



Εύρηκα

利用運動傳感器
MPU6050
操控機械手臂

李耀宇

指導老師 賴奕帆

指導教授 李尉彰



李尉彰教授

- Ph.D., Electrical Engineering & Computer Sciences, University of California, Berkeley, 2015
- M.S., Graduate Institute of Electronics Engineering, National Taiwan University, 2005
- B.S., Electrical Engineering, National Taiwan University, 2003



李尉彰教授

- Integrated Micro/Nanoelectromechanical Resonators & Resonant Switches
- Zero Quiescent Power Radio
- Ultra-Low Power Sensing Technologies
- Advanced Materials & Micro/Nanofabrication Technologies for High-Q MEMS
- Sensing System for Smart Manufacturing



Εύρηκα

研究動機

- 機械手臂代勞
- 操作方式
 - 往往以按鍵操控
- 與手臂同步運轉



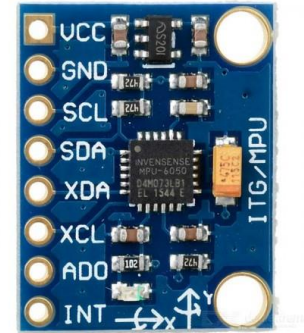
研究目的

- 利用MPU6050求出手之姿態
- 令機械手臂Tinkerkits Braccio與手同步運動
- 自製一臺六軸機械手臂
- 進一步設計一款特殊用途的機械手臂



研究設備及器材

- 六軸傳感器 MPU6050
- 機械手臂 Tinkerkit Braccio
- Arduino UNO 開發板、杜邦線、漆包線
- 電腦
 - (Python 、 VPython 、 Arduino)



研究方法：利用MPU6050求得當前姿態

（一）旋轉角度大小

由角速度 ω ，以及間隔時間 Δt ，即可推算其旋轉的角度大小：

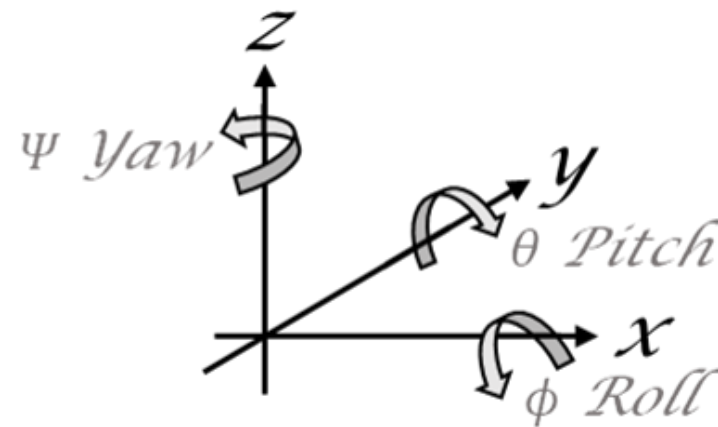
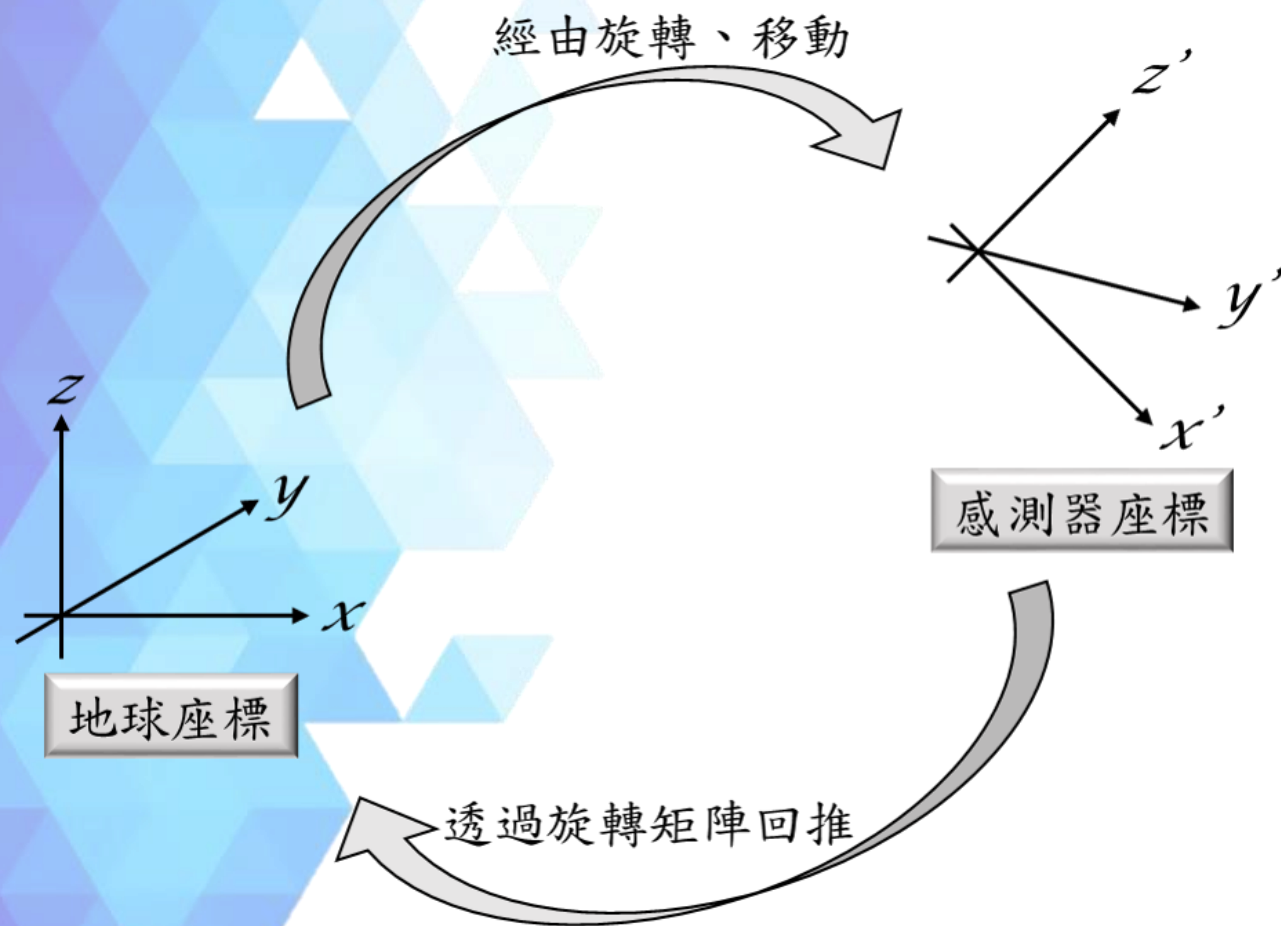
$$\Delta\theta = \omega \times \Delta t$$

