

Dispositivo de bajo costo para la enseñanza-aprendizaje de la robótica Low-cost device for teaching-learning robotics

C. R. Alcántara-Sánchez ^a, M. S. Islas-Ortiz ^b, N. Hernández-Oliva ^{1,*}, M. Alejandre-Flores ¹, M. Pedroza-Villalba ^d

^aInstituto Politécnico Nacional ESIME Zacatenco, 07738, Ciudad de México, México

^bInstituto Politécnico Nacional UPIITA, 07340, Ciudad de México, México.

^cInstituto Politécnico Nacional CECyT 2 Miguel Bernard, 11260, Ciudad de México, México

^dHwaseung Automotive Alabama LLC, 36330, Enterprise, Alabama, Estados Unidos

Resumen

En este trabajo se presenta la implementación de un dispositivo, integrado por un modelo de robot a escala obtenido mediante impresión 3D de tipo FDM (Fuse Deposition Modeling) tomando como base un robot ABB IRB 460 industrial y una tarjeta programable Arduino UNO R3, el cual puede ser utilizado para la enseñanza de la robótica a nivel superior. Para ello se obtuvo el modelo cinemático del robot a escala, el cual se usó como dispositivo de prueba cuando no es posible tener acceso a un robot real. Por ese motivo se programó en Matlab el cálculo de la cinemática directa propia del robot empleando el método de Denavit-Hartenberg, misma que puede ser programada en la tarjeta Arduino conectado al robot. Una vez concluidas las pruebas, se demostró que los resultados obtenidos son similares. Además, es posible realizar dichas validaciones a la par, a partir de la modificación de los parámetros del robot en cada uno de los respectivos programas, permitiendo a los alumnos la interacción con un modelo a escala y su manipulación cuando no es posible tener el robot real. También permite incluir otro tipo de programas conforme avance su curso y de igual forma ser validados con este mismo esquema desarrollado.

Palabras Clave: Robot, Cinemática, Arduino UNO, Dispositivo, Enseñanza.

Abstract

This work presents the implementation of a device, integrated by a scaled robot model obtained by 3D printing FDM (Fuse Deposition Modeling) based on an industrial robot ABB IRB 460 also an Arduino UNO R3 programmable board, which can be used for teaching robotics at middle-superior and superior level. For this purpose, the kinematic model of the scaled robot was obtained, which was used as a test device when it is not possible to get access to a real robot. For this reason, the calculation of the robot's own direct kinematics was programmed in Matlab using the Denavit-Hartenberg method, which can be programmed on the Arduino card connected to the robot. Once the tests were completed, it was shown that the obtained results are similar. In addition, it is possible to develop such validations at the same time, from the modification of the robot parameters in each of the respective programs, allowing students to interact with a scale model and its manipulation when it is not possible to handle a real robot. It also allows the inclusion of other types of programs as the course progresses and, in the same way, to be validated with this same developed scheme.

Keywords: Robot, Kinematics, Arduino UNO, Device, Education

1. Introducción

Derivado de la pandemia que se ha estado viviendo, el uso de nuevas tecnologías, como es la denominada STEM (acrónimo en inglés de Science, Technology, Engineering and Mathematics; en español Ciencia, Tecnología, Ingeniería y Matemáticas), ya que es un área en continuo desarrollo. Es por este

motivo, que resulta interesante el desarrollo de plataformas diseñadas para la enseñanza en esta área sea de forma didáctica y con la facilidad de poder interactuar de forma remota a través de una conexión a internet.

Algunas aplicaciones interesantes para este trabajo, destacan las siguientes debido a que han sido aplicadas en la en-

* Autor para correspondencia: nhernandez@ipn.mx

Correo electrónico: alcantarachristianr@gmail.com (Christian Rodrigo Alcántara Sánchez), shamedarduino@gmail.com (Mihael Shamed Islas Ortiz), nhernandez@ipn.mx (Noemi Hernández Oliva), malejandref@ipn.mx (Marisol Alejandre Flores), apdl.dn88@gmail.com (Martin Pedroza Villalba).

