

基于 FPGA 的语音信号实时处理

姜元亭, 樊昌元, 杨军华

(成都信息工程学院 电子工程学院, 四川 成都 610225)

摘要: 介绍一种在语音识别系统中运用 FPGA 技术对语音信号进行前期实时处理的方法。利用 DSP Builder 设计信号处理算法的图形化电路模块, 运用硬件环 (HIL Hardware in the Loop) 技术对模块进行软硬件协同仿真。满足设计要求后, 再用 Signal Compiler 将模块转换成 VHDL 语言和 Quartus II 工程文件下载至目标芯片。结果表明此方法可以快速灵活地设计出语音处理模块, 语音数据能在要求的时间范围内处理完毕, 达到了实时处理的目的。

关键词: 语音识别; FPGA; 实时; 信号处理

中图分类号: TP391

文献标识码: A

文章编号: 1674-7720(2011)02-0029-03

Real-time processing of speech signal on FPGA

Jiang Yuanting, Fan Changyuan, Yang Junhua

(School of Electronic Engineering, Chengdu University of Information Technology, Chengdu 610225, China)

Abstract: This paper introduces a real-time voice signal pre-processing method based on FPGA technology in speech recognition systems. DSP builder design was used in a graphical signal processing circuit model, meanwhile using HIL (hardware in the loop) technology for the collaborative simulation software modules. After meeting the design requirements, reoccupy signal compiler will convert VHDL language and module Quartus II project files downloaded to the target. The results show that this method can quickly and flexibly to design voice processing module, voice and data requirements can be processed within the time frame, to achieve real-time processing purposes.

Key words: speech recognition; FPGA; real-time; signal processing

随着语音识别技术的应用越来越广, 对其实时性的要求也越来越高。专用的 DSP 语音芯片虽然有硬件加速功能, 但其指令依然是串行计算, 在实时性方面有所欠缺。如今, 具有并行运算能力的 FPGA 主频不断提高, 加上其设计灵活、功耗低、体积小等优点^[1], 可以满足语音信号实时处理的要求。目前很多语音处理算法都是基于软件平台的, 真正的语音处理硬件实现很少。本文针对非特定人的语音信号, 研究当前主流的语音处理算法, 并将这些基于软件平台的算法“硬件化”。在保证一定精度的前提下将浮点运算转换成便于 FPGA 实现的定点运算^[2]。

本文以通过对语音信号滤波、分帧、加窗、能量计算等模块的设计为例, 介绍语音信号实时处理的方法, 需要运用到 MATLAB、DSP Builder、QUARTUS II、ModelSim 等 EDA 工具联合设计^[3]。语音信号经过模数转换进入 FPGA 以后, 对其滤波, 因为要对信号进行实时处理, 需要采用动态分帧, 最后计算出每帧的能量为语音信号的下一步处理如端点检测、特征提取^[4]等做好前期准备。

1 实时处理算法分析

语音数据经过 A/D 转换之后进入芯片, 首先对其进行滤波。为了使信号的频谱趋向平坦, 需要对其进行预加重滤波, 这里采用一阶 FIR 滤波器^[5]:

$$H(z) = 1 - \mu z^{-1} \quad \mu \approx 0.9375 \quad (1)$$

差分方程表示为:

$$s[i] = s[i] - \mu \times s[i-1] \quad 0 \leq i \leq N \quad (2)$$

语音信号虽然是一种非平稳信号, 但在短时内 (10 ms ~ 30 ms) 可以看作是平稳的^[2], 这样就可以对其进行分帧处理。在实时系统中无法确定语音的长度和大小, 只能对其进行动态分帧。考虑到帧的连续性, 采用交叠分帧, 帧移取 0.5, 硬件中可以用两个 FIFO 实现, 其中 FIFO1 的读时钟频率是写时钟的两倍, 且 FIFO2 的读写时钟频率与 FIFO1 读时钟频率相同。

分帧后的数据需要窗函数对其加权, 加窗后的语音信号为 $s_w(n) = s(n) \times \omega(n)$ 。由于汉明窗在语音频段的平滑特性, 因此本文采取汉明窗^[4]:

《微型机与应用》2011 年 第 30 卷 第 2 期

硬件纵横 Hardware Technique

$$w(n) = \begin{cases} 0.54 - 0.46\cos[2\pi n/(N-1)] & 0 \leq n \leq N-1 \\ 0 & \text{其他} \end{cases} \quad (3)$$

