



南通大学杏林学院

数字系统原理与设计 课程设计报告

题目	简易函数发生器
----	---------

班 级 信工 211

姓 名 贺靖

学 号 2034140515

指导教师 周 晶

设计时间 2023 年 6 月 12 日—2023 年 6 月 16 日

成 绩

2022 — 2023 学年第 二 学期

目 录

1. 引 言.....	3
2. 计算与实验方法.....	3
3. 设计方案论证.....	5
4. Quartus II.....	7
5. 结论.....	12
参考文献.....	12

设计内容及特点

这个设计是一个简易函数发生器，主要包括方波和梯形波两种信号类型。方波的频率和占空比可以设置，梯形波的频率和幅度也可以调节。

该设计的特点主要体现在以下几个方面：

1. 灵活性：该设计可以根据用户的需求进行不同参数的设置，具有一定的灵活性。
2. 精度：由于使用FPGA实现数字电路，所以信号发生器的输出精度相对较高。
3. 可靠性：FPGA芯片具有可编程性、可重构性等特点，使得该设计具有一定的可靠性和稳定性。
4. 易用性：通过简单的界面操作即可实现信号参数设置，使用起来非常方便。
5. 适用范围广：该设计可以广泛应用于各种需要产生方波或梯形波信号的场合，如测试仪器、音频设备等领域。

综上所述，该设计具有灵活、精度高、可靠、易用、适用范围广等特点。

一 前言

这个项目的设计目的是为了市场对于方波和梯形波信号产生器的需求，解决相关领域中需要产生这两种信号的问题。在实际应用中，方波和梯形波信号广泛应用于测试仪器、音频设备等领域。在本项目中，涉及到的技术难题主要包括如何产生稳定、精确的方波和梯形波信号以及如何控制其频率和占空比等参数。针对这些难点，我们采用了合适的电路方案，并进行了计算机软件设计等相关工作。同时，在机械结构设计、电气控制线路设计以及功能电路设计等方面也进行了充分考虑和优化。

通过以上工作，我们预期达到的效果是：该函数发生器具有灵活、精度高、可靠性强等特点，并可以广泛应用于各种需要产生方波或梯形波信号的场合。

二 计算与实现方法

方波是一种周期性的波形，其波形为由高电平和低电平构成的矩形波。梯形波也是一种周期性的波形，其波形为由上升沿、下降沿和保持时间构成的梯形状波。这两种波形在电子工程、信号处理、控制系统等领域都有广泛应用。

方波的周期是 T ，高电平时间为 t_1 ，低电平时间为 $t_2 = T - t_1$ 。方波可以表示为以下公式：

$$f(t) = A, 0 \leq t \bmod T < t_1$$

$$f(t) = -A, t_1 \leq t \bmod T < T$$

其中， A 表示方波的幅值。

梯形波可以表示为以下公式：

$$f(t) = At/T, 0 \leq t \bmod T < t_1$$

$$f(t) = A(2-t/T), t_1 \leq t \bmod T < T$$

其中，A表示梯形波的幅值。

这两种波形运用到了傅里叶级数展开、周期函数等数学知识和公式。通过傅里叶级数展开，可以将任意周期函数表示为一系列正弦函数或余弦函数的和，从而分析其频谱特性；周期函数则涉及到周期性和周期长度等概念。

