

---

## TAREA 4 (MOVIMIENTO DE UN ROBOT)

---

DINAMICA DE ROBOTS



**UNIVERSIDAD POLITÉCNICA**  
**DE LA ZONA METROPOLITANA DE GUADALAJARA**

ALUMNO: FRANCISCO JAVIER HERNANDEZ MORALES  
MAESTRO: CARLOS ENRIQUE MORAN GARABITO

## DINAMICA DE ROBOTS

OBJETIVO: MOVER UN MOTOR PAP POR GRADOS UTILIZANDO UN MICROCONTROLADOR QUE NO SEA ARDUINO.

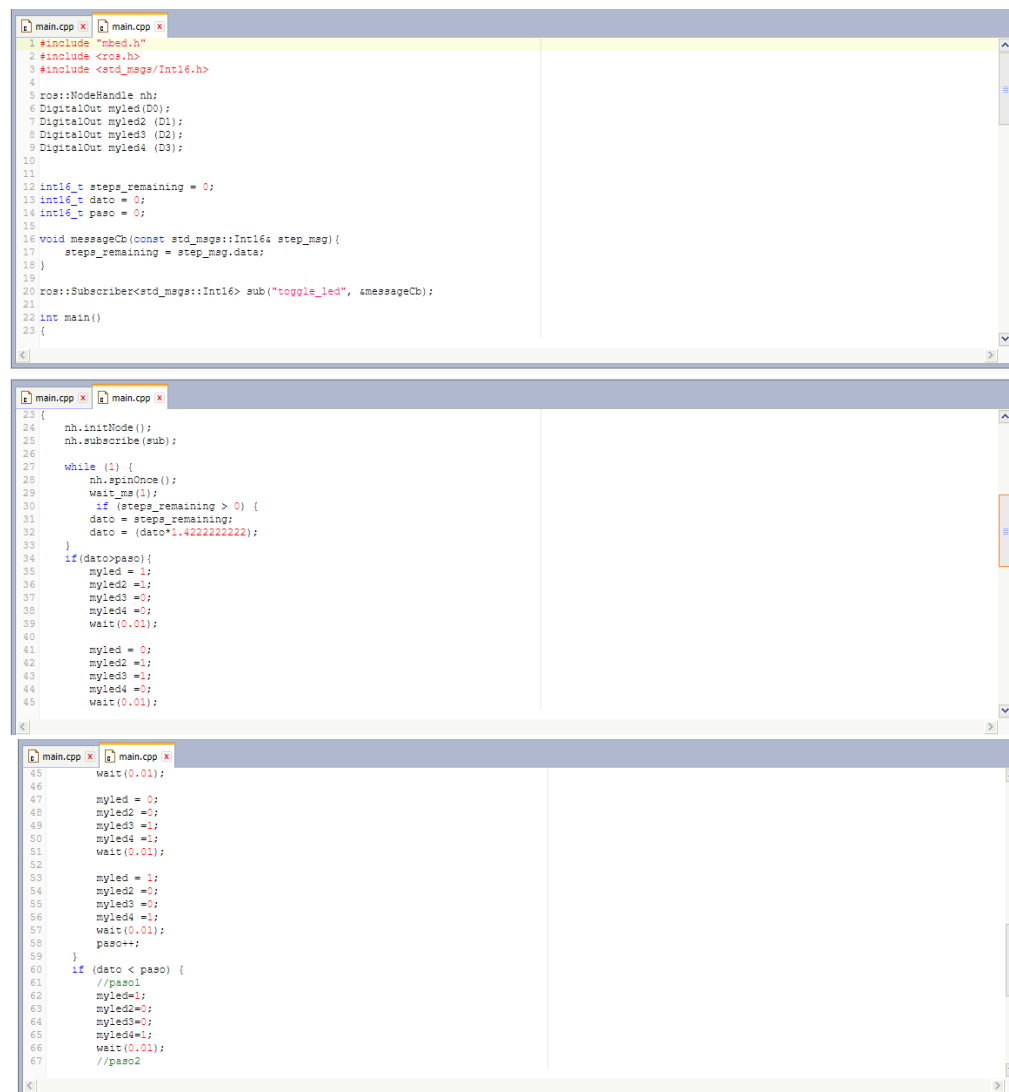
MATERIAL:

- COMPUTADORA
- CABLES DE CONEXIÓN
- FREESCALE
- MOTOR PAP

PROCEDIMIENTO:

1.- PRIMERAMENTE, CREAMOS EL CODIGO EN MBED CON NUESTRA LIBRERÍA DE ROS PARA QUE AL EL PROGRAMA SE PUEDA CORRER DESDE ROS.

CODIGO:



```
1 #include "mbed.h"
2 #include <ros.h>
3 #include <std_msgs/Int16.h>
4
5 ros::NodeHandle nh;
6 DigitalOut myled(D0);
7 DigitalOut myled2 (D1);
8 DigitalOut myled3 (D2);
9 DigitalOut myled4 (D3);
10
11
12 int16_t steps_remaining = 0;
13 int16_t dato = 0;
14 int16_t paso = 0;
15
16 void messageCb(const std_msgs::Int16& step_mag){
17     steps_remaining = step_mag.data;
18 }
19
20 ros::Subscriber<std_msgs::Int16> sub("coggle_led", &messageCb);
21
22 int main()
23 {
24
25     nh.initNode();
26     nh.subscribe(sub);
27
28     while (1) {
29         nh.spinOnce();
30         wait_ms(1);
31         if (steps_remaining > 0) {
32             dato = steps_remaining;
33             dato = (dato*1.42222222222);
34         }
35         if (dato > paso) {
36             myled = 1;
37             myled2 = 1;
38             myled3 = 0;
39             myled4 = 0;
40             wait(0.01);
41
42             myled = 0;
43             myled2 = 1;
44             myled3 = 1;
45             myled4 = 0;
46             wait(0.01);
47
48             myled = 1;
49             myled2 = 0;
50             myled3 = 0;
51             myled4 = 1;
52             wait(0.01);
53
54             myled = 1;
55             myled2 = 0;
56             myled3 = 1;
57             myled4 = 1;
58             wait(0.01);
59             paso++;
60         }
61         if (dato < paso) {
62             //paso1
63             myled=1;
64             myled2=0;
65             myled3=0;
66             myled4=1;
67             wait(0.01);
68             //paso2
```

## DINAMICA DE ROBOTS

```
main.cpp x main.cpp x
66 wait(0.01);
67 //paso2
68 myled1=1;
69 myled2=1;
70 myled3=0;
71 myled4=0;
72 wait(0.01);
73 //paso3
74 myled=0;
75 myled2=1;
76 myled3=1;
77 myled4=0;
78 wait(0.01);
79 //paso4
80 myled=0;
81 myled2=0;
82 myled3=1;
83 myled4=1;
84 wait(0.01);
85 paso--;
86 }
87 }
88 }
```

- EL PROGRAMA ESTA DISEÑADO PARA QUE EL MOTOR GIRE EN SENTIDO HORARIO Y ANTIHORARIO.

2.- para correr para correr el programa cargado en el microcontrolador freescale en ros usaremos los siguientes comandos:

```
roscore
```

después ubicamos el puerto y ponemos:

```
roslaunch roserial_python serial_node.py /dev/ttyUSB0
```

y por ultimo ponemos los grados que deseamos moverlo:

```
rostopic pub toggle_led std_msgs/Empty --120
```

RESULTADO:

