

案例：鱼缸自动水位控制

学习目标：

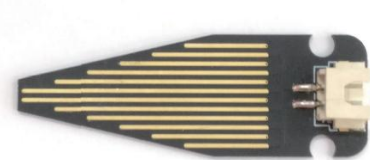
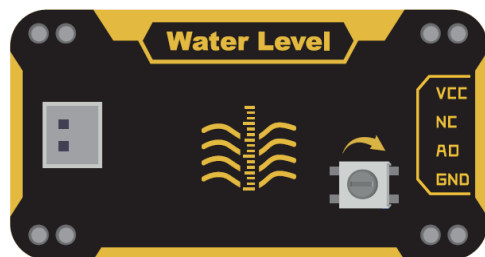
- 1.了解水位传感器和驱动器的使用方法
- 2.通过鱼缸自动水位控制案例，理解物联网中控制与反馈的实现过程
- 3.基于 python 编程，实现利用传感器实时监测鱼缸水位，若低于预设的阈值，便启动水泵工作，使水位上升到预定范围的效果。

器材清单：

① 掌控板；② 通用拓展板；③ 水位传感器；④ 水位探头；⑤ 驱动器；⑥ 4 水泵；⑦ PIN 线*2；⑧ Type-C 数据线。

器材说明：

水位传感器用于检测鱼缸中的水位,当鱼缸水位下降时,传感器输出值将减小,反之将增大。可以使用这个传感器制作一款鱼缸自动水位控制装置。



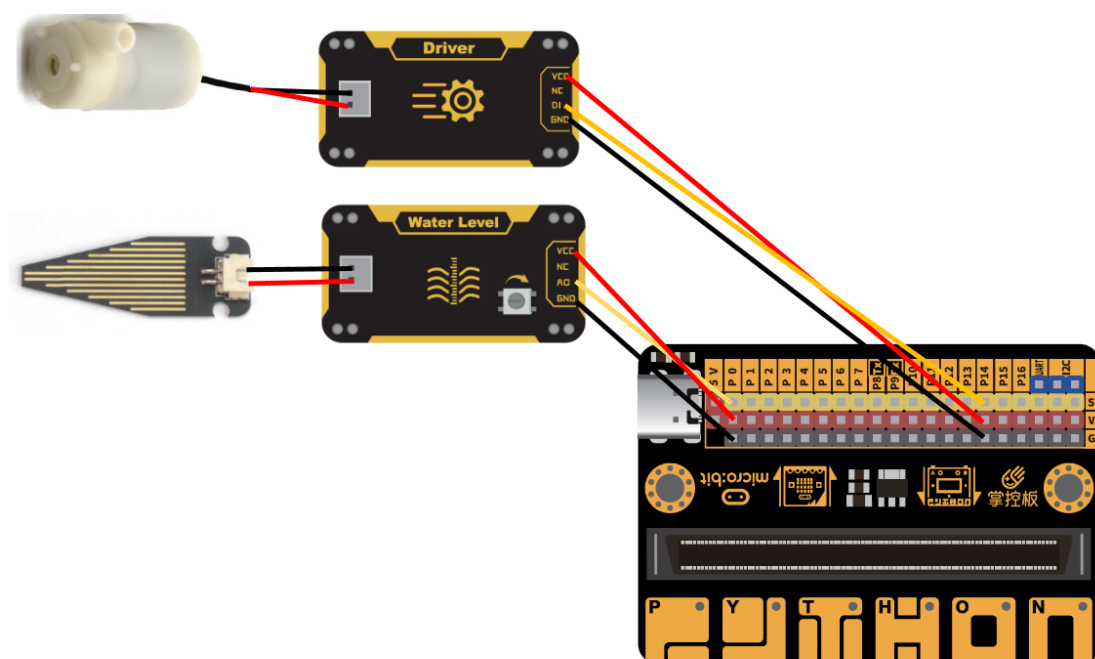
驱动器用于切换开关控制，提供 4PIN 写入接口和 2PIN 输出接口。



操作步骤：

(1) 硬件连接

首先用 4PIN 线将水位传感器连接到通用拓展板 P0 引脚，然后用 4PIN 线将驱动器连接到通用拓展板 p14 引脚，最后按照拓展板的提示将掌控板插入拓展板卡槽，如下图所示：



(2) 程序编写并刷入

在 mPython 软件中，输入如图所示的代码，点击“刷入”按钮，等控制台显示刷入成功，即可测试。当鱼缸中水位传感器返回值低于 3000 时，驱动器写入高电平，接通电源，水泵开始工作，从外面往鱼缸注水。

代码截图

硬件编程

文件

教程

water le... * x

```

1  from mpython import *
2  p0 = MPythonPin(0,PinMode.ANALOG)
3  p14 = MPythonPin(14,PinMode.OUT)
4  while True:
5      oled.fill(0)
6      oled.DispChar('水位值: '+str(p0.read_analog()),0,0,1)
7      oled.show()
8      if p0.read_analog()<3000:
9          p14.write_digital(1)
10         oled.DispChar('水泵开始工作',0,16,1)
11         oled.show()
12     else:
13         p14.write_digital(0)
14         oled.DispChar('水泵不工作',0,32,1)
15         oled.show()
16

```

#导入mpython库

#将0号引脚设置为模拟输入模式

#将14号引脚设置为数字输出模式

#循环

#清屏

#在oled屏上显示当前水位值

#将缓存发送至oled屏显示

#如果水位传感器获取的值小于3000

#14号引脚设置为高电平，接通电源

#在oled屏上显示“水泵开始工作”

#将缓存发送至oled屏显示

#否则

#14号引脚设置为低电平，断开电源

#在oled屏上显示“水泵不工作”

#将缓存发送至oled屏显示

图形化截图