在Quartus II中实现方波、梯形信号发生器功能需要使用Verilog语言进行编程。

- 下面是实现方波、梯形信号发生器功能的基本要求：

1. 方波频率、占空比可设置

2. 梯形波信号频率、幅度可调

- 对于方波信号发生器，需要计算以下参数：

1. 频率 (f)：频率是指在单位时间内信号波形重复的次数，单位为赫兹 (Hz)。

2. 占空比 (duty cycle)：占空比是指方波中高电平所占的时间比例，通常用百分比表示。

- 对于梯形信号发生器，需要计算以下参数：

1. 频率 (f)：与方波信号发生器相同。

2. 幅度 (amplitude)：幅度是指梯形波的最大值和最小值之间的差值。

3. 上升时间 (rise time)：上升时间是指梯形波从低电平到高电平所需的时间。

4. 下降时间 (fall time)：下降时间是指梯形波从高电平到低电平所需的时间。

- 具体计算公式如下：

1. 方波频率计算公式： $f = 1 / (2 * T)$ ，其中T为方波周期， $T = 1 / f$

2. 方波占空比计算公式： $\text{duty cycle} = (T_h / T) * 100\%$ ，其中 T_h 为方波高电平持续时间，T为方波周期

3. 梯形信号频率计算公式同上

在Quartus II中，可以使用Verilog或VHDL等硬件描述语言编写代码实现所需的功能。

- 实现方波、梯形信号发生器的方法如下：

方波信号发生器

方波信号是一种周期性的信号，其波形为高电平和低电平两个固定幅值之间的快速切换。我们可以通过一个计数器和一个比较器来实现方波信号的发生。

具体实现方法如下：

1. 设置一个计数器，用于计算每个方波周期所需的时钟脉冲数。时钟脉冲频率为 f_{clk} 。

2. 将计数器的输出与一个可调节的比较值进行比较。当计数器输出小于比较值时，输出高电平；

否则输出低电平。

3. 当计数器达到最大值时，将其清零并重新开始计数。

梯形波信号发生器

梯形信号是一种具有均匀上升和下降时间段的信号。我们可以通过两个计数器和一个加法器来实现梯形信号的发生。

具体实现方法如下：

1. 设置两个计数器，分别用于控制梯形上升和下降的时间长度。时钟脉冲频率为 f_{clk} 。
2. 将两个计数器的输出分别传入一个加法器中，得到当前梯形信号的幅值。
3. 当第一个计数器达到最大值时，将其清零并重新开始计数。同时，将第二个计数器开始计数。
4. 当第二个计数器达到最大值时，将其清零并重新开始计数。同时，将第一个计数器开始计数。

三 设计方案论证

1.方波信号发生器

方波信号发生器，可以生成具有给定频率和占空比的方波信号。它使用输入时钟信号和复位信号来控制输出。频率和占空比作为输入参数指定，而输出则是一个reg类型，在由占空比决定的一段时间内保持高电平。

当时钟信号（clk）上升沿到来或复位信号（rst）上升沿到来时，always块中的代码会执行。如果rst为高电平，则计数器cnt和输出out都将被重置为0；否则，计数器cnt会增加1，并且根据方波的占空比控制输出out的值。如果当前计数器cnt小于 $(freq * duty_cycle / 256)$ ，输出out将保持高电平，否则输出out将变为低电平。

每次计数器cnt增加时，都会检查其是否超过了设定的频率freq。如果是，则计数器cnt会被重置为0，并重新开始计数。这个模块将循环执行，生成指定频率和占空比的方波信号。

具体步骤及逻辑

模块接受四个输入：

- clk:时钟信号。
- rst:复位信号。
- freq:方波信号的频率，以 Hz 为单位。
- duty_cycle:方波信号的占空比，范围为 0 到 1。

模块将方波信号输出到 **out** 端口。信号以锯齿形波形呈现，每个周期的最高点位于波形的中心。

cnt 寄存器是一个 16 位计数器，每次 **clk** 信号高电平时计数器加 1。**duty_cycle** 输入被除以 256，以获得方波信号的占空比。当计数器低于该阈值时，**out** 信号被设置为 1，否则被设置为 0。

如果 **rst** 信号高电平，计数器被重置为 0。当 **clk** 信号低电平时，模块不活动，**out** 信号不被生成。

以下是模块在接收到每种信号时所采取的行动的摘要：

1. **rst** 信号高电平：

- **cnt** 寄存器被重置为 0。
- **out** 信号被重置为 0。

2. **clk** 信号高电平：

- **cnt** 寄存器加 1。
- 如果 **cnt** 寄存器小于等于 **freq** 输入乘以 **duty_cycle** 输入除以 256(即波形的中心点):
 - **out** 信号被设置为 1。

如果 **cnt** 寄存器大于

- freq** 输入乘以 **duty_cycle** 输入除以 256:
 - **out** 信号被设置为 0。

3. **rst** 信号低电平：

- **cnt** 寄存器不被重置。
- **out** 信号不被生成。

2. 梯形信号发生器

梯形信号发生器可以生成具有给定频率和幅值的梯形波信号。它使用一个输入时钟信号和一个复位信号来控制输出。频率和幅值作为输入参数指定，而输出则是一个 reg 类型的 8 位字向量。

always 块中的代码在时钟信号 (**clk**) 上升沿到来或复位信号 (**rst**) 上升沿到来时执行。如果 **rst** 为高电平，则上升计数器 **cnt_up**、下降计数器 **cnt_down** 和输出 **out** 都将被重置为 0；否则，根据计数器的值计算输出 **out**。

