

Integrantes del equipo: Expediente:

* Gómez Limón Lucia Elizabeth ie694006
* Hernández Cortés Rubén Salvador ie695262
* Davila Reyes Mynor Francisco ie696993
* Rizo de la Torre Miguel Andrés ie696930

Sistemas de Control Automático | LUIS ENRIQUE GONZÁLEZ

2016

# Objetivo de control

Lograr el control cinético en tres dimensiones de un brazo robótico de cuatro grados de libertad para realizar una trayectoria establecida por el usuario utilizando como sensor de visión un Kinect V2.

# Identificación de variables del sistema

## Variables manipuladas

* Ángulos entre los eslabones del robot para cada unión.

## Variables controladas

* Posición del efector final.

# Componentes del sistema

* Brazo robótico impreso en 3D, con motores reductores para el movimiento de las uniones. La configuración del brazo es de cuatro grados de libertad.
* Actuadores; motores de corriente directa en conjunto con sus drivers (puente H).
* Utilizaremos Matlab, para realizar el procesamiento de los datos entregados por el sensor.
* Matlab se comunicará por medio de UART a una tarjeta con el microcontrolador FRDM-K64F que generará una salida de PWM que será la entrada al puente H.
* El sensor que utilizaremos es la cámara de profundidad y de color de la cámara Microsoft Kinect V2, junto con su adaptador para Windows.

# Cinemática Directa: Método Denavit Hartenberg

El objetivo de aplicar cinemática directa en el manipulador de 4GDL es para determinar la posición y orientación del efector final, usando los valores de los ángulos que hay entre cada unión.

Para la obtener las matrices de transformación que describen los eslabones de nuestro manipulador aplicamos la ecuación:

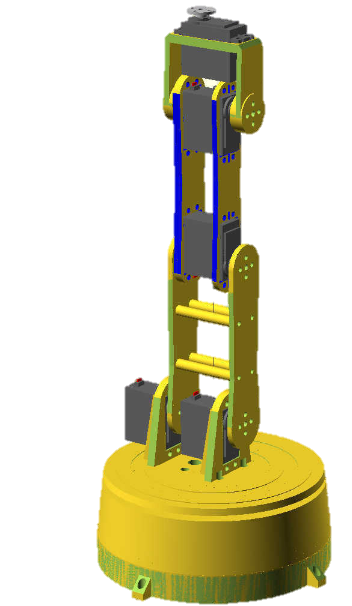
El modelo de las uniones por medio del método Denavit Hartenberg en nuestro manipulador está definido por una matriz de [4,4] las transformaciones y rotaciones por la cantidad de eslabones.