信号经过以上处理后,便可以对其每一帧计算。设第 n 帧语音信号为 $s_n(m)$ 。短时能量 E_n 表征语音信号能量大小,其定义为:

$$E_n = \sum_{m=1}^N s_n^2(m) \quad (N \text{ 为帧长}) \quad (4)$$

第 n 帧的短时平均幅度定义为:

$$E_n = \sum_{m=1}^N |s_n(m)| \quad (5)$$

2 硬件模块的实现

仿真时通过读取 hex 文件来模拟实时的数据流。通过 MATLAB 将采样频率 16 kHz, 宽度 8 bit 的 wav 格式音频文件转化成 hex 文件的数据。部分代码如下^[6]:

```

.....
[y,fs,n]=wavread('speech.wav');
y1=int8(y*(2^n-1)+128);
[a,b]=size(y1);
fid=fopen('speech.txt','wt');
for i=1:a;
line=[num2str(i-1),':',num2str(y1(i)),','];
fprintf(fid,'%s\n',line);
end
fclose(fid);
.....
    
```

在模块中通过地址计数器将 ROM 中的数据不断读出,然后对数据流进行滤波。其 DSP Builder 模块实现如图 1 所示。

考虑到语音信号的短时平稳性,将 256 点数据分为一帧写入 Dual-Clock FIFO,写入 128 点后以两倍的写入速度读出,同时以两倍速度写入深度为 128 的 FIFO2。如此循环便可以实现帧的交叠。具体实现如图 2 所示,左半部分为时钟控制模块。

为了使每帧的数据点与窗函数的数据点一一对应,

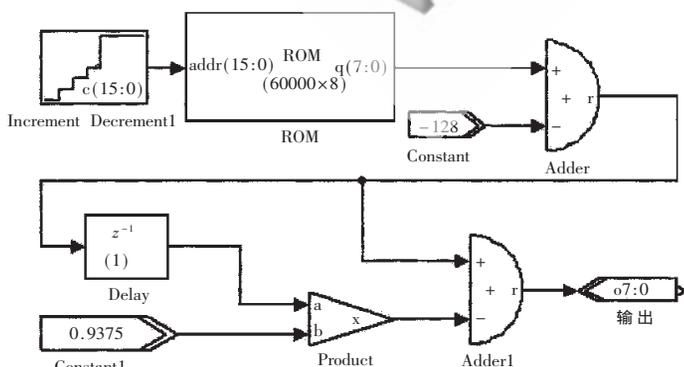


图 1 数据产生及滤波模块

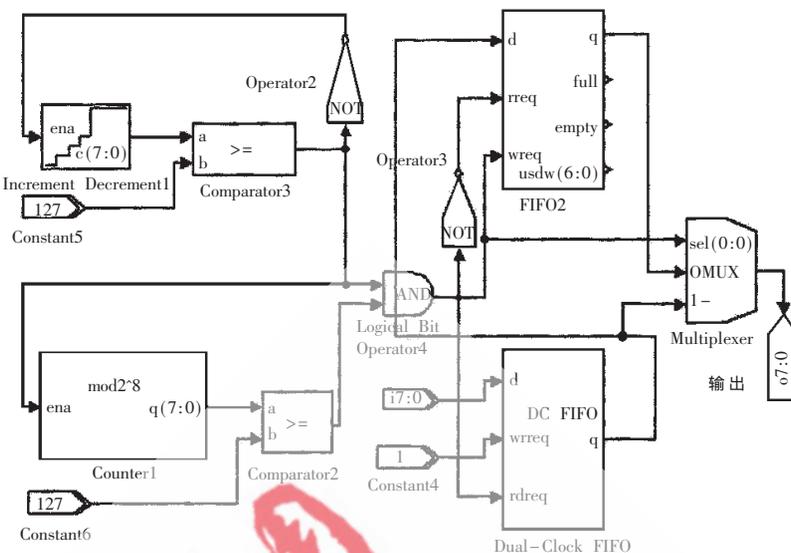


图 2 分帧模块

在加窗之前搭建了一个时序控制模块。Constant1 控制模块延时 384 个时钟周期,counter 是模为 256 的计数器。将 LUT 设为 $[7964 - 15073 \times \cos([0:2 \times \pi / 255:2 \times \pi])]$ ^[6]。分帧后的信号取模然后与窗函数相乘再累加便得到其能量,由 Multiply Accumulate 模块实现^[2]。Clock 提供基础时钟,PLL 产生模块所需要的两个时钟,Signal Compiler 对模块进行编译,转化成 VHDL 语言。具体模块如图 3 所示。

3 仿真测试
将上述三个子模块和 Simulink 中的模拟示波器 Scope 连接在一起,读取 ROM 中语音“1、2、3”的数据流。结果显示在示波器上,如图 4 所示。从上到下依次为原始信号、滤波信号、分帧信号、能量信号。

从图中可以看到设计模块已经可以实时处理数据,达到了设计要求。接下来便可以将其转换成 VHDL 语言在 QUARTUS II 中进行仿真,生成 pof 文件下载到 FPGA 里面。打开 Signal Compiler,Family 选择 Cyclone II,Device 选择 ALTERA 公司的 EP2C5T144C6 芯片。点击 compile,便可以生成工程文件、VHDL 代码及配置文件^[1]。

