# Report of Musical Windmill 谭凯文

Goal   ¶

The goal of the project is to build a music windmill and program it. The windmill has to play the music according to the music score we wrote. It also need to light up and rotate.

Process

The process of the project can be mainly divided into 2 parts: buling up the windmill, and programming. To build up the windmill, we use "ITTY BITTY CITY" to make a simple windmill with a LED light, a buzzer and a electrical machine one it. Then we use mixly to program it. The program in mixly is as follows. It enables the windmill to play different musical notes, rotate or light up based on the order it receives.
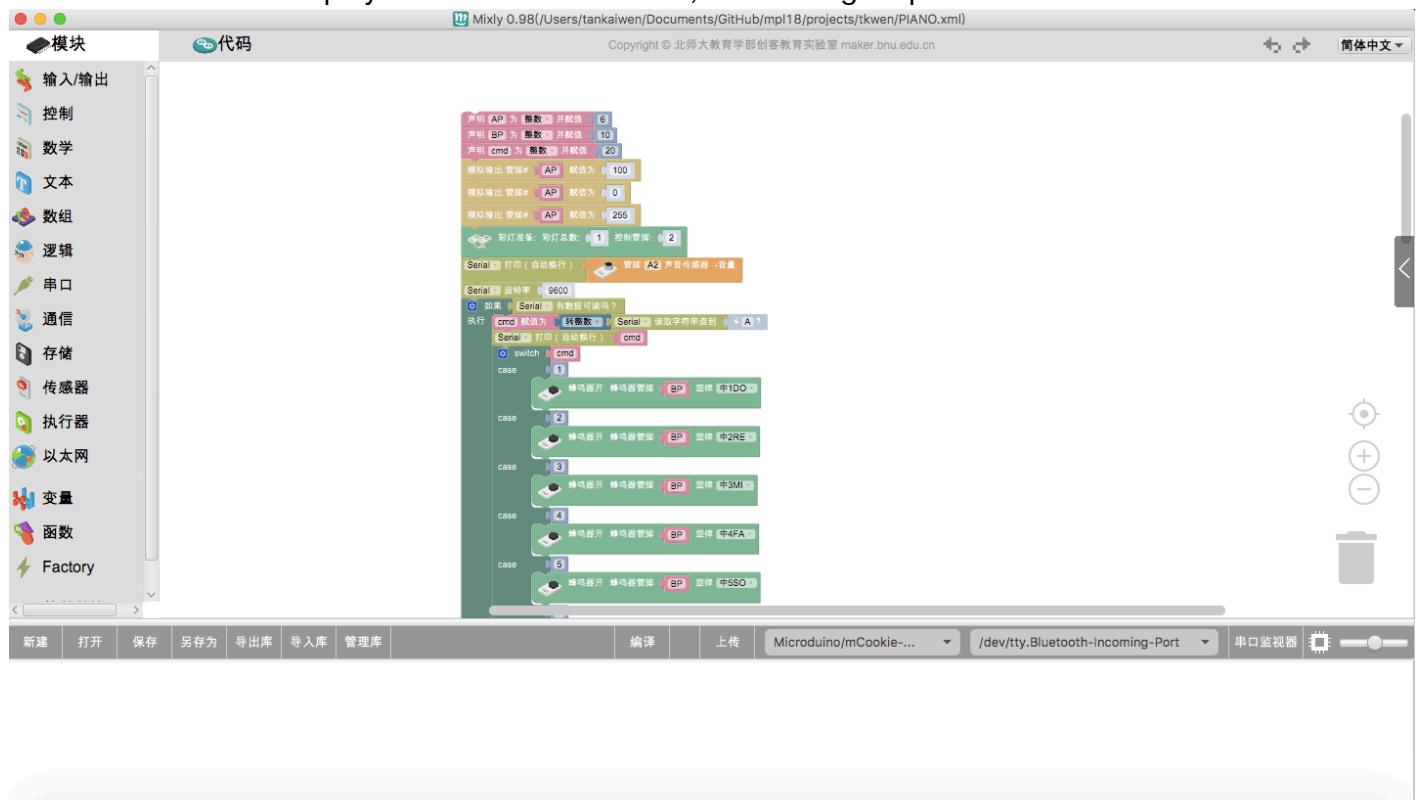

