



宁波大学
NINGBO UNIVERSITY

2021-2022 学年第 2 学期

数字系统设计工程实践

(课号: 103D47B01)

实 验 报 告

实验名称: DDS 波形发生器

学 院 信息科学与工程学院

班 级 阳明 2 班

姓 名 熊 康

学 号 206001232

指导教师 张 跃 军

完成时间 2022.03.24

目 录

一、实验任务与要求	3
1.1 实验目的.....	3
1.2 实验设备与元器件.....	3
1.3 实验内容.....	3
二、系统原理与方案设计	3
2.1 DDS 原理	3
2.1.1 工作过程	4
2.1.2 改变输出信号的频率.....	4
2.1.3 相位累加器.....	4
2.1.4 正弦查询表.....	4
2.1.5 DDS 参数计算	4
2.1.6 DDS 的特点.....	5
2.2 方案设计.....	5
2.2.1 方案设计结构.....	5
2.2.2 正弦查询表	6
2.2.3 计算器.....	6
2.2.4 信号频率控制器	7
2.2.5 输出控制器	7
2.2.6 方案参数的确定	7
2.3 方案仿真.....	8
三、系统实现与实验结果分析	8
四、结论.....	9
五、附录.....	10

一、实验任务与要求

1.1 实验目的

学习利用 EDA 技术和 FPGA 技术实现 DDS 波形发生器的设计。

1.2 实验设备与元器件

- 1) DSE-V 数字电路实验平台 1 台；
- 2) 计算机 1 台；
- 3) Quartus II 软件 1 套。

1.3 实验内容

利用 FPGA+DAC，设计一个 DDS 信号发生器。

- 1) 分辨率优于 10Hz；
- 2) ROM 表长度 10 位、位宽 8 位；
- 3) 时钟频率：10MHz；
- 4) 输出最高频率时，每个输出周期不少于 20 个数据点；
- 5) 显示信号频率/频率控制字（可切换,十六进制显示）；
- 6) 直接输入频率控制字设置频率；
- 7) 用最少的 N 和 M 实现。

二、系统原理与方案设计

2.1 DDS 原理

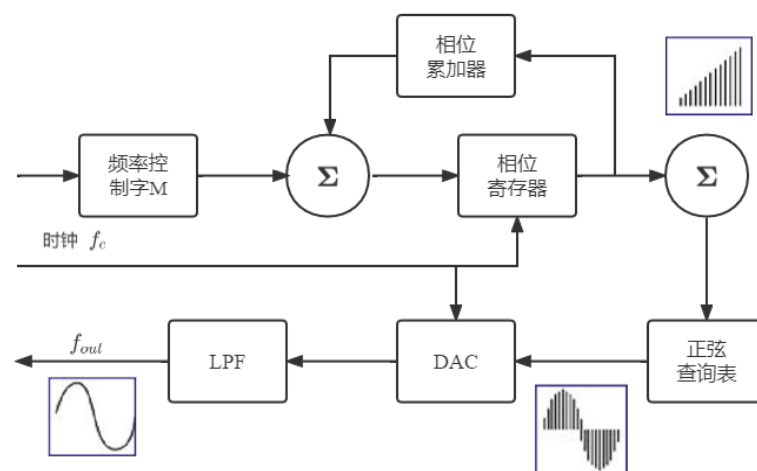


图 1 DDS 基本结构框图

2.1.1 工作过程

将存于数表中的数字波形,经 D/A 转换器和滤波, 形成模拟量波形。

2.1.2 改变输出信号的频率

- 1) 改变查表寻址的时钟频率, 改变输出波形的频率;
- 2) 改变寻址的步长, 改变输出信号的频率—DDS;
- 3) 步长即为对数字波形查表的相位增量.由累加器对相位增量进行累加, 累加器的值作为查表地址;
- 4) 设相位累加器的位宽为 N, Sin 表的大小为 2^P , 累加器的高 P 位用于寻址 Sin 表;
- 5) 时钟频率为 f_c , 若累加器按步进 M 累加直至溢出, 称 M 为频率控制字。