Historial del manuscrito: recibido el 21/10/2022, última versión-revisada recibida el 21/10/2022, aceptado el 07/11/2022, publicado el 11/11/2022. **DOI:** <https://doi.org/10.29057/icbi.v10iEspecial5.10101>



señanza de la robótica. En el primer trabajo, se presenta una metodología para la elección de robots educativos a partir del conocimiento de los parámetros que consideraron más importantes en la selección de una plataforma robótica a través de un amplio estudio bibliográfico, como se detalla en (Barrientos *et al.*, 1997). Por otro lado, en (Wang y Song, 2021) se presenta una plataforma de competición, la cual está integrada por una interfase hombre máquina en la que es posible observar la imagen y posición en tiempo real, mientras que el robot se programa por medio de la aplicación Scratch todo bajo el esquema de educación STEAM. Mientras en (Arenas-Rosales *et al.*, 2021) robots virtuales y los sistemas de control se implementaron con software de código abierto y plataformas de hardware de bajo costo, mediante la implementación de un robot virtual para el aprendizaje práctico de los métodos de control del manipulador robótico UR5. En (Chatzopoulos *et al.*, 2022) desarrollaron una plataforma robótica de código abierto para aplicaciones educativas bajo el Modelo de Aceptación de la Tecnología (en inglés TAM), específicamente en términos de facilidad de empleo para profesores. El desarrollo de "SPArK-Bot", un robot educativo que combina la versatilidad de la Raspberry Pi con la simplicidad del Arduino, se puso en práctica para tratar de mejorar las habilidades de programación en los alumnos y elevar su aprendizaje en esta área de estudio como se muestra en (Parikh *et al.*, 2022). Para (Mohanavel *et al.*, 2022) se diseña un robot móvil controlado por medio de un protocolo de comunicación bluetooth entre un teléfono inteligente con sistema operativo android y el robot, con lo cual es posible que los alumnos repliquen el propio a bajo costo. En (Feijoo-Almonacid y Rodríguez-Garavito, 2022) pretende animar a los niños colombianos a desarrollar habilidades STEM, incentivándolos a la obtención de resultados científicos, tecnológicos, matemáticos y artísticos; para ello se emplea un robot-elefante (Eli) y una interfaz de usuario llamada "Experiencia Chamali" formada por tres módulos. Finalmente, en (Zhong *et al.*, 2022) desarrollan una propuesta para el seguimiento de objetivos submarinos por medio de un árbol de decisión de detección centralizado en menos ramas, rápida convergencia y alta pureza mediante la realización de experimentos de seguimiento de objetivos guiados por peces robóticos.

En el presente trabajo se implementa una plataforma que está integrada por un brazo robot impreso en 3D escala 1:7 y un controlador programado en una placa de desarrollo Arduino, con el fin de permitirle al usuario la validación de diferentes aspectos involucrados en el estudio de la robótica, ya que puede ser adaptable por medio de la reestructuración de la programación de la tarjeta desde Simulink de Matlab; además esta misma plataforma puede permitirle al usuario final la emulación del funcionamiento de un robot industrial sin ser necesario contar con el real. La organización del documento es la siguiente: en la Sección 2 se describe las características del robot seleccionado y su modelo cinemático programado en Matlab, en la Sección 3 se muestra el desarrollo de la programación en Arduino para la manipulación de los eslabones del robot; en la Sección 4 se analizan los resultados obtenidos de las diferentes pruebas realizadas; finalmente, en la Sección 5 se exponen las conclusiones y trabajos a futuro.

2. Diseño propuesto

El dispositivo propuesto, como ya se ha descrito, está compuesto por el robot impreso en 3D a escala y del programa que se encuentra descargado directamente en el Arduino, que a su vez sirve como un proceso de validación para los cálculos obtenidos en el software Matlab R2019a. La selección de los componentes tanto brazo robótico como tarjeta fueron a partir de una selección de modelos con base a accesibilidad, complejidad estructural en el caso de la impresión 3D y de programación. El modelo de robot mencionado fue seleccionado de los modelos disponibles en páginas como Thinkgiverse y similares. El modelo de robot fue tomado de EEZYbotARM MK2 by daGHIZmo en Thinkgiverse con sus respectivas actualizaciones, mientras la tarjeta programable elegida dependió del análisis de tarjetas a bajo costo similares, como las que se presentan en la Tabla 1 siguiente:

Tabla 1: Comparativo de tarjetas electrónicas

Tarjeta	Costo	Características
Arduino (Arduino, 2022)	< 45 dólares	$V_{IN} = 7 - 12V$ I/O digitales 14 (6 PWM) Entradas analógicas 6 Programación en C
Hercules LaunchPad (TI, 2022)	> 50 dólares	$V_{IN} = 5 - 12V$ GPIO 45 Entradas analógicas 41 Programación en C
Ophyra (Intesc, 2022)	> 70 dólares	$V_{IN} = 3,3 - 5V$ GPIO 34 9 puertos GPIO Programación en C
STM32F406VG Discovery Kit (STM, 2022)	> 50 dólares	$V_{IN} = 5 - 12V$ 9 puertos GPIO 3 ADC Programación en C

Como ya se mencionó, se optó por la tarjeta Arduino UNO R3 debido a sus características y prestaciones de software y de hardware, si bien es una tarjeta a bajo costo, eso no significa que exista alguna limitante o no sea suficiente para desarrollar la programación del robot, ya que puede considerarse como un dispositivo de entrada, en otras palabras básico con opción de ser mejorado o sustituido por alguna otra tarjeta electrónica.

Para empezar, el estudio de la robótica es primordial en varias áreas de la ingeniería, sin embargo, en muchas ocasiones su estudio y análisis se limita al uso de simuladores propios para ello y solo en algunos casos con el robot físico a través de la programación de puntos por medio de un control manual programable conocido como teach pendant. Otra cuestión de interés cuando se estudian estos temas tienen que ver con el tipo de configuración del robot, el cual se presenta ampliamente en (Kelly y nez, 2003), (Barrientos *et al.*, 1997), (Reyes, 2011) por mencionar algunos autores. Para el presente artículo se propuso la implementación de un dispositivo, integrado por el robot a escala 1:7 impreso en 3D, disponible en (Franciscone, 2018), el cual corresponde con el modelo de robot industrial de la marca ABB modelo IRB 460 (ver Figura 1), del cual se pueden ve-

rificar todas sus características técnicas en (ABB, 2022). En la Figura 2 se muestra el robot impreso utilizado.



Figura 1: Robot industrial IRB460 (ABB, 2022)

del robot para n grados de libertad (GDL) utilizando el algoritmo de Denavit Hartenberg (DH) para la obtención del modelo cinemático directo, debido a que se establecen los sistemas de referencia para cada uno de los eslabones con los que cuenta el robot. El primer paso fue obtener los parámetros DH del robot seleccionado, usando la ecuación (1) siguiente:

$$DH = \begin{bmatrix} c(\theta_i) & -c(\alpha_i)s(\theta_i) & s(\alpha_i)s(\theta_i) & a_i c(\theta_i) \\ s(\theta_i) & c(\alpha_i)c(\theta_i) & s(\alpha_i)c(\theta_i) & a_i s(\theta_i) \\ 0 & s(\alpha_i) & c(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

La cual relaciona los 4 parámetros del robot (θ , d , a y α), los cuales corresponden con:

1. Rotación alrededor del eje, z_i un ángulo θ_i .
2. Traslación a lo largo de z_i una distancia d_i ; vector $d_i(0, 0, d_i)$.
3. Traslación a lo largo de x_i una distancia a_i ; vector $a_i(a_i, 0, 0)$.
4. Rotación alrededor del eje, x_i un ángulo α_i .

Una vez obtenidos los parámetros, el siguiente paso corresponde con su codificación en el software Matlab versión 2019a con el fin de verificar que los datos calculados corresponden con los previamente obtenidos para el robot. Para ello, en el diagrama de flujo de la Figura 3 muestran los pasos que se siguieron para su codificación.



Figura 2: Modelo CAD e impresión 3D

Existen principalmente dos modelados aplicables a la robótica, los modelos respecto a la cinemática de robot y los referentes a la dinámica del robot. Este trabajo está enfocado en el modelo cinemático de un robot de 3 grados de libertad (3GDL) con configuración 3R, es decir, sus articulaciones son de tipo revoluta, dado que es la configuración con mayor uso en la literatura especializada como ejemplo para la enseñanza de la robótica. Por lo que, es de interés conocer la relación que existe entre la posición y orientación del extremo final del robot con los valores que toman sus articulaciones.

La forma de solución consiste en tratarlo como dos problemas, el primero llamado problema cinemático directo y el segundo llamado problema cinemático inverso. El enfoque inicial para esta propuesta consiste en determinar respecto a ciertas coordenadas de referencia cuál es la posición y orientación del extremo final del robot, considerando las medidas, la geometría y los valores de las articulaciones del robot.