以上属于软件仿真,具有速度慢、内容不易控制等缺点。ALTERA 的 DSP Builder 提供的 HIL 模块可以在 Simulink 模型与 FPGA 开发板之间通过 JTAG 通信口建立联系,从而实现基于 MATLAB/DSP Builder 平台的硬件仿真。打开 HIL 模块,设置好工程文件 speech.pof 路径,连接上 FPGA 开发板,点击 Configure FPGA 便可以进行硬件仿真。打开示波器查看仿真结果与软件仿真结果吻合。在 QUARTUS II 中对生成的工程文件进行编译。整个系统使用了 306 个 LE、214 个寄存器、62 个管脚,非常节省资源。

通过 DSP Builder 进行 FPGA 设计无论是建模还是仿真都非常方便快捷,并可以在外部硬件测试平台不够完善的条件下引入 HIL 模块进行软硬件联合仿真。相对

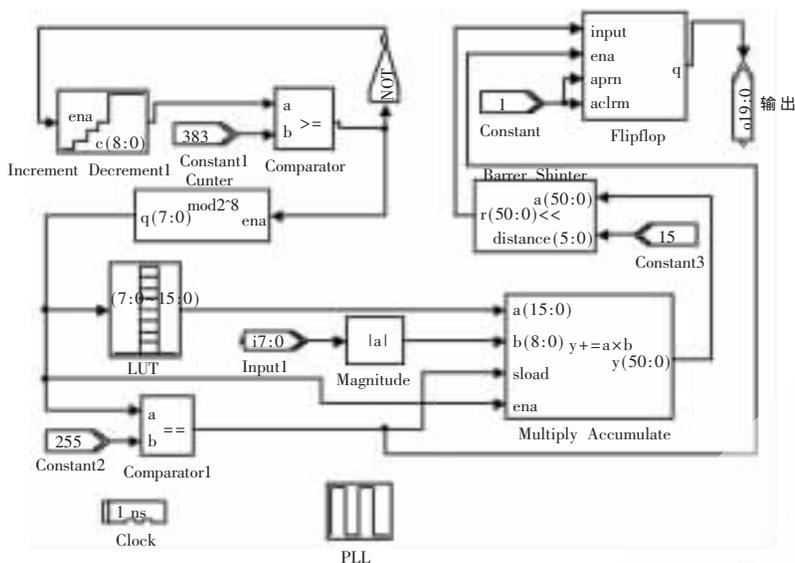


图3 加窗及能量计算模块

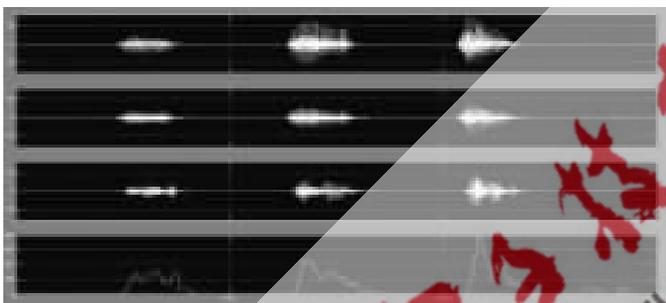


图4 仿真结果

于传统开发方式,具有更大的优势。在时序仿真时可以看出从语音输入到能量的输出占用 640 个周期,在 100

MHz 的工作频率下仅耗时 6.4 μ s,是在 MATLAB 下运行速度的 50 多倍,实时性得到了充分的体现。

参考文献

[1] 潘松,黄继业.EDA 技术与 VHDL(第 2 版)[M].北京:清华大学出版社,2007:12-19.
 [2] 刘军海,基于 DHMM 非特定人孤立词语音识别及硬件设计研究[D].上海:上海大学,2007:35-37.
 [3] 潘松,黄继业,王国栋.现代 DSP 技术[M].西安:西安电子科技大学出版社,2003:237-242.
 [4] 王炳锡,屈丹,彭煌.实用语音识别基础[M].北京:国防工业出版社,2005:56-59.
 [5] 刘静萍,姜占财,德熙嘉措.语音信号的预处理技术探讨[J].甘肃联合大学学报(自然科学版),2006,20(5):61-64.
 [6] 谢秋云.基于 FPGA 的语音识别技术研究[D].南京:江苏大学,2007:42-48.

(收稿日期:2010-08-19)

作者简介:

姜元亭,男,1985 年生,硕士研究生,主要研究方向:信号获取与处理。

樊昌元,男,1968 年生,教授,硕士生导师,主要研究方向:大气探测信息处理、通信系统与信号处理。

杨军华,男,1984 年生,硕士研究生,主要研究方向:北斗信号的调制与接收。