

文章编号: 1004-4353(2019)04-0356-05

基于 FPGA 和 UART 的乐曲演奏系统设计

吕兆承, 傅小兰, 李莹, 罗靖宇

(淮南师范学院 电子工程学院, 安徽 淮南 232038)

摘要: 采用 UART 串口通信协议,借助硬件描述语言 Verilog 和 Quartus 开发工具,设计了一款基于 FPGA 的简易乐曲演奏系统. 首先根据系统的总体功能要求,进行系统级设计;然后结合 UART 通信协议和乐曲演奏原理,分别设计了 UART 串口接收模块和乐曲演奏模块,并通过嵌入式逻辑分析仪 SignalTap II 验证了设计方案的有效性;最后采用 EP4CE10F17C8N 控制芯片对硬件进行调试,结果显示 FPGA 可成功驱动蜂鸣器演奏乐曲. 该方案设计简单,便于修改和扩展,具有很好的实用性.

关键词: FPGA; 模块化; UART; 乐曲

中图分类号: TN710 **文献标志码:** A

Music playing system design based on FPGA and UART

LYU Zhaocheng, FU Xiaolan, LI Ying, LUO Jingyu

(School of Electronic Engineering, Huainan Normal University, Huainan 232038, China)

Abstract: According UART serial communication protocol, a simple music playing system based on FPGA was designed by the hardware description language Verilog and Quartus development tools. Firstly, the system-level design was carried out according to the overall functional requirements to the system. Secondly, the UART serial interface receiving module was designed by UART communication protocol, and then the music playing module was designed combined with music playing principle, which were verified by embedded logic analyzer SignalTap II. Finally, the hardware debugging was carried out with EP4CE10F17C8N control chip. The results showed that FPGA could successfully drive buzzer to play music. The scheme has the advantage of simple structure, which is easy to modify and expand, and has good practicability.

Keywords: FPGA; modularity; UART; music

0 引言

在电子系统设计中,处理器与外部设备之间进行数据的发送与接收需要遵循一定的通信协议来保证数据传输的正确性. 常见的通信协议有 UART、SPI、I²C、USB 等,其中 UART(universal asynchronous receiver and transmitter)是一种异步收发传输器,其在数据发送时将并行数据转换成串行数据来传输,在数据接收时将接收到的串行数据转化成并行数据,以此实现全双工传输和接收,具有操作方便、抗干扰性强以及易于实现等优点^[1]. 目前,UART 被广泛地应用在电子设备的底层调试和一些电路模块间的通信中.

目前,具有乐曲演奏功能的电子产品大多采用单片机微处理器来实现乐曲演奏,但这些产品因受硬件资源的限制,难以对原设计进行变更或升级^[2]. FPGA 技术是一种新型的电路设计技术,其通过软件

收稿日期: 2019-06-27 **作者简介:** 吕兆承(1983—),男,实验师,研究方向为 EDA 技术.

基金项目: 淮南师范学院支持百名优秀学生课外科技实践创新活动基金资助项目(2019XS046);淮南师范学院校级科研项目(2018xj37)

编程设计硬件电路,可极大地提高电路设计的效率和可靠性,被广泛使用在 5G 通信、大数据处理、人工智能、工业控制等领域^[3]. 目前,有学者尝试性地将 FPGA 技术运用于乐曲演奏中. 例如:崔智军等^[4]在 FPGA 器件上设计了一个具有简易触摸屏的电子琴电路,并实现了触摸控制弹奏和自动播放乐曲的功能. 王丽君等^[5]采用 VHDL 编程技术设计了一种基于 FPGA 的简易电子琴,该电子琴具有手动弹奏、自动演奏和简谱码显示等功能. 基于上述研究,本文利用 Altera 的 FPGA 芯片,在 Quartus 平台下,采用 top-down 设计方法,设计一款基于 UART 通信接口的乐曲硬件演奏电路,并通过嵌入式逻辑分析仪 SignalTap II 对系统的有效性进行仿真验证.