（二）旋轉方向與座標轉換

透過旋轉矩陣 M ，推知各時刻運動傳感器回傳的地球座標下的加速度和角速度



研究方法：利用MPU6050求得當前姿態



$$\begin{cases} R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \\ P = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \\ Y = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{cases}$$

研究方法：利用MPU6050求得當前姿態

(一) 旋轉角度大小

由角速度 ω ，以及間隔時間 Δt ，即可推算其旋轉的角度大小：

$$\Delta\theta = \omega \times \Delta t$$

(三) Arduino程式控制

以[4] MPU6050_raw.ino為基礎，在部分輸出方面稍加修改，作為Arduino UNO控制MPU6050的程式

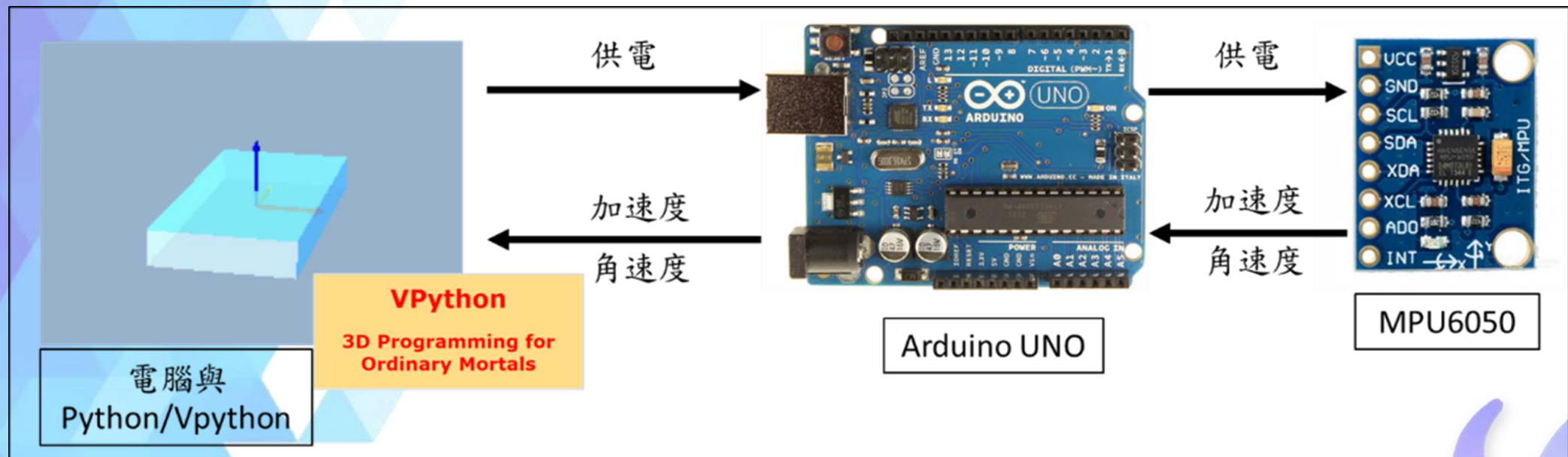
(二) 旋轉方向與座標轉換

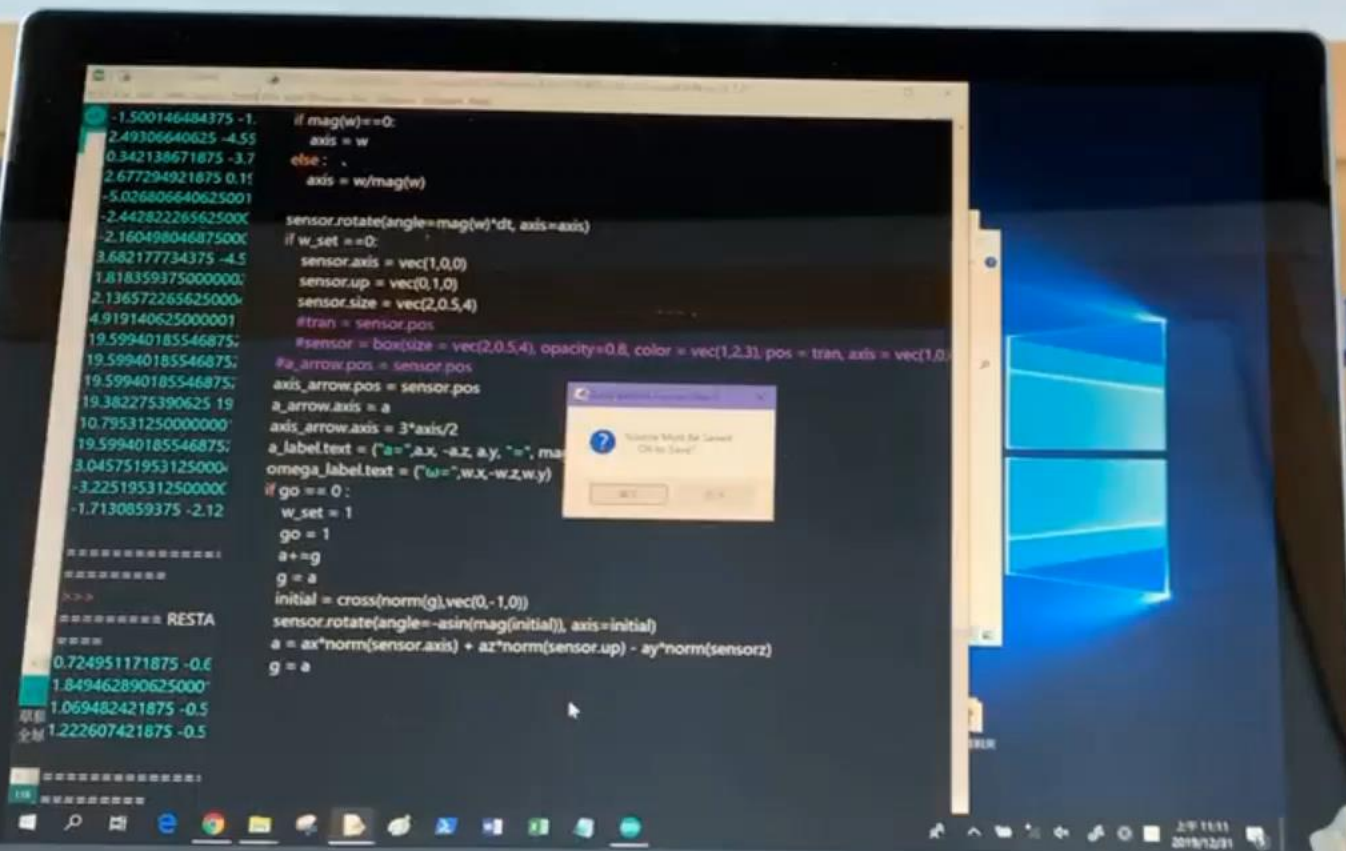
透過旋轉矩陣M，推知各時刻運動傳感器回傳的地球座標下的加速度和角速度

(四) Python處理數據

引入Vpython視覺化繪圖模組，在螢幕上畫出大小適當的模型，依照回傳的資訊，同步繪出物體在空間中的姿態

研究方法：利用MPU6050求得當前姿態





```
-1.500146484375 -1.
2.49306640625 -4.55
0.342138671875 -3.7
2.677294921875 0.1
-5.026806640625001
-2.442822265625000
-2.160498046875000
3.682177734375 -4.5
1.818359375000000
2.136572265625000
4.919140625000001
19.59940185546875
19.59940185546875
19.59940185546875
19.382275390625 19
10.795312500000000
19.59940185546875
3.045751953125000
-3.225195312500000
-1.7130859375 -2.12

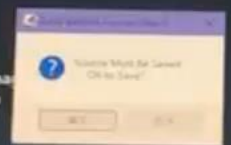
=====
=====
===== RESTA
=====
0.724951171875 -0.6
1.849462890625000
1.069482421875 -0.5
1.222607421875 -0.5

=====
=====
```

```
if mag(w)==0:
    axis = w
else:
    axis = w/mag(w)

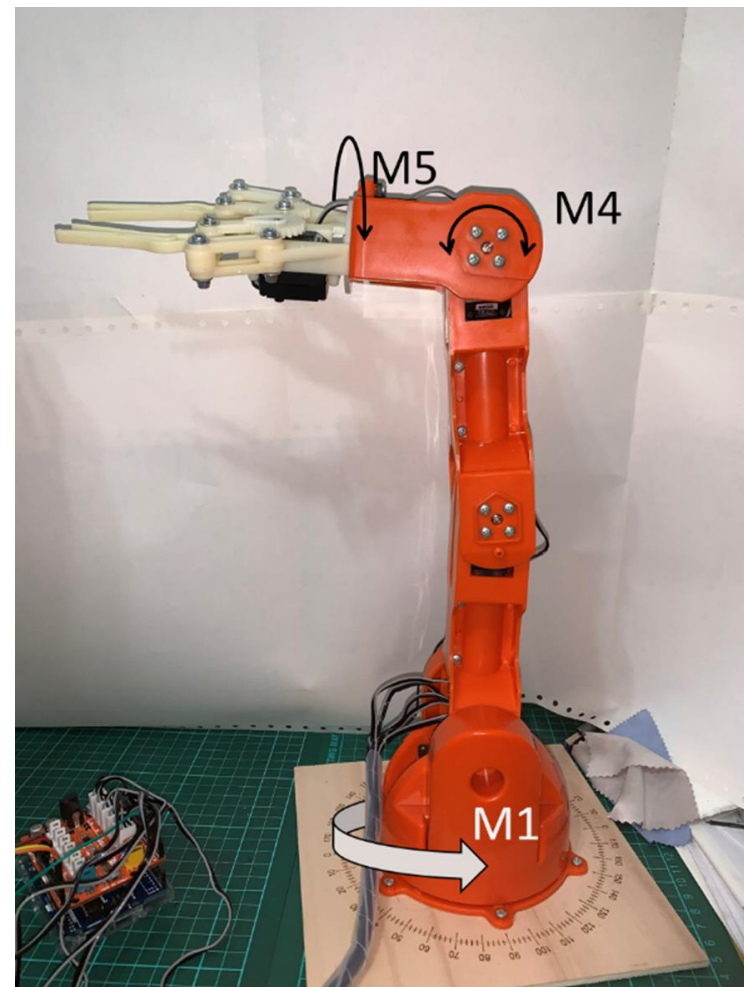
sensor.rotate(angle=mag(w)*dt, axis=axis)

if w_set == 0:
    sensor.axis = vec(1,0,0)
    sensor.up = vec(0,1,0)
    sensor.size = vec(2,0.5,4)
    #tran = sensor.pos
    #sensor = box(size = vec(2,0.5,4), opacity=0.8, color = vec(1,2,3), pos = tran, axis = vec(1,0,0))
    #a_arrow.pos = sensor.pos
    axis_arrow.pos = sensor.pos
    a_arrow.axis = a
    axis_arrow.axis = 3*axis/2
    a_label.text = ("a=",a.x,-a.z,a.y,"=",mag(a))
    omega_label.text = ("w=",w.x,-w.z,w.y)
    if go == 0:
        w_set = 1
        go = 1
        a+=g
        g = a
        initial = cross(norm(g),vec(0,-1,0))
        sensor.rotate(angle=-asin(mag(initial)), axis=initial)
        a = ax*norm(sensor.axis) + az*norm(sensor.up) - ay*norm(sensor.z)
        g = a
```



