count1 > 21'd2_000)
    count1 <= 21'd0;
  else
    count1 <= count1 + 1'b1;
  end

always @(posedge clk or negedge rst_n)
] begin
  if(!rst_n)
  ] begin
    // #10 count1 <= 21'b0;
    led <= 1'b1;
  end
  else if(count1 > 21'd500_000 && count1 < 21'd1_000_000)
    led <= 1'b0;
  else
    led <= 1'b1;
  end
```

[yuechenping](#) 答:

在 verilog 中, 几个 always 块是并行同时运行的

[xi0126](#) 答:

同一变量不能在两个以上 always 中赋值, 这个 verilog 语言的语法知识 还是要多看下书啊
[查看更多回答与讨论>>](#)

19、选择 VHDL 还是 verilog HDL?

[chenghuitong](#) 问:

一直在用 VHDL 语言, 但是在网上看到好多人用 Verilog。请问大家我有必要转型学 Verilog 吗?

[smarteebit](#) 答:

各有优劣, 不过目前知名企业中, ZTE 和 Huawei 内部的 FPGA 开发都是统一规定使用 Verilog。供参考。

[逍遥散人](#) 答:

两个差不多的, 转很容易的, 多学一个也无所谓
[查看更多回答与讨论>>](#)

20、新手, 关于 vga 的疑惑啊。

[飞絮忆](#) 问:

在特权老师 vga 显示的代码中都没有消隐信号的啊。

[ilove314](#) 答: HSY 和 VSY 是行场同步信号, 应该包含你说的所谓消隐信号

[查看更多回答与讨论>>](#)

21、【语言求助】请教一个 verilog 程序设计的问题!

[mimmimhu](#) 问:

才开始接触学习 verilog 遇到了解决不了的问题。

```
module ttt(out,rst,clk,start);  
output [127:0] out;  
input rst,clk,start;  
reg [127:0] out;  
always @(posedge clk or negedge rst)  
begin  
if(rst==0) out<=128'b0;  
else if(start==0)  
out<=out+1;  
else  
out<=out;  
end  
endmodule
```

如上的程序是个计数器, 现在想提高它的工作频率, 只能对程序进行修改, 布局布线不能改变。该如何修改?

用 case 语句代替 if 语句?

如果用 pipeline 结构的话, 该如何实现? 怎样分割? 谢谢大家!!

[ys3663391](#) 答:

个人认为改成 case 语句可能会改善综合的效果, 但是是否会提高电路性能, 不好说。至于楼主想用 pipeline 的话我觉得不怎么好划分

[陈涛](#) 答:

分成低 64 位和高 64 位加法, 分两拍就可以

MIMI, 你在劈腿那

[andyandy](#) 答:

start 是触发信号, 不能改。

原程序只在上升沿累加, 可改为在上升沿和下降沿均累计, 计数频率可提高 1 倍。只需将
always @(posedge clk or negedge rst)

改为

always @(posedge clk or negedge clk or negedge rst)

即可。

[paradoxfx](#) 答:

用 case 语句代替 if 语句布局布线, 最终结果是没有区别的; 用流水线的话布局布线倒是肯定改变了; 提高输入时钟频率吧

[查看更多回答与讨论>>](#)

22、Verilog 除法的问题

[sunpei0337115](#) 问:

想请教大家一个有关 Verilog 除法的问题, 由于项目碰到要计算除法, 而且是实时计算, 下面是程序代码:

```
/*  
*  
// Company :  
// Engineer : Andy  
// Create Date : 2011.05.09  
// Design Name :  
// Module Name : data_calculate  
// Project Name :  
// Target Device: Cyclone EP2C8Q208C8N  
// Tool versions: Quartus II 9.0  
// Description : 数据计算模块  
//  
// Revision : V1.00  
// Additional Comments :  
//  
*****  
*/  
module data_calculate  
(
```

```

Clk, //输入时序信号
Reset, //异步复位信号
// V_0, //初速度 mm/s
// Acc, //加速度 mm/s^2
// S, //固定行走距离 um
Full, //FIFO 满标志