1 系统设计

该演奏系统主要包括串口调试工具、UART 串口接收模块、乐曲演奏模块和蜂鸣器,其总体结构框图如图 1 所示. 串口调试工具的功能主要是控制指令的发送;UART 串口接收模块的功能主要是控制指令的接收,并将接收到的控制指令发送至乐曲演奏模块;乐曲演奏模块的功能主要是输出使乐曲能连续演奏所需的基本数据;蜂鸣器的功能主要是演奏乐曲. 其中 UART 串口接收模块和乐曲演奏模块在 FPGA 中实现. 由此可以看出,该系统只需很少的外围器件即可演奏音乐,从而可提高系统的可靠性.

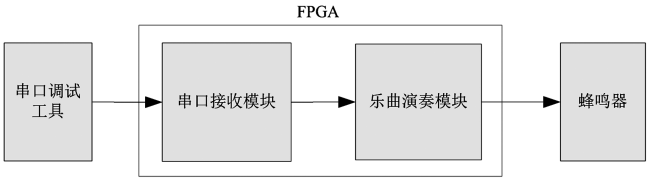


图 1 演奏系统的结构框图

2 UART 串口接收模块设计

2.1 UART 的工作原理

UART 串口的通信主要由两根信号线来实现,一根用于串口发送,另外一根负责串口接收. 数据在传输过程中不需要时钟,只需通信双方在数据传输过程中约定好一个特定的波特率即可,典型的波特率有 9 600、19 200、115 200 bps 等. 数据的传输有起始位、停止位、中间位. 起始位表示数据传输的开始,停止位表示数据传输的结束,中间位的个数可以根据需要设定,并可以设定奇偶校验位(可选)^[6]. UART 的数据传输格式如图 2 所示. 数据格式为 1 位起始位、8 位数据位、1 位停止位,无校验位.

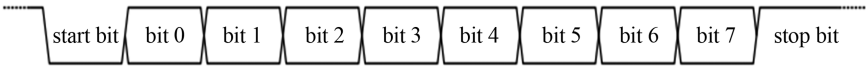


图 2 UART 的数据传输格式

2.2 UART 串口接收模块程序的设计

根据 UART 通信原理,本文将 UART 串口接收模块划分为波特率发生器、数据接收模块,如图 3 所示. 波特率发生器的功能是产生与 UART 通信同步的时钟,通过该功能 UART 串口接收模块可方便地按照 UART 通信的时序要求进行数据接收或者发送. 数据接收模块的主要功能是接收和转换数据. 数据接收模块在完成接收数据后,输出结束标志信号^[7]. 本文将 UART 串口接收模块的端口定义为:

- 1) clk 为系统时钟输入端,输入频率为 50 MHz;
- 2) rst_n 为系统复位信号输入端,低电平有效;
- 3) uart_rx 为接收模块串行数据接收端;
- 4) out_rx_dara 为接收模块并行数据输出端;
- 5) rx_bps_start 为数据接收标志信号,高电平表示接收数据.