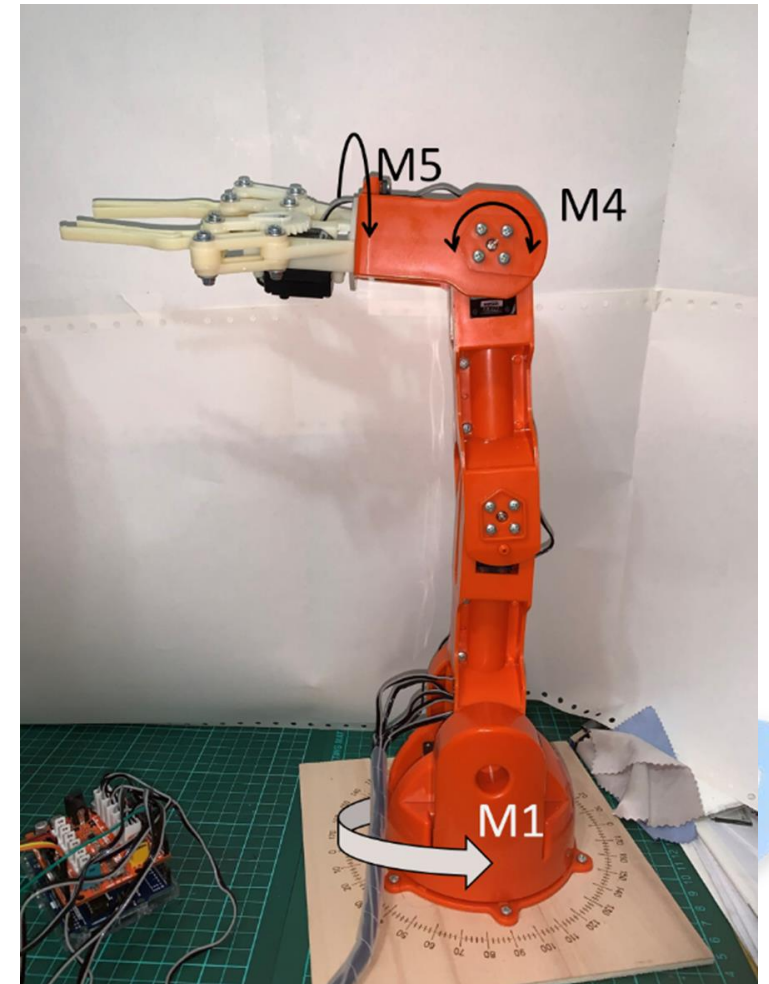
研究方法：以MPU6050操控Braccio

- 連接Braccio
- M1、M4、M5
 - 依序對應z、x、y方向



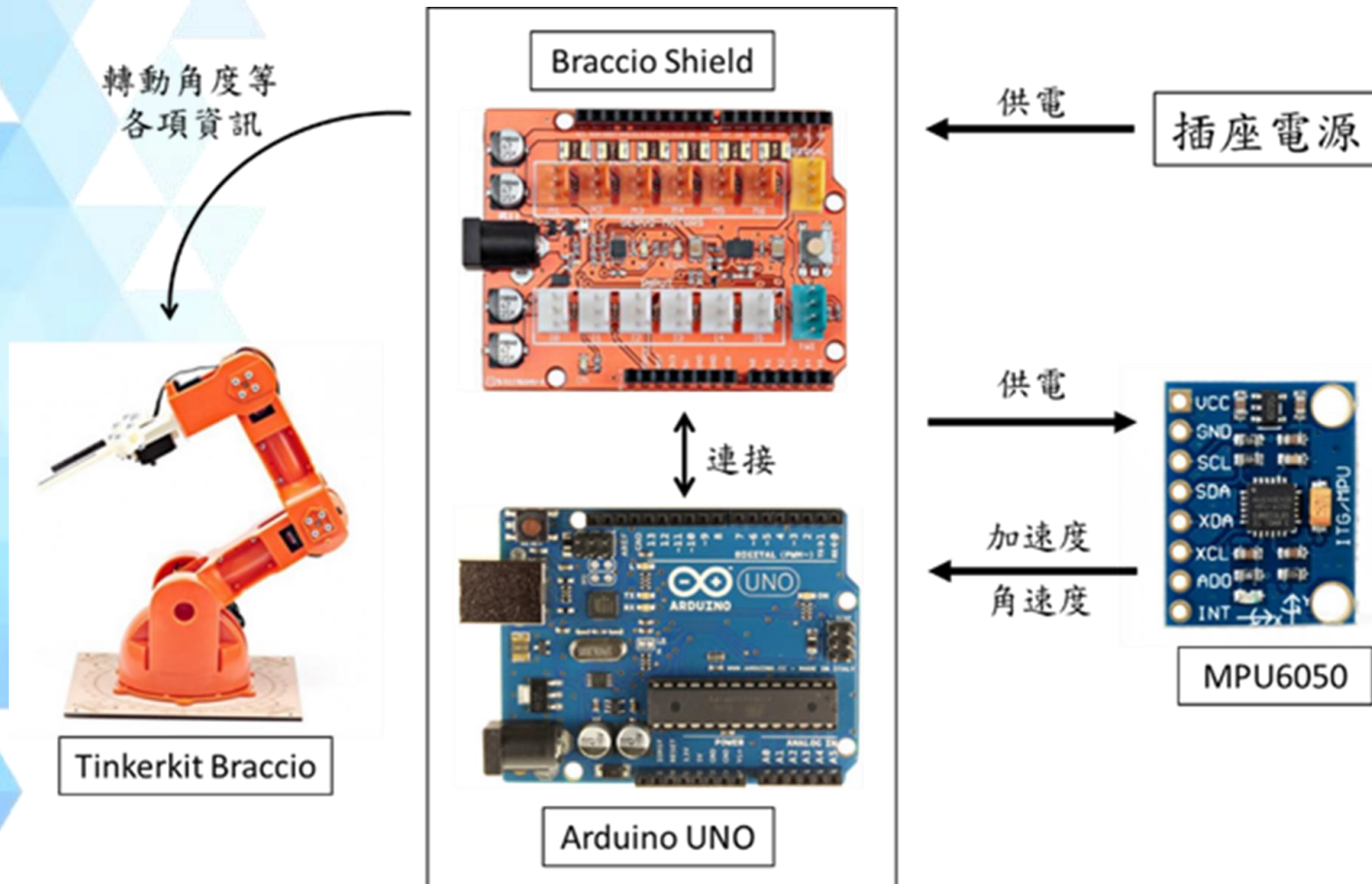
研究方法：以MPU6050操控Braccio

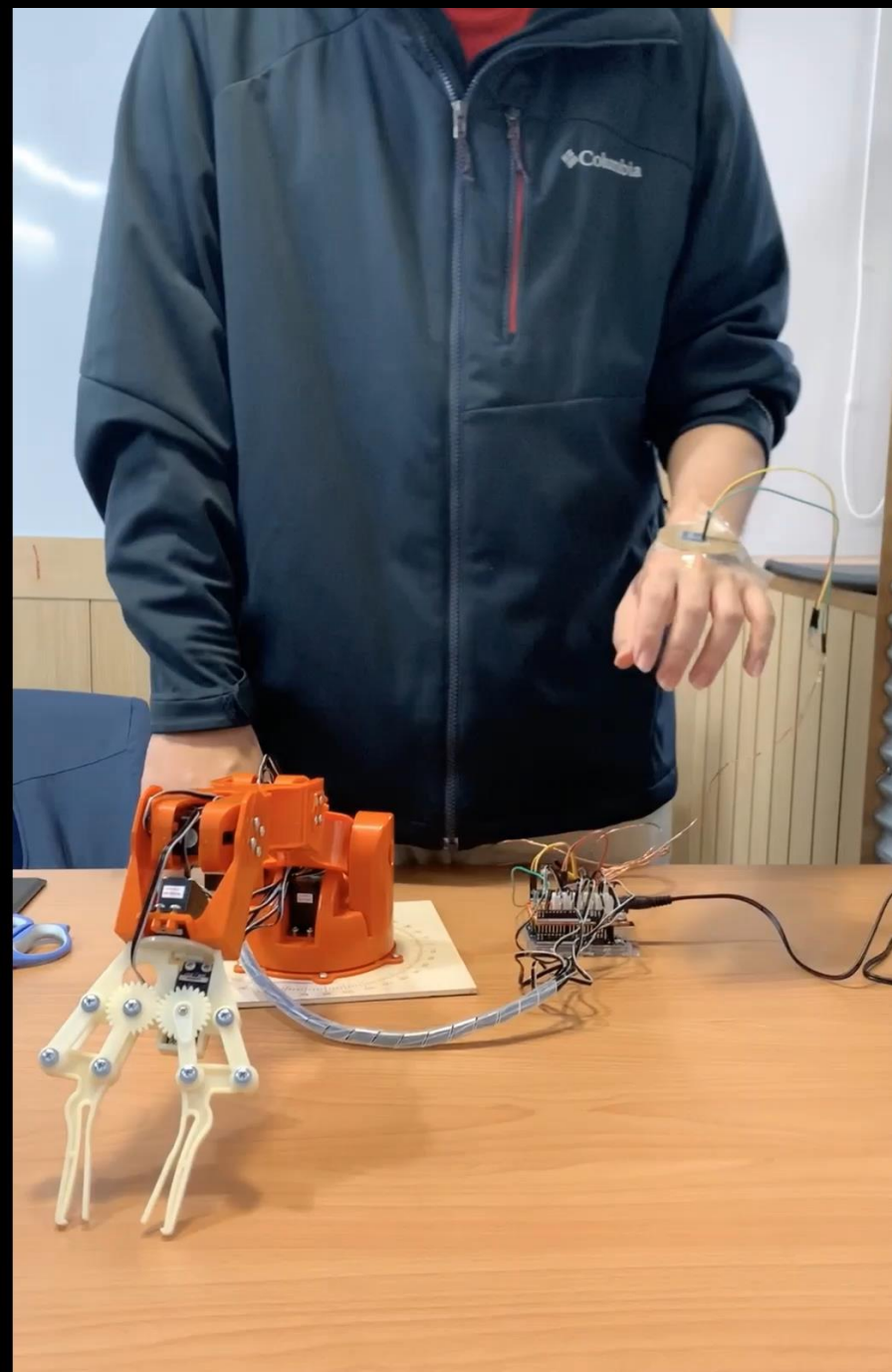
```
1. void Bracciomove() {  
2.     M1 = 90 + rz;  
3.     M4 = rx;  
4.     M5 = 90 + ry;  
5.     //(step delay M1, M2, M3, M4, M5, M6);  
6.     Braccio.ServoMovement( 0, M1, 90, 90, M4, M5, 73);  
7. };
```



