Escuela Universitaria Politécnica
La Almunia de Doña Godina
Zaragoza

NUM. TFG 424,16,136
Noviembre 2016

Grado en Ingeniería Mecatrónica

ARTICULATED ELEMENT OF 2DOF DEGREES OF FREEDOM CONTROLLED BY A CAMERA

FIN DE GRADO carried out at the SCHOOL OF INNOVATION, DESIGN AND ENGINEERING (IDT) --MÄLARDALEN UNIVERSITY SWEDEN—

Autor: JAVIER QUINTÍN RICO

Director: GIACOMO SPAMPINATO

Universidad Zaragoza
1542

OBJECTIVE OF THE PROJECT---

The final goal of this project is to build a robot arm of 2DOF (2 degrees of freedom) using a couple of stepper motors and 2 bars. Implement the inverse kinematics of the robot to be able to move the TCP as desired. Apart of the position control is required to implement speed control to the robot arm. For the movement is needed to detect by camera the TCP of the robot by a led (recommended) or any other component. For this, the detected point has to be always in the center of the camera and if you move the robot arm or the camera, the movement will correct the position to be again in the middle of the camera.

PARTS--

BUILD THE ROBOT ARM

To build the robot arm I used for the motors the Dynamixel AX-12 stepper actuator, I will connect to the motors the arms bioloid frame F2 of Dynamixel and for the feet bioloid frame F12 of Dynamixel too. It is needed something for the camera to detect, knowing that the camera with MatLab can detect colors I used a red tape of a bottle and attached it to the final part of the robot arm to work as the TCP.

