

分频计数实验

这个实验可以说是verilog入门最基础的实验了，我们不做太多的理论分析，实践是硬道理。蜂鸣器与CPLD的接口如图5.2所示，当CPLD的I/O口（FM）为低电平时，三极管截至，蜂鸣器不发声；当CPLD的I/O（FM）为高电平时，三极管导通，蜂鸣器发声。

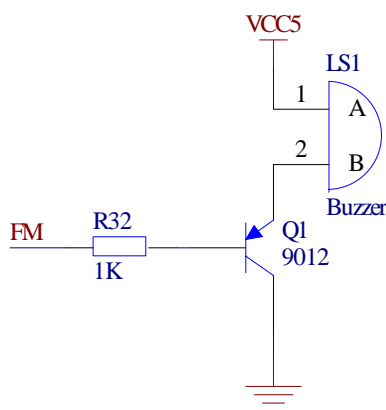


图5.2 蜂鸣器接口

在verilog代码设计中，我们把分频输出的信号clk_div与FM管脚对应，大家就可以真真切切的感受到什么是分频了。在代码里，用了20bit的计数器cnt，循环的计数，所以说一个周期有2的20次幂即大约1M分频。因为主时钟是50MHz（周期是20ns），所以大约20ms为一个计数周期。蜂鸣器就以大约20ms的周期发声，如果大家希望蜂鸣器的发声频率改变，那么可以改变cnt的值看看效果。

表5.3 分频计数实验接口定义

信号名称	方向	描述
clk	input	时钟信号，50MHz
rst_n	input	复位信号，低电平有效
clk_div	output	分频信号，连接到蜂鸣器

```

module clkdiv(
    clk,rst_n,
    clk_div
);

input clk;      //50MHz
input rst_n;    //低电平复位信号

-----

reg[19:0] cnt;  //分频计数器

always @ (posedge clk or negedge rst_n) //异步复位
    if(!rst_n) cnt <= 20'd0;
    e output clk_div;  //分频信号，连接到蜂鸣器

//lse cnt <= cnt+1'b1; //寄存器 cnt 大约 20ms 进行循环计数

//-----
reg clk_div_r;  //clk_div 信号值寄存器

always @ (posedge clk or negedge rst_n)
    if(!rst_n) clk_div_r <= 1'b0;
    else if(cnt == 20'hffff) clk_div_r <= ~clk_div_r;
                                //每 20ms 让 clk_div_r 值翻转一次

assign clk_div = clk_div_r;

endmodule

```

第一个always语句里实现了20位计数器cnt的循环计数。第二个always语句对cnt的计数值做判断，满足条件（每一个计数周期）时就会相应的对clk_div_r进行翻转。如此以来，就实现了clk_div的输出为每一个计数周期（大约20ms）变化一次，从而达到了控制蜂鸣器发声频率的目的。