

Recibido: 19 de octubre de 2019
Aceptado: 16 de diciembre de 2019

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL
CINEMÁTICO CON OPEN-HARDWARE DEL MANIPULADOR DIDÁCTICO
PEGASUS DE MARCA AMATROL UBICADO EN TECNOACADEMIA
CÚCUTA**

**DESIGN AND IMPLEMENTATION OF A CINEMATIC CONTROL SYSTEM
WITH OPEN-HARDWARE OF THE PEGASUS DIDACTIC MANIPULATOR OF
THE AMATROL BRAND LOCATED IN TECNOACADEMIA CÚCUTA.**

MSc. Oscar Manuel Duque Suárez^{*,}, Ing. Jhonatan Ferreira Pacheco^{*},
MSc. Andrés Mauricio Puentes Velásquez^{**}**

*** Universidad De Pamplona**, Facultad de Ingenierías y Arquitecturas, Ingeniería
Mecatrónica
Autopista Internacional Vía Los Álamos - Villa Antigua, Villa del Rosario – Norte de
Santander
Tef: (57+7) 5706966 (Fax) - 5703742 (57+7) 5685303 Ext: 600
E-mail: omduke@hotmail.com, jhonatanferreira0616@gmail.com

**** SENA**, Tecnoacademia Cúcuta, Regional Norte de Santander.
Dirección, Ciudad, Departamento, País.
Teléfono y Fax, con indicativos internacional y nacional.
E-mail: apuentesv@sena.edu.co

Resumen: Tecnoacademia Cúcuta es un centro de formación didáctico, donde los estudiantes de bachillerato se les enseña el uso de los dispositivos tecnológicos, electrónicos y de diseño que se encuentran en sus instalaciones. Entre estos, cuentan con una célula de trabajo robotizado que consta de un manipulador robótico Pegasus de marca AMATROL que incluye su propio hardware y software proporcionado por el proveedor.

En el desarrollo de este trabajo se encuentra con el proceso en el que se diseñara un sistema de control cinemático modelado al robot Pegasus de AMATROL que funciona a base de motores DC con encoder para determinar su posición, los cuales se controlaran por medio de un dispositivo Open-Hardware realizado con Arduino, y una interfaz de control. Para esto se realizó el modelado del manipulador, y se calculó su cinemática directa e inversa para ser usado en un algoritmo generador de trayectorias y que controle la velocidad de cada articulación del manipulador, donde; cada actuador cuenta con un u propio controlador PID para garantizar el cumplimiento de la posición articular.

Abstract: Tecnoacademia Cúcuta is a didactic training center, where high school students are taught the use of technological, electronic and design devices found in their facilities. Among these, they have a robotic work cell consisting of a Pegasus robotic manipulator of the AMATROL brand that includes its own hardware and software provided by the supplier.

Keywords: Kinematic control, Open-Hardware, DC motor, robotic manipulator, trajectories.

1. INTRODUCCION

En este trabajo se realiza el diseño de un sistema de control cinemático para el manipulador robótico PEGASUS de la marca AMATROL que se implementó en la Tecnoacademia del SENA sede Cúcuta, Regional Norte de Santander; (J Plaza, M Núñez, 2017).

Para el diseño del sistema de control se estableció el modelado cinemático y dinámico del manipulador, robótico y de esta forma poder trabajar en el diseño de un controlador cinemático. Se usa principalmente, la cinemática inversa del robot (Reyes Cortes, 2011); Avanzada, ISSN: 1692-7257; pues en este proyecto se utiliza la posición final del manipulador como el punto de referencia del controlador cinemático y también se diseñó una aplicación para computador desde donde se dan las ordenes (Murillo-Soto, 2014), (T Velásquez, E Espinel & G Guerrero, 2016) y cuenta con una interfaz gráfica didáctica para que sea de simple uso para los aprendices de Tecnoacademia Cúcuta en el SENA.

2. MATERIALES Y MÉTODOS

2.1 Modelado Cinemático

El modelado cinemático y dinámico del manipulador robot PEGASUS de marca AMATROL es necesario para poder conocer el comportamiento del sistema, determinando las fórmulas que permitan hallar la localización final del efector final en función de los ángulos de sus articulaciones o viceversa, también su matriz Jacobiana y su modelo dinámico mediante la formulación de Lagrange (Gutierrez, 2013). Se emplean los dos tipos de modelado cinemático que son la cinemática directa y la inversa (Lopez, 2009), que estas explicadas a continuación.