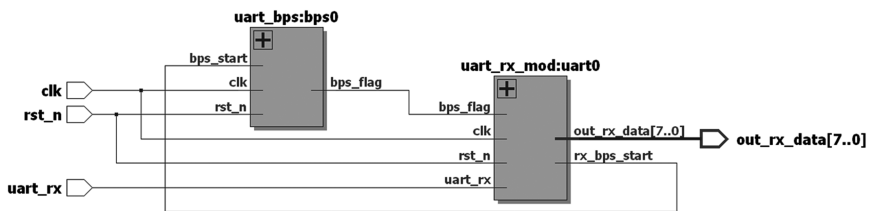


图 3 UART 串口接收模块的 RTL 级视图

2.3 仿真测试

本文用 Quartus 自带的嵌入式逻辑分析仪 SignalTap II 来捕获和显示信号,并用其来分析系统硬、软件之间的互相作用^[8].使用 SignalTap II 的一般流程为:新建调试文件,添加信号,设置信号触发,设置时钟相关参数,下载仿真. UART 串口接收模块的仿真结果如图 4 所示.由图 4 可以看出:串行数据接收端 uart_rx 上的数据序列为 0100011001,其起始位为“0”,数据位为“10001100”,停止位为“1”(最后一位).在数据接收标志信号 rx_bps_start 和波特率时钟计数器 bit_cnt 的控制下,UART 串口只采集输入数据(串行数据)的中间 8 位有效位,并将其逐位存储到寄存器 shift_data 中.当 UART 串口接收到所有(9 个)的波特率时钟之后,寄存器 shift_data 将接收到的数据并行输出给数据输出端 out_rx_data.因 UART 串口在数据通信时首先发送的是最低位数据,因此接收的数据为“00110001”.由表 4 中的数据栏可以看到,当 UART 串口接收完数据后,其数据接收标志信号 rx_bps_start 为“0”,数据输出端 out_rx_data 为“00110001”,与串行数据接收端 uart_rx 接收的数据一致.这表明, UART 串口接收模块的功能正常.

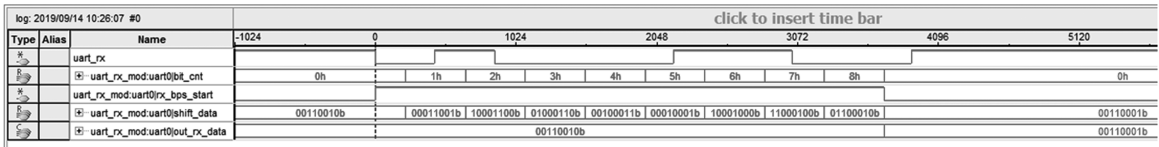


图 4 UART 串口接收模块的仿真时序

3 乐曲演奏模块设计

3.1 乐曲演奏的原理和实现

组成乐曲的每个音符的频率值(音调)及其持续的时间(音长)是乐曲能够连续演奏所需的两个基本数据. ①音调.因乐曲都是由一连串的音符组成,当演奏电路按照乐曲的乐谱依次输出这些音符所对应的频率时,蜂鸣器即可连续地发出音符的音调^[9].这些音符的频率是通过对基准频率进行分频得到的. ②音长.音长是由乐曲的速度及每个音符的节拍数来确定的.为了实现节拍,需选择一个时钟频率来控制音长.如:若将全音符的持续时间设为 1 s,只需提供一个 4 Hz 的时钟频率即可产生四分音符的时长;对于节拍较长的音符,如二分音符,则连续读取 2 次即可.因此,只要控制好输出到蜂鸣器的激励信号(包括频率的高低和持续时间)就可以使蜂鸣器发出连续的乐曲声^[10].

根据上述乐曲演奏的原理,本文将乐曲演奏模块划分为 4 个子模块:音符数据 ROM 模块、地址发生器模块、乐谱码查表模块、数控分频器模块.

3.2 音符数据 ROM 模块

音符数据 ROM 模块的功能是存放乐曲中的音符数据.该功能是利用 Quartus 软件自带的 IP 核 LPM-ROM 来实现的,即将乐谱中相应的音符按照一定的规则放在一个连续的地址上.将音符转换为代码的规则为:采用 8 位二进制数表示音符,低 4 位用 1(do)、2(re)、3(mi)、4(fa)、5(sol)、6(la)、7(si)代表音阶中的 7 个基本音级,高 4 位用 1、2、3 分别表示每个音级的低音、中音和高音,休止符用 0 表示.本文选用的 ROM 的容量为 8×512 bits,并选择 3 首乐曲(《两只老虎》《送别》和《布娃娃和小熊跳舞》)

分别储存在 ROM 中,各乐曲的地址值分别为 0~71、88~215、240~367。

3.3 地址发生器模块

本文将一个 9 位二进制计数器(计数最大值为 512)作为音符数据 ROM 模块的地址发生器,以此实现按节拍读取乐谱。该模块每收到 1 次时钟信号,计数器就会计数 1 次;计数结果停留的时间越长,该音符演奏的时间就越长。

