

基于FPGA的乐曲演奏器的设计

张亮 罗小巧 董继承 华中师范大学 信息技术系

摘要: 本文详细阐述了利用VHDL 语言和模块化的设计方法,在EDA开发工具Quartus II 软件平台上,通过音符编码的设计思想实现了音乐自动播放、琴键演奏、配有随音乐节奏而闪烁变化的LED以及乐谱显示等功能的乐曲演奏器系统。使基于FPGA芯片的乐曲演奏器数字电路得到了优化,提高了它的灵活性和可扩展性。

关键字: 电子琴 VHDL 可编程门阵列

1. 引言

电子琴因其价格合理,功能强大而且结构简单,广泛的应用于各个领域。而作为专用集成电路(ASIC)概念上的一个新型范畴和门类,FPGA以其自上至下的设计方法,功耗小,能够现场编程和反复改写的新颖功能,体现出了十分诱人的应用前景。本文采用ALTERA公司生产的最具性价比的Cyclone系列FPGA,利用VHDL语言来实现电子琴的设计。整个设计采用自顶向下的设计思路,配以外围电路,实现了用键盘演奏乐曲、音乐自动播放、音符的显示及高低音的显示。

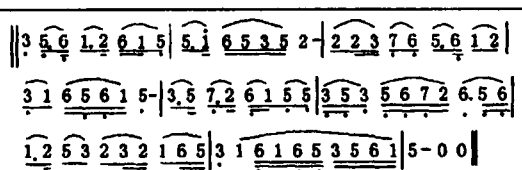


图1 《梁祝》中化蝶部分简谱

本设计一共采用16个按键,其中12个琴键,3个高中低音选择键和1个音乐自动播放开关按键。音乐自动播放的乐曲为梁祝的化蝶部分,其具体的乐谱如图1。

2. 系统工作原理

2.1 系统原理图

图2就是整个系统的整体电路结构框图,从这个图中就可以看到整个系统的工作过程。系统分为两个流程:键盘演奏和音乐的自动播放。这两个流程可以通过音乐自动播放开关按键来实现切换。

首先,FPGA芯片读取音乐自动播放开关按键信号。如果音乐自动播放开关按键关闭,则FPGA等待琴键信号,待琴键信号发出后,FPGA就根据琴键信号输出相应的频率信号以及驱动显示模块的信号;如果音乐自动播放开关按键打开,则FPGA开始自动查找乐曲梁祝的乐谱,然后产生相应的频率信号以及显示信号。显示模块和扬声器得到驱动信号后开始工作。这样,音乐就自然的演奏出来了,音符也得到了显示。

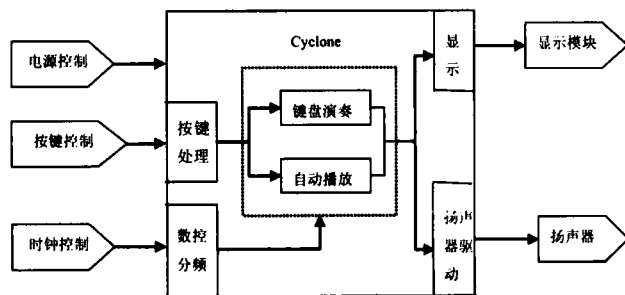


图2 系统整体电路结构框图

在这里,显示模块有一个七段数码显示管和三个LED组成。

七段数码显示管主要显示音符,而LED显示高中低音。时钟信号为12MHz。时钟通过分频器后可以得到1MHz的音符基准频率和4Hz节奏控制时钟来控制音符的节奏。

我们知道,要使扬声器发出美妙的音乐就需要控制好每一个音符的频率以及每一个音符演奏时所持续的时间。因此,音符频率和节奏控制是演奏音乐的两个关键因素。下面从这两个方面来讨论。

2.2 音符频率的获得

由乐曲的12平均率可知:每2个八度音(如简谱中的中音1与高音1)之间的频率相差1倍。在2个八度音之间,又可分为12个半音,每2个半音的频率比为21/12。另外,音符A(简谱中的低音6)的频率为440 Hz,音符B到C之间、E到F之间为半音,其余为全音。由此可以计算出简谱中从低音1至高音1之间每个音符的频率。