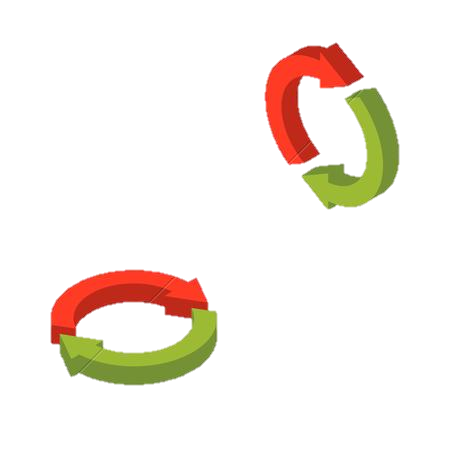
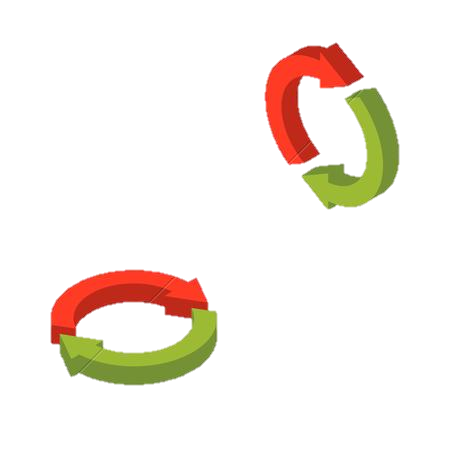
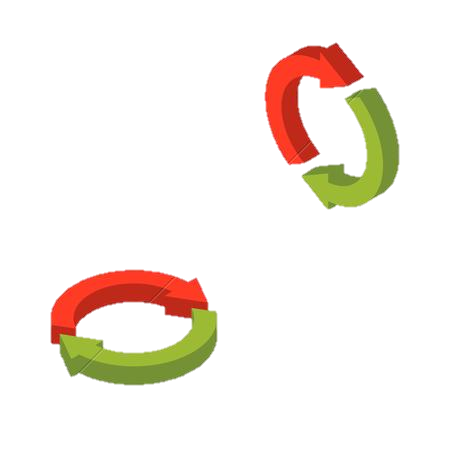
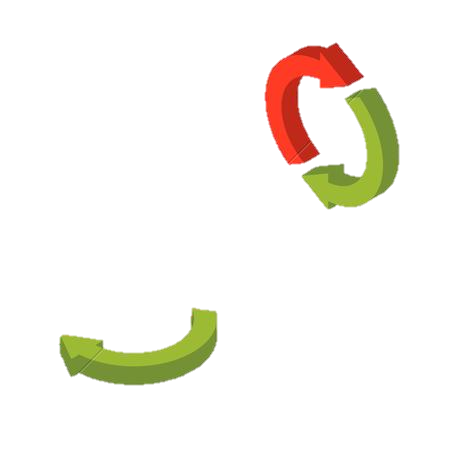
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Link |  |  |  |  |
| 1 |  |  | 0 |  |
| 2 |  | 0 |  | 0 |
| 3 |  | 0 |  | 0 |
| 4 |  | 0 |  | 0 |

Las matrices de transformación quedan de la siguiente manera:

La transformación homogénea que define el efector final en el sistema de coordenadas inercial queda como la multiplicación de todas las matrices de transformación.

# Ecuaciones diferenciales





La ecuación diferencial está compuesta por el jacobiano de velocidad angular y velocidad lineal del efector final.

Todas nuestras uniones son rotativas por lo que la velocidad angular está definida como

Velocidad angular global del efector final está compuesto por:

Considerando que todas las uniones son rotativas siempre será 1 y el eje de rotación está dado por

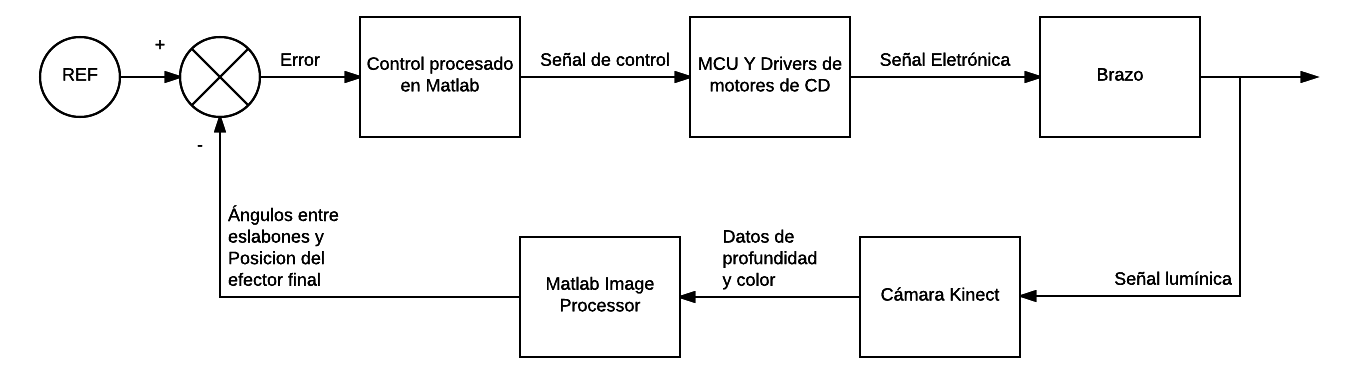
k es un vector columna unitario.

El Jacobiano angular general está dado por:

Calculamos los ejes de rotación para cada unión utilizando las transformaciones entre los eslabones de nuestro manipulador obtenidas al sacar los parámetros de Denavit Hartenberg.