2.1.1 Cinemática directa

(Barrientos, 2007) Con este modelado cinemático se obtiene la posición y orientación final del brazo manipulador, con respecto al sistema de referencia base, conociendo previamente los valores angulares de las articulaciones y longitudes morfológicas que lo componen.

Al hallar la cinemática directa, se puede modelar el sistema en una matriz de transformación homogénea con el método Denavit-Hartenberg;

el cual implementa un algoritmo para determinar esta matriz (Fernandez, 2015). Ver tabla 1.

Tabla 1: Matriz Denavit Hartenberg Amatrol

q	θ	d	a	alfa
1	q1	L1	0	-90
2	q2	0	L2	0
3	q3	0	L3	0
4	q4-90	0	0	-90
5	q5	L4	0	0

2.1.2 Cinemática inversa

Se realiza la cinemática inversa del manipulador AMATROL por el método geométrico, se escoge este método; dado que la morfología del manipulador permite realizar proyecciones, que facilitan el despeje de los valores articulares (Ollero, 2001); (L Fernández, L Mesa & W Pérez, 2017)

2.2 Interpolación de trayectorias

Se usan cuatro trayectorias para ir de una posición a otra con dos tipos de interpoladores. El interpolador trapezoidal o a tramos se usa en las trayectorias eje a eje, simultánea y coordinada. Y el interpolador cubico se usa para la trayectoria continua que traza una línea recta (Bautista, Guerrero, 2010); (J Pérez, J Castro, 2018).

2.3 Reconocimiento de elementos, actuadores y sensores en el manipulador Pegasus Amatrol.

Se puede observar que el manipulador PEGASUS de AMATROL cuenta con seis motores que se encargan mover cada una de las articulaciones y la pinza de agarre. Esos motores tienen integrados un encoder que sirve para medir la posición angular, así que se tendrían seis encoders por todo el sistema (CALENDAMAIA, 2014). El brazo robot PEGASUS, de la figura 2; también tiene cinco sensores infrarrojos ubicados en cada una de las cinco articulaciones rotatorias lo que implementa un sistema de iniciación a los encoders (Carbonetto, 2010) y así siempre las mediciones sean iguales ya que la lectura de los encoders se hacen mediante un contador de pulsos y se debe determinar en qué posición se debe empezar a contar los pulsos que luego serán escalados a la medición del ángulo (Cruz, 2018) y (Encoder

Products Company, 2016). Partes del manipulador :

- Motores
- Encoder
- Sensores infrarrojos



Fig. 1 Manipulador PEGASUS. Fuente: Propia.

El control de los motores se realiza con una conexión maestro esclavo entre un Arduino mega y tres Arduino, como se ve en la figura 3; uno para que cada uno realizar la medición de la posición de las cuatro articulaciones que se mueven en simultáneo.

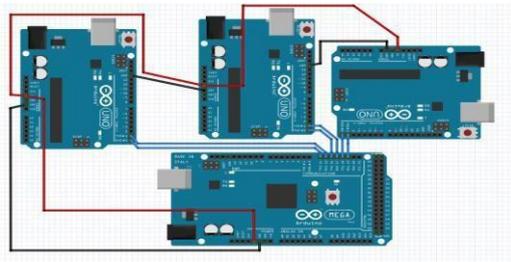


Fig. 2 Comunicación maestro-esclavo Arduino MEGA y Arduino UNO.

También se selecciona el Modulo Control Motor L_298n Puente H Motor Driver Arduino para la etapa de potencia de los motores que son de 24V. Ver figura 4.



Fig. 3 Modulo Control Motor L_298n Puente H Motor Driver Arduino. Fuente: ifitronica.

Se sintonizan controladores clásicos de tipo PID, PI y P para cada articulación (Fusario- Crotti, 2013). En este proyecto se evalúa el uso de un método empírico debido a la gran complejidad que tiene el modelo dinámico establecido por eso lo primero que se consideró para determinar el método a aplicar en este caso fue el análisis de la respuesta del sistema en lazo abierto (Spong, 2010). Como ejemplo, ver la figura 5 donde se muestra el controlador de la articulación 4.

Usando el segundo método Ziegler-Nichols. Sin embargo, no dieron tan buenos resultados, así que se ajustaron las constantes de los controladores de forma manual de tal modo que se consiguiera un resultado deseado (R C González, 2011).

