# 用Seeeduino XIAO做一个特效钢琴灯条

## 1 项目背景

网友老徐最近入了一架某品牌的智能钢琴，但是该智能钢琴的亮灯功能还是比不上另一类MIDI乐器——Launchpad。于是他先用Unity引擎原创开发了一个酷炫的特效钢琴程序。虽然现实世界里不能耍酷，虚拟世界里耍一下特效还是能做到的（虚拟世界里的测试视频见<https://www.bilibili.com/video/BV1KE411y76H/>）。下面这个截图是从老徐的程序实时跑的过程中捕获的，可以看出特效已经很花哨了。



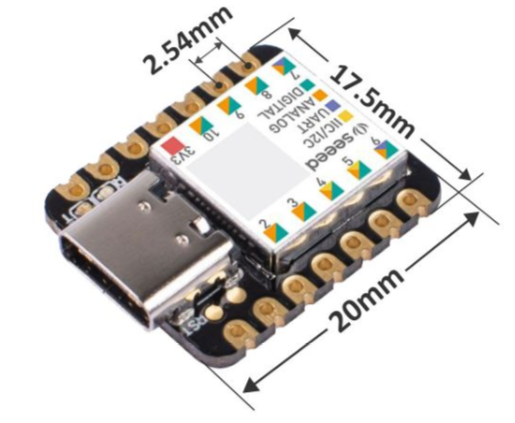
虚拟世界里面能炫特效以后，老徐还觉得不过瘾，于是又把launchpad上最具亮点的RGB LED给加到琴上了。这下总算打破了次元壁，能在现实世界里耍宝了（现实世界的测试视频见<https://www.bilibili.com/video/BV1NC4y1h7Ha/>）。下图的截图序列就是来自老徐的现场演奏。

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

这里最重要的就是那个特效钢琴灯条。今天的这篇文章里，将会介绍一下这个特效钢琴灯带的制作方法。

## 2 材料准备

机缘巧合下，老徐得到了一块由开源社区SeeedStudio 提供的XIAO开发板，那么我们就基于这个小板子来做。XIAO开发板基于SAMD21G18单片机，内核是ARM Cortex M0的，32位，主频48MHz，带有256KB Flash和32KB SRAM。尺寸只有20毫米X17.5毫米，真的是非常非常迷你。

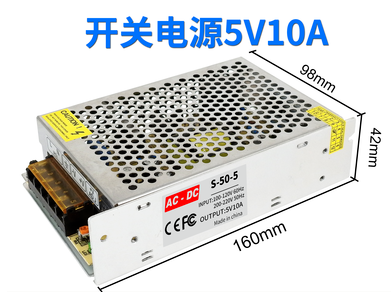


我们的项目是特效钢琴的灯条。既然是要点灯，LED灯带肯定是必不可少的。你还需要准备WS2812灯带。最低规格的每米30灯的灯条用在钢琴键盘上，因为间距太宽会有漏灯。间距更小，位置分辨率更高的144灯，可以获得更好的效果。一般不会把两个相邻的半音同时按下，所以老徐选用的是60灯每米的灯条。



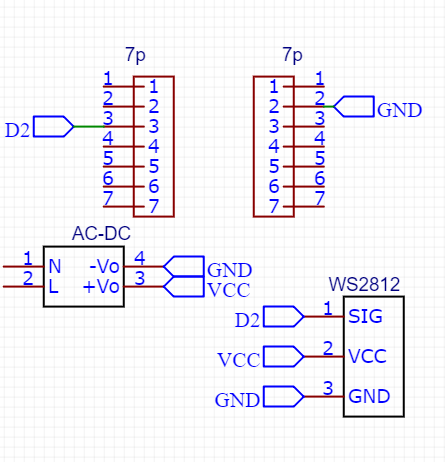
我们来计算一下，琴键每组有12个半音，琴键每组距离164毫米；换算下来，每米73.17个灯就可以保证每个音都能对应一个灯。实际情况中，我们弹流行音乐，这164毫米的一组琴键里只会用到7个音；再换算一下，如果需要给每个音分配一个灯，每米需要42.68灯。这也就是为什么我给你推荐60灯每米的原因。

你还需要一个5V的AC-DC开关电源来给WS2812灯带供电。



## 3 接线原理图

XIAO开发板的USB接到电脑；XIAO的GND和D2分别接到WS2812灯带的SIG和GND；WS2812灯带的GND和VCC分别接到AC-DC开关电源的V-和V+。聪明的你应该已经发现，XIAO的GND，通过WS2812，与开关电源的负极输出连在一起了。因为都是数字信号，倒是不太怕环地和共模干扰。当然，如果你心情好，还可以给D2和SIG之间加个高速光耦隔离，这样就不用把XIAO开发板与WS2812共地了。原理图如下所示，其中两个7P端口正是XIAO开发板。



