 ING.MECATRÓNICA.



Ev\_2\_2\_Movimiento de un robot.

Jessica Nayeli Lozada Canizal 9-B Profesor: Carlos Morán.

Objetivo:

Mover un motor a pasos con un microcontrolador de 32 bits.

Materiales:

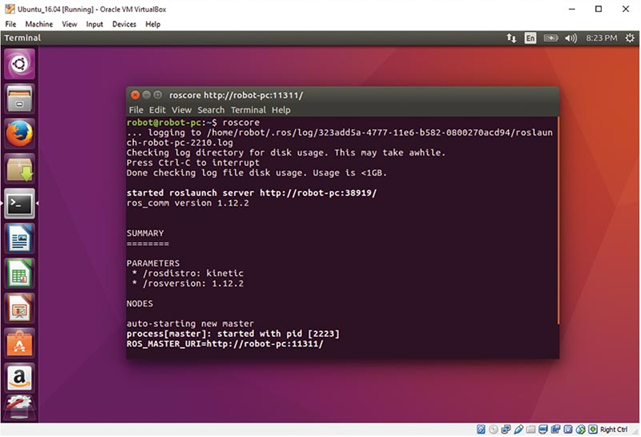
* Ubuntu con ROS
* Microcontrolador (KL25Z)
* Driver para motor a pasos.
* Cables

Requisitos:

* Tener instalado el programa ROS en el ordenador.
* Instalar la librería rosserial.

Desarrollo

1. Inicio de Ros con el comando *roscore*



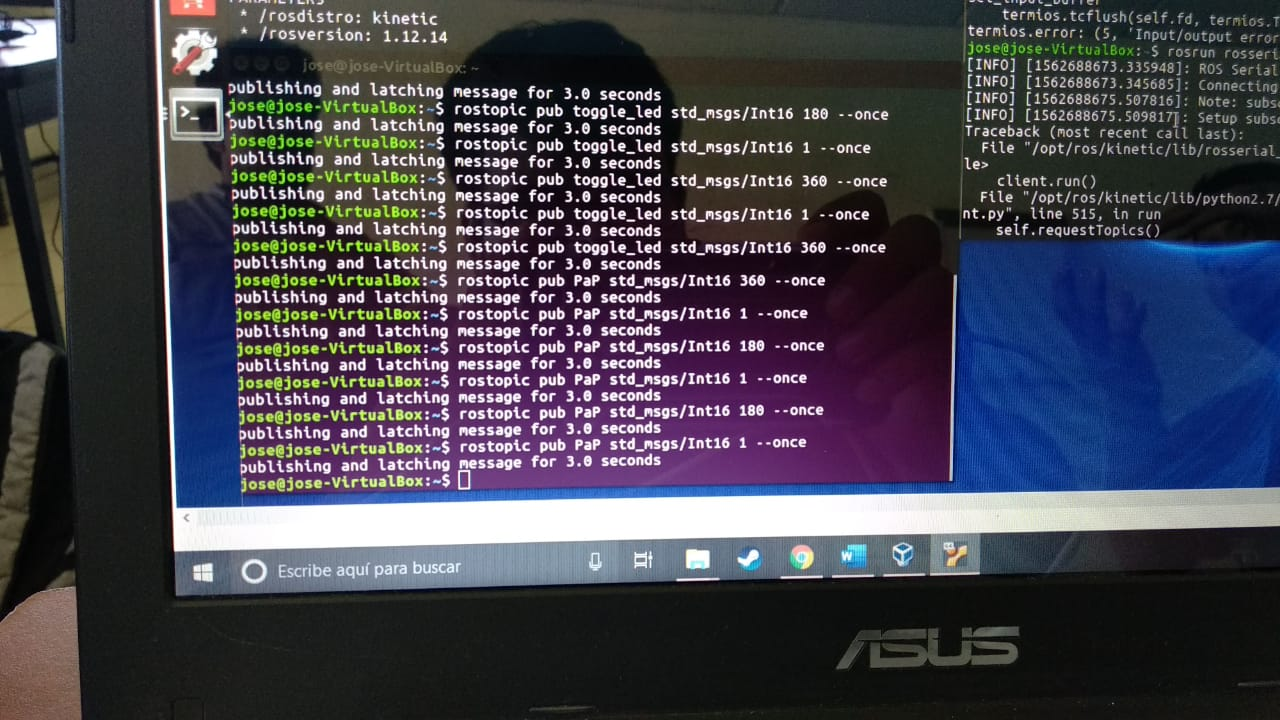
1. Una ves iniciado ROS se abre una nueva terminar el la cual se ingresará el siguiente comando, para que el microcontrolador sea reconocido por linux