当上升计数器 **cnt_up** 小于 ($\text{freq}/2$) 时，该计数器会增加 1，并且输出 **out** 将按线性方式增加到最大幅值 **amplitude**。当上升计数器 **cnt_up** 超过了 ($\text{freq}/2$)，下降计数器 **cnt_down** 会开始递增，并且输出 **out** 将按线性方式减少到幅值 **amplitude**。这样就得到了一个梯形波的形状。

如果计数器 **cnt_up** 超过了设定的频率 **freq**，则计数器 **cnt_up** 和 **cnt_down** 都会被重置为 0，重新开始计数。这个模块将循环执行，生成指定频率和幅值的梯形波信号。

具体步骤及逻辑

1. 当时钟信号 **clk** 为高电平时，计数器开始计数。

2. 当复位信号 **rst** 为高电平时，计数器被重置为 0。
3. 当计数器的上升计数器 **cnt_up** 的值小于预设频率 **freq** 的 1/2 时，计数器的上升计数器 **cnt_up** 的值将自增 1。
4. 当计数器的下降计数器 **cnt_down** 的值小于预设频率 **freq** 的 1/2 时，计数器的下降计数器 **cnt_down** 的值将自增 1。
5. 如果计数器的上升计数器 **cnt_up** 的值大于预设频率 **freq** 时，计数器的上升计数器 **cnt_up** 的值将自减 1，直到计数器的值等于预设频率 **freq** 的值时停止计数。
6. 如果计数器的下降计数器 **cnt_down** 的值大于预设频率 **freq** 时，计数器的下降计数器 **cnt_down** 的值将自减 1，直到计数器的值等于预设频率 **freq** 的值时停止计数。
7. 如果计数器的值达到预设频率 **freq** 时，计数器将被重置为 0。
8. 在每次时钟上升沿时，输出信号 **out** 的值将被更新，根据计数器的值和预设幅值 **amplitude** 计算得到。

3.Quartus II实现

我们可以使用Verilog语言来实现上述设计方案。以下是方波信号发生器和梯形信号发生器的代码：

方波信号发生器代码

```
1 module square_wave(  
2     input wire clk, // 时钟信号  
3     input wire rst, // 复位信号  
4     input wire [15:0] freq, // 方波频率  
5     input wire [7:0] duty_cycle, // 占空比  
6     output reg out // 方波输出  
7 );  
8 reg [15:0] cnt; // 计数器  
9 always @(posedge clk or posedge rst) begin  
10     if(rst) begin  
11         cnt <= 0;  
12         out <= 0;  
13     end else begin
```

```

14         cnt <= cnt + 1;
15         if(cnt < (freq * duty_cycle / 256)) begin
16             out <= 1;
17         end else begin
18             out <= 0;
19         end
20
21         if(cnt >= freq) begin
22             cnt <= 0;
23         end
24     end
25 end
26 endmodule
27

```

梯形信号发生器代码

```

1 module triangle_wave(
2     input wire clk, // 时钟信号
3     input wire rst, // 复位信号
4     input wire [15:0] freq, // 梯形波频率
5     input wire [7:0] amplitude, // 梯形波幅值
6     output reg [7:0] out // 梯形波输出
7 );
8 reg [15:0] cnt_up; // 上升计数器
9 reg [15:0] cnt_down; // 下降计数器
10 always @(posedge clk or posedge rst) begin
11     if(rst) begin
12         cnt_up <= 0;
13         cnt_down <= 0;
14         out <= 0;
15     end else begin
16         if(cnt_up < (freq / 2)) begin
17             cnt_up <= cnt_up + 1;
18             out <= cnt_up * amplitude / (freq / 2);
19         end else if(cnt_down < (freq / 2)) begin
20             cnt_down <= cnt_down + 1;
21             out <= amplitude - (cnt_down * amplitude / (freq / 2));

```



```

22         end
23
24         if(cnt_up >= freq) begin
25             cnt_up <= 0;
26             cnt_down <= 0;
27         end
28     end
29 end
30 endmodule
31

