

电子技术基础实验第十三周实验报告

王磊 2022012972

2023 年 12 月 20 日

1 task1

本次任务与上周任务相比，唯一不同是换用了ROM中的数据，图1展示了新的正弦数据生成代码。该代码的作用是生成了三次谐波数据。

```
% 设置参数
width = 24; % 每个数据元素的位宽为24位
depth = 256; % 存储器的深度，即数据元素的数量为256个
%phase = 0;……% 相位初始为0，这里被注释掉
phase = 0; % 相位设置为0，即90度

% 打开文件并写入文件头信息
fid = fopen('sin_phase0_24bit_harmony.mif', 'w');
fprintf(fid, 'WIDTH=%d;\n', width);
fprintf(fid, 'DEPTH=%d;\n', depth);
fprintf(fid, 'ADDRESS_RADIX=DEC;\n');
fprintf(fid, 'DATA_RADIX=HEX;\n');
fprintf(fid, 'CONTENT BEGIN\n');

% 循环生成24位正弦波数据并写入文件
for i = 0:depth - 1
    sin_data_base = floor((sin(2 * pi * i / depth + phase)) * 0.5 * (2 ^ (width - 1) - 1));
    sin_data_third = floor((sin(2 * pi * 3 * i / depth + phase)) * 0.3 * (2 ^ (width - 1) - 1));
    sin_data = sin_data_base + sin_data_third;
    if (sin_data < 0)
        sin_data = 2 ^ width + sin_data;
    end
    fprintf(fid, '%d:%x;\n', i, sin_data);
end

% 写入文件尾部信息并关闭文件
fprintf(fid, 'END ;\n');
fclose(fid);
```

图 1: matlab代码

最终示波器上显示的波形如图2所示。



图 2: 50Hz正弦波与三次谐波波形

2 task2

本任务所涉及代码均已在网络学堂给出或在其他任务完成，因此只需给出顶层文件如下：

```

1  'include "rom_data_tri.v"
2  'include "rom_base.v"
3  'include "rom_harmony.v"
4  'include "dac_controller_new_1213.v"
5  'include "mypll2.v"
6  'include "adc_controller_new_1213.v"

7
8 module task2_top(
9     input clk,
10    input rst,
11    input sdata_adc,
12    output lrck_dac,
13    output sclk_dac,
14    output mclk_dac,
15    output sdata_dac,
16    output mclk_adc,
```

```
17     output sclk_adc,
18     output lrck_adc,
19     output reg buzz,
20     output [23:0] dataL_adc,
21     output [23:0] dataR_adc
22 );
23
24 initial begin
25     buzz = 1'b1;
26 end
27
28 wire[23:0] data_dac_chL;
29 wire[23:0] data_dac_chR;
30 wire[7:0] addr_chL;
31 wire[7:0] addr_chR;
32 wire c0;//6553600hz
33 wire c1;//1.28Mhz
34
35 rom_data_tri uut1(
36     .lr_ch_tri_clk(lrck_dac),
37     .rst(rst),
38     .addr_chL(addr_chL),
39     .addr_chR(addr_chR)
40 );
41
42 rom_base uut2(
43     .clock(clk),
44     .address(addr_chL),
45     .q(data_dac_chL)
46 );
47
48 rom_harmony uut3(
49     .clock(clk),
50     .address(addr_chR),
51     .q(data_dac_chR)
52 );
53
54 dac_controller_new_1213 uut4(
55     .clk(c0),
56     .rst(rst),
```

```
57     .data_dac_chL(data_dac_chL),
58     .data_dac_chR(data_dac_chR),
59     .mclk_dac(mclk_dac),
60     .sclk_dac(sclk_dac),
61     .lrck_dac(lrck_dac),
62     .sdata_dac(sdata_dac)
63 );
64
65 adc_controller_new_1213 uut6(
66     .clk(c1),
67     .rst(rst),
68     .sdout_adc(sdout_adc),
69     .mclk_adc(mclk_adc),
70     .sclk_adc(sclk_adc),
71     .lrck_adc(lrck_adc),
72     .dataL_adc(dataL_adc),
73     .dataR_adc(dataR_adc)
74 );
75
76 mypll2 uut5(
77     .inclk0(clk),
78     .c0(c0),
79     .c1(c1)
80 );
81
82 endmodule
```

