SOFTWARE

Instalaciones

Como estamos usando el driver de la marca adafruit modelo pca9685, la libreria que hay que usar es: Adafruit\_PCA9685 por lo tanto tendremos que instalar dicha librería en la raspberry pi.

El enlace de la librería es el siguiente: https://github.com/adafruit/Adafruit\_Python\_PCA9685

Además, como vamos a usar el giroscopio mpu6050 he encontrado una librería que tiene funciones bastante interesantes, y que nos facilita bastante las cosas, que es mpu6050-raspberrypi. El enlace de la librería es el siguiente: https://github.com/Tijndagamer/mpu6050

Hay que instalar la librería de github puesto que es mas facil clonar el repositorio con todo el codigo que pasarlo a mano

Direcciones

Como vamos a usar dos aparatos diferentes conectados al puerto I2c de nuestra placa, vamos a necesitar saber que dirección se asigna a cada “cacharro”, en principio es bastante facil, simplemente ers un comando de terminal (pero hay que hacerlo)

Antes de comprobar en que dirección se encuentra el driver hay que activar el protocolo i2c en la placa ya que por defecto viene desactivada.

Al tener que utlizar dos drivers, para el control de los servos, es necesario cambiar la direccion de cada uno con un soldador, de forma que no interfieran con el giroscopio, ni entre ellos.

Además una vez hecho esto hay que actualizar el objeto robot en el codigo de forma que coincidan.

Permisos

A no ser que ejecutemos nuestros programas desde el usuario root tendremos que darle permiso sobre los puertos i2c al resto de usuarios que no son root puesto que por defecto viene desactivado, una buena guía de cómo hacerlo es la siguiente https://www.abelectronics.co.uk/kb/article/1061/i2c--smbus-and-armbian-linux

Pines

El driver que hemos cogido no trabaja con el puerto sda\_0 ni scl\_0, si no que la librería esta preparada para trabajar con el pin scl\_1 y sda\_1. Es algo a tener en cuenta puesto que si no la libraria no encuentra el driver de forma correcta

Función salida pwm

El driver escogido da un pulso se entre 450, que es su mínimo, mientras que 2100 parece ser un límite seguro para que los servos no se sobrecarguen. Por lo tanto, hay que hacer una función que permita recibir un ángulo deseado, y después pasarlo a un pulso pwm. El rango del pulso hay que comprobarlo experimentalmente.

COMUNICACIÓN

Para comunicarse con robot, se va a utilizar wifi.

Para ello la polaca se va a poner como punto de acceso wifi, de forma que el ordenador que la controle tendrá que conectarse a dicho wifi

Usuario: humanoide

Contraseña: humanoide2018

El método de comunicación inicial será a través de socket, librería que viene implementada en Python. En posteriores fases del proyecto se creará un servidor web en el que posiblemente se colocará una página web realizada en html o en css.