3.4 乐谱码查表模块

乐谱码查表模块中包含乐曲全部音符所对应的频率,如表 1 所示。表 1 中的计算结果是四舍五入所得,因此与每个音符的实际频率相比会出现小于 1 Hz 的误差。

表 1 简谱中的音符与频率的关系

音符名	频率/Hz	音符名	频率/Hz	音符名	频率/Hz
低音 1	261	中音 1	523	高音 1	1 045
低音 2	293	中音 2	587	高音 2	1 174
低音 3	329	中音 3	659	高音 3	1 318
低音 4	349	中音 4	699	高音 4	1 396
低音 5	392	中音 5	784	高音 5	1 568
低音 6	440	中音 6	880	高音 6	1 760
低音 7	493	中音 7	987	高音 7	1 975

3.5 数控分频器模块

数控分频器模块的功能是通过对某个基准时钟频率进行分频,获得与各个音符所对应的频率输出。

3.6 乐曲演奏模块程序的设计

图 5 为乐曲演奏模块的 RTL 级视图。乐器演奏模块的端口定义如下:

- 1) clk 为系统时钟输入端,输入频率为 50 MHz;
- 2) rst_n 为系统复位信号输入端;
- 3) key_cnt 为控制信号输入端,用来选择 ROM 中不同的地址段。选择第 1 首乐曲, key_cnt 的值为“8'h30”,计数器从位置 0~71 循环计数,由此实现乐曲的循环演奏;
- 4) beep 为驱动蜂鸣器的输出信号。

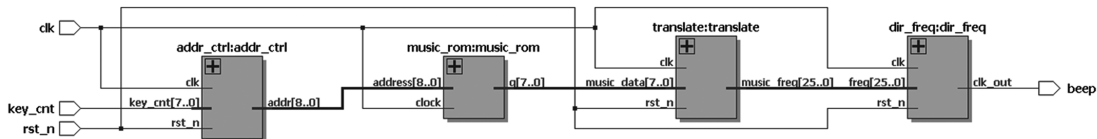


图 5 乐曲演奏模块的 RTL 级视图

为了实现乐曲的播放,首先需要将曲谱数据存储到音符数据 ROM 模块中,然后按照一定的节拍从 ROM 中读出曲谱。在时钟的驱动下,音符数据 ROM 模块依次输出该曲谱的音符数据并发送至乐谱码查表模块;乐谱码查表模块输出音符的相应频率,并将此频率输送至数控分频器模块;数控分频器模块通过对 50 MHz 时钟进行分频,输出此音符所对应的发声频率,由此驱动蜂鸣器发出连续的乐曲声。当乐曲被演奏完一遍后,乐曲演奏模块能自动从头开始循环演奏。乐曲演奏模块根据 key_cnt 取值,选择的地址段不同,演奏不同的乐曲。乐曲演奏的实现过程如图 6 所示。

3.7 仿真测试

图 7 为乐曲演奏模块的时序仿真图。图 7 中输入的控制信号 key_cnt 为“30”,演奏的乐曲为《两只老虎》。其中第 1 个音符是中音 do,它的输出频率是 523 Hz,占了 2 个状态的时间(不包括 0 时刻之前的状态);第 2 个音符是中音 re,它的输出频率是 587 Hz,占了 2 个状态的时间;第 3 个音符是中音 mi,它的输出频率是 659 Hz,占了 2 个状态的时间;第 4 个音符是中音 do,占了 4 个状态的时间。对照《两只老

虎》乐曲的乐谱可知,仿真结果正确.