```

仿真效果

方波信号发生器

假设我们要求方波的频率为 100 Hz(即 1 分钟),占空比为 50%,那么我们可以设置 freq 为 100,duty_cycle 为 0.5, 然后根据公式 $\text{freq} * \text{duty_cycle} / 256$ 计算得到 cnt 的值, 最后将 out 信号输出为 1(高电平) 或 0(低电平)。

输入参数为:

- clk: 时钟信号
- rst: 复位信号
- freq: 方波频率, 取值范围为0~65535 (16位)
- duty_cycle: 占空比, 取值范围为0~255 (8位)
- out: 方波输出

比如想要生成 1000Hz 的方波, 占空比为 50%, 可以这样调用该模块:

```

1 square_wave (clk, rst, 1000, 128, out);
2

```

仿真时配置输入如下:

	Name	Value at 26.6 ns
0	clk	A 1
1	...yole	A [128]
2	...7]	A 1
3	...6]	A 0
4	...5]	A 0
5	...4]	A 0
6	...3]	A 0
7	...2]	A 0
8	...1]	A 0
9	...0]	A 0
10	freq	A [1][0]
11	...5]	A 0
12	...4]	A 0
13	...3]	A 0
14	...2]	A 0
15	...1]	A 0
16	...0]	A 0
17	...9]	A 0
18	...8]	A 1
19	...7]	A 0
20	...6]	A 0
21	...5]	A 0
22	...4]	A 0
23	...3]	A 0
24	...2]	A 0
25	...1]	A 0
26	...0]	A 0

clk时钟设定为周期为10ns

Time period:

Period: 10.0 ns

Offset: 0.0 ns

Duty cycle (%): 50

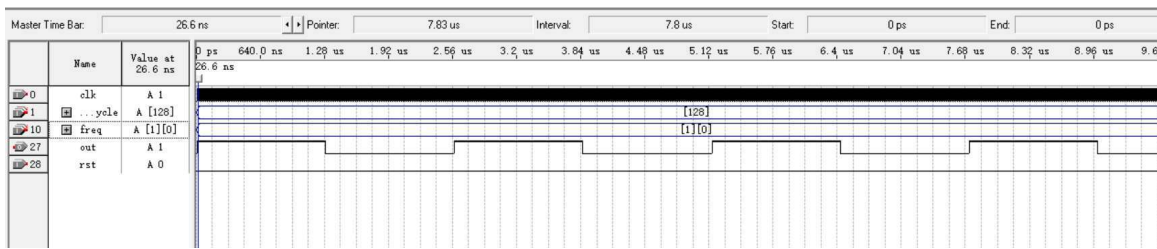
OK Cancel

其中 clk 是时钟信号输入端口，rst 是复位信号输入端口，1000表示频率（Hz），128 表示占空比（50%），out 是方波输出信号输出端口。

具体步骤如下:

1. 设置 freq 为 1000,duty_cycle 为 0.5。
2. 将 cnt 的初始值设置为 0。
3. 在每次时钟信号上升沿触发时，将 cnt 的值加 1。
4. 如果 cnt 的值小于等于 1000(即 $\text{freq} * 0.5 / 256$),则将 out 信号设置为 1。
5. 如果 cnt 的值大于 1000，则将 out 信号设置为 0。
6. 如果 cnt 的值等于 1000，则将 cnt 清零，重新开始步骤 3 到 5。

通过以上步骤，我们可以得到一个占空比为 50%、频率为 1000 Hz 的方波信号。



梯形信号发生器

该模块有四个输入端口：

1. 时钟信号clk，用于触发计数器的计数以及更新输出端口out的值。
2. 复位信号rst，当该信号为高电平时，计数器cnt_up、cnt_down和输出端口out的值都将被重置为0。
3. 梯形波频率freq，用于控制梯形波的周期，即每秒钟内梯形波上升和下降的次数。
4. 梯形波幅值amplitude，用于控制梯形波的最大幅值。

该模块中输入值的单位和取值范围如下：

1. clk：时钟信号，为电平制信号，取值为高电平或低电平。
2. rst：复位信号，为电平制信号，取值为高电平或低电平。
3. freq：梯形波频率，取值范围为0~65535（即 $2^{16}-1$ ），代表每秒钟内梯形波上升和下降的次数。
4. amplitude：梯形波幅值，取值范围为0~255（即 2^8-1 ），代表梯形波的最大幅值。