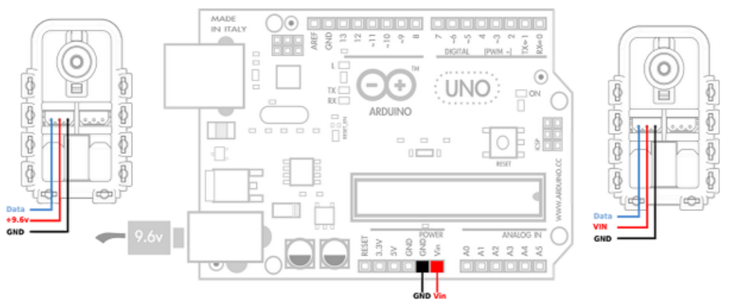
AX-12 stepper actuator

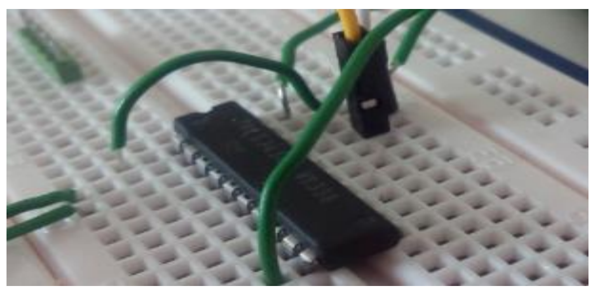
Bioloid frame F2

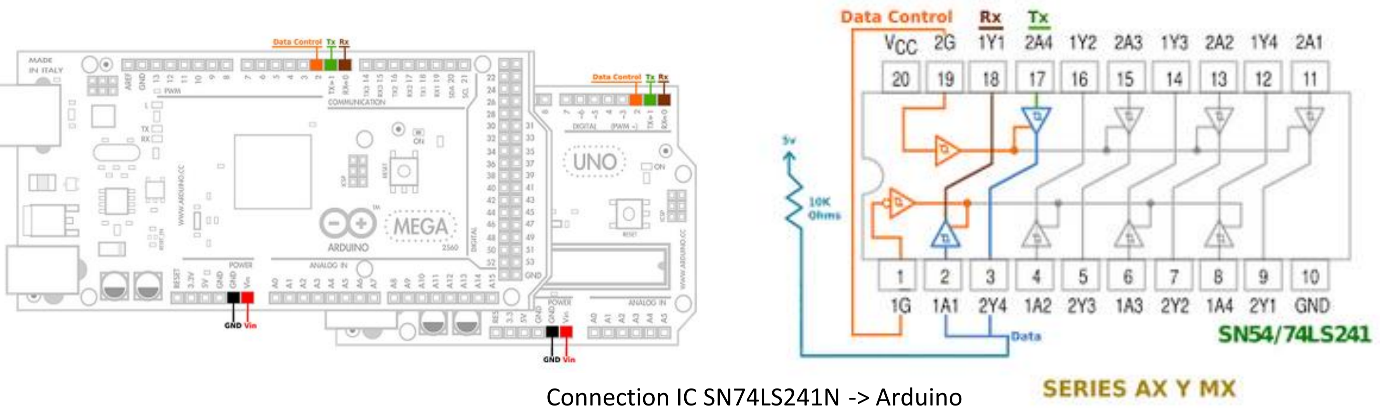
Bioloid frame F12

Now we have the robot built but still without movement. For the movement I will use a microcontroller specifically Arduino Uno because I have already worked with it.

Even with the Arduino Uno and the robot it is needed something so they can understand each other i.e. driver/buffer tristate. Searching in internet for different ones I found one that worked for this communication, IC SN74LS241N.


Connection AX-12 -> Arduino

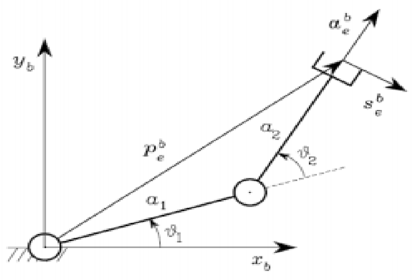

IC SN74LS241N



Connection IC SN74LS241N -> Arduino

INVERSE KINEMATICS

Use of the kinematics equations of a robot to determine the joint parameters that provide a desired position of the end-effector. Specification of the movement of a robot so that its end-effector achieves a desired task.



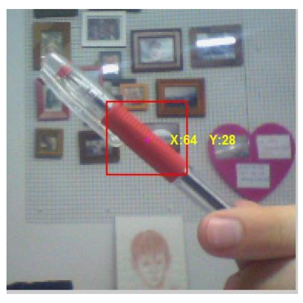
COMUNICATION MATLAB AND ARDUINO

Both Arduino and Matlab coding environment we have specific commands that enables the serial communication, but the core behind its operation is built on the same few principles:

- Arduino and Matlab should both know the rate (BAUD RATE) at which they share info and the physical serial port they are talking through, otherwise they are not going to understand each other.
- They both have commands to send (WRITE) data to the other part through the serial port and to receive and READ data from the serial buffer.