可以通过一个分频电路来产生各音符所需的频率。但是分频系数应为整数,而每个音符频率一般存在小数,这样就存在一个四舍五入的过程来取整。分频系数又与时钟基准频率有关。如果分频器时钟过低,那么分频系数小,取整后的误差大;过高的时钟虽然能够减小误差,但是又增加了分频器的级数,继而增加系统成本。实际的设计应处理好这一对矛盾,在尽量减小频率误差的前提下取合适的时钟频率。本文设计的电子琴选取由12MHz分频得到的1MHz的基准频率。其实,只要各个音符间的相对频率关系不变,演奏出的乐曲听起来都不会“走调”。

还有一点需要强调:对基准频率分频后获得的输出信号,是一些脉冲极窄的尖脉冲信号(占空比=1/分频系数),这些信号是没有足够的功率来驱动扬声器发生的。一般是在这些信号后加一个D触发器,将分频器的输出脉冲均衡为对称方波(占空比=1/2),但这时的频率是原来的1/2。在计算分频系数时应该考虑。

2.3 节奏控制

每个音符持续的时间是乐曲能连续演奏所需的另一个基本要素。我们可以在每一个按键的后面连接一个D触发器,用来存储每一个琴键以及辅助按键的状态。在第一次按下琴键时,启动最小节拍计时器,每隔一个音乐的最小节拍将所有D触发器的数值读入到系统内部进行处理。例如,我们要演奏“3”这个音符2个拍子,在相应琴键上按下2个拍子的过程中,系统也读了两次“3”这个音符。这里,我们假设通过电子琴演奏的乐曲最小节拍为1/4拍(当然,这个最小节拍是可以修改的,这在后面将要讲到),将1拍的时间长度定为1 s,则只需要再提供一个4 Hz的时钟频率即可产生1/4拍的时长。

音乐自动演奏的时间控制通过记录来完成,对于占用时间较长的节拍一定是1/4拍的n倍(n为整数)。我们就可以用一个存储器来存储音符。1/4拍的音符存储电路一次,n/4拍的音符存储电路n次。在音乐自动演奏的时候,用一个加法器来计算存储器的地址,每隔1/4秒计算一次,取出一个音符。这样乐曲的节奏也就可以控制了。

3. FPGA内部电路结构设计

我们可以利用结合LPM宏单元,将自动播放的乐曲音符存储电路在定制的ROM中,这样就可以随时改变乐曲,提高了系统的灵活性;而琴键的值可以通过查表来获得。

FPGA芯片内部电路结构图如图3所示。整个电路需要控制音符频率和节奏频率的两个时钟,可以通过对12MHz进行分频

得来。这里音符频率我们用12MHz，而节奏频率用4Hz。首先，FPGA芯片查询开关键。如果开关键为高电平，则音符存储电路模块工作。这时可以通过一个二进制加法计数器来查询乐谱存储器的地址，获得音符。加法计数器的频率即为节奏频率，在每一个计数值上停留0.25秒。如果开关键为低电平，则琴键音符查表电路模块开始工作。琴键音符查表电路模块读取琴键产生的Key[14..0]来进行查表取出音符。

取出来的音符通过音符频率对应关系即音符频率查表电路模块可以得到它所对应的频率及其相关的分频系数和计数初值。这样，通过分频器驱动频率产生电路就可以将12MHz的频率分出不同的频率来驱动扬声器发声。这就是电子琴电路的工作过程。

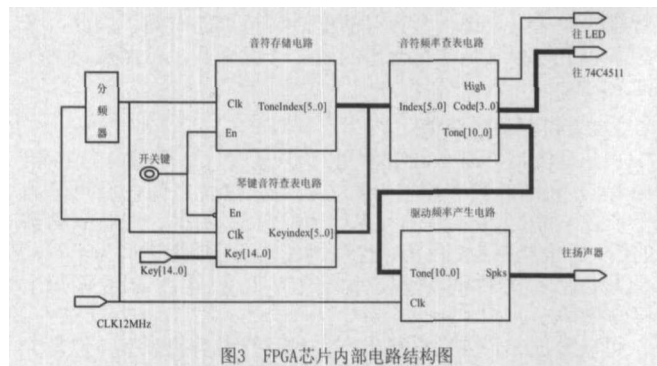


图3 FPGA芯片内部电路结构图

下面我们再讨论各个模块的实现方法：

3.1 分频器模块

这是整个系统的第一个分频器，主要的目的是获得4Hz的频率用来产生最小音乐节拍，通过改变这个分频器的分频系数就可以改变最小音乐节拍。这里要注意的是，如果改变了最小音乐节拍，则音符存储电路中音符的存储以及琴键音符查表电路中琴键节奏的读取都要做相应的改变。