*rosrun rosserial\_python serial\_node.py /dev /ttyAMC0*

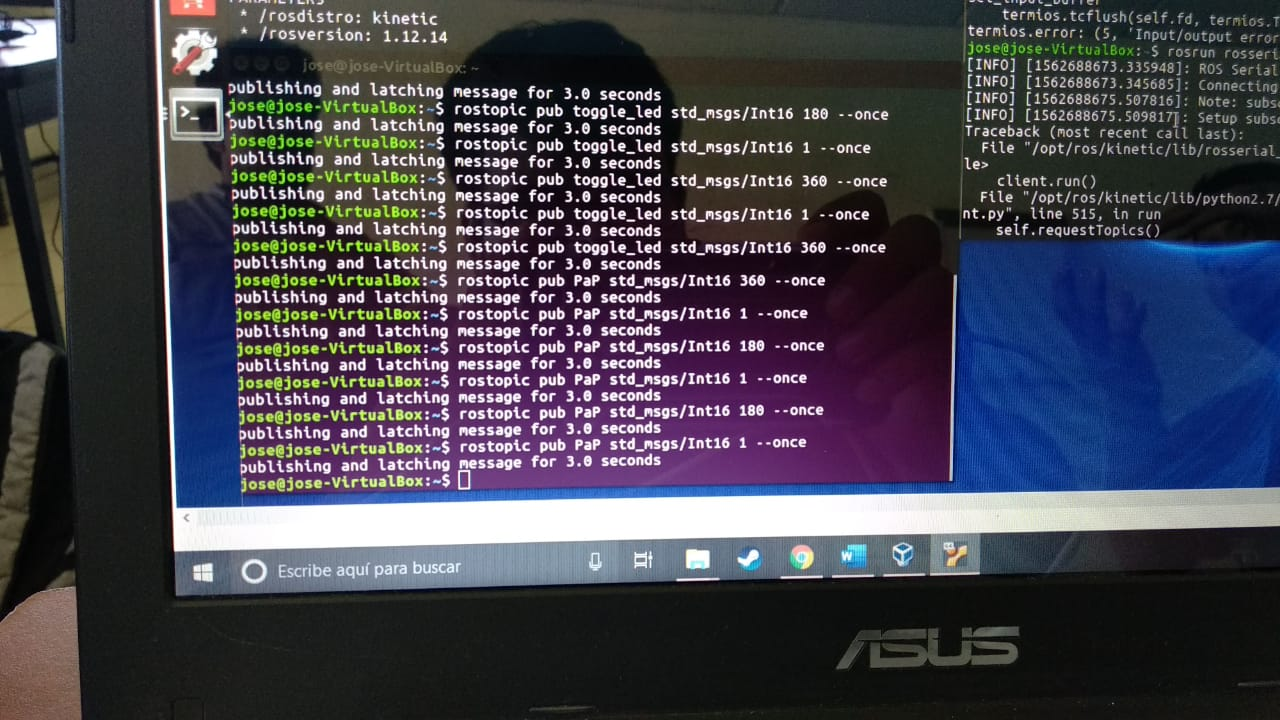
*Luego* el siguiente comando:

*sudo chmod 666 /dev/ttyACM0 or ttyS0, para que Ros tenga permisos de puerto.*

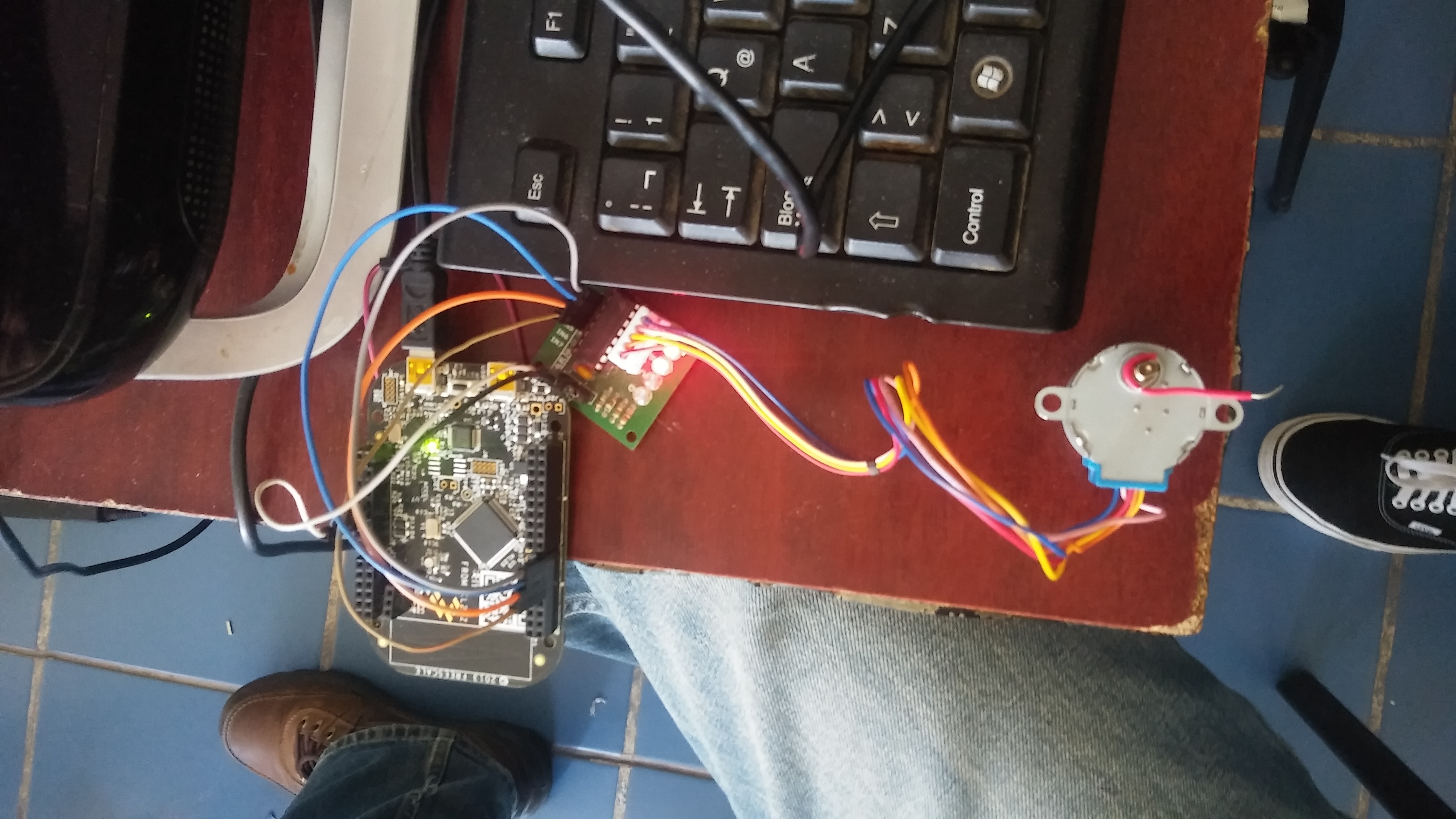
1. El comando para realizar la comunicación serial entre el micro y ROS, es el siguiente.



Resultados:



Comando de comunicación entre ROS y la KL25Z , se ingresa el número de grados.



Código usado en MBED

#include "mbed.h"

#include <ros.h>

#include <std\_msgs/Int16.h>

ros::NodeHandle nh;

DigitalOut IN1(D8);

DigitalOut IN2 (D9);

DigitalOut IN3 (D10);

DigitalOut IN4 (D11);

int16\_t step = 0;

int16\_t data = 0;

int16\_t paso = 0;

void messageCb(const std\_msgs::Int16& step\_msg){

step = step\_msg.data;

}

ros::Subscriber<std\_msgs::Int16> sub("PaP", &messageCb);

int main() {

nh.initNode();

nh.subscribe(sub);

while (1) {

nh.spinOnce();

wait\_ms(1);

if (step > 0) {

data = step;

data = (data\*1.4222222222);

}

if(data > paso){

IN1= 1;

IN2 =1;

IN3 =0;

IN4 =0;

wait(0.01);

IN1= 0;

IN2 =1;

IN3 =1;

IN4 =0;

wait(0.01);

IN1=0;

IN2=0;

IN3=1;

IN4=1;

wait(0.01);

IN1=1;

IN2=0;

IN3=0;

IN4=1;

wait(0.01);

paso++;

}

if (data < paso) {

//paso1

IN1=1;

IN2=0;

IN3=0;

IN4=1;

wait(0.01);

//paso2

IN1=0;

IN2=0;

IN3=1;

IN4=1;

wait(0.01);

//paso3

IN1=0;

IN2=1;

IN3=1;

IN4=0;

wait(0.01);

//paso4

IN1=1;

IN2=1;

IN3=0;

IN4=0;

wait(0.01);

paso--;

}

}

}

Conclusión:

Tuve dificultades al compilar el código para la KL25Z, sin embargo, resultó más sencillo de lo que esperaba y creo que este tipo de tareas, va a ayudar a nuestro equipo a posicionar el brazo robótico.