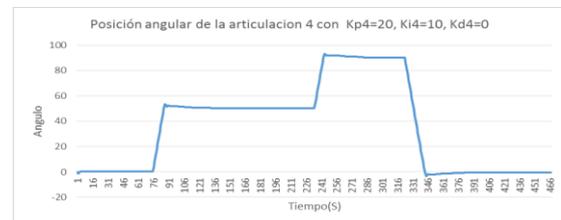


Fig. 4 Grafica controlador posición Articulación 4. Fuente: Propia

2.4 Diseño de algoritmo del control cinemático y HMI

Se diseña el algoritmo para el control desde el computador por medio de un aplicativo donde se muestre en una interfaz HMI didáctica las opciones para realizar la secuencia de movimientos del manipulador robótico. Teniendo que realizar la selección del software donde se pueda generar el aplicativo y especificando con mayor detalle las pautas que debe seguir el algoritmo (Pablo, 2014), que finaliza en él enviaron de datos al dispositivo de control que aplicara el controlador diseñado.

Primero se toman los valores de la posición y orientación final del manipulador ya sea por medio de coordenadas XYZ y orientaciones de cada efector final o por medios los ángulos de cada articulación a través de la interfaz gráfica (Cuenca-Salt, 2005) y (P.M. TAYLOR, 1992).

Estos valores serán almacenados y visualizados en una tabla que definirá el orden del movimiento por los diferentes puntos que el usuario quiera ingresar con todos los parámetros

necesarios para realizar la trayectoria entre punto y punto.

Para desarrollo de la interfaz y toda la programación necesaria para el controlador cinemático y generador de trayectorias se utilizó el IDE de NetBeans usando lenguaje de programación en java (García., 2000) y (Perry, 2019). El resultado obtenido fue la interfaz que se visualiza en la figura 6, que es bastante intuitiva y fácil de usar para el público en cuestión (García González, 2016).

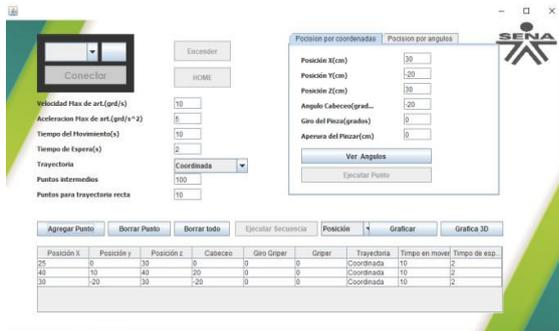


Fig. 5. Interfaz gráfica del algoritmo.
Fuente: Propia:

2.5. Implementación del sistema de control

Se plantea primero la estructura que compone la conexión de los diferentes componentes electrónicos con la computadora y se hace mediante la tarjeta de control con un esquema de conexión como se muestra en la figura 7.

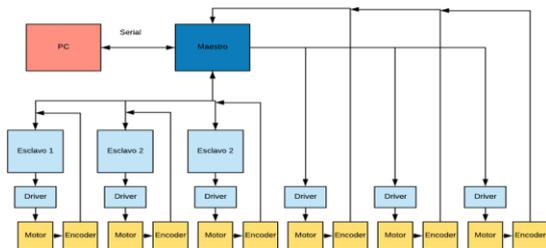


Fig. 6. Diagrama general del nuevo controlador
Fuente: Propia.

Para implementar el sistema de control cinemático fue necesario diseñar una tarjeta de conexión que funcionara como un accesorio que acoplara los cuatro Arduino y los conectores de los motores y sensores del brazo robot, incluyendo las conexiones de los Drivers de los motores. Así se realizan las conexiones dibujando las líneas cumpliendo con la norma IPC 2221B donde se especifican medidas para el

paso de la corriente. También se encapsula el circuito para darle más protección en su uso dado como resultado la tarjeta electrónica mostrada en la figura 8.

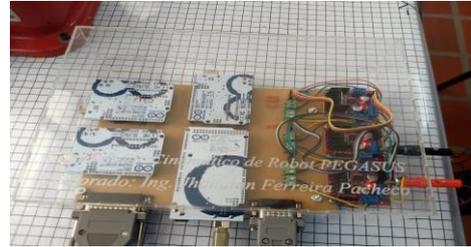


Fig. 7 Circuito de conexión físico para el control de mando de actuadores.
Fuente: Propia

Se implementa una lona donde se puede encontrar una cuadrícula de un centímetro por cuadro y el punto central del sistema de referencia con los ejes X y para poder facilitar la medición, como se ve en la figura 9.

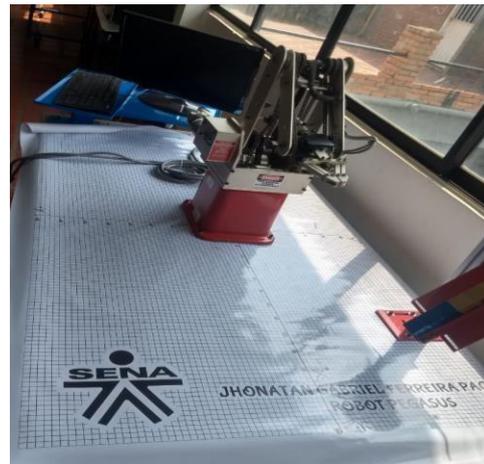


Fig. 8 1 Mesa de trabajo del manipulador PEGASUS con la lona de medición.
Fuente: Propia

3. RESULTADOS Y DISCUSIÓN

Para la validación de la cinemática directa e inversa se procede a generar pruebas, que harán al brazo ejecutar una posición en el plano XYZ, el cual fue adecuado con una lona hecha de tela PVC e impresa con una cuadrícula de un centímetro para una mejor práctica. Ver las figuras 10 y 11 para ver el error de los modelos cinemáticos directo e inverso



Fig. 9 Porcentaje de error histograma, con cinemática directa. Fuente: Propia

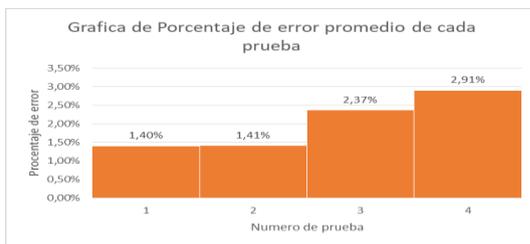


Fig. 10 Porcentaje de error histograma, con cinemática inversa Fuente: Propia

3.1 Validación del Interpolador trapezoidal

El interpolador trapezoidal se usó para las trayectorias en las que se ordenaba al brazo robot ir de un punto a otro sin pasar por posiciones intermedias por eso se usó en las trayectorias eje a eje, coordinad y simultanea ya que irán de un punto inicial a un punto final directamente, ejecutando la trayectoria seleccionada.

3.2 Validación trayectoria eje a eje y simultanea

Para las trayectorias eje a eje y simultánea se puede hacer una validación similar ya que utilizan el mismo interpolador trapezoidal a la que no se le ingresa el tiempo final del movimiento, sino que este determina el tiempo que dura cada movimiento a partir de la velocidad máxima y la aceleración ingresada.

Estas dos trayectorias se diferencian una de la otra en que el eje a eje siempre realiza el movimiento de una articulación a la vez y nunca dos o más al mismo tiempo por eso, mueve cada articulación de la posición inicial hacia la final en orden. Por otro lado, la trayectoria simultanea

mueve todas las articulaciones al mismo tiempo sin embargo el tiempo que dura cada movimiento es distinto por eso no llegan todas al punto final de igual forma.

3.3 Validación trayectoria coordinada.

La trayectoria coordinada utiliza el interpolador trapezoidal en el que el usuario define el tiempo inicial y el final del movimiento por eso el algoritmo tiene unas pequeñas variaciones para lograrlo. Como ya se sabe esta trayectoria mueve todas las articulaciones desde la posición inicial a la finas de forma coordinada ya que el movimiento es al mismo tiempo y todas las articulaciones tardan el mismo tiempo. Por esto es la trayectoria entre dos únicos puntos que se recomienda utilizar pues se visualiza un movimiento coordinado (Vera, 2017).

3.4 Validación del Interpolador Cúbico

Una de las maneras más optimas de validar la trayectoria de un manipulador, es sometiéndolo a ejecutar una trayectoria en línea recta (Vera Luzuriaga, 2017), por eso entre dos puntos del plan cartesiano se adapta para suspender una cuerda la cual hace de línea recta y al probar el brazo tiene que ejecutar estas coordenadas de una manera muy exacta, ver figuras 12.