Wr_en, //FIFO 写信号
T_average //每个脉冲平均周期 取 100us 为基准时间
);
input Clk;
input Reset;
//input [7:0] V_0;
//input [7:0] S;
//input [15:0] Acc;
input Full;
output [31:0] T_average;
output Wr_en;
reg [31:0] V_temp; //中间速度寄存器
reg [31:0] T_sample_S; //采样时间寄存器
reg [31:0] T;
reg write_sig; //写信号寄存器
reg [2:0] i;
parameter V_0 = 5; //参数设置初始值
parameter Acc = 1000;
parameter S = 350; //每个脉冲走过的距离为 3.5um, 取 10 个脉冲, 换算成整型单位为 1/10um

//每一个上升沿计算一次采样时间
always @(posedge Clk or negedge Reset)
begin
if (!Reset)
begin
T_sample_S = 0;
write_sig = 0;
V_temp = 0;
T = 0;
end

else
begin
case (i)

3'd0:
if (!Full) //判读 FIFO 是否满, 若不满则写入数据

```

```

begin
write_sig = 1; //写信号有效
//T_sample_S = V_0 / Acc; //计算初速度占用的时间，单位为 100us
i = i + 1'b1;
end

3'd1:
if (!Full)
begin
T = T + T_sample_S; //统一时间单位 100us
V_temp = V_0 * 10000 + Acc * T; //mm/s * 10000 = mm/100us，避开小数，同时放大 1000 倍
T_sample_S = S * 10_000 / V_temp; //转换成 100us 为基础时间，统一单位为 mm， S *
100_000_000 / (V_temp * 1000)
i = 3'd1;
if (T_sample_S < 2) //直到 200us，结束采样时间计算 (仿真时取 29，实际运行时取 2)
begin
i = i + 1'b1;
T_sample_S = 2;
end
end

3'd2: //FIFO 满，停止写入数据
if (!Full)
begin
write_sig = 0;
end
/*
else
begin
T_sample_S = 0;
//i = 2'd2;
end
*/
default:
begin
T = 0;
V_temp = 0;
T_sample_S = 0;
i = 3'd0;
end
endcase
end
end
assign T_average = T_sample_S;

```

```
assign Wr_en = write_sig;
endmodule
```

综合后，用掉了近 1100 门，我查看了一下，基本都是除法器用掉的，这里 Quartus 自动综合成 IP 核了，由于这里的除法数据位为 32 位的，我单独测试了一下 LPM 的 32 位除法器 IP 核，确实很占资源，不知道大家有何建议，谢谢！

[446041452](#) 答：

是的啊一般的 像乘法除法这些都比较占用资源的

[sunpei0337115](#) 答：

恩，是的，16 位和 8 位的就少了很多，但是这里我为了避开小数，将数据放大了，所以不得已用到 32 位，甚至超过 32 位了，主要还是为了数据的精度，不知道您有没有什么高见能够避开除法呢？

[查看更多回答与讨论>>](#)

23、关于 MicroBlaze 上数组问题

[chk1989](#) 问：

想要从串口接收一帧图片，就开一个比较大的数组来接收，但是似乎数组太大而导致程序在调试的时候不能执行。下面程序将 IMAGESIZE 改为 100，程序就能正常运行，为什么？

```
#include "xparameters.h"
#include "stdio.h"
#include "xutil.h"
#include "xuartlite.h"
#include "xgpio.h"
#include "xstatus.h"

#define IMAGESIZE 76800
u8 DataBuffer[IMAGESIZE];

int main()
{
    XUartLite UartLite;
    XGpio LED;
    int RecvNum,i,j;

    int state = XUartLite_Initialize( &UartLite , XPAR_UARTLITE_0_DEVICE_ID );
    if(state!=XST_SUCCESS ){
        return state;
    }

    state = XGpio_Initialize(&LED ,XPAR_LEDS_8BIT_DEVICE_ID);
    if(state!=XST_SUCCESS ){
        return state;
    }
}
```

```

}

XGpio_DiscreteSet(&LED , 1,0xffffffff);
XGpio_SetDataDirection(&LED,1,0);

