

实习报告

标题：音游大师

作者：刘臻、孙佳芮

摘要：本作品利用 micro:bit 制作了模拟打击类音乐游戏的互动开源硬件，创造性地结合了 micro:bit 上的磁力计与加速度传感器来确定，给穿戴者带来虚空弹琴的体验。

一、选题及创意介绍

组内两个成员都是生科专业，同时对音乐较为感兴趣，在选题时主要考虑这两个方向。考虑到 micro:bit 自带传感器精度有限，最终选择制作音乐相关的互动硬件设备。打击类音乐游戏（俗称音游）是现代年轻人非常喜欢的一类平板/手机游戏，游戏通常以点击的方式互动，跟随音乐节奏，在对应标志到达判定线上时进行点击、长按等操作，部分音游根据鼓点节奏制作乐谱，还有一些根据音调制作。我们受此启发，发现 micro:bit 上的磁力传感器和加速度传感器可以作为信号输入，通过检测操作者的手部动作实现“虚空音游”，使用时只需带上固定好 micro:bit 的手套，即可在空中弹出不同的音调，结合投影、VR 等设备可以达到更好的游戏体验。

二、设计方案和硬件连接

我们设计用 micro:bit 中的加速器（accelerometer）模块判定左右手，磁力传感器（compass）模块判断单片机实体的朝向与相应音符进行一一对应，再通过加速度模块判定单片机朝向判定发生点从而进行乐器的模拟演奏。通过与手套结合可以更方便地模拟虚空拍击功能，同时给使用者更好的游戏体验。

三、实现方案及代码分析

● 实现方案及程序特点：

- 开始-欢迎环节
- 准备校正环节

- 1) 左右手：通过加速器模块检验单片机放置的朝向，使得当玩家掌心相对前平举时可确认左右手佩戴方式
- 2) 玩家面向：通过磁力传感器模块确认玩家游玩时的初始面向（范围大致为东南方向（ $135^{\circ} \pm 5^{\circ}$ ）），以便后续游戏时找准音符在虚空中的位置；当单片机指向不在规定范围内时通过现实左右箭头提示校正

● 游戏主体

最终方案通过左右手区分音域（c4,d4,e4,f4 和 g4,a4,b4,c5），当手向下拍击（手腕下压）时通过单片机水平判定发出音域对应的音符从而进行演奏

● 游戏结束

```
1. # Imports go at the top
2. from microbit import *
3. import music
4. import time
5.
6. #original menu
7. flag = 0
```

```

8.      #0:初始界面
9.      #1: 准备(校准) 界面-左右手
10.     #2: 游戏界面
11.     #6: 准备(校准) 界面-方向校准
12.
13.
14. music.play(music.NYAN,wait=False)
15. display.scroll('Welcome to Music Master! Shake to begin calibration!',delay=80,loop=False,wait=True)
16.
17. #通过 shake 手势开始准备环节
18. while flag == 0:
19.     display.show(Image.HAPPY)
20.     if accelerometer.was_gesture('shake'):
21.         flag=1
22.
23. #flag2=0#direction
24. flag3=-1#music
25. flag4=0#左右手
26.
27.
28. while flag==1:#确认所佩戴机器位于左手或右手
29.     if accelerometer.was_gesture('left'):#左手初始判定
30.         display.show(Image.ARROW_W)
31.         flag4=1
32.     elif accelerometer.was_gesture('right'):#右手初始判定
33.         flag4=2
34.         display.show(Image.ARROW_E)
35.     time.sleep(2)#间隔二秒进行二次判定确认所佩戴手
36.     if flag4==1 and accelerometer.was_gesture('left'):
37.         display.show('L')
38.         time.sleep(2)
39.         flag=6
40.     elif flag4==2 and accelerometer.was_gesture('right'):
41.         display.show('R')
42.         time.sleep(2)
43.         flag=6
44.
45.
46. while flag==6:
47.     if 130<=compass.heading() <=140:#初始位于 135°±5°范围内—一次判定
48.         flag2=0
49.         if flag2+flag3!=0:
50.             flag2=0
51.             flag3=0