CAMERA DETECTION

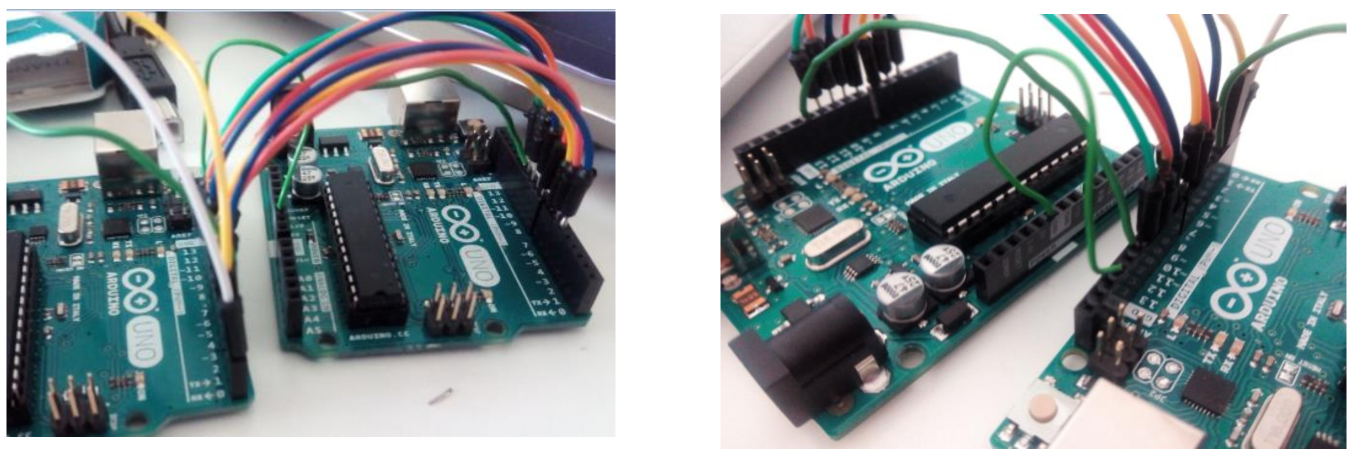
The robot has to be controlled by a camera so I am going to use a program in MatLab to detect the color red (can work with other colors, blue and green). I used a program that detect the color red and gives you the coordinates in the video having the 0.0 in the center of the video.

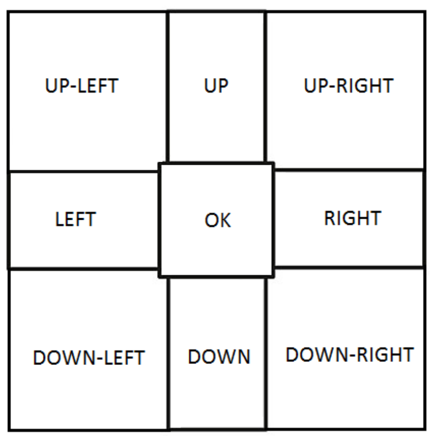


UNITING THE PARTS------------------

Now it is time to put all this parts together and made them work at the same time. Combining the MatLab parts was not a problem it is just to use the program to detect the color and add to it functions of the library to communicate with Arduino.

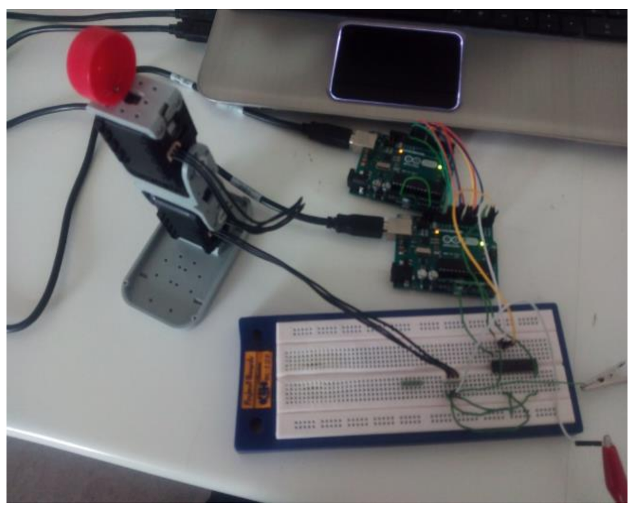
The problem was when I tried to put both Arduino programs together since they used the same pins to communicate one with MatLab and the other with the motors. It is actually possible to change those pins in both programs with some libraries but I didn't manage to made them work in other pins so I arrived to one point where I have 2 programs incompatible and 1 Arduino Uno. Until the point I know I just need to make with the robot 8 types of movement: up, down, right, left, up-right, up-left, down-right and down-left, so I decided to use 2 Arduino Uno and make this 8 instruction communication with wires using the digital pins of both Arduino.