RecvNum = 0;

while(1)
{
    XGpio_DiscreteWrite(&LED,1,0 );
    RecvNum = XUartLite_Recv(&UartLite, DataBuffer , IMAGESIZE );

    if(RecvNum == IMAGESIZE)
        XGpio_DiscreteWrite(&LED,1,0xffffffff );
    else
        XGpio_DiscreteWrite(&LED,1,0x0000000f );

    for(j=0;j<100000;j++);
    XGpio_DiscreteWrite(&LED,1,0 );
}

return 0;
}

```

[wangshaosh](#) 答:

变量+程序=本地 memory

变量定义太多肯定不行!

如果把本地 memory 设置大一些的话就可以了 比如 256K 或者更大

[参与更多回答与讨论>>](#)

24、急救! Error: Can't elaborate user hierarchy "I2C:inst" 什么意思?

[jenny4594](#) 问:

Error: Can't elaborate user hierarchy "I2C:inst"这句话是编译时产生的一个错误,请各位帮帮忙看下是什么意思,急用,麻烦各位了

[sdjntl](#) 答:

是 Quartus 软件产生的错误吧。看不到工程,像是 I2C 的问题。

[jenny4594](#) 答:

是啊,最近在写那个 I2C 的程序,搞了几天了,没有什么很大的进展。

[参与更多回答与讨论>>](#)

25、quartus ii 出现 “can't find design entity ” 求助

[yiyiduiying](#) 问:

quartus ii 在新建工程是出现 can't find design entity 不知哪位大侠知道原因,还有要怎样解

决啊???
救救小弟，谢谢！

[wrhwindboy](#) 答：

这个是你的 quartus 的设置问题，要在 quartus 的创建新项目时指定的 top module 的名字和你的.v 文件的最上层的名字一样。不能有中文路径。

[paradoxfx](#) 答：

我明白了就是说工程的名字要和顶层文件的名字一样

[查看更多回答与讨论>>](#)

26、vhdl 高手帮帮忙，程序编译有警告，影响仿真结果

[mincheng](#) 问：

诸位 vhdl 高手，程序在编译时出现警告，仿真时数码管显示始终为 0，这是教材上的程序，想了半天无法解决，请大家帮帮忙。出现警告如下：Warning (10631): VHDL Process Statement warning at plcl.vhd(82): inferring latch(es) for signal or variable "seg7", which holds its previous value in one or more paths through the process

Warning: Latch seg7[0]\$latch has unsafe behavior

Warning: Ports D and ENA on the latch are fed by the same signal cnt[13][code]library ieee;

use ieee.std_logic_1164.all;

use IEEE.std_logic_arith.all;

use ieee.std_logic_unsigned.all;

```
entity plcl is
```

```
port ( sysclk : in std_logic; --10MHZ 频率输入
```

```
inclk : in std_logic; --待测频率信号输入
```

```
seg7:out std_logic_vector(6 downto 0); --7 段显示控制信号 (abcdefg)
```

```
scan:out std_logic_vector(7 downto 0)); --数码管地址选择信号
```

```
end;
```

```
architecture one of plcl is
```

```
signal cnt:std_logic_vector (24 downto 0);
```

```
signal clk_cnt:std_logic;
```

```
signal cntp1,cntp2,cntp3,cntp4,cntp5,cntp6,cntp7,cntp8:std_logic_vector (3 downto 0);
```

```
signal cntq1,cntq2,cntq3,cntq4,cntq5,cntq6,cntq7,cntq8:std_logic_vector (3 downto 0);
```