3.2 音符存储电路模块

模块音符存储电路是音乐自动播放中的乐谱查表和音乐节拍控制电路，它决定了当前输出哪一个音符以及该音符的持续时间。可以在其中设置了一个二进制加法计数器作为音符ROM的地址发生器，这个计数器的位数与用来记录音符的ROM空间有关，如果用了256个ROM单元，则位数就是8；计数频率就是通过12MHz分频过来的4Hz。这样在每一计数值上的停留时间为1/4秒，也就是一个最小音乐节拍。若当前音符的节奏为n/4拍，在ROM中存储n次，也就是在n个计数值上输出该音符。这样，随着音符存储电路中的计数器按4Hz的时钟速率作加法计数时，梁祝乐曲就自然演奏起来了。

音符存储电路内部有一个存储器MUSIC用来存储乐谱，我们可以通过Quartus II在芯片上定制一个ROM，这样就方便我们存储多首歌曲以及查表，然后通过元件例化语句进行调用。具体的操作步骤如下：

1. 建立ROM数据文件。在File菜单中选择“New”，点击“Other files”项，并选中“Memory Initialization File”，点击OK后产生ROM数据文件大小选择窗。ROM的数据数Number为256，数据宽Word size取4位。点击“OK”，将出现如图4的mif数据表格，表格中的数据为10进制表达方式，数据对应的地址为左列于行行数之和。将波形数据填入此表中，完成后在File菜单中点击“Save as”，保存此数据文件，在这里不妨取名为data1.mif。

2. 定制ROM元件件（MUSIC.VHD）。在Tools菜单中选择“MegaWizard Plug-In Manager”项，选择“Create a new custom...”项，即定制一个新的模块。点击“Next”后，产生对话框，在左栏选择“Storage”项下的LPM_ROM，再选“cyclone”器件和VHDL语言方式，最后键入ROM文件存放的路径和文件名，点击“Next”。接下来是要选择ROM控制线和地址、数据线。在图5所示的对话框中设置地址与数据的位宽分别为256和4，选择地址锁存控制

信号，点击“Next”，然后选择数据文件data1.mif。最后完成ROM文件MUSIC.VHD的生成。打开此文件MUSIC.VHD就可以看到其中调用初始化数据的语句。

Addr	+0	+1	+2	+3	+4	+5	+6	+7
0	0	3	3	3	5	5	5	8
8	8	8	8	8	8	8	5	5
16	12	12	12	15	13	12	10	12
24	9	8	9	9	9	9	9	12
32	9	9	9	10	7	7	8	8
40	5	5	5	5	8	8	8	8
48	3	3	8	8	8	5	8	8
56	5	5	5	5	5	5	5	5
64	10	10	10	12	7	7	9	9
72	8	8	5	5	5	5	5	5
80	3	5	3	3	5	8	7	9
88	8	8	8	8	8	8	5	8
96	8	8	8	8	12	12	12	10
104	9	8	10	9	8	8	8	5
112	3	3	3	3	8	8	8	8

图4 将波形数据填入mif文件表中

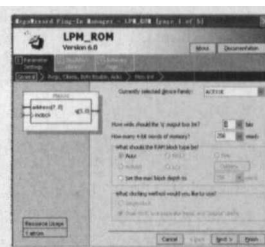


图5 选择模块数据线和地址线宽度

3.3 琴键音符查表模块和音符频率查表电路模块

琴键音符查表电路模块和音符频率查表电路模块都是查表电路，电路的结构都是一样的。不同的是琴键音符查表电路模块是琴键与所对应音符的查表电路，其功能是通过琴键找到相应的音符后输出给音符频率查表电路模块；而模块音符频率查表电路是乐曲简谱码对应的分频预置数查表电路。这里通过模块音符频率查表电路来进行说明。