研究方法：以MPU6050操控Braccio

Εύρηκα



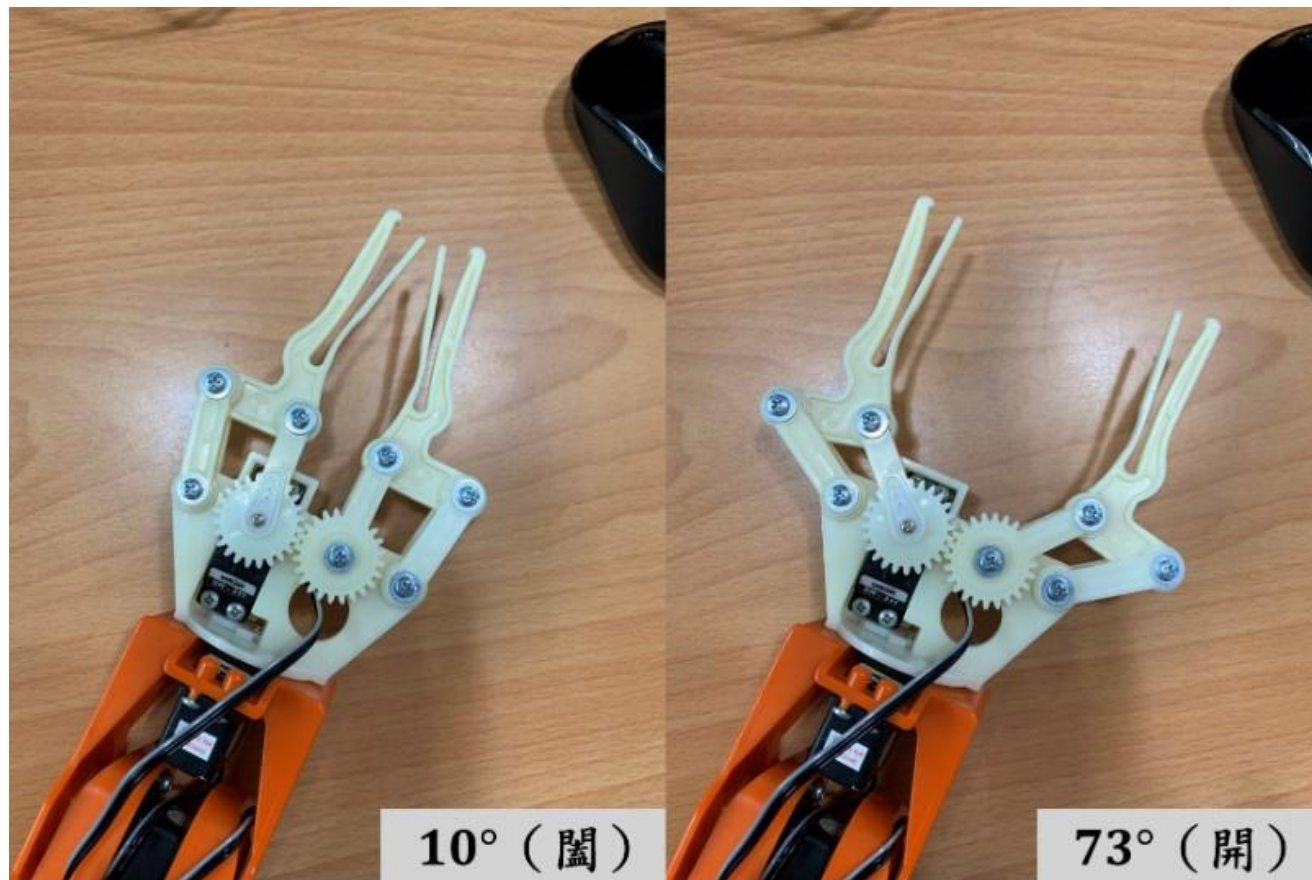


未來計畫

- 完全操控六軸
- M6 (夾子)



Εύρηκα



未來計畫

- 完全操控六軸
 - M2、M3的操控
 - 多使用二組MPU6050



未來計畫

- 以MPU6050操控之便利程度
 - ✓ 利用MPU6050
 - ✓ 利用按鈕/搖桿
- 比較操作時間



Εύρηκα

未來計畫

- 自製六軸機械手臂
- 為特殊用途設計一款不同的機械手臂



結論

本研究已具備以下能力：

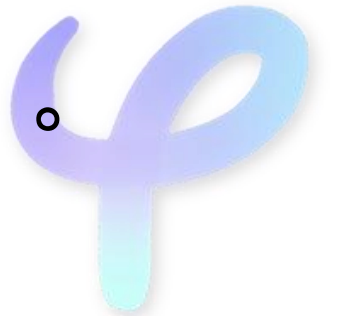
- 透過MPU6050求得物體姿態
- 令機械手臂與手部同步運動



結論

本研究今後的目標：

- 完全操控六軸
 - 壓力、彎曲感測器
 - 增加二組MPU6050
- 自製一臺六軸機械手臂。
- 為特殊用途設計一款不同的機械手臂。



參考資料

Εύρηκα

- [1] Industrial Robots and Robot System Safety. United States Department of Labor. from https://www.osha.gov/dts/osta/otm/otm_iv/otm_iv_4.html
- [2] MPU-6050 Six-Axis (Gyro + Accelerometer) MEMS MotionTracking™ Devices. (2019, October, 18). TDK - InvenSense. Retrieved October 18th, 2019 from <https://www.invensense.com/products/motion-tracking/6-axis/mpu-6050/>