The next step is to write a program in atom. The program need to read the input of both number of song score, but also the song name. In order to do this, we have to use the function of "dictionaty" and "function". Because the code can only connect one number to one song, not one word to one song, so we create a function which turns the input song name into corresponding song number. This process allows us to input both song number and song name to get a some. The program is as follows.

```python
60
61
62  #songs_dictionary={'tinklestar':1,'dadaotuhao':2,'RadetzkyMarsch':3,'xjbsong':4,'clash royale':5}
63
64  song_dic={'tinklestar':1,'dadaotuhao':2,'RadetzkyMarsch':3,'RadetzkyMarsch2':4,'xjbsong':5,'clash royale':6}
65  #ser=serial.Serial(port='COM4')
66  #ser=serial.Serial(port)='/dev/ttymodem542')
67  #ifha;oifhad;oifh
68  def run():
69
70      action = "empty"
71      while action != "q":
72          print ('select 1.input song sequence, number,select 2 , input song namen , q and others for quit')
73          action = input("> ")
74          if action=='1' :
75              print ('select in which song do you want to play:for example: 1,2,3,4,5, q and others for quit')
76              song_number = int(input("> "))
77              print("song number is:")
78              print(song_number)
79              for notes in songs[song_number]:
80                  ser.write(notes.encode())
81                  print ("send:"+notes)
82                  ser.write("A".encode())
83                  print("send:A")
84                  time.sleep(1)
85          elif action == "2":
86              print ('select in which song do you want to play:tinklestar,dadaotuhao,RadetzkyMarsch,xjbsong,clash royale,q and
    others for quit')
87              song_name = input("> ")
88              print("songs name is:")
89              print(song_name)
90              song_number=get_song_dictionary[song_name]
91              print("song number is:")
92              print(song_number)
93              for notes in songs[song_number-1]:
94                  ser.write(notes.encode())
95                  print ("send:"+notes)
96                  time.sleep(1)
97          else :
98              return
99  run()
100
```

```python
1   import serial
2   import serial.tools.list_ports
3   import time
4
5   f = open('mysongs.csv', 'r')
6   data = f.read()
7   rows = data.split('\n')
8   print(rows[0:5])
9
10  print("Hello")
11
12  songs=[]
13  for row in rows:
14      song=row.split(',')
15      songs.append(song)
16  print(songs)
17
18  lines = []
19
20  print("Hello Again")
21  for xxx in songs:
22      lines.append(xxx[0])
23  print(lines[3])
24
25
26
27  def get_song_dictionary(xxxx):
28      dictionary = {}
29      for n in range(len(songs)):
30          song = songs[n]
31          dictionary[song[0]] = n
32      return dictionary
33
34
35
36
37
38  print ('hello')
39  ports = list(serial.tools.list_ports.comports())
40  print (ports)
41  for p in ports:
42      print (p[1])
```

Result

Due to the short of time, we have not build a complete windmill. Instead, we just build a easy model with LED light, buzzer and electricity machine on it. Then we test the program for times till it can work in the right way. The final result is an easy windmill which can play several songs. It can also rotate and light up. But the windmill can only play simple music, which means it can only play one musical note at one time, and all the musical notes last for the same time.