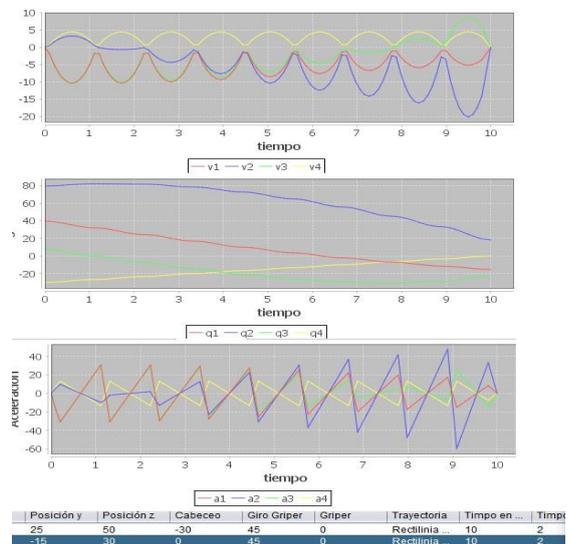


Fig. 11. Trayectoria en línea recta en el HMI. fuente: Propia

3.5. Guías para el uso y aprendizaje.

Para la generación de guías de aprendizaje para el uso del manipulador robot PEGASUS se utiliza el formato de guías proporcionado por las oficinas del SENA el cual utiliza una metodología de competencias para ir cumpliendo los propósitos de las guías.

4. CONCLUSIONES

- Al establecer el modelo cinemático del manipulador robot se logra tener la matemática necesaria para el control cinemático usado en este proyecto. Pero también se puede usar el modelo dinámico que se estima para sintonizar otro tipo de controlador que puede ser más preciso y robusto.
- Al comparar los resultados obtenidos por los controladores sintonizados a través del segundo método de oscilación de onda Ziegler-Nichols se observa que en ocasiones producen varias oscilaciones antes de estabilizarse en el Setpoint. Por tanto, es necesario realizar ajustes a los valores obtenidos de las ecuaciones basados en los principios de influencia de cada constante en la salida del bloque PID, en la mayoría de las articulaciones solo un control proporcional cumplía satisfactoriamente con la función esperada (Vivas Albán, 2011).
- El Algoritmo con el que se realizó el control cinemático y de las trayectorias fue correctamente realizado a través de la creación de una aplicación ejecutable desde el ordenador con una interfaz gráfica y que fue programado usando el lenguaje JAVA lo que permite un manejo intuitivo para la operación del robot. Esta aplicación es totalmente abierta a configuraciones por cualquier persona que conozca de ese tipo de programación y se espera que se pueda seguir mejorando e integrando nuevas funciones.
- Por medio de la implementación del sistema de control que se diseñó en este proyecto se comprueba que la aplicación y controladores programados cumplían sus funciones correctamente en cuanto a la forma en que efectuaba los movimientos el manipulador PEGASUS, sin embargo, siempre es posible optimizar el sistema durante este procedimiento para mejorar las calidades del servicio.
- A partir de la validación del controlador cinemático de las trayectorias se observa que se

trabaja de una forma más cómoda usando la trayectoria coordinada ya que con esta se puede establecer perfectamente el tiempo de ejecución del movimiento al igual que la trayectoria continua, pues en las trayectorias eje a eje y simultánea el tiempo del movimiento lo definen la aceleración y velocidad máxima que ingresa el usuario en este caso, aunque eso puede ser cambiado y se puede usar el mismo interpolador que en la trayectoria coordinada dándole el tiempo que se desea que tarde en ejecutar el movimiento cada articulación (Bullo-Cortés, 2009).

6. REFERENCIAS

- Barrientos. (2007). Fundamentos de robótica segunda edición.
- Bautista Peña, G. J., Guerrero Guadarrama, E. E., & Jiménez Ledesma, A. (2010). Rehabilitación de un robot manipulador de 5 GDL "Pegasus" para uso didáctico derivado del proyecto de investigación. Mexico D.F.: tesis de grado de ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA.
- CALENDAMAIA. (2014). NetBeansNetBeans. GANBETA. Recuperado el 21 de enero de 2019, de <https://www.genbeta.com/desarrollo/netbeans-1>
- Carbonetto, S. H. (2010). Tutorial de Octave. Buenos Aires.
- Cruz, J. M. (2018). Encoder, Aplicaciones en la Industria. <http://www.logicbus.com.mx/encoder-app-industria.php>.
- Encoder Products Company. (16 de junio de 2016). Recuperado el 20 de abril de 2018, de <http://encoder.com/blog/encoder-basics/que-es-un-encoder/>
- Fernandez, V. (2015). Desarrollo de una aplicación gui para control de actuadores basado en un sistema empujado (Tesis de pregrado). Madrid, España.
- García González, A. (23 de noviembre de 2016). panamahitek. Obtenido de Arduino + Java: Fácil y Rápido: <http://panamahitek.com/arduino-java-facil-y-rapido/>

