

	ใบงานที่ 8	ครั้งที่ 13
	หน่วยที่ GPIO การใช้งาน stepper motor บน raspberry pi ด้วย Python	รวม 9 ชั่วโมง
เรื่องการใช้ stepper motor บน raspberry pi ด้วย Python ของ RPi		จำนวน 90 นาที
ชื่อ.....ชั้น วันที่/...../255.....		

วัตถุประสงค์ทั่วไป

เพื่อให้ผู้เรียนมีความรู้ ความเข้าใจ การใช้งานGPIO ติดต่อกับ stepper motor บน raspberry pi โดยใช้ภาษา Python

วัตถุประสงค์เชิงพฤติกรรม

เมื่อผู้เรียน เรียนจบแล้วสามารถ เชื่อมต่อกับเครื่องคอมพิวเตอร์ เขียนโปรแกรมและประกอบวงจรดังนี้

- 1.เขียนโปรแกรมและประกอบวงจรโดย GPIO ของ RPi โดยใช้ภาษา Python ใช้งาน stepper motor ได้
- 2.เขียนโปรแกรมและประกอบวงจรโดย GPIO ของ RPi โดยใช้ภาษา Python ใช้งาน servo motor ได้
- 3.มีคุณธรรม จริยธรรม และคุณลักษณะอันพึงประสงค์

เครื่องมือและอุปกรณ์

- | | |
|---------------------------------|-----------------|
| 1. บอร์ดทดลอง Raspberry Pi | จำนวน 1 บอร์ด |
| 2. แหล่งจ่ายไฟ 9 โวลต์ | จำนวน 1 เครื่อง |
| 3. เครื่องคอมพิวเตอร์ | จำนวน 1 ชุด |
| 4. บอร์ดทดลองวงจรอิเล็กทรอนิกส์ | จำนวน 1 บอร์ด |
| 5. แอลอีดี | จำนวน 4 ตัว |
| 6. ตัวต้านทาน 330 Ω | จำนวน 8 ตัว |
| 7. ไอซี ULN2003 | จำนวน 1 ตัว |
| 8. Stepper motor | จำนวน 1 ตัว |
| 9. servo motor | จำนวน 1 ตัว |

ลำดับขั้นตอนการทดลอง

1. โปรแกรมทดลองใช้งาน Stepper Motor มีดังนี้

```
import time
import RPi.GPIO as GPIO          # Use BCM GPIO references
GPIO.setmode(GPIO.BCM)          # instead of physical pin numbers
StepPins = [24,25,8,7]          # Define GPIO signals to use
for pin in StepPins:             # Set all pins as output
    print "Setup pins"
    GPIO.setup(pin,GPIO.OUT)
    GPIO.output(pin, False)

StepCounter = 0                  # Define some settings
WaitTime = 0.5                   # Define simple sequence
StepCount1 = 4
Seq1 = []
Seq1 = range(0, StepCount1)
Seq1[0] = [1,0,0,0]
Seq1[1] = [0,1,0,0]
Seq1[2] = [0,0,1,0]
Seq1[3] = [0,0,0,1]
```

```

StepCount2 = 8 # Define advanced sequence
Seq2 = [] # as shown in manufacturers datasheet
Seq2 = range(0, StepCount2)
Seq2[0] = [1,0,0,0]
Seq2[1] = [1,1,0,0]
Seq2[2] = [0,1,0,0]
Seq2[3] = [0,1,1,0]
Seq2[4] = [0,0,1,0]
Seq2[5] = [0,0,1,1]
Seq2[6] = [0,0,0,1]
Seq2[7] = [1,0,0,1]

Seq = Seq1 # Choose a sequence to use
StepCount = StepCount1 # Start main loop
while 1==1:
    for pin in range(0, 4):
        xpin = StepPins[pin]
        if Seq[StepCounter][pin]!=0:
            print " Step %i Enable %i" %(StepCounter,xpin)
            GPIO.output(xpin, True)
        else:
            GPIO.output(xpin, False)
    StepCounter += 1
    if (StepCounter==StepCount): # If we reach the end of the sequence start again
        StepCounter = 0
    if (StepCounter<0):
        StepCounter = StepCount
    time.sleep(WaitTime) # Wait before moving on
    sudo nano stepper.py แล้ว copy code ข้างบน มาวาง sudo python stepper.py เพื่อรัน