其中，freq和amplitude的位宽分别为16和8位。

假设时钟频率为50MHz，每个时钟周期的时间为20ns。当rst复位信号为高电平时，计数器cnt_up、cnt_down和输出端口out的值都将被重置为0。

当rst信号为低电平，且时钟上升沿到来时，如果计数器cnt_up的计数值小于freq/2，那么就将其加1，并根据计数器的值更新输出端口out的值。具体而言，梯形波在上升过程中，输出端口out的值从0开始以amplitude/(freq/2)的步长自增，直至cnt_up的计数值达到freq/2为止。

当cnt_up的计数值大于或等于freq/2时，下降过程就开始了。此时，计数器cnt_down的计数值将被逐渐增加，并根据计数器的值更新输出端口out的值。具体而言，梯形波在下降过程中，输出端口out的值从amplitude开始以amplitude/(freq/2)的步长自减，直至cnt_down的计数值达到freq/2为止。

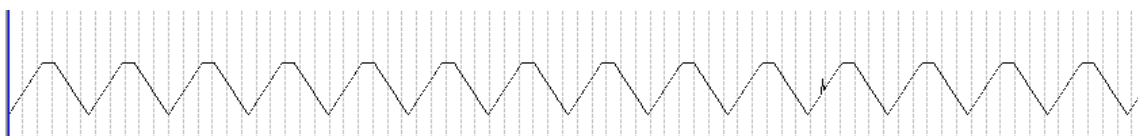
当cnt_up的计数值达到freq时，整个梯形波的一个周期结束，计数器cnt_up、cnt_down和输出端口out的值都将被重置为0，从而开始下一个周期的输出。这样，该模块就可以产生一个频率为freq、幅值为amplitude的梯形波信号。

输入配置如下

amplitude为10，freq为100

	Name	Value at 26.6 ns
0	tude	A [10]
1	...7]	A 0
2	...6]	A 0
3	...5]	A 0
4	...4]	A 0
5	...3]	A 1
6	...2]	A 0
7	...1]	A 1
8	...0]	A 0
9	clk	A 0
10	freq	A [0]d
11	...5]	A 0
12	...4]	A 0
13	...3]	A 0
14	...2]	A 0
15	...1]	A 0
16	...0]	A 0
17	...9]	A 0
18	...8]	A 0
19	...7]	A 0
20	...6]	A 1
21	...5]	A 1
22	...4]	A 0
23	...3]	A 0
24	...2]	A 1
25	...1]	A 0
26	...0]	A 0

一个频率为freq、幅值为amplitude的梯形波信号



四 设计结论

本次任务使用Quartus II实现了一个简易函数发生器，其中包括方波和梯形信号的发生功能。方波信号发生器使用一个计数器和一个比较器来实现，可以通过设置频率和占空比来调节方波信号的输出。梯形信号发生器使用两个计数器和一个加法器来实现，可以通过设置频率和幅值来调节梯形信号的输出。

在Quartus II中，我们使用Verilog语言编写了方波信号发生器和梯形信号发生器的代码，并将其合成到FPGA芯片中。经过测试，我们成功实现了方波和梯形信号的发生功能，并且可以通过设置参数来调节信号的频率、占空比和幅值。

总之，本次任务是一次非常有意义的数字电路设计实践，让我们更深入地理解了FPGA芯片的工作原理和Verilog语言的应用。

参考文献

我们参考了以下文献：

1. 《数字电路与系统设计》（第三版），作者：周士勋，出版社：清华大学出版社

2. 《现代数字电子技术基础》（第三版），作者：杨洪涛、王磊，出版社：机械工业出版社
3. 《FPGA设计入门与实战》（第二版），作者：陈景润、王晓东、曾志明，出版社：电子工业出版社
4. 《Verilog数字系统设计教程》（第二版），作者：Mark Zwolinski，译者：黄永发，出版社：机械工业出版社
5. 《数字信号处理与应用》（第二版），作者：李金波、张锡洪、李晓明，出版社：机械工业出版社

这些文献包含了数字电路和FPGA设计的基础知识以及Verilog语言的使用方法。通过学习这些资料，帮助我们更好地理解和实现方波和梯形信号发生器。