8 Instruction wire connections between both Arduino + power supply



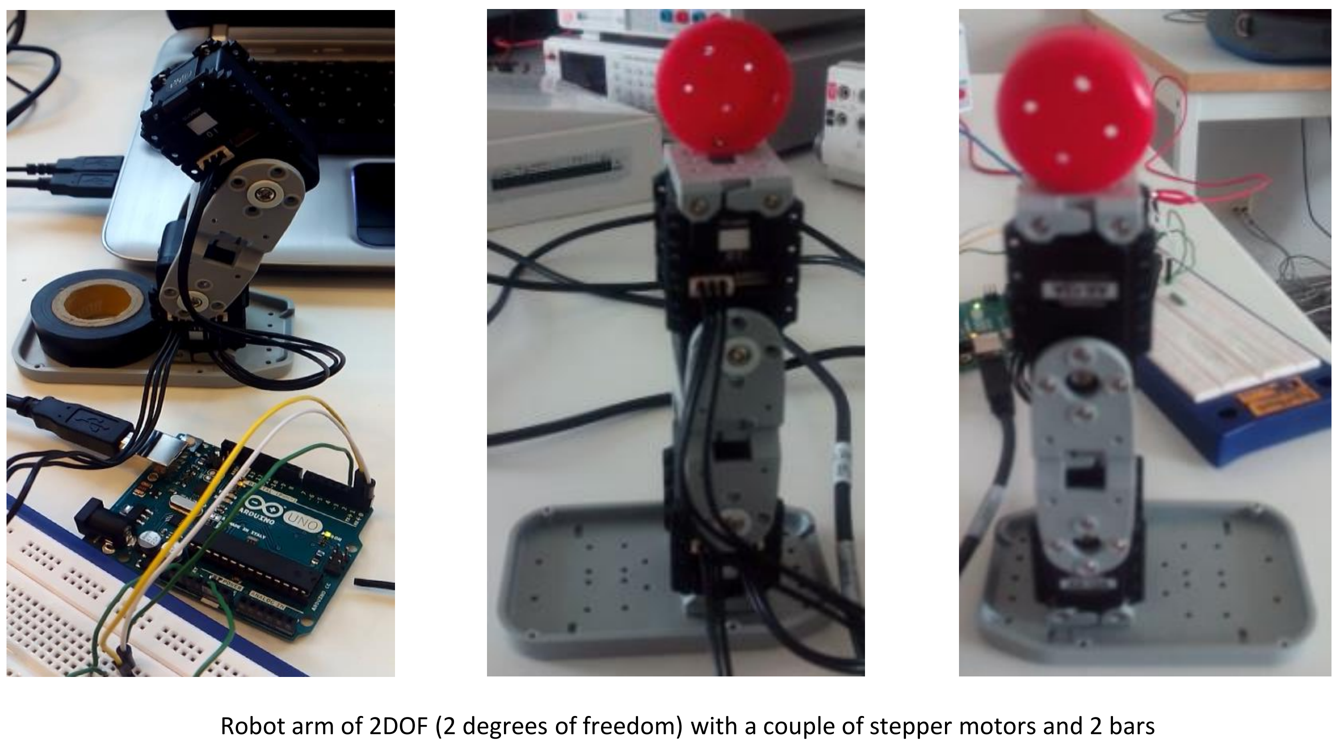
UP-LEFT	UP	UP-RIGHT
LEFT	OK	RIGHT
DOWN-LEFT	DOWN	DOWN-RIGHT

8-types robot movement



STEPS------------------------------------

1. The camera detect in which point of the video is the red tape.
2. MatLab says to the first Arduino where is the red tape (up, down, up-right,…)
3. The first Arduino sends a signal (High-Low) through the digital pins to the second Arduino.
4. The second Arduino changes the coordinates for the robot depending on the signal.
5. The second Arduino makes the inverse kinematics with those new coordinates.
6. The second Arduino says to the motor the new angles and the motors move.
7. The steps start again until the red tape is in the center.


Robot arm of 2DOF (2 degrees of freedom) with a couple of stepper motors and 2 bars