

电子技术基础实验最终实验报告

王磊 2022012972

2024 年 1 月 7 日

1 系统实现功能、架构和主要参数指标

1.1 实现功能

最终构建的测量系统实现主要功能如图1所示。FPGA作为下位机，控制DAC实现双路数模转换，控制ADC实现双路模拟测量。使用simulink搭建上位机，通过串口与FPGA通信，实现数据的传输与显示。

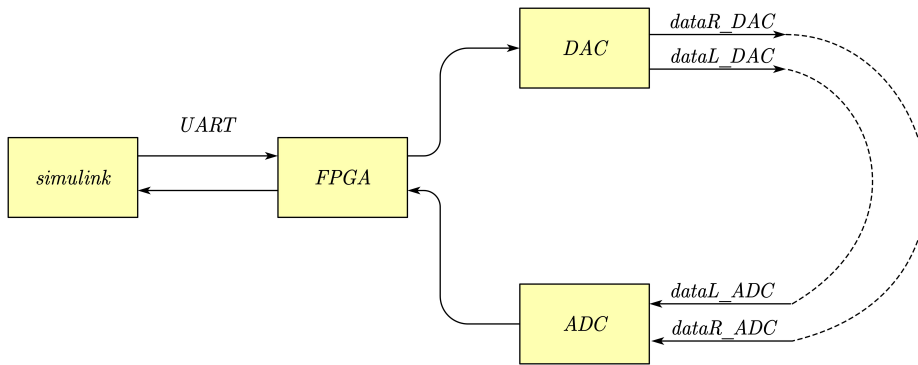


图 1: 测量系统功能图

1.2 系统架构与参数指标

实现系统功能的RTL电路图如图2所示。

其中各模块实现的功能为：

- **uart_rx_byte:** 串口接收模块，以字节为单位接收上位机发送的数据。
- **uart_rx_Nbyte_controller:** 数据锁存模块，功能为锁存上位机发送的最近两个字节的的数据，输送给乘法器。
- **rom_data_tri:** ROM触发模块，功能为根据DAC的lrck信号，更改输送给ROM模块的地址，实现数据的读取。

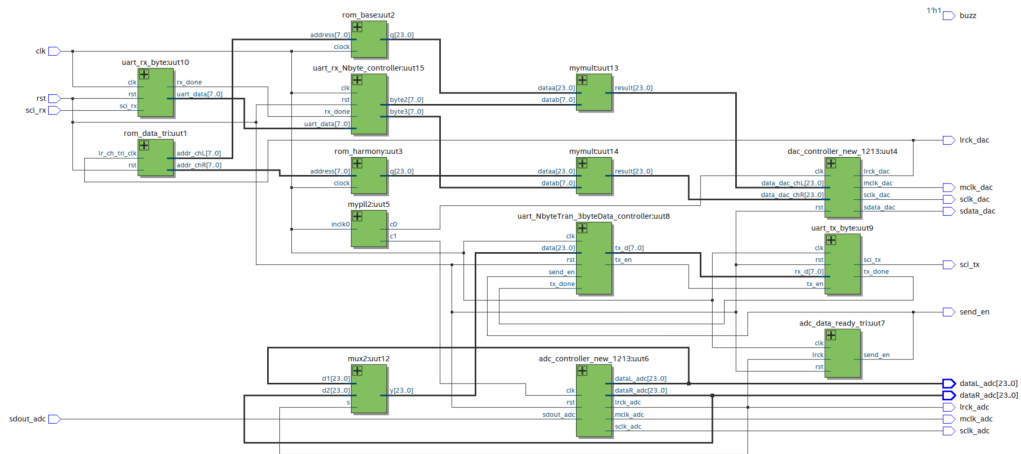


图 2: 系统RTL电路图

- **rom_base&harmony**: ROM模块, 分别储存了基波和三位谐波的数据, 根据rom_data_tri模块的地址, 输出对应的数据。
- **mypll2**: 锁相环模块, 负责产生6.5535MHz的时钟信号, 作为DAC的mclk信号; 产生1.28MHz的时钟信号, 作为ADC的mclk信号。
- **mymult**: 乘法器模块, 将上位机发送的数据与ROM模块输出的数据相乘, 输出给DAC。
- **dac_controller_new_1213**: DAC控制模块, 负责控制DAC的工作模式, 以及接收来自乘法器的数据, 输出给DAC。
- **adc_controllenew_1213**: ADC控制模块, 负责控制ADC的工作模式, 以及接收来自ADC的数据, 输出给MUX模块。
- **mux2**: MUX模块, 根据ADC的lrck信号, 选择输出ADC的左通道或右通道的数据。
- **adc_data_ready_tri**: 串口使能信号产生模块, 根据ADC的lrck信号, 使得当ADC采样后, 给出串口使能信号, 使得数据可以传输给上位机。
- **uart_NbyteTran_3byteData_controller**: 串口发送控制模块, 负责将24位的数据分为3个字节, 依此交给uart_tx_byte模块, 实现数据发送, 同时根据设定数据包的长度加入帧头和帧尾。
- **uart_tx_byte**: 串口发送模块, 以字节为单位, 将数据发送给上位机。

