

### 实验三 串口收发器和存储器的使用

**实验目的：**了解和掌握 UART 的工作原理，掌握仿真验证方法，为后续设计做准备

**实验原理：**

1. 串口基本原理：

UART (Universal Asynchronous Receiver/Transmitter) 是一种通用串行数据总线，用于异步通信。该总线双向通信，可以实现全双工传输和接收。在嵌入式设计中，UART用来与PC进行通信，包括与监控调试器和其它器件。UART是计算机与嵌入式系统中最常用的串行通信协议，速率有规定的9600等波特率。

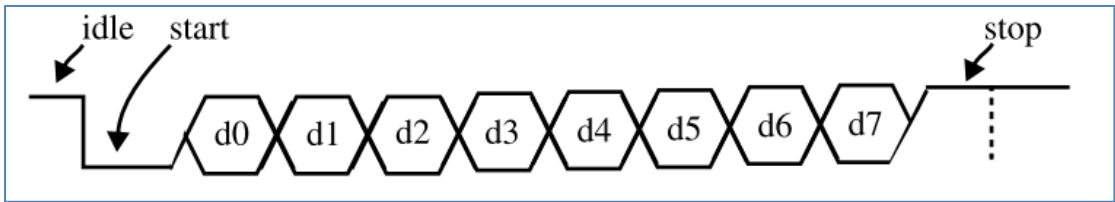


图 3.1 串口时序示意图

图3.1表明在异步传送中串行发送一个数据字节的位定时关系(图中没有包括奇偶校验位)。发送一个完整的字节信息，首先是一个作为起始位的逻辑“0”位，接着是8个数据位，然后是1个、1+1/2个或2个停止位逻辑“1”位，数据线空闲时呈现为高或“1”状态。在字符的8位数据部分，先发送数据的最低位（LSB），最后发送最高位（MSB）。每位持续的时间是固定的，由发送器本地时钟控制，每秒发送的数据位个数，即为“波特率”。

起始位和停止位起着很重要的作用。显然，他们标志每个字符的开始和结束，但更重要的是他们使接收器能把局部时钟与每个新开始接收的字符再同步。异步通信没有可参照的时钟信号，发送器随时都可能发送数据，需要从任何边沿的出现时刻开始正确地采样紧接着的 10~11位(包括开始位、数据位和停止位)。接收器的时钟与发送器的时钟不是同一个，因此，接收器采样点的间隔跟由发送器时钟所确定的位间隔时间不同，接收器设计不好可能会导致采样错误。

目前的个人计算机在硬件上很少有专用的UART(RS232)接口了，其UART功能已经融合在USB接口中。EGo1的串口模块采用了FT2232芯片，将UART转换加载到USB传输接口上，其管脚约束如表3.1所示。

电路端口	FPGA 管脚
UART_RX	N5
UART_TX	T4

表 3.1 EGo UART 管脚约束

在PC机一侧通过串口调试助手选择对应的USB COM端口，可以将数据通过USB连接线和EG01开发板完成通信。

## 2. 实验设计原理：

串口收发器包括发送器和接收器两个模块。首先，通过串口接收器模块从外部接收数据，并将接收到的数据送给控制器模块，同时控制器模块根据接收的串口数据产生发送数据，并通过串口发送器模块将数据发送到外部。

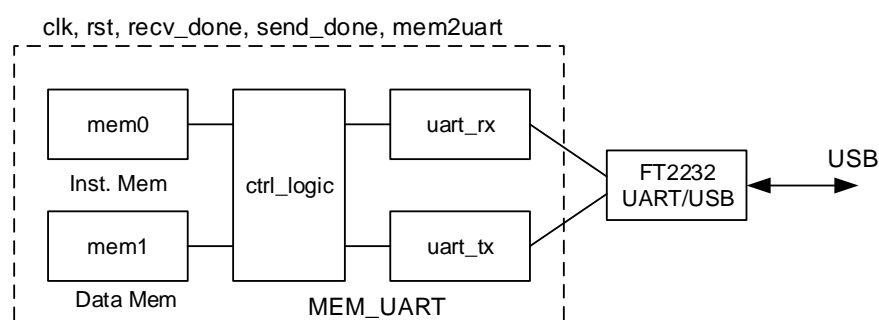
### (1) 实现的功能

串口→存储器

从串口读取512\*32\*2bit的数据到FPGA中的指令存储器和数据存储器，其中指令存储空间为512\*32bit，数据存储空间为512\*32bit。数据接收完成，拉高recv\_done信号。