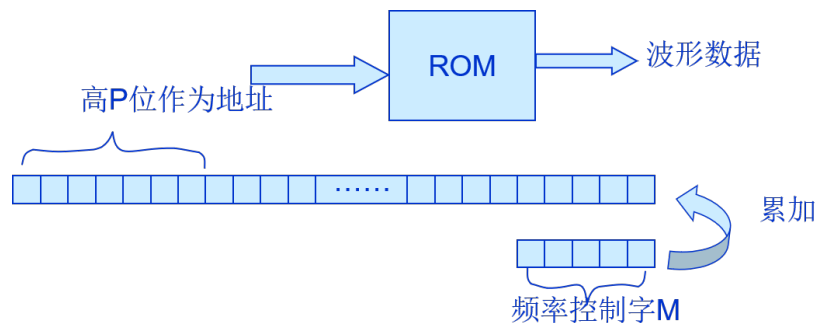


图 2 信号频率控制原理图

2.1.3 相位累加器

DDS 系统的核心是相位累加器, 它由一个加法器和一个相位寄存器组成; 每来一个时钟, 相位寄存器以步长增加, 相位寄存器的输出与频率控制字(M)相加, 然后输入到正弦查询表地址上。

2.1.4 正弦查询表

正弦查询表包含一个周期正弦波的数字幅度信息, 每个地址对应正弦波中 $0\sim360^\circ$ 范围的一个相位点。查询表把输入的地址相位信息映射成正弦波幅度的数字量信号, 驱动 DAC, 输出模拟量。

公式:

$$x = \sin(a \times \pi/180) \times 1024 + 128 \quad (2.1)$$

2.1.5 DDS 参数计算

相位寄存器每经过 $2^N/M$ 个 f_c 时钟后回到初始状态, 相应地正弦查询表经过一个循环回到初始位置, 整个 DDS 系统输出一个正弦波。

- 1) 输出正弦波频率:

$$f_{out} = M \frac{f_c}{2^N} \quad (2.2)$$

2) M 与输出 f_{out} 和 f_c 之间的关系:

$$M = (f_{out} \times 2^N) / f_c \quad (0 < M < 2^N - 1) \quad (2.3)$$

3) **DDS 的最小分辨率**通常用频率增量来表示频率合成器的分辨率。

$$\Delta f_{min} = \frac{f_c}{2^N} \quad (M = 1) \quad (2.4)$$

这个增量也就是最低的合成频率。

4) **DDS 的最大合成频率**最高的合成频率受奈奎斯特抽样定理的限制。

$$f_{max} = \frac{f_c}{2} \quad (2.5)$$

在实际运用中,为了保证信号的输出质量,输出频率不要高于时钟频率的 33%,以避免混叠或谐波落入有用输出频带内。

2.1.6 DDS 的特点

存在以下优点:

- 1) 超宽的相对宽带;
- 2) 超高的捷变速率(可实现跳频);
- 3) 超细的分辨率;
- 4) 相位的连续性;
- 5) 输出波形灵活;
- 6) 可编程全数字化。

但存在杂散大的缺点。相位累加器相位舍位误差造成的杂散;幅度量化误差(由存储器有限字长引起)造成的杂散和 DAC 非理想特性造成的杂散。

频率上限目前还只能达到数百兆(主要是受 DAC 速度的限制)。

2.2 方案设计

2.2.1 方案设计结构

主体根据图 1 进行具体设计,此外为实现 DAC,将 DAC 基准时钟与系统基准时钟信号 clk 相连。

clk	系统基准时钟信号输入
rst	系统重置信号输入
clkm	频率控制字脉冲信号输入
car	频率控制字位运算控制
control	数码管显示控制
out	数码管输出
q	数字波形输出