## 4 程序代码的说明

程序代码开源在github站点，地址<https://github.com/xukejing/NeonPiano>

你能在SeeedStudioXiao目录里找到一个seeedxiao.ino文件，用arduino编译并烧写到XIAO开发板上便可。

如果你想要在我的代码的基础上做二次开发，或调整键盘和灯带参数，建议继续阅读下面我为您写的代码说明。

从18行到22行，有一堆宏，分别定义LED数量、琴键键盘的按键数、键盘最左边琴键的音符号、键盘最右边琴键的音符号、用于控制灯带的信号引脚编号。

第30行，float radius=8.33，这是个定义影响半径的参数。当一个琴键被按下或抬起时，这个琴键“附近”的LED会做出相应。怎么表示这个“附近”的大小呢，于是我设计了这个radius变量。其中值8.33是每米60灯灯条的LED间距的一半，单位毫米。在这个radius值下，理论上，每个琴键对应1个LED。如果你希望一每琴键能多点亮几个临近的LED，可以设置radius值为16.67或更大。

第35行的NoteOn函数用于处理琴键按下时候的灯条控制变量。我是这样设计的，当leds[i] 等于1时候，把对应i编号的LED点亮；当leds[i]等于0时，对应i编号的LED熄灭。第48到51行，是计算灯与按键的距离的绝对值，然后把这个值与radius的比较，确定哪些灯在影响范围内。在影响范围内就赋值1。另外，为了让灯光颜色能像霓虹灯一样变化，我设计了一个变量pixelHue用来控制色相变化，这个变量值域0~65535，它会在每次按键按下时被加256。每次加256的设计意图是这样的，按键每按（65535/256）次为一个颜色变化循环周期，换算下来刚好每按256次按键颜色换一轮。如果你想要颜色变化再快一点，可以每次按键按下时把变量pixelHue加1024。反之，如果要颜色变慢点，就每次少加点，比如加64，或32。

第61行的NoteOff函数用于处理琴键松开时候的灯条控制变量，其中leds[i]的说明见上小节。

setup函数中的第93行到95行，开机时候把所有灯都设置为灭的状态。这里我实际设置的并不是完全熄灭，而是颜色0x0A0A0A，处于微微亮的状态。

loop函数里第104行到107，根据leds[i]决定第i个led亮还是灭。灭的话，给个0x0A0A0A的颜色让它微微亮。亮的话，设置颜色为strip.gamma32(strip.ColorHSV(pixelHue)，其中pixelHue的定义看上上上小节。

最后一个myinit函数，从111行开始。我假设灯带的第一个灯为坐标参考系原点，即x=0毫米。我把第一个灯布置到最左边第一个琴键更左点的位置。我的第一个琴键对应的位置实际是第二灯所在坐标，即x=16.66毫米的位置。后面114到124的参数不要改，这是计算标准琴键各个键距离的算法，改了反而就错了。第129行是各个LED灯的坐标的计算方法，这个与灯带型号有关，我的是每米60灯，所以每次加（1000毫米除以60等于16.66666毫米）。

## 5 灯条使用方法

现在我们要把电钢琴的MIDI信号给灯条控制器XIAO，怎么做呢？这里我给大家推荐个免费的软件，叫MIDI-OX，它除了可以做MIDI信号分析，还可以用于多个MIDI设备的信号转发。演奏时，电钢琴把MIDI信号发送给电脑上的MIDI-OX，然后MIDI-OX把信号发给XIAO开发板。

其实，能实现MIDI信号转发的软件有很多，大多数带MIDI输入输出的谱曲软件或音乐制作软件都附带这个功能了，比如Cubase、FL Studio、Ableton Live等。

最后一个问题，也许大家是想问，琴怎么弹？或者，小白怎么弹出一些不难听的旋律呢？老徐要给你传授压箱底的划水绝招了。

第一步：从右往左（从高音到低音）两只手像撸猫一样连续刮揍黑键，每次刮过6到7个黑键，先右手，后左手，再右手。。。（技巧：每次换手时候，后手比前手的最后一个键位更靠右一点点，也就是说把一部分刮揍区域重叠）。

第二步：当向左到达黑键最低音位置以后，用右手把黑键从左往右（从低音到高音）一气呵成刮一遍，匀速刮上去。好了，你现在学会制作花果山水帘洞风格的音乐了。