- [3] Li W. C. (2014). Introduction to Arduino. Institute of Applied Mechanics, National Taiwan University, Taipei, Taiwan, ROC.
- [4] Jeff Rowberg (2011, October, 07). MPU6050_raw.ino. Retrieved October 07th, 2011 from <https://github.com/jrowberg/i2cdevlib>

參考資料

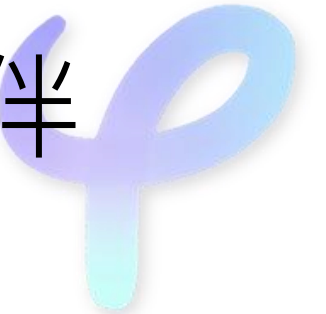
Εύρηκα

- [5] Getting Started with the TinkerKit Braccio Robot. (2019, March, 22). ARDUINO. Retrieved March 22nd , 2019, from <https://www.arduino.cc/en/Guide/Braccio>
- [6] Bo-Chun Chiu. (2012). Golf Swing Analysis Using Body Sensor Networks. Department (Institute) of Network Engineering National Chiao Tung University.
- [7] G. D. Lee et al. (2012). Arm exoskeleton rehabilitation robot with assistive system for patient after stroke. Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, ROC.

Εὐρηκα

感謝

- 感謝 賴奕帆老師 指導
- 感謝 李尉彰教授 指導
- 感謝 高君陶導師&姚志鴻導師的鼓勵
- 感謝 家人的支持與鼓勵
- 感謝 各位同學和物專好夥伴們的陪伴



Εύρηκα

Thanks For Listening

報告到此結束