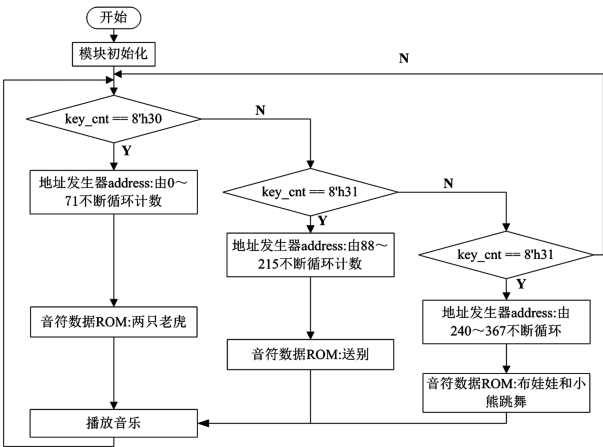


图 6 乐曲演奏流程

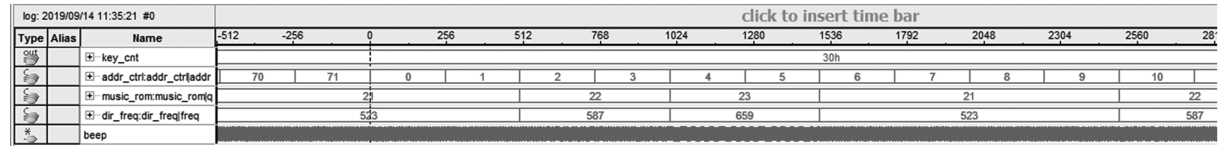


图 7 乐曲演奏模块的仿真时序

4 系统调试

测试完 SignalTap II 后,还需进行硬件测试.本设计采用芯航线 AC620 开发板(Altera Cyclone IV EP4CE10F17C8N)进行系统的调试和功能验证.调试过程为:首先打开串口调试工具助手,配置串口通信参数;然后根据用户需求发送控制指令给 FPGA,FPGA 根据此命令控制乐曲演奏器播放不同的乐曲.经测试,系统可实现智能化控制演奏过程,表明本文系统符合设计要求.

5 结语

本文在 EDA 开发平台上,利用 FPGA 技术设计了一款基于 UART 数据传输的乐曲演奏电路.测试结果表明:通过 PC 机串口调试助手发送控制指令给 FPGA,FPGA 可以准确地接收相应指令,并确定出与该指令相对应的乐曲,然后通过蜂鸣器即可完成音乐播放.本文方案稳定可靠,使用便捷,因此具有很好的实用性.另外,本系统在实际运行中可根据外设和处理器的不同需要,通过调整波特率和改变数据帧的格式设置等即可将本系统运用于不同的数字演奏系统.

参考文献:

[1] 袁玉卓,曾凯锋,梅雪松. FPGA 自学笔记:设计与验证[M]. 北京:北京航空航天大学出版社,2017:153-157.

[2] 陈强,陈守满. 基于 FPGA 的多功能触摸屏电子琴系统设计[J]. 无线互联科技,2016,12:38-70.

[3] 李营,殷小航,吕兆承,等. 基于 FPGA 的 VGA 汉字显示器设计[J]. 延边大学学报(自然科学版),2018,44(4):365-368.

[4] 崔智军,刘昭元,张瑜. 基于 FPGA 的简易触摸电子琴设计[J]. 现代电子设计,2014,37(22):114-116.

[5] 王丽君,李萌. 基于 FPGA 的简易电子琴设计[J]. 电子科技,2017,30(5):62-64.

[6] 魏亚昉,李振华,卢雪萍. 基于 FPGA 的 UART 控制器的设计[J]. 绍兴文理学院学报,2011,31(7):66-70.

[7] 鲍存会. 基于 FPGA 的多通道并行 UART 接口设计[J]. 陕西理工学院(自然科学版),2012,28(6):19-23.

[8] 吕兆承,李营,全贵英,等. 基于 FPGA 的简易信号发生器设计与实现[J]. 江苏科技信息,2018,21:43-45.

[9] 蒋小艳. 基于 FPGA 的多路音频采集与接口系统设计与实现[J]. 科学技术与工程,2012,12(27):7089-7093.

[10] 王金明. 数字系统设计与 Verilog HDL[M]. 7 版. 北京:电子工业出版社,2019:266-271.