- García., J. (2000). Aprenda Java como si estuviera en primero. San Sebastian, España.
- Gutierrez, G. (2013). Estudio cinemático del cuerpo humano mediante kinect (Tesis de pregrado. Madrid, España: Universidad Politécnica de Madrid Escuela Técnica de telecomunicaciones.
- JEG Plaza, MAR Nuñez, (2017) Formación en competencias específicas para la industria del software colombiano. Experiencias del uso del aprendizaje basado en proyectos. Revista Colombiana de Tecnologías de Avanzada, ISSN: 1692-7257.
- J Pérez, J Castro. (2018). LRS1: Un robot social de bajo costo para la asignatura “Programación 1”. Revista Colombiana de Tecnologías de Avanzada, ISSN: 1692-7257.
- L Fernández, L Mesa, W Pérez (2017) Investigación formativa para estudiantes de ingeniería utilizando robótica. Revista Colombiana de Tecnologías de Avanzada, ISSN: 1692-7257.
- Lopez, L. (2009). Modelación y simulación dinámica de un brazo robótico de 4 grados de libertad para tareas sobre un plano horizontal (Tesis de pregrado). Lima, Perú: Pontificia Universidad Católica del Perú.
- Murillo-Soto. (2014). Diseño del programa de control para una celda de manufactura flexible didáctica.
- Ollero, A. (2001). Robótica: Manipuladores y robots móviles. . Barcelona: <https://doi.org/10.1111/1755-0998.12480>.
- O Suarez, C Vega, E Sánchez, A Pardo. (2018) Degradación anormal de p53 e inducción de apoptosis en la red P53-mdm2 usando la estrategia de control tipo pin. Revista Colombiana de Tecnologías de Avanzada, ISSN: 1692-7257
- Pablo. (1 de Noviembre de 2014). Instalar PyDev en Eclipse. Recuperado el 21 de Enero de 2019, de Protoinformatic: <http://protoinformatico.com/language/es/2014/11/01/instalar-pydev-en-eclipse/>
- Perry, J. S. (21 de Enero de 2019). Conceptos básicos del lenguaje Java. Obtenido de IBM: <https://www.ibm.com/developerworks/sa/java/tutorials/j-introtojava1/index.html>
- Vera Luzuriega, J. G. (2017). Control de trayectoria de la simulación de un brazo robot de 5 grados de libertad. valencia: tesis de grado.
- Vera, J. (2017). ontrol de trayectoria de la simulación de un brazo robot de 5 grados de libertad, controlado mediante la plataforma C2000 Piccolo LAUNCHXL-F28027F (Tesis maestría). . Valencia, España: Universidad Politecnica de Valencia.
- Reyes Cortes Fernando , 2011. Robótica: Control De Robots Manipuladores Editorial: S.A. MARCOMBO
- Fusario Rubén Jorge; Patricia S. Crotti; Andres P.M. Bursztyn , 2013. Teoría De Control Para Informáticos Editorial: S.A. MARCOMBO
- Spong Mark W.; Seth Hutchinson , 2010. Robot Modelling And Control Editorial: JOHN WILEY & SONS INC.
- T Velásquez, E Espinel, G Guerrero (2016). Estrategias pedagógicas en el aula de clase. Revista Colombiana de Tecnologías de Avanzada, ISSN: 1692-7257
- K S FY; R C González, 2011. Robótica Control Detección Visión e Inteligencia. Editorial: MCGRAW-HILL
- Cuenca Lacruz Ángel; Salt Llobregat, 2005. Automática Industrial Y Control Editorial: UNIVERSIDAD POLITECNICA
- P.M. TAYLOR , 1992. MANUAL DE CONTROL ROBOTICO, Editorial: CEAC.
- Vivas Albán Oscar Andrés, 2011. Diseño Y Control De Robots Industriales. Teoría Práctica. Editorial: ELALEPH Bullo Francesco, Cortés Jorge, Martínez Sonia, 2009. Distributed Control Of Robotic Networks Ebook. Editorial: Princeton University Press