```

ทดลองแก้ จาก

```

Seq1[0] = [1,0,0,0]
Seq1[1] = [0,1,0,0]
Seq1[2] = [0,0,1,0]
Seq1[3] = [0,0,0,1]

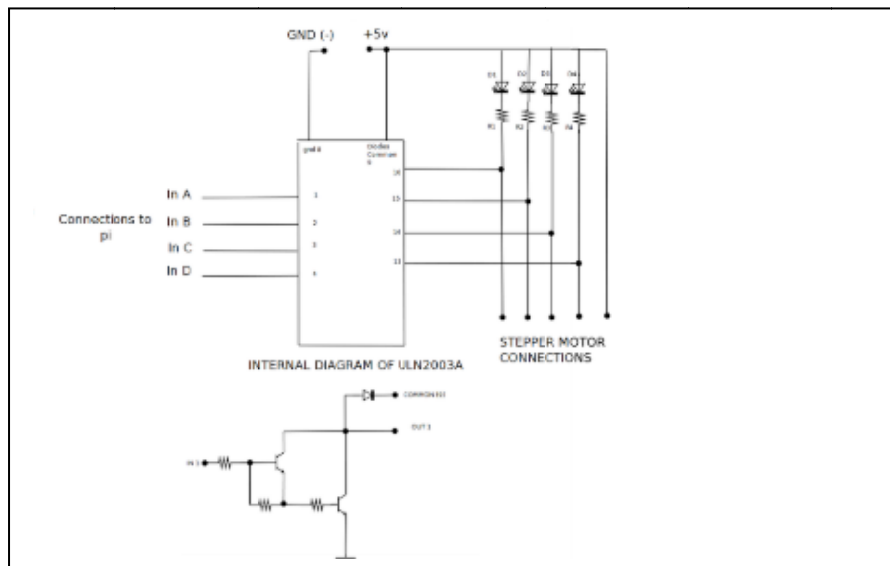
```

เป็น

```

Seq1[3] = [1,0,0,0]
Seq1[2] = [0,1,0,0]
Seq1[1] = [0,0,1,0]
Seq1[0] = [0,0,0,1]

```



1. Step 1:
Connect in A with Pin 11 (GPIO 17)
2. Step 2:
Connect in B with Pin 12 (GPIO 18)
3. Step 3:
Connect in C with Pin 13 (GPIO 21)
4. Step 4:
Connect in D with Pin15 (GPIO 22)
5. Step 5:
Connect 5 V with Pin 6 (Gnd) and Pin 2 (+5 V)

The motor connects to the controller board with a pre-supplied connector. The controller board has 4+2 pins that need to be connected to the Pi header (P1).

The P1-XX references above represent the Pi header pins I used. These parameters are defined in the Python code example below in the StepPins list so if you use different pins be sure to update the Python list as well. You can use other GPIO pins if required, just remember to update your Python script.

Python Script

Here is a copy of the script I used to rotate the stepper motor. It uses version 0.2.0 of the RPi. GPIO library and defines a 4-step and 8-step sequence.

import time	#Import required libraries
import RPi.GPIO as GPIO	
GPIO.setmode(GPIO.BCM)	#Use BCM GPIO references
Pins 18,22,24,26	Instead of physical pin numbers
GPIO24,GPIO25,GPIO8,GPIO7	Define GPIO signals to use
StepPins = [24,25,8,7]	Set all pins as output for pin in StepPins:
print "Setup pins"	
GPIO.setup(pin,GPIO.OUT)	
GPIO.output(pin, False)	
StepCounter = 0	Define some settings
WaitTime = 0.5	

```
StepCount1 = 4
```

Define simple sequence

```
Seq1 = []
```

```
Seq1 = range(0, StepCount1)
```

```
Seq1[0] = [1,0,0,0]
```

```
Seq1[1] = [0,1,0,0]
```

```
Seq1[2] = [0,0,1,0]
```

```
Seq1[3] = [0,0,0,1]
```

```
StepCount2 = 8
```

Define advanced sequence as shown in manufacturers datasheet

```
Seq2 = []
```

```
Seq2 = range(0, StepCount2)
```

```
Seq2[0] = [1,0,0,0]
```

```
Seq2[1] = [1,1,0,0]
```

```
Seq2[2] = [0,1,0,0]
```

```
Seq2[3] = [0,1,1,0]
```

```
Seq2[4] = [0,0,1,0]
```

```
Seq2[5] = [0,0,1,1]
```

```
Seq2[6] = [0,0,0,1]
```

```
Seq2[7] = [1,0,0,1]
```

```
Seq = Seq1
```

Choose a sequence to use

```
StepCount = StepCount1
```

```
while 1==1:
```

Start main loop

```
    for pin in range(0, 4):
```

```
        xpin = StepPins[pin]
```

```
        if Seq[StepCounter][pin]!=0:
```

```
            print " Step %i Enable %i" %(StepCounter,xpin)
```

```
            GPIO.output(xpin, True)
```

```
        else:
```

```
            GPIO.output(xpin, False)
```

```
    StepCounter += 1
```

```
    # If we reach the end of the sequence
```

```
    # start again
```

```
    if (StepCounter==StepCount):
```

```
        StepCounter = 0
```

```
    if (StepCounter<0):
```

```
        StepCounter = StepCount
```

```
    # Wait before moving on
```

```
        time.sleep(WaitTime)
```

คำถาม

1. ไอซี MCP23017 มีความสามารถคือ

.....

.....

.....

.....

.....

2. การเปิดใช้งานพอร์ต I²C ของ RPi

1.
2.

3. การทดสอบการติดต่อกับอุปกรณ์ I²C

.....

.....

4. MCP23017 byte mode register address นั้น

- 00h หมายถึง
- 14h หมายถึง