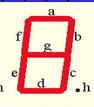
数电实验综合设计实验报告

1. 实验课题

在电路板上实现节奏大师

1. 使用说明

* 将数码管横过来，让其依次根据音乐节奏掉落不同按键，在最右边的数码管上设置判定线，当按键落到判定线上时按下对应的按键
* 按键分为三种，长键，短键和拐弯键位，短键按一下即可，长键需要按着不动直到长键结束，拐弯键位在长键的末端需要同时按压对应的两种键位
* 

如图，长键只会出现在a和d键位上，短键只会出现在b和c号键位上，拐弯键会出现在长键末端类似于de亮，af亮这种，比如de亮，则需要在d结束的时候同时按压短键位，用四个按钮对应四个键位，两个长键位两个短键位，左长键btnl，右长键btnu，左短键btnd，右短键btnr，中间btns属于开始或复位信号

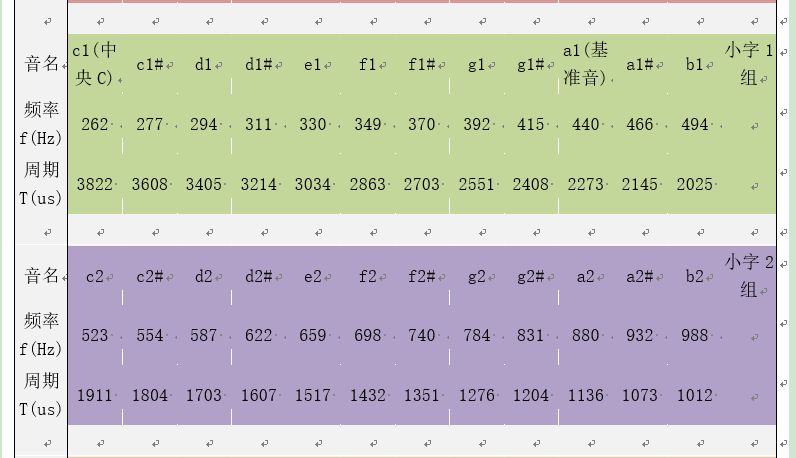
* 对按键的判定有三个级别，短键按压时间差小于0.1s判定为perfect，加50分，时间差大于0.1s但是小于3个timeunit判定为great，加25分，按下时间迟于应当按下的时间判定为miss，不加分。对于长键的判定是根据按下的有效时间，可以提前按按键，但是不计入有效时间，可以提前结束按键，但是不会得全分，根据长键长度，每长占50分，按比例算分。
* 判定等级实时显示在三个LED灯上，perfect亮三盏灯，great两盏，miss一盏。
* 分数按等级显示在后5个LED灯上，初始为0分，不亮灯，否则小于100分亮一盏灯，到300分两盏，700分三盏，1000分4盏，2000分5盏。
* 接外设蜂鸣器，根据按键节奏同步播放音乐。设计中的音乐为《世上只有妈妈好》
* 设计中有4个开关，低位三个开关供调速，开关0置1为低速，开关1为中速，开关2为高速，其余开关组合为变态倍速（基本没有办法玩），并且调速之后音乐的速度也会随之改变。开关3为暂停，置1时暂停，音乐也随之暂停，置0时继续游戏，音乐也继续

1. 代码解释

Cntms 起系统时间的作用，记录的是复位信号按下到现在经过的毫秒数，start复位，stop暂停

蜂鸣器播放音乐的模块music.v

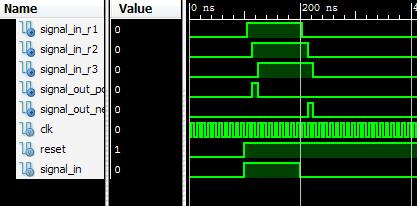
这首歌按照每小节8拍算，每拍一个timeunit，用cnt1作基频。



由频率计算开始值origin，然后计数到统一的32767，再从origin开始，用cnt2改变origin相当于改变音调。即先用节拍频率把演奏时间划分成一段一段，再根据不同的音调在每一段内部继续划分确定音调。

按键信号转化为脉冲信号的模块signal\_generate.v，利用信号的移位延迟以及与或操作实现

仿真：



输出数码管信号的模块Mom.v

从复位信号开始计时，没过一个timeunit将一个Y信号输出。写法是将根据谱子的节拍，每个音调确定相应的timeunit数（比如长音3拍，持续三个timeunit），然后设计在长键键位上，再根据同时期的短音等确定该timeunit的数码管需要亮哪几段，输出Y以及使能OE。先是在Y变化后使OElast变化，再变化OE，相当于延时一个clock，才能让OE输出当前Y。

判定模块judge.v

对应于四个键位，建立四个队列queue，四个持续或延迟时间数组tdist，四个状态机state，对于长按键有拐弯键的还有建立两个turn数组。

调用了signal\_generate模块转化为8个脉冲信号（4key上升下降），用btn\_pos\_p(pulse)的wire类型输出脉冲，用btn\_pos\_r保存信号。

首先说明设计中的一个难以解决的问题：长按键不得超过4拍。因为长按键需要记录该长按键的全部长度，然后清空队列，然而按键有可能会提前放开，但是这时还不知道长键什么时候结束，但是放开按键就需要判定并进行清队列了，所以只能规定长键不超过4拍保证该长键在判定时长度已知。根据这种情况在四个数码管中同时出现的长键信号不会超过两个，因此有后面的处理。

每次接受到OE的脉冲，分析哪些亮并且插入队列，由于接受到Y和Y到达判定线被击中有3个timeunit’的延迟，所以插入队列时插入的是当前系统时间+3\*timeunit，表示标准应该被击中的时间，短键的queue也是此功能，但是长键的queue【0】，【2】记录的是时间，而【1】【3】记录的分别是【0】【2】按键的持续长度，如果接受到0，表示该长键已结束，准备计数下一长键的长度，但是接受到1有两种情况，该长键在继续计数或者该长键已计数完毕，此长键为下一长键的第一个。所以讲长键队列的queue【4】作为状态，0表示正在为第一长键计数，1表示正在为第二长键计数。

长键（1、4）按下，用sttime1 、4记录按下时的系统时间。如果结算时如果晚于标准时间则为miss。

短键比较好处理，其queue存储4个按下时的系统时间并且与标准时间作比较。

还有是queue多设置了一个数，因为用for语句时非要报溢出，虽然没有访问溢出的那一位数

关于状态机的注释和其他注释写在源代码中。

还有一点是关于拐弯按键的检测，检测和删除队里只能放在短按键被按下时。规定拐弯按键的操作方法是按住长按键并在长按键的尽头按下对应的短按键，所以短按键被按下同时长按键被按下同时拐弯队列中有值则判定拐弯按键成立，单独清队列并且为拐弯部分加50分，非拐弯部分在长键判定中单独计算

数码管显示模块display.v 就是用分频显示