表 1 输入输出量说明

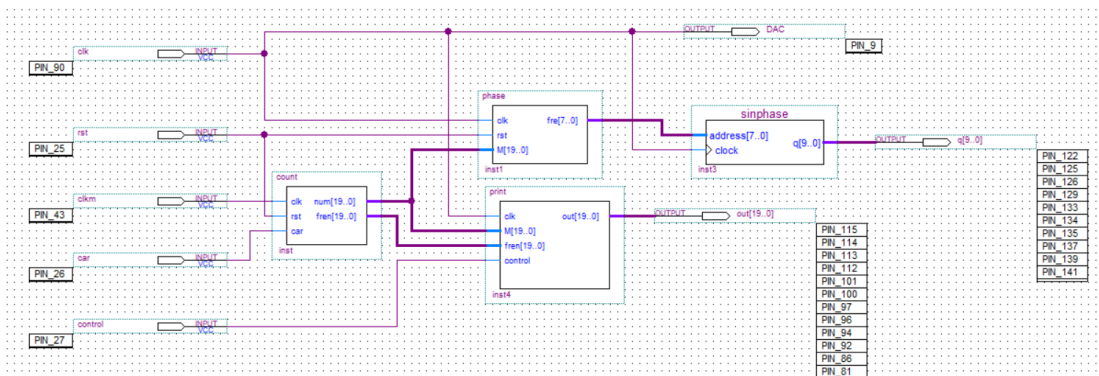


图3 方案设计结构图

2.2.2 正弦查询表

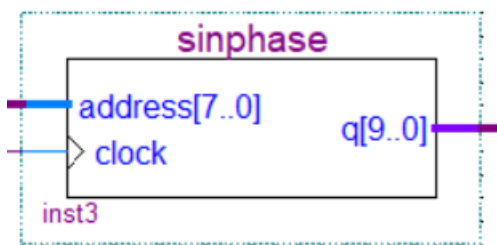


图4 功能模块正弦查询表

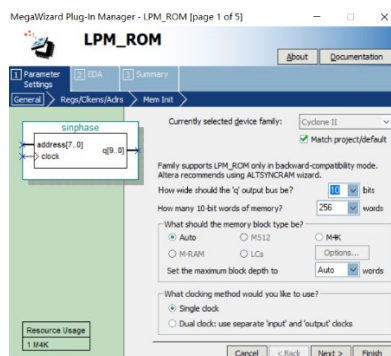


图5 .mif 文件属性

Addr	+0000	+0001	+0010	+0011	+0100	+0101	+0110	+0111	+1000	+1001	+1010	+1011	+1100	+1101	+1110	+1111
00000000	524	537	549	562	574	587	599	611	624	636	648	660	672	684	696	707
00010000	719	730	742	753	764	775	785	796	806	816	826	836	846	855	865	874
00100000	882	891	899	907	915	923	930	937	944	951	957	963	969	974	980	985
00110000	989	994	998	1001	1005	1008	1011	1014	1016	1018	1020	1021	1022	1023	1023	1023
01000000	1023	1023	1022	1021	1020	1018	1016	1014	1011	1008	1005	1001	998	994	989	985
01010000	980	974	969	963	957	951	944	937	930	923	915	907	899	891	882	874
01100000	865	855	846	836	825	816	806	796	785	775	764	753	742	730	719	707
01110000	696	684	672	660	648	636	624	611	599	587	574	562	549	537	524	512
10000000	499	486	474	461	449	436	424	412	399	387	375	363	351	339	327	316
10010000	304	293	281	270	259	248	238	227	217	207	197	187	177	168	158	149
10100000	141	132	124	116	108	100	93	86	79	72	66	60	54	49	43	38
10110000	34	29	25	22	18	15	12	9	7	5	3	2	1	0	0	0
11000000	0	0	1	2	3	5	7	9	12	15	18	22	25	29	34	38
11010000	43	49	54	60	66	72	79	86	93	100	108	116	124	132	141	149
11100000	158	168	177	187	197	207	217	227	238	248	259	270	281	293	304	316
11110000	327	339	351	363	375	387	399	412	424	436	449	461	474	486	499	512

图6 正弦查询表.mif 文件

通过公式 (2.1)，运用 excel 表格操作得到正弦数据后，新建.mif 文件，导入数据，得到正弦查询表。

2.2.3 计算器

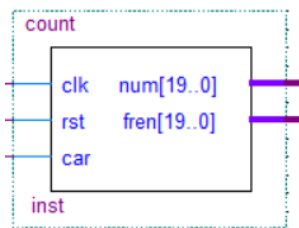


图7 功能模块计算器

通过寄存器和公式 (2.2) 运算实现计算频率控制字和输出频率。clk 频率控制字脉冲输入，rst 模块重置，car 频率控制字进位控制，num 显示频率控制字，fren 显示输出频率。

