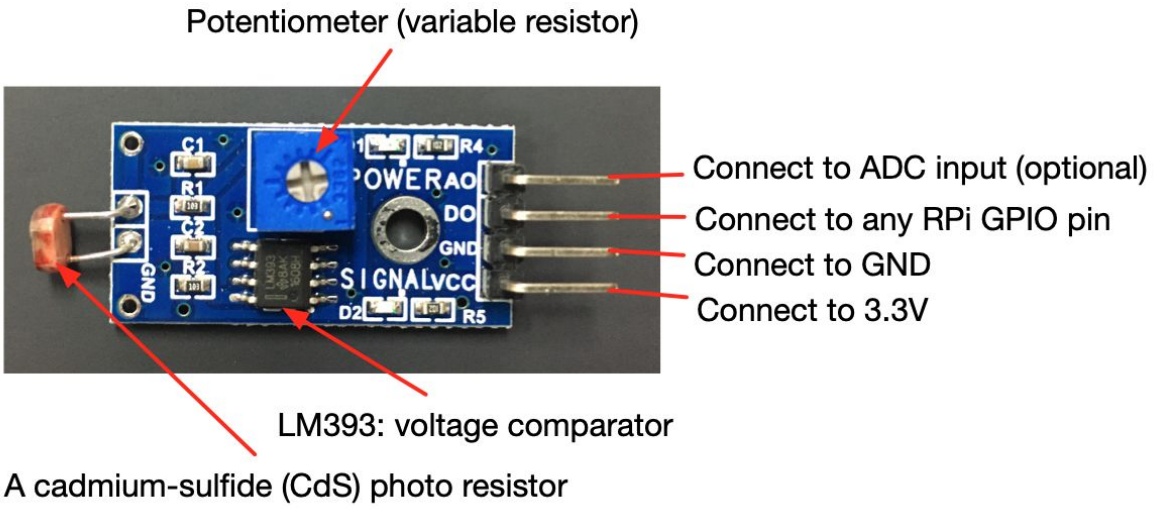
2019-2 Embedded System Design Practice Lab #4

email: ysjoo@cs.kookmin.ac.kr

Light Sensor

0. Pin description
   - VCC: 3.3V
   - GND: ground (0V)
   - DO: digital out
     - You can increase/decrease the threshold voltage by adjusting the potentiometer (use a philips “+” shaped screwdriver).
   - AO: analog out

1. Make a prototype for the below circuit.



   -

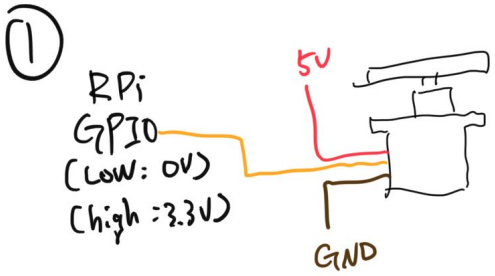
2. Make a C/C++ or python code to perform light detection as below:
   - For every 100ms, read the light sensor value.
   - If the light status is changed from the previous value, display either “Light turned on.” or “Light turned off.” accordingly.
   - If the status is not changed, do not display any message.

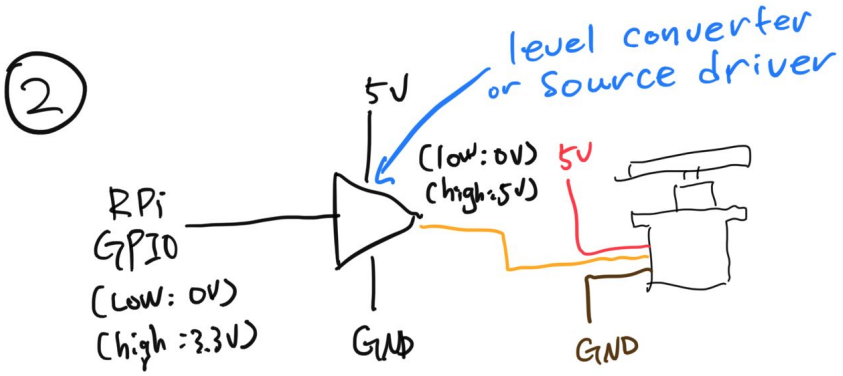
3. The above example uses a polling method to monitor light status. Use a interrupt instead of polling to perform the same function described above.