De acuerdo con la literatura, para obtener este modelo se tienen dos opciones diferentes, el método geométrico y los basados en cambios de sistemas de referencia; para este artículo se seleccionó el método de matrices de transformación homogénea, ya que permite obtener el modelo cinemático directo

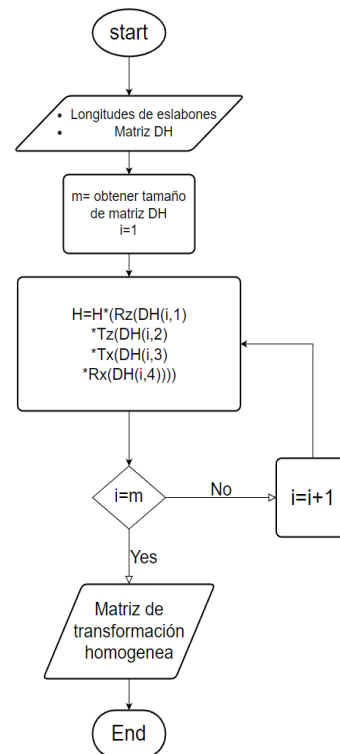


Figura 3: Diagrama de flujo del programa realizado

En general, en el programa principal se declaran como variables simbólicas que están asociadas a las rotaciones con el eje z , además se introducen como constantes las longitudes de cada uno de los eslabones. Posteriormente, para obtener los parámetros de Denavit Hartenberg (DH), que definen al robot por medio de la matriz DH, estos son introducidos a la función de Matlab *forwardKinematicsDH*, con esta se obtiene la matriz

de transformación homogénea que se requiere para obtener el resultado del robot, obteniendo la matriz correspondiente con el robot seleccionado, los parámetros DH, se muestran en la ecuación (2) siguiente:

$$DH = \begin{bmatrix} \Theta_i & d_i & a_i & \alpha_i \\ q_1 & l_1 & 0 & \pi/2 \\ q_2 + \pi/2 & 0 & l_2 & 0 \\ q_3 & 0 & 0 & 0 \\ q_4 - \pi/2 & 0 & l_2 + l_4 & 0 \end{bmatrix} \quad (2)$$

2.1. Programación en Arduino

Para la electrónica se utilizó una fuente de 6,3V a 3A, así como dos capacitores de 15000 μ F a 16 V conectados en paralelo. El robot a escala trabaja de forma interactiva a través de 3 botones pulsadores (en configuración *pull down*) conectados a la tarjeta Arduino UNO, cada botón pulsador hace que un eslabón del robot recorra todo su espacio de trabajo de acuerdo al esquema de la Figura 5.



Figura 4: Dispositivo propuesto para la enseñanza de la robótica

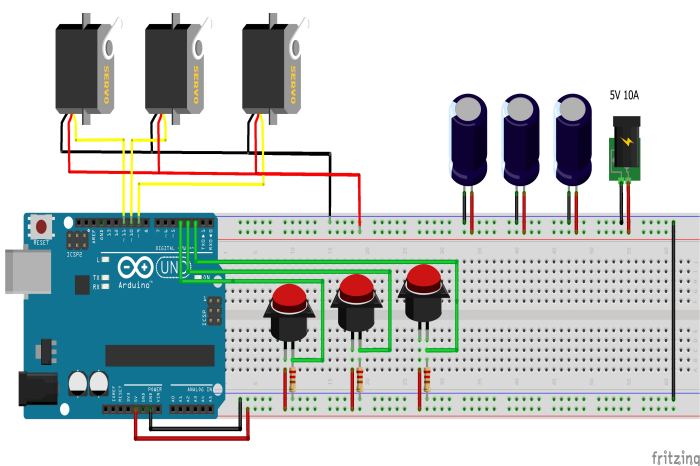


Figura 5: Layout de conexiones del dispositivo

Con el fin de tener una comparación entre los resultados obtenidos a partir de Matlab, se realizó la implementación del