2.2.4 信号频率控制器

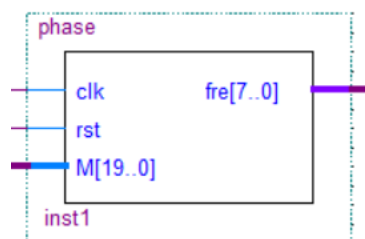


图 8 功能模块信号频率控制器

根据原理 2.1.2 设计得到信号频率控制器。通过输入基准时钟频率 clk 和频率控制字 M ，得到当前输出频率 fre 。 rst 实现模块重置。

2.2.5 输出控制器

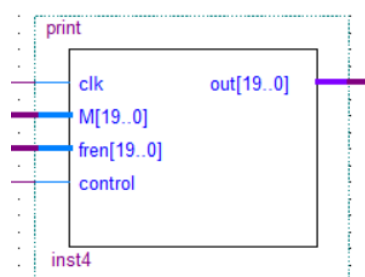


图 9 功能模块输出控制器

通过 $control$ 输出控制端，来实现对频率控制字 M 和输出频率 $fren$ 的选择性输出。 rst 实现模块重置。 clk 为基准时钟信号。

2.2.6 方案参数的确定

一、相位累加器的位宽 N ：

在本次设计中，通过改变频率控制字 M 来实现改变输出频率。所以相位累加器的位宽 N 在设计时就被固定了。为了实现最小 N 和 M ，也为了实现分辨率优于 10Hz ，则根据式(2.2)得当 $M = 1$ 时，使 $f_{out} < 10$ ，即 $\frac{10^7}{2^N} < 10$ ，所以 $N \geq 20$ 。就是说 N 最小可取 20。

二、输出频率的显示：

以 Hz 为单位，最低频率时 $M=1$ 约 9.5Hz ，用一个数码管；最高频率高 8 位采样周期中每次采样应至少经过 $\frac{2^8}{20} = 12.8$ 个周期，也就是说， M 应为 52428 时，最高输出频率为 500KHz ，需要使用 5 个数码管，则在输出低频时不能显示小数位。

那么如果考虑在输出低频时，多用 1 个数码管显示小数位，就需要有一个进位标志。但在 DSE-EP205 目标板上，具有 DAC 模块的模式三和模式五只有 5 个数码管可供显示输出频率。而且 LED 灯不受系统控制，所以进位标志无法实现，即只能显示整数位。

2.3 方案仿真

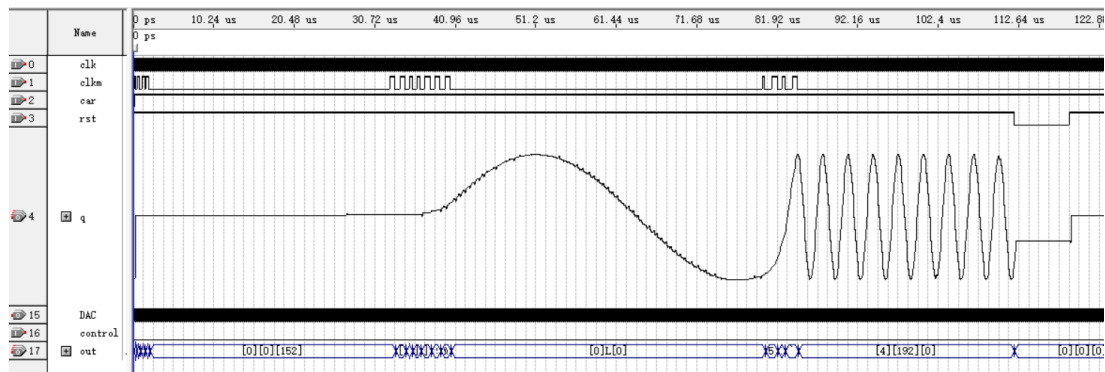


图 10 方案仿真图

分析：由图可知，通过输入基准时钟信号 clk，q 输出正弦波。当 clkm 输入脉冲后，频率控制字 M 增加，输出频率增加，输出波形变密。当 rst 为 0 时，系统重置，符合要求。

