

文章编号: 1000-5013(2010)06 0641- 04

利用 FPGA 的数字语音系统的设计与实现

吴昊, 黄华灿

(华侨大学 信息科学与工程学院, 福建 泉州 362021)

摘要: 采用基于 AMBE 算法的专用语音压缩编/解码芯片, 设计并实现一种基于 FPGA 的语音压缩编/解码系统. 整个控制程序包括复位及时钟分频程序、发送端控制程序和接收端控制程序 3 个功能模块, 控制程序的接收端和发送端都使用先进先出(FIFO)电路存储器. 仿真结果表明, 控制程序能够完成相应的功能, 符合设计的目的和要求.

关键词: 语音压缩; 芯片; 控制程序; 现场可编程门阵列; 先进先出存储电路

中图分类号: TN 912.3

文献标识码: A

通过 FPGA 对语音专用压缩芯片进行控制, 可使得设计的语音压缩系统^[1]具有码率较低、编码速率可选择的特点. 此外, 可以通过修改 FPGA 中控制程序, 进一步调整系统的码率等. 在实际的语音通信中, 为了语音质量满足实际的通话要求, 本文设计与实现一种基于 FPGA 的数字语音系统.

1 系统组成及原理

系统选用美国 Xilinx 公司的 FPGA 产品^[2]中 Spartan- II 系列的 XC2S100 其门数为 10 万门, 有两组工作电压: 内核工作电压为 2.5 V, 外部工作电压为 3.3 V. 整个系统包括 MicroPhone, 编译码器 (CODEC, A/D 和 D/A 转换器)、编/解码专用芯片、控制芯片 FPGA^[2]、集成运放、Earphone, 以及晶振、电源等元器件. 系统为全双工工作方式^[3], 其编解码过程是并行完成的. 整个系统的原理如图 1 所示.

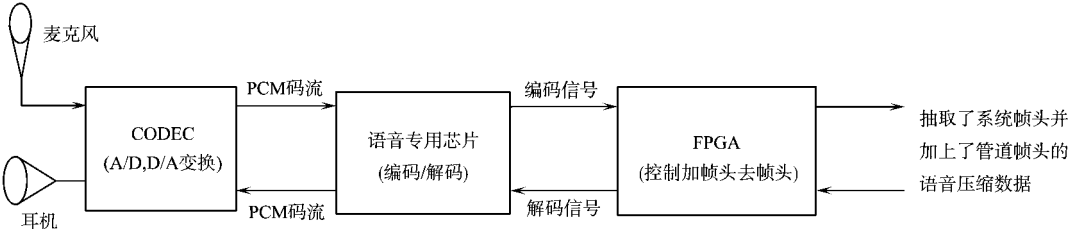


图 1 系统组成框图

Fig.1 Block diagram system

话音经 MicroPhone 后得到模拟信号, 通过 CODEC 将模拟信号进行 A/D 变换得到脉冲编码调制 (PCM) 信号, 并送入语音芯片进行编码. 输出的语音编码信号是具有一定帧格式的连续帧的数字信号. 然后, 控制芯片 FPGA 接收语音编码信号进行控制处理, 即去掉该帧纯语音数据前的系统帧头并加上信道帧头. 最后, 将处理后的语音编码信号送到信道加以传输.

控制芯片 FPGA 接收信道传来的编码信号, 将其去掉信道帧头并在纯语音数据前加上系统帧头, 然后, 发送到语音芯片进行解码. 解码后的 PCM 再由 CODEC 进行 D/A 变换得到语音模拟信号, 并经过音频功放送到耳机. 通过在 FPGA 中作相应的控制及修改 FPGA 中的控制程序, 可以调整系统的码

收稿日期: 2009-04-17

通信作者: 黄华灿 (1948-), 男, 教授, 主要从事视频处理与传输及专用集成电路设计的研究. E-mail: hchuangqz@yahoohoo.com.cn.

基金项目: 福建省科技计划项目 (2006T0006); 福建省泉州市科技计划项目 (2006G3)

率,以降低传输码率.这种效果是直接传输难以达到的.

2 控制程序设计及仿真^[4]

整个控制程序包括复位及时钟分频程序、发送端控制程序和接收端控制程序 3 个功能模块.控制程序的接收端和发送端都使用了先进先出(First In First Out, FIFO) 电路存储器.来自主芯片的已编码帧是高速率($512\text{ kbit} \cdot \text{s}^{-1}$) 的,但它不是连续的数据流,而是周期性(每次 20 ms) 间断出现的,没有数据帧时全为零.为此,需要将到来的每帧数据(去帧头后) 暂存于 FIFO 中,并且用 4 kHz 的低速率时钟,将 FIFO 中的存储数据连续地发送出去.