mismo problema utilizando Arduino, el cual se presenta en la Figura 6 donde se muestra una parte del código. Dentro de la programación del Arduino se establecieron y midieron valores angulares de los motores, los cuales nos representan la posición de cada eslabón del robot a escala, con esto comprobamos la cinemática directa obtenida en MATLAB. Además, se realizó una comunicación serial en la IDE de Arduino, en donde el monitor serial muestra que eslabón se está moviendo y en que ángulo se encuentra el motor de dicho eslabón. Por último, se consideró dentro de la programación que el robot a escala no colisione entre el eslabón 2 y 3, ya que estos eslabones pueden colisionar entre ellos si el robot llega a entrar en algún punto de singularidad, para ello se puede conocer probables puntos donde estos podrían colisionar a partir del conocimiento del espacio de trabajo propio del robot que se encuentra en la hoja de especificaciones del robot real.

```
49 #include <Servo.h> //Include the Servo Library
50 int degree=0;
51 String words;
52 Servo servol; // "Servos" objects are created
53 void setup() {
54 // put your setup code here, to run once:
55 Serial.begin(9600);
56 Serial.println("Servos - Arduino");
57 //servol.attach(5); //Default
58 servol.attach(5,544,2400); // 544 a 2400 equivale con 0 a 180
59 servol.write(0);
60 }
```

Figura 6: Programa IDE de Arduino

3. Pruebas y resultados obtenidos

La obtención de la matriz de transformación que describe al robot en el entorno del dispositivo que en un principio fue programada en Matlab fue programada en el IDE de arduino, con lo que se corroboró que es posible implementarse para ambos casos, pero, en Arduino pueden enviarse las posiciones a diferentes puntos dados con el fin de poder validar de forma física como es el movimiento de cada uno de los eslabones que integran al robot. Los cuales están dados por los ángulos que alcanzan cada uno durante la realización de una tarea, en este caso el posicionamiento del robot.

En la Figura 7 se muestra el entorno de programación que se tiene implementado.

```
Editor - D:\Nube matlab\DH.m
DH.m
1 - clear
2 - clear
3 - syms q1 q2 q3 q4
4 - L1 = 105;
5 - L2 = 104;
6 - L3 = 152;
7 - L4 = 30;
8 - q = [q1; q2; q3; q4];
9
10
11 DenH = [0 0 0 0
12         q(1) L1 0 pi/2
13         q(2)+pi/2 0 L2 0
14         q(3) 0 0 0
15         q(4)-pi/2 L3+L4 0 0];
16
Command Window
[ 0, 0, 0, 0]
[ q1, 105, 0, pi/2]
[ q2 + pi/2, 0, 104, 0]
[ q3, 0, 0, 0]
[ q4 - pi/2, 182, 0, 0]
fx >>
```

Figura 7: Programa en Matlab para la obtención de la cinemática del robot

El resultado obtenido de la implementación del programa en el Arduino UNO, con la posibilidad de mostrar la posición que se tienen cada uno de los eslabones del robot desde el monitor serie de su propia IDE, como se ve en la Figura 8.

```

pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:500 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:473 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90
pot1:501 servodeg_1:90 pot2:472 servodeg_2:90 pot3:472 servodeg_3:90

```

Figura 8: Monitor serie de Arduino para observar las posiciones de los eslabones del robot

Para probar el correcto funcionamiento del robot se realizan lecturas de las señales enviadas a los servomotores, como se muestra en la Figura 9, en el cual el eje x representa el eje de tiempo dado en s y el eje y representa la amplitud del valor de frecuencia ($1kHz$); esto con ayuda de MATLAB, ya que, debido a que el robot se encuentra en lazo abierto, fue necesario validar que los trenes de pulso correspondieran a los ángulos de los eslabones mostrados en el monitor serie de la IDE de Arduino.

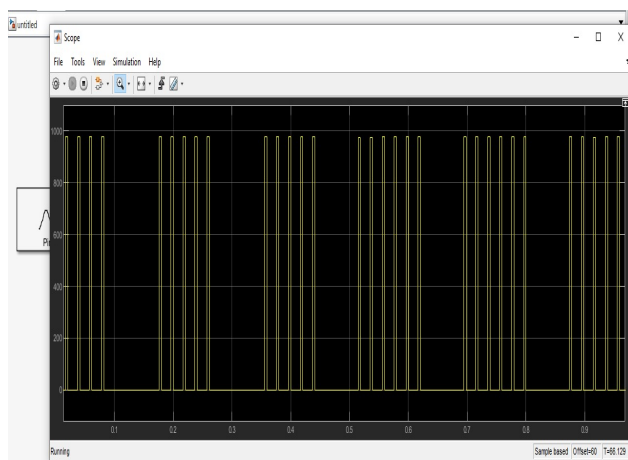


Figura 9: Señal para el control de los servomotores

4. Conclusiones y trabajos a futuro

El desarrollo del robot a escala del modelo industrial IRB 460 permite la enseñanza de la robótica a un costo accesible, ya que la construcción mediante impresión 3D es rentable y, además, permite que los estudiantes de nivel superior puedan relacionar el manejo y construcción del robot industrial sin que se tenga dicho robot industrial disponible. Otra cuestión importante de resaltar radica en la posibilidad de que la adecuación de nuevos cálculos o de temas más avanzados en el estudio de