对应的RTL电路图如图3所示。

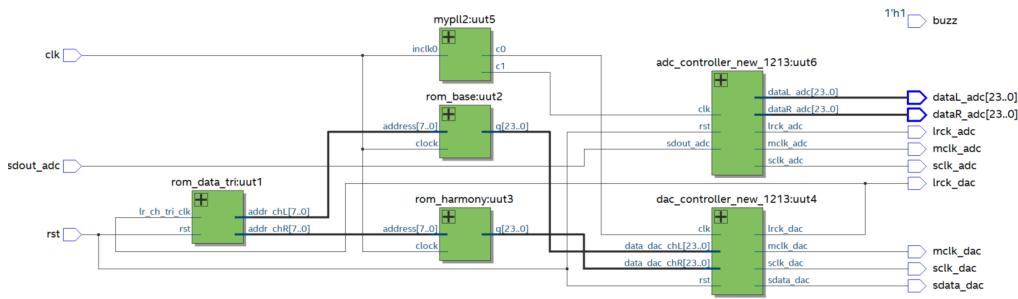


图 3: task2RTL电路图

通过signalTap观察到的波形如图4所示。

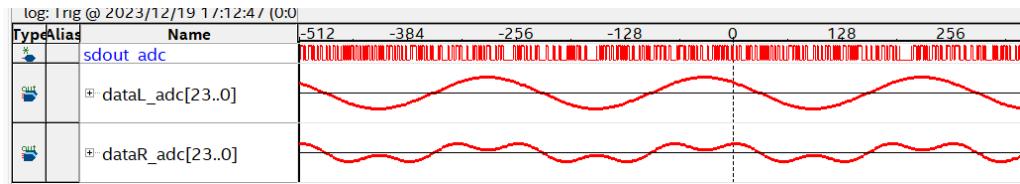


图 4: signalTap波形

3 task3

本任务所涉及代码均已在网络学堂给出或在其他任务完成，因此只需给出顶层文件如下：

```

1  'include "rom_data_tri.v"
2  'include "rom_base.v"
3  'include "rom_harmony.v"
4  'include "dac_controller_new_1213.v"
5  'include "mypll2.v"
6  'include "adc_controller_new_1213.v"
7  'include "adc_data_ready_tri.v"
8  'include "uart_NbyteTran_3byteData_controller.v"
9  'include "uart_tx_byte.v"

10
11 module task3_top(
12     input clk,
13     input rst,
14     input sdout_adc,
```

```
15     output lrck_dac,
16     output sclk_dac,
17     output mclk_dac,
18     output sdata_dac,
19     output mclk_adc,
20     output sclk_adc,
21     output lrck_adc,
22     output reg buzz,
23     output [23:0] dataL_adc,
24     output [23:0] dataR_adc,
25     output send_en,
26     output sci_tx
27 );
28
29 initial begin
30   buzz = 1'b1;
31 end
32
33 wire[23:0] data_dac_chL;
34 wire[23:0] data_dac_chR;
35 wire[7:0] addr_chL;
36 wire[7:0] addr_chR;
37 wire c0;//6553600hz
38 wire c1;//1.28Mhz
39
40 wire tx_done;
41 wire tx_en;
42 wire [23:0] data;
43 wire [7:0] tx_d;
44 wire send_done;
45
46 rom_data_tri uut1(
47   .lr_ch_tri_clk(lrck_dac),
48   .rst(rst),
49   .addr_chL(addr_chL),
50   .addr_chR(addr_chR)
51 );
52
53 rom_base uut2(
54   .clock(clk),
```

```
55     .address(addr_chL),
56     .q(data_dac_chL)
57 );
58
59 rom_harmony uut3(
60     .clock(clk),
61     .address(addr_chR),
62     .q(data_dac_chR)
63 );
64
65 dac_controller_new_1213 uut4(
66     .clk(c0),
67     .rst(rst),
68     .data_dac_chL(data_dac_chL),
69     .data_dac_chR(data_dac_chR),
70     .mclk_dac(mclk_dac),
71     .sclk_dac(sclk_dac),
72     .lrck_dac(lrck_dac),
73     .sdata_dac(sdata_dac)
74 );
75
76 adc_controller_new_1213 uut6(
77     .clk(c1),
78     .rst(rst),
79     .sdout_adc(sdout_adc),
80     .mclk_adc(mclk_adc),
81     .sclk_adc(sclk_adc),
82     .lrck_adc(lrck_adc),
83     .dataL_adc(dataL_adc),
84     .dataR_adc(dataR_adc)
85 );
86
87 mypll2 uut5(
88     .inclk0(clk),
89     .c0(c0),
90     .c1(c1)
91 );
92
93 adc_data_ready_tri uut7(
94     .clk(clk),
```

```

95     .rst(rst),
96     .lrck(lrck_adc),
97     .send_en(send_en)
98 );
99
100    uart_NbyteTran_3byteData_controller uut8(
101        .clk(clk),
102        .rst(rst),
103        .send_en(send_en),
104        .data(dataR_adc),
105        .tx_d(tx_d),
106        .tx_en(tx_en),
107        .tx_done(tx_done)
108 );
109
110    uart_tx_byte uut9(
111        .clk(clk),
112        .rst(rst),
113        .rx_d(tx_d),
114        .tx_en(tx_en),
115        .tx_done(tx_done),
116        .sci_tx(sci_tx)
117 );
118

```

对应的RTL电路图如图5所示。

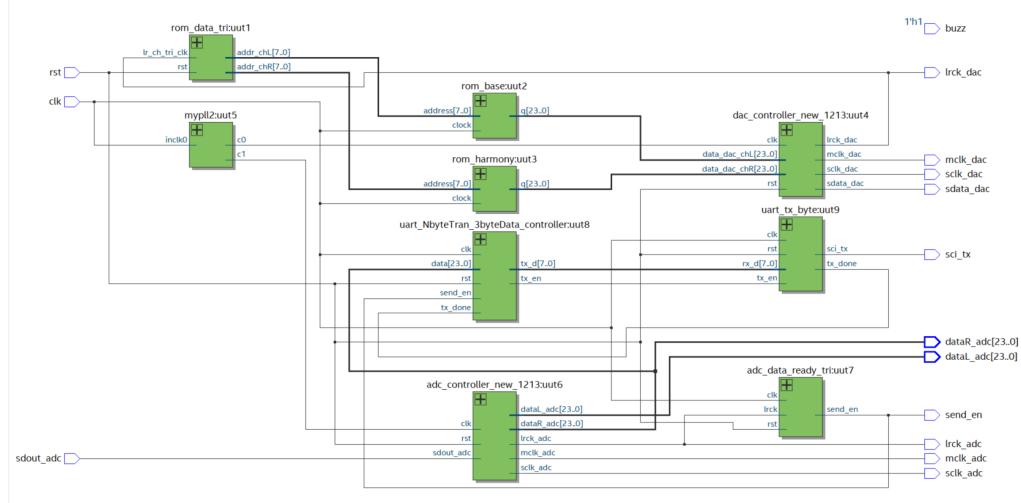


图 5: task3RTL 电路图

simulink设置如图6所示。

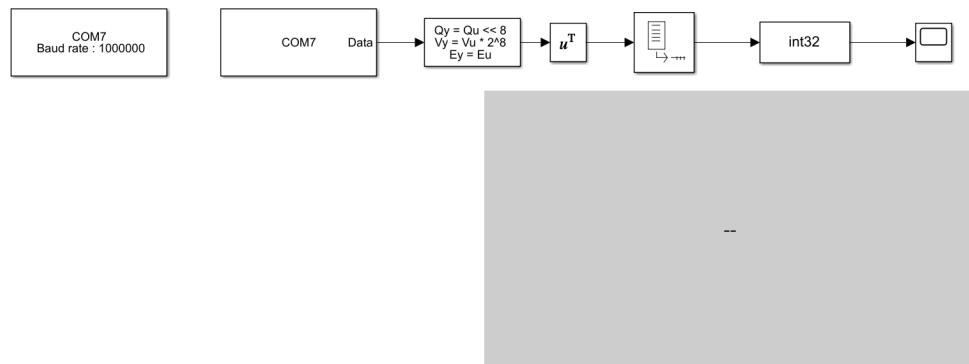


图 6: simulink设置

simulink接受到的波形如图7所示。



图 7: simulink接受到的波形