由于 $512\text{ kbit} \cdot \text{s}^{-1}$ 的高速已编码帧在去除帧头后,其 1 s 内速率平均值正好是 $4\text{ kbit} \cdot \text{s}^{-1}$, 因此用 4 kHz 时钟发送数据,可实现 FIFO 输入、输出的动态平衡.即 FIFO 既不会溢出也不会读空.通过以上过程,可以将高速但不连续的帧数据变为低速但连续的数据流发送到信道,实现低码率的传输.

2.1 复位及时钟分频程序

系统上电后,按照顺序对 CODEC 和主芯片进行复位.先由 FPGA 产生一个负脉冲,送至 CODEC,此后,CODEC 的复位脚一直保持高电平.以同样方式对主芯片进行复位操作,主芯片复位完成后即向 CODEC 的 SDI 口发送控制字.设定采样频率及帧同步信号频率,使其按照主芯片的要求工作.

主芯片和 CODEC 的主时钟都是 16.384 MHz.系统工作时,还需要一些其他的时钟信号,而这些时钟信号都是在 FPGA 中采用计数器对主时钟分频后得到.各时钟信号如图 2 所示.

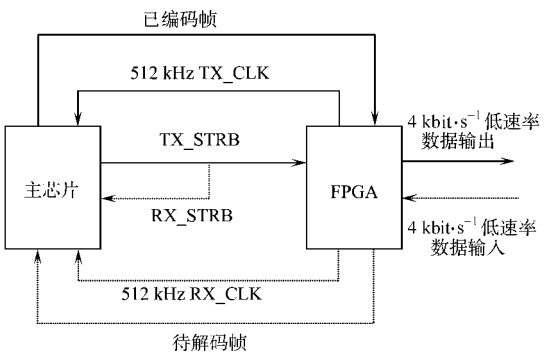


图 2 时钟信号
Fig. 2 Clock signal

复位及时钟分频程序仿真波形,如图 3 所示.其中, MCLK1684 是 16.384 MHz 的主时钟, TX_CLK 和 RX_CLK 分别是 512 kHz 的发送时钟和接收时钟, 4 kHz 是在传输信道上传输信号的时钟.

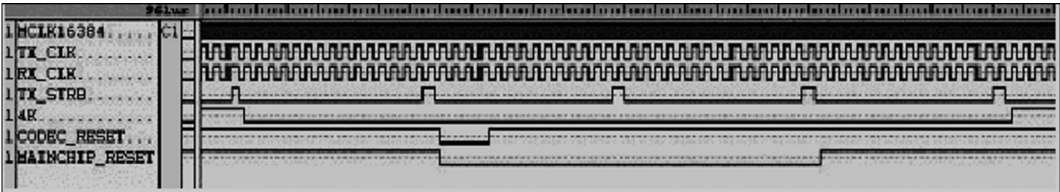


图 3 复位及时钟分频程序仿真波形

Fig. 3 Simulation waveform of the reposition and clock division

2.2 发送端控制程序^[5]

在发送端,每 20 ms 都有一帧来自主芯片的已编码的数据送到 FPGA.在 FPGA 中,使用移位寄存器检测帧头中的 Header(同步码,长度为 16 bit,十六进制为 13EC).若检测到该同步码,则输出一个有效高电平,使计数器 A 开始对 TX_CLK 计数.

由于每来一个 TX_CLK,时钟升沿对应 1 bit 数据,故计数器的长度应为 12 个字,192 位.计数完毕则表示帧头已经完毕,进入语音数据区.此时,将 FIFO 的写使能变为有效,开始把语音数据写入 FIFO 中.将计数器 A 清零,同时启动计数器 B,对 TX_CLK 进行新的计数.以上是跳过帧头而只写入语音数据,即去除帧头.

由于编码速率选择的是 $4\text{ kbit} \cdot \text{s}^{-1}$,此时每帧中的语音数据的长度为 5 个字,80 位.因此,计数器 B 计数值到 80 时,表示语音数据已经全部写入 FIFO 中.此时,将计数器 B 清零,完成一帧的处理;隔 20 ms 后下一帧到来,再重复上述过程.

当 FIFO 半满时, WR_COUNT 变为有效高电平.它将 FIFO 的读使能变为有效高电平,此时,开始将 FIFO 存储的数据用 4 kHz 的时钟读出并发送.