```

```

52.         music.play(music.BA_DING,wait=False)
53.     display.show(Image.HEART)
54.     time.sleep(2)
55.     if 130<=originPoint<=140:#二次判定
56.         flag2=1
57.         music.play(music.ODE,wait=False)#判定成功提示音
58.         display.scroll('Ready! Raise your hands to Play',delay=80,loop=False,wait=True)#举起双手
-准备开始演奏
59.         while flag == 6:
60.             display.show(Image.HAPPY)
61.             time.sleep(1)
62.             display.show(Image('00900:'
63.                 '09000:'
64.                 '99999:'
65.                 '09000:'
66.                 '00900'))
67.             if accelerometer.was_gesture('up'):
68.                 flag=2#激活成功
69.                 display.show(Image.ARROW_N)
70.                 music.play(music.BA_DING)
71.         else: #增加背景音乐和旋转提示
72.             flag2=1
73.             if flag2+flag3!=2:
74.                 flag2=1
75.                 flag3=1
76.                 music.play(music.BLUES,loop=True,wait=False)
77.             if 0< compass.heading() < 130 or 315<compass.heading()<360:#
78.                 display.show(Image('00900:'
79.                     '00090:'
80.                     '99999:'
81.                     '00090:'
82.                     '00900'))
83.             elif 140 < compass.heading() < 315:
84.                 display.show(Image('00900:'
85.                     '09000:'
86.                     '99999:'
87.                     '09000:'
88.                     '00900'))
89.
90.
91. dic={0:['c'],1:['d'],2:['e'],3:['f'],4:['g'],5:['a'],6:['b'],7:['c5']}#音阶字典
92. flag6=0#标记打击状态
93. while flag==2:
94.     display.show(Image.HAPPY)

```

```

95.     sleep(400)
96.     if accelerometer.was_gesture('face up'):
97.         flag6=1
98.     if flag6:
99.         if 0 < compass.heading() < 270:#音符激发范围为 0°-270°
100.             rg=int((compass.heading())/270*4+4)%4+(flag4-1)*4
101.             #int((compass.heading())/270*4+4)%4 表示四分区,
102.             #(flag4-1)*4 为左右手区分
103.             music.play(dic[rg])
104.         flag6=0
105.     if pin_logo.is_touched():#当 logo 被触摸时游戏暂停
106.         display.scroll("Paused. Press A to Continue Press B to Quit",delay=80,loop=True,wait=False)
107.         flag5=1
108.         while flag5:
109.             if button_a.was_pressed():#按下 A 键游戏继续
110.                 flag=2
111.                 flag5=0
112.             elif button_b.was_pressed():#按下 B 键游戏结束
113.                 flag=0
114.                 flag5=0
115.                 display.show(Image.HAPPY)
116.                 time.sleep(2)
117.                 display.scroll("Goodbye!",delay=80,loop=True,wait=False)
118.                 music.play(music.PYTHON)

```

●历史版本（游戏主体~探索过程）

version1: 通过检测拍击桌面的分贝检测到达阈值来发出声音

abandoned: 分贝检测难以调整（要么很轻易到达要么很难到达因此难以控制）

```

1. while flag==2:
2.     display.show(Image.HAPPY)
3.     sleep(400)
4.     if microphone.sound_level(>60:#待定
5.         flag4=1
6.     if flag4:
7.         rg=int(compass.heading()/360*7)%7
8.         music.play([dic[rg]])
9.         flag4=0

```

version2: 通过单手正反扩大音域（即，正手对应一段音域，反手对应另一段音域）

abandoned: 经过实测和磁场探测器原理分析后发现, 正反手转换过程中内部的小磁针(所猜测的磁场探测器本体) 的指向会随着单片机朝向的反转进行大角度转向, 因而发出声音时

小磁针指向不稳定导致发出的音符错误。

```
1. while flag==2:
2.     display.show(Image.HAPPY)
3.     sleep(400)
4.     # if microphone.sound_level()>40:#待定
5.     if accelerometer.was_gesture('face up'):
6.         flag4=1
7.     elif accelerometer.was_gesture('face down'):
8.         flag4=2
9.     if flag4:
10.        if flag4==1 and 0 < compass.heading() < 180:
11.            display.scroll(compass.heading())
12.            sleep(1000)
13.
14.            rg=int((compass.heading())/180*4+4)%4
15.            music.play(dic[rg])
16.        if flag4==2 and 180 < compass.heading() < 360:
17.            display.scroll(compass.heading())
18.            sleep(1000)
19.
20.            rg=int(360-(compass.heading())/180*4+4)%4
21.            music.play(dic[rg+4])
22.        display.show(Image('00900:'
23.                            '09990:'
24.                            '99999:'
25.                            '09990:'
26.                            '00900'))
27.        flag4=0
```

version3: final ver (如上)

四、后续工作展望

- 1.希望可以优化产品外观结构，外接电池盒并和手套紧密固定；
- 2.希望可以做出来一些谱，并结合谱面加入选曲和自动评分功能；

五、小组分工合作

源代码：

欢迎页&校准：刘臻、孙佳芮

主体：孙佳芮

退出：刘臻

实习报告撰写：刘臻、孙佳芮

作品 Poster 设计：孙佳芮

作品照片拍摄：刘臻

运行和介绍视频：刘臻、孙佳芮