模块音符频率查表电路中设置了全部音符所对应的分频预置数。其功能就是为驱动频率产生电路模块提供发音的分频预置数。每一音符的停留时间由音乐节拍和音调发生器模块音符存储电路的clk的输入频率决定。音符频率查表电路的输出由对应的6位输入确定，而Index[5..0]记录36个音符（高中低音各有12个音符）。音符频率查表电路的输入值Index[5..0]与持续的时间由琴键音符查表电路模块或音符存储电路模块的输出值决定。其内部结构图如图6：

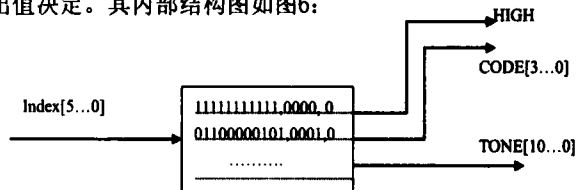


图6 音符频率查表电路模块内部电路结构

由于音符频率查表电路是一个查表电路，它所对应的分频预置数是固定的，因此可以通过case语句来实现。同时该模块还可以输出乐谱显示时需要的数值。

3.4 驱动频率产生电路模块

驱动频率产生电路模块是一个可预置数的数控分频器，用来获得各个音符的频率。其CLK12MHz端输入一个频率为12MHz的时钟信号，通过驱动频率产生电路分频后由SPKOUT输出。驱动频率产生电路对CLK12MHz输入信号的分频比由11位TONE[10...0]决定，SPKOUT的输出频率就有了一一对应关系。

驱动频率产生电路模块可由三个分频器串联形成，前面的为分频系数12的固定分频器，12MHz的频率经它分频后变为1MHz。第二个的分频系数由分频预置数Tone确定（Tone就是从音符频率表中取出的数值）。最后一个分频器为2分频器。经过第一个分频器，12MHz的频率被分成1MHz，然后经过第二个分频器后形成与每一个音符频率两倍的频率，最后通过第三个分频器就成了能过驱动扬声器的方波。其结构框图如图7。

其中，2048这个数值不是固定的，由音符频率所对应的分频系数的最大值决定。比如说，这个最大值能够用11位二进制表示，这时这个数值就是211=2048。

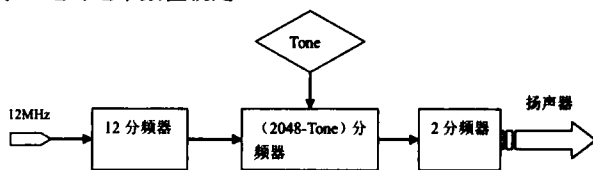


图7 驱动频率产生电路A模块内部电路结构

4. 结束语

本设计采用VHDL语言层次化和模块化的设计方法,实现了基于VHDL的电子琴的设计。通过仿真和下载可以看到,系统能够通过琴键演奏乐曲和按键选择播放指定乐曲,具有可移植性和可扩展性。而且不必变化顶层文件即可随意变更或增加乐曲,更好地优化了电子琴数字电路的设计,提高了设计的灵活性,使整个系统可集成在一个芯片上,体积小、功耗低、可靠性高,能够达到设计要求。

参考文献:

- [1] 潘松,黄继业.EDA技术实用教程(第二版)[M].科学出版社,2004.
- [2] 潘松,黄继业,曾毓.SOPC技术实用教程[M].清华大学出版社,2005.
- [3] 杨碧波,刘吉.基于FPGA的数字频率计[J].
电子测量技术,2006 Vol.29 No.4 P.85-86
- [4] 陈燕东,李颖.可编程器件EDA应用开发技术[M].
北京:国防工业出版社,2005.
- [5] 樊国梁,张晓燕.基于VHDL的键盘扫描及显示电路设计[J].
电子世界,2005 No.2, P.25-26
- [6] 潘晓利,陈学煌,刘永志.基于MIDI模块的音乐发生器设计.[J].
电子测量技术,2007 Vol.30 No.2 P.108-110
- [7] 吴晨光,彭安金,王奔.SVPWM信号发生器的VHDL实现.[J].
电子设计应用,2006 No. P.69-71