No necesitamos el Jacobiano de velocidad angular directamente pero lo utilizamos para obtener el Jacobiano de velocidad lineal del efector final.

# Diagrama a bloques del sistema



# Esquemas de control

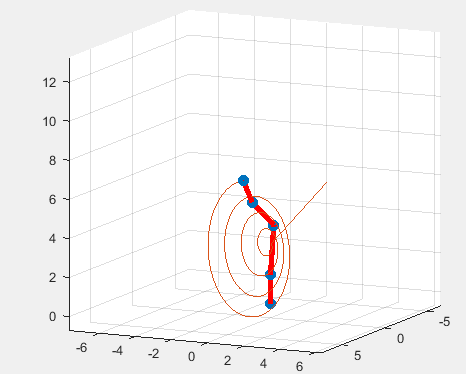
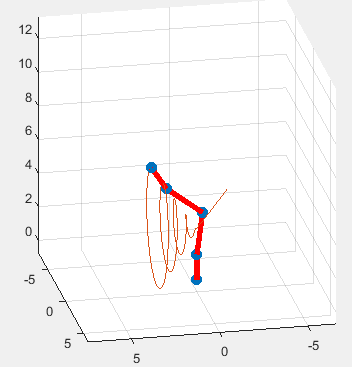
El primero es un control por retro de error para poder seguir referencias constantes y variables.

A partir del código de Matlab que codificamos creamos dos referencias base para comprobar su funcionamiento en simulación.

Ejemplo 1.: Seguimiento de referencia de una espiral

Referencia = [ 0.1\*t ; 0.1\*t\*sin(t) ; 3 + 0.15\*t\*cos(t) ]

Derivada de la referencia = [ 0.1 ; 0.1\*sin(t) + 0.1\*t\*cos(t) ; -0.15\*t\*sin(t) + 0.15\*cos(t) ];

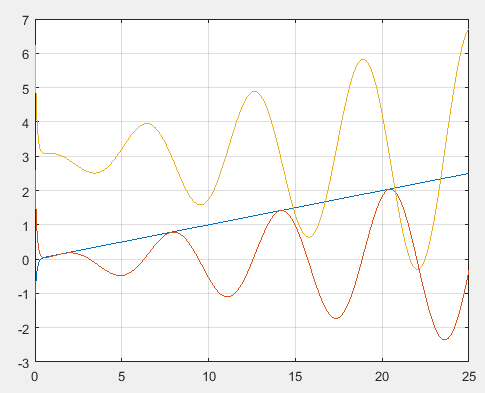


Ilustración de ejes con control z [amarillo], x[azul], y[rojo]

Ejemplo 2.: Seguimiento de referencias constantes

Referencia = [3; 1; 2];

Derivada de la referencia = [0; 0; 0];

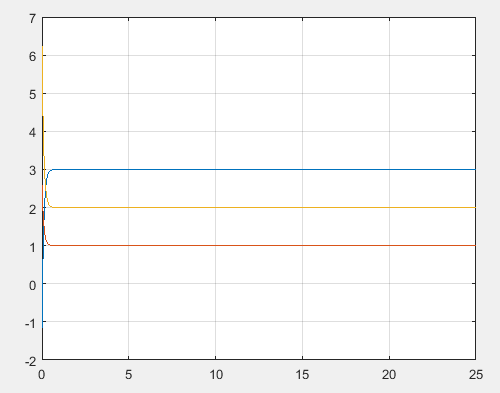
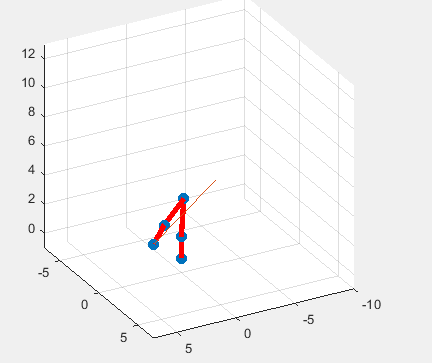
 

Ilustración de ejes con control z [amarillo], x[azul], y[rojo]