|  |
| --- |
| Universidad Autónoma de Madrid. Escuela Politécnica Superior |
| Proyecto: Rubik Solver |
| Plataformas para sistemas empotrados |

|  |
| --- |
| Marcos Asenjo González – Jon Larrea Martínez  27-4-2019 |

# Descripción del proyecto a realizar

El proyecto Rubik Solver consiste en la construcción de un robot que sea capaz de resolver un cubo de rubik de 3x3x3 gracias a la utilización de 8 servomotores y de una cámara para al inicio detectar el estado del cubo.

El procesador principal encargado de detectar los diferentes colores de cada pieza de las caras del cubo y de realizar el cálculo de los movimientos para resolver el cubo mediante un algoritmo que garantiza menos de 25 movimientos residirá en una Raspberry Pi 2 a la que se le ha acoplado el módulo PiCamera. Por otro lado, es necesario un controlador encargado de gestionar el movimiento de todos los motores para el cual hemos utilizado un Arduino Uno, el cual a través de diversas librerías (creadas específicamente para este proyecto y otras readas por terceros), nos permite girar los 8 servomotores en el rango [0, 180] grados.

# Actuadores necesarios

Los actuadores que necesitaremos para este proyecto son 8 servomotores que nos permitirán realizar movimientos sobre las seis caras del cubo de Rubik:

* **4 x DS3218:** Es el servomotor con más fuerza de los dos (20 Kg) y por tanto es el encargado de realizar el movimiento más exigente, girar 4 caras del cubo (cada servo girará una de las caras).
* **4 x S3003:** Este servomotor tendrá la función de acercar y alejar cada uno de los cuatro servomotores DS3218 a las caras del cubo, permitiendo cambiar la cara que van a rotar.

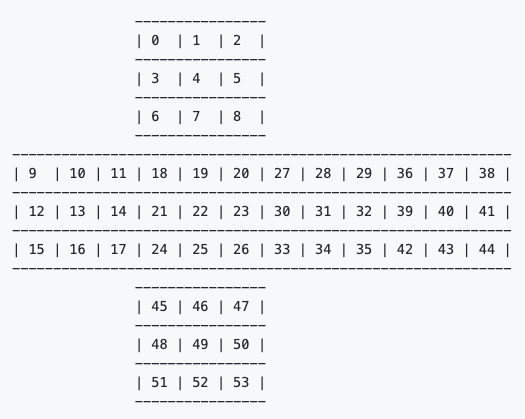
Todos ellos estarán gestionados por el mismo controlador, que a su vez responde a las órdenes del procesador principal. Por tanto, disponemos de un sistema que con cuatro servomotores principales y cuatro secundarios es capaz de controlar la posición de las seis caras del cubo.

# Sensores necesarios

El único sensor utilizado en esta práctica es el módulo *PiCamera* el cual consiste un módulo que integra cámara con una lente de ojo de pez y con una resolución de 8MP y que será el responsable de tomar las 6 imágenes correspondientes a cada cara del cubo para posteriormente en la Raspberry Pi procesar dichas imágenes para obtener los colores de las piezas de cada cara. Este módulo ha sido seleccionado debido a su facilidad de uso mediante la librería PiCamera de Python.

# Control, procesamiento de los datos y comunicaciones

Para la identificación de los colores de las piezas que forman cada cara aprovechamos que la cámara se encuentra en una posición fija respecto al cubo para definir unas secciones cuadradas en las imágenes en cada una de las piezas que forman cada cara de manera que se realiza la media de los tres colores de cada uno de los píxeles de dicha sección para obtener un color representativo de esa cara. Seguidamente se calcula la distancia en el espacio de colores a los 6 colores del cubo y el color con la distancia que sea mínima corresponde con el color de dicha cara. Una vez identificado el vector con los colores de las diferentes caras es necesario aplicar ciertas transformaciones para simular que el cubo tiene la siguiente estructura:



Siendo los intervalos, los colores mostrados a continuación:

* [0, 8]: Cara amarilla
* [9, 17]: Cara azul
* [18, 26]: Cara roja
* [27, 35]: Cara verde
* [36, 44]: Cara naranja
* [45, 53]: Cara blanca

Se utilizará la Raspberry Pi 2 para procesar las imágenes captadas por la cámara y calcular los pasos necesarios para resolver el cubo utilizando el algoritmo de Kociemba, el cual es uno de los algoritmos óptimos para resolver el cubo. Se ha decidido utilizar la placa Raspberry Pi 2 debido a su reducido tamaño, el cual nos permite integrarlo fácilmente en la estructura final.

Para actuar sobre los servos y hacer que estos giren las diferentes caras del cubo de acuerdo a la solución aportada por el algoritmo de Kociemba ya calculado utilizaremos un microcontrolador Arduino modelo Uno el cual nos permitirá con la librería servo.h que ya trae por defecto su IDE de programación junto con una librería propia diseñada para controlar los 8 servomotores de una manera sencilla.

Para realizar la conexión entre el Arduino y la Raspberry utilizaremos un cable USB (El que se utiliza en la carga del código en el Arduino) con una conexión serie para el envio de comandos y recepción de ACKs desde la Raspberry Pi 2.