发送端控制程序流程图及仿真波形,如图 4、5 所示.图 5 中: CO_FRAME 是已编码帧的波形,作为

仿真时要输入的激励信号; HEADERWREN 为存储已编码帧, 作为本地帧头到 FIFO 的写使能信号; DATARWR_EN 为存储语音数据 FIFO 的写使能信号; RD_EN1 是从该 FIFO 读出数据的读使能信号; LOWDATAOUT 则是最后发送的 $4\text{ kbit} \cdot \text{s}^{-1}$ 的低码率数据.

仿真时, 设置激励信号帧数据 CO_FRAMED 的第 13 字至第 17 字(即 5 个字, 80 bit 的语音数据)十六进制表示为: EE00, FF00, FF00, FF00, F0DD, 如图 3 所示. 从图 5 可以看出, 发送的数据输出 LOWDATAOUT 与这 5 个字的语音数据完全相同, 只是速率降低为 $4\text{ kbit} \cdot \text{s}^{-1}$, 程序无误.

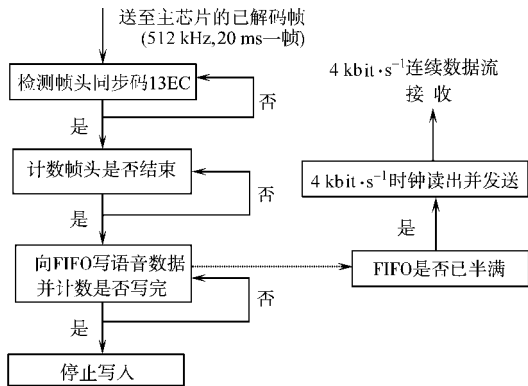


图 4 发送端控制程序流程图

Fig. 4 Flow chart of the control programme in sender

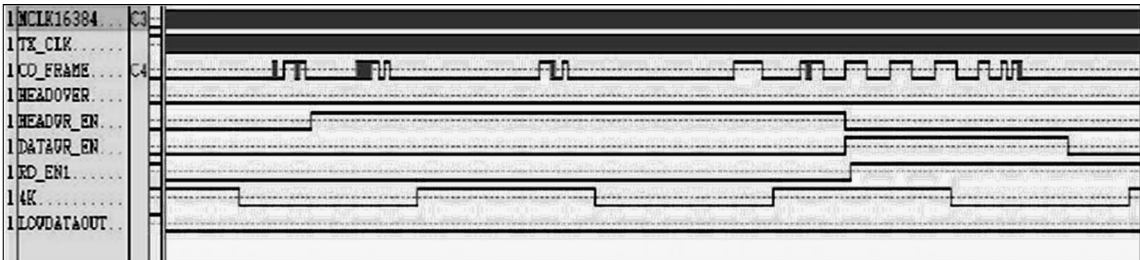


图 5 发送端控制程序仿真波形

Fig. 5 Simulation waveform of the control programme in sender

2.3 接收端控制程序^[5]

FPGA 接受来自信道的 $4\text{ kbit} \cdot \text{s}^{-1}$ 的连续数据流, 并将该语音数据流存储到 FIFO 中. 为了重新组帧, 在接收程序中, 还要产生一个本地帧头, 它可以另行产生. 因为主芯片是全双工的, 可以同时编码和解码, 也可以把已编码帧中的帧头暂存后作为本地帧, 程序中实际使用的就是此方法.

作相应控制, 每 20 ms(通过分频得到一个 50 Hz 的信号实现)一次, 先用 512 kHz 的时钟 RX_CLK 读出暂存的本地帧头, 读完后, 立刻(仍用 RX_CLK)从 FIFO 中读出 80 bit 语音数据(同发送程序中一样, 也是通过计数实现). 这样就重新组合成一个格式与已编码帧相同, 完整的待解码数据帧, 将其送至主芯片进行解码.

需要注意的是, RX_STRB 要与输入主芯片的待解码帧同步, 即满足图 5 中所示的时序关系才能够正确解码. 这是由于主芯片采用的是主动模式, RX_STRB 是直接连接到 tx_strb 的, 二者为同一个信号. 为此, 需要作相应处理, 使得在 RX_STRB 的升沿时才开始组合一帧数据, 以完成同步. 以上过程每 20 ms 重复一次, 即恢复各待解码帧, 完成解码.

接收端控制程序流程图及仿真波形, 如图 6, 7 所示. 图 7 中: LOWDATAIN 是接收到的 $4\text{ kbit} \cdot \text{s}^{-1}$ 低码率语音数据流; RD_EN2 是存储语音数据的 FIFO 的写使能信号, 该信号为有效高电平时, 从 FIFO 中读出语音数据(80 bit); DE_FRAME 为重组后得到的待解码帧. 由于主芯片采用的是主动模式, RX_STRB 直接连接到 TX_STRB, 二者是同一信号, 该信号与重组得到的待解码帧 DE_FRAME 同步时, 才能正确解码(为了方便观察, 一些显示状态的中间信号未在图中列出.).