系统的主要参数指标如表1所示。

表 1: 系统主要参数指标

| 参数 | 指标 |
|---------|---------|
| DAC采样率 | 12800Hz |
| DAC量化精度 | 24bit |
| DAC通道数 | 2 |
| ADC采样率 | 5000Hz |
| ADC量化精度 | 24bit |
| ADC通道数 | 2 |
| 通信协议 | UART |
| 数据包长度 | 1000 |

2 设计思路与具体实现

除去老师提供的参考模块和由ip核自动生成的模块，由自己设计并实现的模块有：

- rom_data_tri
- adc_data_ready_tri
- mux2

下面分别介绍这三个模块的设计思路与具体实现。

2.1 rom_data_tri

该模块的代码如下：

```

1  module rom_data_tri (
2      input lr_ch_tri_clk,
3      input rst,
4      output reg [7:0] addr_chL,
5      output reg [7:0] addr_chR
6  );
7      always @(posedge lr_ch_tri_clk or posedge rst) begin
8          if (rst) begin
9              addr_chL <= 8'd0;
10         end else begin
11             if (addr_chL == 8'd255) begin
12                 addr_chL <= 8'd0;
13             end else begin
14                 addr_chL <= addr_chL + 1'b1;

```

```

15         end
16     end
17 end
18
19 always @(negedge lr_ch_tri_clk or posedge rst) begin
20     if (rst) begin
21         addr_chR <= 8'd0;
22     end else begin
23         if (addr_chR >= 8'd255) begin
24             addr_chR <= 8'd0;
25         end else begin
26             addr_chR <= addr_chR + 1'b1;
27         end
28     end
29 end
30
31 endmodule

```

该模块的功能是根据DAC的lrck信号，更改输送给ROM模块的地址，实现数据的读取。由于ROM模块的数据是以8位为单位读取的，因此需要两个地址，分别对应左通道和右通道。当lrck信号为上升沿时，左通道的地址加1；当lrck信号为下降沿时，右通道的地址加1。当地址达到255时，地址清零。

2.2 adc_data_ready_tri

该模块的代码如下：

```

1 module adc_data_ready_tri(
2     input clk,
3     input rst,
4     input lrck,
5     output reg send_en
6 );
7
8 reg pulse1, pulse2, pulse3;
9 wire clk_edge;
10 always @(posedge clk, posedge rst) begin
11     if(rst) begin
12         pulse1 <= 1'b0;

```

```

13         pulse2 <= 1'b0;
14         pulse3 <= 1'b0;
15     end
16     else begin
17         pulse1 <= lrck;
18         pulse2 <= pulse1;
19         pulse3 <= pulse2;
20     end
21 end
22 assign clk_edge = (pulse2 & ~pulse3) | (~pulse2 & pulse3);
23
24
25 always @(posedge clk, posedge rst) begin
26     if (rst) begin
27         send_en <= 1'b0;
28     end
29     else begin
30         if(clk_edge) begin
31             send_en <= 1'b1;
32         end
33         else begin
34             send_en <= 1'b0;
35         end
36     end
37 end
38
39
40 endmodule

```

该模块的功能是根据ADC的lrck信号，使得当ADC采样后，给出串口使能信号，使得数据可以传输给上位机。当lrck信号发生跳变时，说明ADC进行了一次采样，打慢两拍以确保数据做好传输准备，给出一个脉冲信号，通知串口模块发送新的数据。

2.3 mux2

该模块的代码如下：

```

1 module mux2(

```

```

2      input  s,
3      input [23:0] d1,
4      input [23:0] d2,
5      output reg [23:0] y
6  );
7
8      always @ (s or d1 or d2)
9          case (s)
10             1'd1: y <= d1;
11             1'd0: y <= d2;
12             default: y <= 24'd0;
13          endcase
14
15      endmodule

```

该模块的功能是根据输入的s信号，选择输出d1或d2。当s为0时，输出d1；当s为1时，输出d2；其他情况输出0。其中ADC的lrck信号作为s信号输入，根据lrck信号的跳变，选择输出左通道或右通道的数据。

3 系统创新

在实现本系统时，我并未采用老师提供的参考思路，即通过修改uart_NbyteTran_3byteData_controller中的状态机实现两路数据的同时发送。

通过阅读ADC芯片的datasheet我了解到，ADC对哪个通道采样是通过lrck信号决定的，当lrck为1时，采样左通道；当lrck为0时，采样右通道。因此，我设计了mux2模块，根据lrck信号，选择输出左通道或右通道的数据。在实现单路数据发送时，我检测了lrck的上升沿作为数据发送的时机，因此只需要增加一行代码使得下降沿也成为数据发送的时机，就可以实现两路数据的同时发送。

4 存在问题与改进方向

目前系统存在最大的问题在数据发送时由于没有更改状态机，因此每个数据包中左通道在前还是右通道在前是随机的，取决于重置后ADC的采样方式，存在一定概率出现左右通道数据错位的情况。解决这个问题的方法是在uart_NbyteTran_3byteData_controller中增加一个或多个状态，根据lrck信号的跳变，判断当前采样的是左通道还是右通道，从而确定数据包中左右通道数据的顺序。