```
signal dat:std_logic_vector(3 downto 0);
```

```
signal latch:std_logic_vector(3 downto 0);
```

```
begin
```

```
-----0.5HZ 分频-----
```

```
process (sysclk)
```

```
begin
```

```
if sysclk'event and sysclk='1' then
```

```
if cnt=4999999 then
```

```
clk_cnt <=not clk_cnt;
```

```
cnt <=(others=>'0');
```

```
else cnt<=cnt+1;
```

```

end if;
end if;
end process;
-----1s 内计数-----
process (inclk)
begin
if inclk'event and inclk='1' then
if clk_cnt='1' then
if cntp1="1001" then cntp1<="0000";
if cntp2="1001" then cntp2<="0000";
if cntp3="1001" then cntp3<="0000";
if cntp4="1001" then cntp4<="0000";
if cntp5="1001" then cntp5<="0000";
if cntp6="1001" then cntp6<="0000";
if cntp7="1001" then cntp7<="0000";
if cntp8="1001" then cntp8<="0000";
else cntp8<=cntp8+1; end if;
else cntp7<=cntp7+1; end if;
else cntp6<=cntp6+1; end if;
else cntp5<=cntp5+1; end if;
else cntp4<=cntp4+1; end if;
else cntp3<=cntp3+1; end if;
else cntp2<=cntp2+1; end if;
else cntp1<=cntp1+1; end if;
else
if cntp1/="0000" or cntp2/="0000" or cntp3/="0000" or cntp4/="0000" or ---对计数值锁存
cntp5/="0000" or cntp6/="0000" or cntp7/="0000" or cntp8/="0000"
then
cntq1<=cntp1;cntq2<=cntp2;cntq3<=cntp3;cntq4<=cntp4;
cntq5<=cntp5;cntq6<=cntp6;cntq7<=cntp7;cntq8<=cntp8;
cntp1<="0000";cntp2<="0000";cntp3<="0000";cntp4<="0000";
cntp5<="0000";cntp6<="0000";cntp7<="0000";cntp8<="0000";
end if;
end if;
end if;
end process;
-----扫描数码管-----
process(cnt(15 downto 13),cntq1,cntq2,cntq3,cntq4,cntq5,cntq6,cntq7,cntq8,dat)
begin
case cnt(15 downto 13) is
when "000"=>scan<="00000001";dat<=cntq1;
when "001"=>scan<="00000010";dat<=cntq2;
when "010"=>scan<="00000100";dat<=cntq3;
when "011"=>scan<="00001000";dat<=cntq4;

```

```

when "100"=>scan<="00010000";dat<=cntq5;
when "101"=>scan<="00100000";dat<=cntq6;
when "110"=>scan<="01000000";dat<=cntq7;
when "111"=>scan<="10000000";dat<=cntq8;
when others =>null;
end case;
end process;
-----数码管显示译码-----
process(dat)
begin
case dat is
when "0000" =>seg7 <= "1111110";
when "0001" =>seg7 <= "0110000";
when "0010" =>seg7 <= "1101101";
when "0011" =>seg7 <= "1111001";
when "0100" =>seg7 <= "0110011";
when "0101" =>seg7 <= "1011011";
when "0110" =>seg7 <= "1011111";
when "0111" =>seg7 <= "1110000";
when "1000" =>seg7 <= "1111111";
when "1001" =>seg7 <= "1111011";
when others =>null;
end case;
end process;
end ;[/code]

```

[paradoxfx](#) 答:

latch

seg7 在锁存器里

[参与更多回答与讨论>>](#)

27、请教一个关于 sdrum 读写的的问题，急需答案！

[jenny4594](#) 问:

我现在编了一个对 sdrum 进行读写的程序，突发读和写都还没有什么问题，主要是全页模式的时候，读的时候从写的第二数据开始读，但是总的还是读的 256 个数据，我想了很久，但是百思不得其解，请各位大侠帮帮忙，不胜感激！

[songchao01](#) 答:

我记得 SDRAM 读写有一个 wrap 的动作，你可以再仔细看一下数据手册

全页猝发一次读出来一行数据，你从第二个数据开始读，读到该行的最后一个数据之后会 wrap 回来，继续读第一个数据，也就是说读出来的第 256 笔数据应该是该行的第一个数据

[louisesunjie](#) 答:

在 SDRAM 的全页模式中，读到最后一个数据的时候如果还没有停止命令，又会回到列 0 地址开始读，循环读操作，也就是上面说的 WRAP。

[查看更多回答与讨论>>](#)

28、请教一个 VHDL 问题

[chenghuitong](#) 问:

```
PROCESS(z, q)
BEGIN
p(z,x,q);
...
END PROCESS;
```

程序中 p(z,x,q)是一个过程语句，z 是输入，x,q 是输出，问题是进程表中为什么要列 q? 只列一个 z 就对了，如果列了 q，是不是 q 变化后，马上又启动进程了呢?

[wangshaolei](#) 答:

process 后括号里写入的是敏感信号，就是这几个信号一旦发生变化就会引起该进程重新计算，至于是不是会导致 p 的变化要看你写的逻辑，一般情况下要把该进程的所有输入都写进括号内，否则容易综合出锁存器(电平敏感)，编译器推荐将所有信号写进括号内

[参与更多回答与讨论>>](#)

[FPGA 经典 100 问 之<HDL 28 问>&<仿真 20 问>下载](#)

[点击下载>>45 篇博文精粹《菜鸟变高手，草根成长记》](#)

[点击下载>>特权同学倾力巨献---FPGA 学习资料下载汇总](#)