Servo Motor

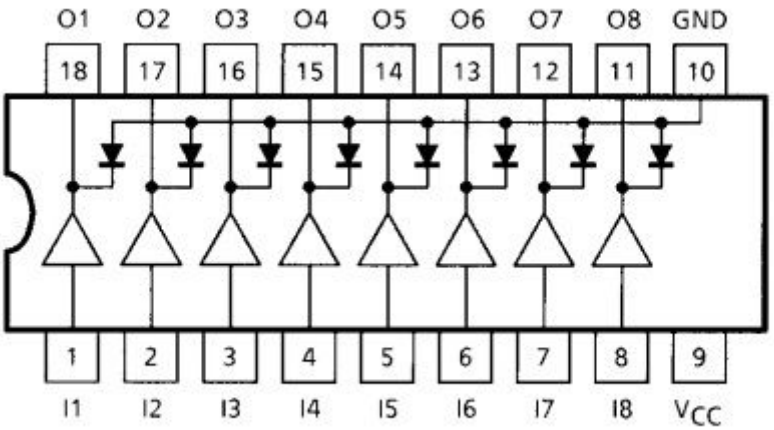
1. Servo pin description
   - Orange: PWM input
   - Red: VCC (typ: 4.8V ~ 6V)
   - Brown: GND
   - **MG-90S datasheet**
     - https://engineering.tamu.edu/media/4247823/ds-servo-mg90s.pdf

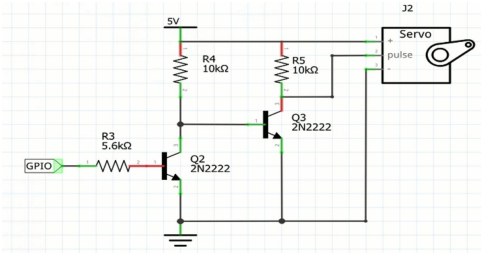
2. How to connect a 5V-based servo to 3.3V GPIO of RPi?
   - (1) direct connection: more susceptible to noise
   - (2) use a source driver (or a level converter)

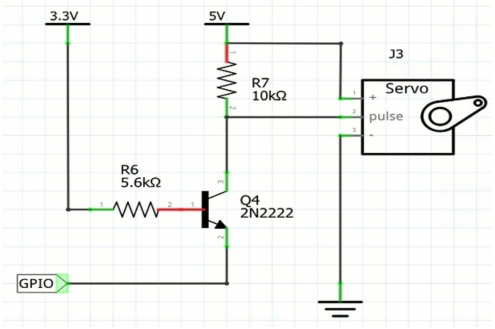




   - A 8-channel source driver pin map example: TD62783APG (available upon request)

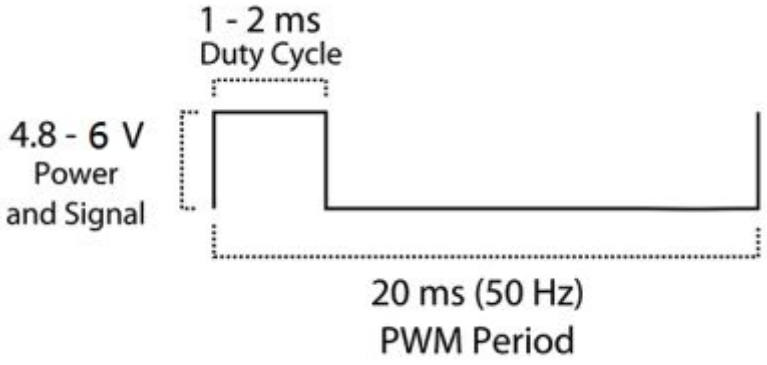


- https://alhkma.com/td62783apg-8-ch-high-voltage-source-driver-dip-18/
- (3) Alternative way: use transistor(s)
- 

or 

- https://www.iot-programmer.com/index.php/books/22-raspberry-pi-and-the-iot-in-c/chapters-raspberry-pi-and-the-iot-in-c/60-raspberry-pi-and-the-iot-in-c-pulse-width-modulation-servos-and-more?start=5

3. How to control a servo motor?
   - Use a PWM signal



   -
   - WiringPi library offers a set of PWM API functions such as start(dutycycle) and ChangeDutyCycle(dutycycle), where dutycycle has a value ranging from 0 to 100 (%).
   - Question 1) For 20ms PWM period, what is a value in % for 1 ms duty cycle (leftmost position)?
   - Question 2) For 20ms PWM period, what is a value in % for 2 ms duty cycle (rightmost position)?
   - Question 2) For 20ms PWM period, what is a value in % for 1.5 ms duty cycle (center position)?
   - Use the above value and the below Python template code to test your servo. Also find the actual range supported by the servo (typically larger than 1-2ms span).

```
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)
GPIO.setup(25,GPIO.OUT)

pos_center = ____ <- fill this using the answer for Question 1.
```

```
pwm = GPIO.PWM(25,50)
pwm.start(pos_center)

dutycycle = input("Duty: ")
while dutycycle != 0:
    pwm.ChangeDutyCycle(dutycycle)
    dutycycle = input("Duty: ")

pwm.stop()
GPIO.cleanup()
```

- Extend the above template such that your servo sweeps between the leftmost position and the rightmost position smoothly. Also, adjust the sweep speed to figure out the speed limit of the tested servo.
- If you are to control multiple servos simultaneously, you may want to use a specially designed library such as pigpio or servoblaster.