三、系统实现与实验结果分析



图 11



图 12

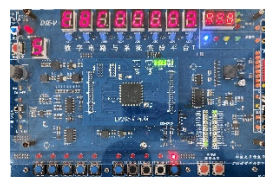


图 13

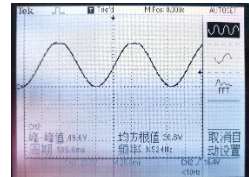


图 14

一、测试输出频率显示：

1. 系统刚开始如图 11 所示，目标板选择模式 5，数码管后 5 位（从右往左数 5 个）显示为 0，按钮输入皆为 0。
2. 图 12 所示，打开系统（最右侧按钮为 1，rst=1），频率控制字加 1（左侧第 3 个按钮按一次），数码管显示控制选择显示频率控制字（右侧第 3 个按钮为 1，control=1），当前显示频率控制字 M=1。
3. 图 13 所示，数码管显示控制选择显示输出频率（control=0），当前显示输出频率为 9Hz。
4. 图 14 所示，实际输出频率为 9.5Hz，在“方案参数的确定”中讨论了小数位的实现，所以只考虑显示整数位，符合要求。

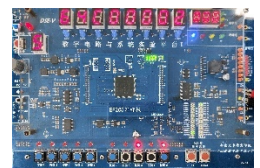


图 15



图 16

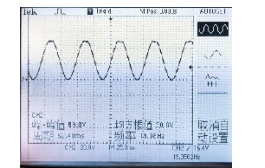


图 17



图 18

5. 图 15 所示，当前显示频率控制字 M=2。

6. 图 16 所示，当前显示输出频率为 19Hz（数码管 16 进制，“13”表示 $16+3=19$ ）。
7. 图 17 所示，实际输出频率为 19.1Hz，相对误差为 0.5%，符合要求。



图 19

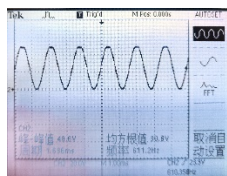


图 20

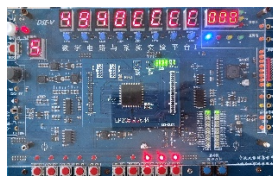


图 21

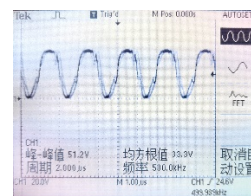


图 22

8. 图 18 所示，当前显示频率控制字 $M=64$ 。
9. 图 19 所示，当前显示输出频率为 610Hz（十六进制 262）。
10. 图 20 所示，实际输出频率为 611.2Hz，相对误差为 0.2%。通过这三组数据可以发现，输出频率越大，数码管显示的频率误差越小。

二、分析系统分辨率：由图 14 可知，当 $N=1$ ， $M=1$ 时，系统最小输出频率为 9.5Hz，即分辨率为 9.5Hz，优于 19Hz。此时 N 为 20， M 为 1。

三、分析输出最高频率时，每个输出周期不少于 20 个数据点：即高 8 位采样周期中每次采样应至少经过 $\frac{2^8}{20} = 12.8$ 个周期，也就是说， M 应为 52428（二进制 1100_1100_1100_1100）（十六进制 CCCC），而此时的最高频率为 $499992.37\text{Hz} \approx 500\text{KHz}$ ，频率字调整如图 21 所示，输出图像如图 22 所示。此时 N 为 20， M 为 52428。

四、结论

通过本次实验熟练掌握 EDA 技术和 FPGA 技术，并能实现 DDS 波形发生器的设计。

可以通过频率控制字改变输出频率，实现分频的目的。再通过查询正弦查询表，能够实现变化信号的不同频率地周期性输出。然后经过 DAC 处理后，从数字量转变成模拟信号。再通过增加计算和输出显示模块，还能增加各种显示输出功能。但唯一遗憾地是，在硬件限制的情况下，不能够实现小数位的显示。