存储器→串口：

当mem2uart为高电平时，FPGA从数据存储器中读取数据，并通过串口发送到上位机。数据发送完成，拉高send\_done信号。



硬件接口及配置性参数

```
module UART_MEM(
    input clk,          // 100MHz
    input rst,          // S1
    input mem2uart,     // SW0
    /*-----MEM-----*/
    output reg recv_done, // led 0
    output reg send_done, // led 1
    /*-----UART-----*/
    input Rx_Serial,
    output Tx_Serial
);

parameter CLKS_PER_BIT = 16'd10417; // 100M/9600
parameter MEM_SIZE = 512;
```

上图所示，系统时钟clk为100MHz，复位按键rst为S1，接收完成信号绑定到LED0。将

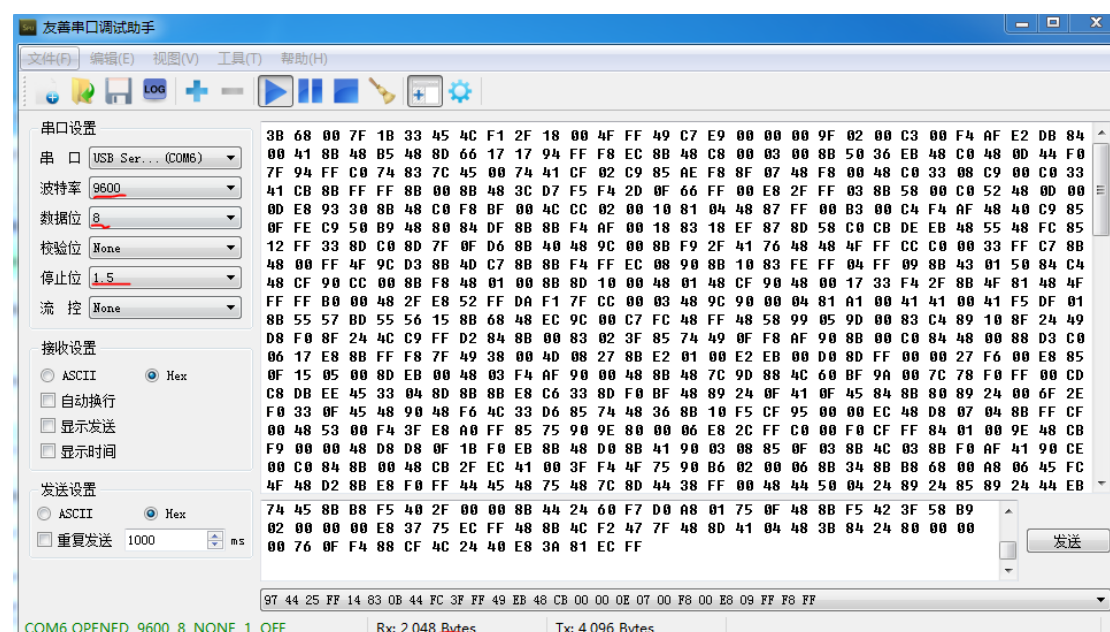
发送存储器中数据的控制信号mem2uart为SW0，发送完成信号绑定到LED1。

参数CLKS\_PER\_BIT可以配置串口的波特率： $\text{CLK\_PER\_BIT} = \text{系统时钟}/\text{波特率}$ 。

参数MEM\_SIZE可以配置指令存储器和数据存储器的大小： $\text{MEM\_SIZE} \times 32\text{bit}$ 。

本程序打通了串口到存储器以及存储器到串口的数据通路，更加具体的功能还需要在此基础上做进一步修改。

## (2) 使用方法



如上图所示，设置波特率9600，数据位8位，注意发送和接收要设置成Hex模式。发送数据之后可以看到Tx为4096Bytes。接收完成数据之后Rx为2048Bytes。如果要重新发送和接收数据，则需要先按下复位按键S1。

## (3) 文件列表

- serial\_port\_utility\_latest.exe: 串口调试助手安装包
- 发送: 保存的串口发送的测试数据，大小为  $512 \times 32 \times 2\text{bit}$ ，前半部分是指令，后半部分是数据
- 理论接收: 保存的是串口接收的理论数据，大小为  $512 \times 32\text{bit}$ ，内容与文件“发送”的后半部分相同
- src: 源文件

## 实验内容:

- (1) 阅读 Verilog 代码，理解电路实现原理。
- (2) 使用 Modelsim 或者在 Vivado 中，对代码进行行为级验证。
- (3) 建立工程，通过综合实现，将比特流加载到开发板上，对设计进行验证。
- (4) 选做: 添加一个信号，当该信号有效时，将数据存储器中的内容取出，逐字取反码，并存回数据存储器中。该信号可绑定在开发板的某个按键上，并通过行为级仿真和开发板上进行实现进行软件和硬件验证。