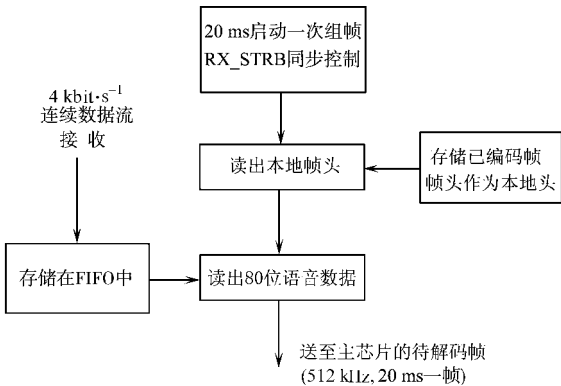


图 6 接收程序流程图

Fig. 6 Flow chart of the receiving programme

比较发现, 图 7 中的待解码帧 DE_FRAME 与图 5 中的已编码帧 CO_FRAME 波形完全相同. 这表明, 发送端产生的编码帧经过控制程序的去帧头、

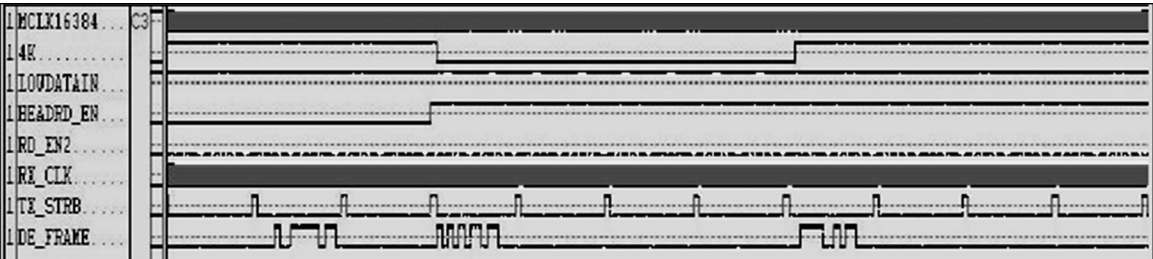


图 7 接收端控制程序仿真波形

Fig.7 Simulation waveform of the control programme in receiver

暂存、传输, 以及在收端加帧头和重组后, 能够无误地恢复出来; 收端的主芯片解码, 完成了语音压缩(低码率传输语音信息) .

3 结束语

整个系统为全双工, 由独立硬件电路实现, 在实际的语音通信中, 语音质量可满足实际的通话要求. 通过复位及时钟分频程序、发送程序和接收程序 3 个功能模块的仿真结果可以看出, 整个控制程序能够完成相应的功能, 符合设计的目的和要求. 系统兼备处理速度快与设计实现灵活的优点, 而且成本低, 使用灵活, 具有较广的应用范围.

参考文献:

[1] RICHARD V C, CANDANCE A K. Speech and language processing for next-millennium communication services [J]. Proc of IEEE, 2000, 88(8) : 1414-1337.
[2] 褚振勇, 翁木云. FPGA 设计与应用[M]. 西安: 西安电子科技大学出版社, 2002
[3] 王秉钧, 孙学军. 现代通信系统原理[M]. 北京: 国防工业出版社, 2001.
[4] 林顺达, 苏武浔. Möbius 变换在模拟通信中的应用与仿真[J]. 华侨大学学报: 自然科学版, 2006, 27(1) : 108-111.
[5] 侯伯亨, 顾新. VHDL 硬件描述语言与数字逻辑电路设计[M]. 西安: 西安电子技术大学出版社, 1999

Design and Implementation of a Speech Compression
System Based on PFGA

WU Hao, HUANG Hua-can

(College of Information Science and Engineering, Huaqiao University, Quanzhou 362021, China)

Abstract: Adopting a special speech compression encoder/decoder chip based on AMBE algorithm, this paper designs and implements a FPGA-based speech compression encoding/ decoding system. The whole control program include three function modules, those are reset and clock frequency, the transmitter and receiver control program. The receiver and transmitter of control procedures use the FIFO memory circuit. The simulation results show that, the control program can complete the corresponding function, and meet the purposes and requirements of design.

Keywords: speech compression; chip; control program; field-programmable gate array; first in first out memory circuit

(责任编辑: 陈志贤 英文审校: 吴逢铁)