五、附录

```
1.  /*****
2.  Author: 熊康
3.  E-mail: 1390000666@qq.com
4.  Device: DSE-EP2C5
5.  Tool:   Quartus 9.0
6.  Function: (计算器) 计算频率控制字和输出频率
7.  Version: 2022-3-17 v1.0
8.  *****/
9.  module count(clk,rst,num,car,fren);
10. input clk,rst,car;
11. output reg [19:0] num;
12. output reg [19:0] fren;
13.
14. always @(posedge clk or negedge rst) begin
15.     //用于控制频率控制字
16.     if(!rst)
17.         num <= 0;
18.     else
19.         if(!car)
20.             num <= num + 1; //M+1
21.         else
22.             num <= {num[18:0],1'b0}; //M*2
23. end
24.
25. always @(negedge clk or negedge rst) begin
26.     if(!rst)
27.         fren <= 0;
28.     else
29.         //计算输出频率
30.         fren[19:0] <= (num * (1000000 >> 18) >>2);
31. end
32.
33. endmodule
```

```
1.  /*****
2.  Author: 熊康
3.  E-mail: 1390000666@qq.com
4.  Device: DSE-EP2C5
5.  Tool:   Quartus 9.0
6.  Function:相位寄存器
7.  Version: 2022-3-17 v1.0
8.  *****/
```

```

9. module phase(clk,rst,M,fre);
10. input clk,rst;
11. input [19:0] M; //M: 频率控制字
12. output reg [7:0] fre; //输出的相位
13. reg [19:0] N; //相位寄存
14.
15. always @(posedge clk or negedge rst) begin
16.     if(!rst)
17.         N <= 0;
18.     else begin
19.         N <= N + M;
20.         fre[7:0] <= N[19:12];
21.     end
22.
23. end
24.
25. endmodule

```

```

1. /*****
2. Author: 熊康
3. E-mail: 1390000666@qq.com
4. Device: DSE-EP2C5
5. Tool: Quartus 9.0
6. Function:输出控制器(用于控制输出频率控制字和输出频率)
7. Version: 2022-3-24 v1.0
8. *****/
9. module print(clk,M,fren,control,out);
10. input [19:0] M;
11. input [19:0] fren;
12. input clk,control;
13. output reg [19:0] out;
14.
15. always @(posedge clk) begin
16.     if(control==1)
17.         out[19:0] <= M[19:0];
18.     else
19.         out[19:0] <= fren[19:0];
20. end
21.
22. endmodule

```

```

1. /*****
2. Author: 熊康
3. E-mail: 1390000666@qq.com

```

```

4. Device: DSE-EP2C5
5. Tool: Quartus 9.0
6. Function:正弦查询表
7. Version: 2022-3-24 v1.0
8. *****/
9.
10. `timescale 1 ps / 1 ps
11.
12. module sinphase (
13.     address,
14.     clock,
15.     q);
16.
17.     input [7:0] address;
18.     input clock;
19.     output [9:0] q;
20. `ifndef ALTERA_RESERVED_QIS
21.
22. `endif
23.     tri1 clock;
24. `ifndef ALTERA_RESERVED_QIS
25.
26. `endif
27.
28.     wire [9:0] sub_wire0;
29.     wire [9:0] q = sub_wire0[9:0];
30.
31.     altsyncram altsyncram_component (
32.         .clock0 (clock),
33.         .address_a (address),
34.         .q_a (sub_wire0),
35.         .aclr0 (1'b0),
36.         .aclr1 (1'b0),
37.         .address_b (1'b1),
38.         .addressstall_a (1'b0),
39.         .addressstall_b (1'b0),
40.         .byteena_a (1'b1),
41.         .byteena_b (1'b1),
42.         .clock1 (1'b1),
43.         .clocken0 (1'b1),
44.         .clocken1 (1'b1),
45.         .clocken2 (1'b1),
46.         .clocken3 (1'b1),
47.         .data_a ({10{1'b1}}),

```

```
48.         .data_b (1'b1),
49.         .eccstatus (),
50.         .q_b (),
51.         .rden_a (1'b1),
52.         .rden_b (1'b1),
53.         .wren_a (1'b0),
54.         .wren_b (1'b0));
55.     defparam
56.         altsyncram_component.clock_enable_input_a = "BYPASS",
57.         altsyncram_component.clock_enable_output_a = "BYPASS",
58.         altsyncram_component.init_file = "demo.mif",
59.         altsyncram_component.intended_device_family = "Cyclone II",
60.         altsyncram_component.lpm_hint = "ENABLE_RUNTIME_MOD=NO",
61.         altsyncram_component.lpm_type = "altsyncram",
62.         altsyncram_component.numwords_a = 256,
63.         altsyncram_component.operation_mode = "ROM",
64.         altsyncram_component.outdata_aclr_a = "NONE",
65.         altsyncram_component.outdata_reg_a = "CLOCK0",
66.         altsyncram_component.widthad_a = 8,
67.         altsyncram_component.width_a = 10,
68.         altsyncram_component.width_byteena_a = 1;
69.
70.
71. endmodule
```