robótica, se puede realizar sencillamente con la modificación del programa principal y de la utilización de algún otro pin disponible en el Arduino.

Como trabajo a futuro, se busca implementar una interfaz didáctica para la enseñanza en donde el alumno pueda manipular el robot y crear subrutinas de forma remota, esto con ayuda de un mando compuesto por más botones pulsadores, joystick y una respectiva pantalla LCD donde se muestre el eslabón y el ángulo que se esté moviendo. También se puede abordar el problema de la dinámica del robot y el seguimiento de trayectoria, incluso la aplicación de técnicas control, por mencionar algunas.

Agradecimientos

Todos los autores agradecen al Instituto Politécnico Nacional el apoyo por medio del proyecto SIP20221986, el segundo autor agradece por la beca BEIFI otorgada.

Referencias

- ABB (2022). Irb 460. <https://new.abb.com/products/robotics/es/robots-industriales/irb-460>. Online, 2022.
- Arduino (2022). Arduino uno rev 3. <https://docs.arduino.cc/hardware/uno-rev3>. Online, 2022.
- Arenas-Rosales, F., Martell-Chavez, F., Sanchez-Chavez, I. Y., y Paredes-Orta, C. A. (2021). Virtual laboratory for online learning of ur5 robotic arm inverse kinematic and joint motion control. *2021 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, pp. 1–6.
- Barrientos, A., Peñín, L. F., Balaguer, C., y Aracil, R., editores (1997). *Fundamentos de robótica*. McGraw-Hill Education / Interamericana de España, España.
- Chatzopoulos, A., Kalovrektis, K., Xenakis, A., Papoutsidakis, M., Kalogianakis, M., y Psycharis, S. (2022). An advanced physical computing - based educational robot platform evaluated by technology acceptance model. *2022 10th International Conference on Information and Education Technology (ICIET)*, pp. 6–10.
- Feijoo-Almonacid, A. y Rodriguez-Garavito, C. H. (2022). Hardware-software platform for the development of stem skills. *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, 17:170–177.
- Franciscone, C. (2018). Eezyrobots, eezybotarm mk2. http://www.eezyrobots.it/eba_mk2.html. Online, 2022.
- Intesc (2022). Tarjeta de desarrollo ophyra. <https://intesc.mx/productos/tarjeta-de-desarrollo-ophyra/>. Online, 2022.
- Kelly, R. y nez, V. S., editores (2003). *Control de movimiento de robots manipuladores*. Editorial Pearson Education, España.
- Mohanavel, V., Diwakaran, S., Maheswaran, U., Anitha, G., y Ramesh, S. (2022). An effective design of wireless android based robotic operation control using 8051 microcontroller. *2022 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, pp. 1–6.
- Parikh, P., Kalamdane, S., Rasal, A., Mundada, K., y Kalshetti, U. (2022). Spark-bot: An educational robotic platform. *2022 IEEE Global Engineering Education Conference (EDUCON)*, pp. 1628–1633.
- Reyes, F., editor (2011). *Robótica, control de robots manipuladores*. Alfaomega Grupo Editor, México.
- STM (2022). Stm32f407g-disc1. <https://www.st.com/en/evaluation-tools/stm32f4discovery.html>. Online, 2022.
- TI (2022). Texas instruments hercules tms5701s12x launchpad development kit. <https://www.ti.com/tool/LAUNCHXL2-TMS57012?> Online, 2022.
- Wang, Y. y Song, J. (2021). Research on robotic competition system for steam education. *Networking and Information Technology (AINIT), 2021 2nd International Seminar on Artificial*, pp. 204–207.
- Zhong, Y., Chen, Y., Wang, C., Wang, Q., y Yang, J. (2022). Research on target tracking for robotic fish based on low-cost scarce sensing information fusion. *IEEE Robotics and Automation Letters*, 7:6044–6051.