Discussion
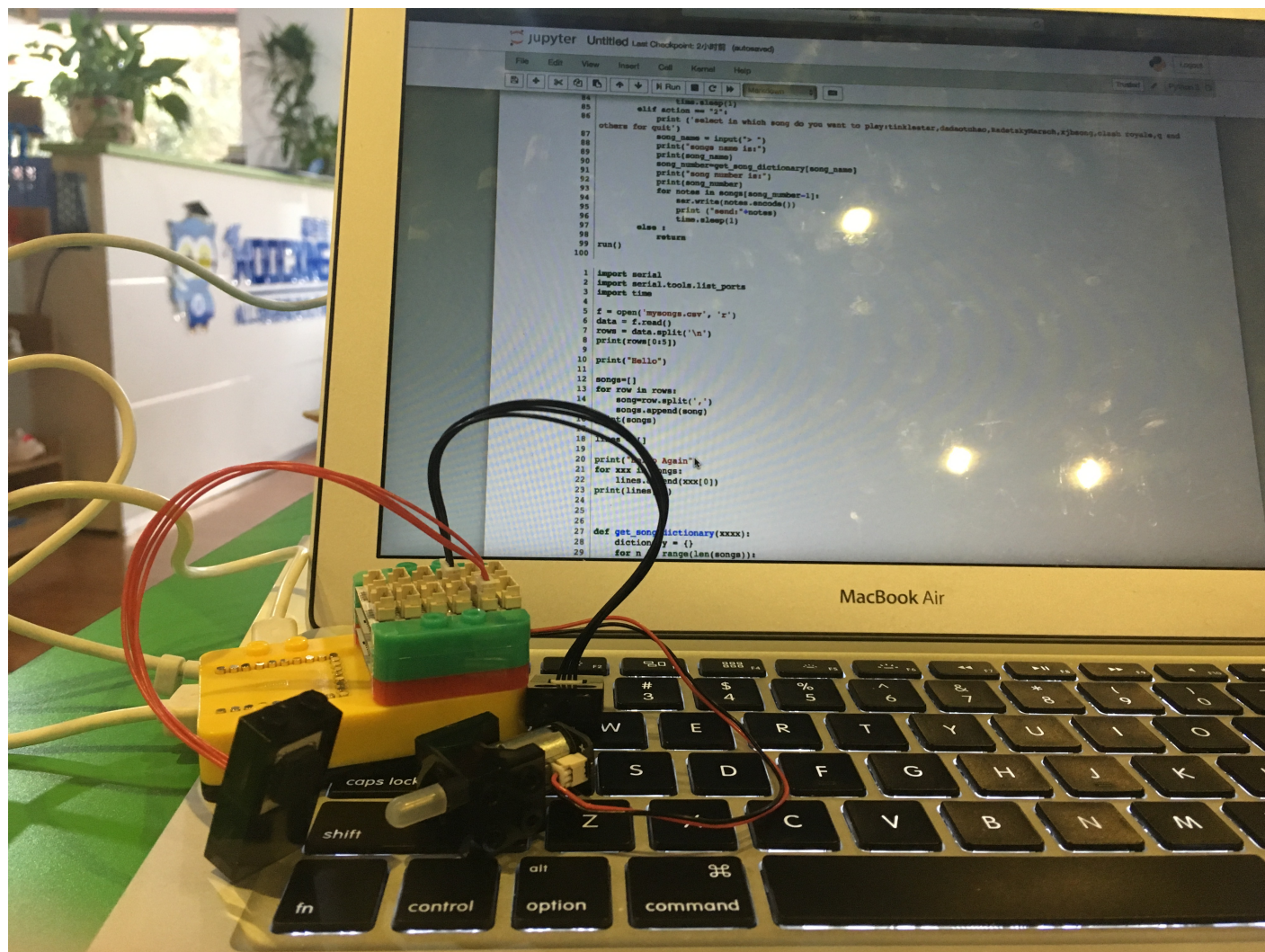
Although the final result does work, it can still improve in some aspect. Firstly, the windmill we build is not complete. Secondly, the songs the windmill play are too simple. By using the function of split, we can actually make the song more complex with different musical note lengthes, but this time, we have not enough time to do this.