

Comunicación Robotstudio-Matlab mediante protocolo TCP/IP en un robot IRB-120 de 6 grados de libertad para uso en actividades didácticas

Robotstudio-Matlab communication through TCP / IP protocol in an IRB-120 robot with 6 degrees of freedom for use in teaching activities

MENDOZA-OLLERVIDES, Rosendo*†, ORTIZ-SIMÓN, José Luis, AGUILERA-HERNÁNDEZ, Martha, CRUZ-HERNÁNDEZ, NICOLÁS y ROJO-VELÁZQUEZ, Gustavo Emilio

Instituto Tecnológico de Nuevo Laredo

ID 1^{er} Autor: *Rosendo, Mendoza-Ollervides* /ORC ID: 0000-0001-8254-957, **Researcher ID Thomson:** S-8583-2018

ID 1^{er} Coautor: *José Luis, Ortiz-Simón* /ORC ID: 0000-0003-2380-9112, **Researcher ID Thomson:** S-4724-2018, **CVU CONACYT ID:** 19115

ID 2^{do} Coautor: *Martha, Aguilera-Hernández* /ORC ID: 0000-0001-8127-190X, **Researcher ID Thomson:** S-7552-2018

ID 3^{er} Coautor: *Nicolás, Cruz-Hernández* /ORC ID: 0000-0001-6118-7831, **Researcher ID Thomson:** S-7552-2018, **CVU CONACYT ID:** 209883

ID 4^{to} Coautor: *Gustavo Emilio, Rojo-Velázquez* /ORC ID: 0000-0002-7792-1436, **Researcher ID Thomson:** S-6815-2018, **CVU CONACYT ID:** 26367

Recibido 6 de Octubre, 2018; Aceptado 19 de Diciembre, 2018

Resumen

El presente artículo se enfoca en usar métodos diferentes para poder comprobar los cálculos realizados del modelado cinemático (Cinemática Directa e Inversa) de un robot IRB-120 de la marca ABB. Uno de los métodos es realizar una conexión TCP/IP entre RobotStudio y Matlab [1]. Otro método es usar un IRB-120 físico para comparar los resultados de un robot físico con el de la simulación. El último método es usando un algoritmo en MatLab diseñado para obtener la cinemática directa e inversa de un IRB-120 para tener una comprobación más.

Interfaz gráfica de Usuario (GUI), Cinemática Directa, Cinemática Inversa, Modelado Cinemático, Algoritmo

Abstract

This article is focused in the use of different methods to check the calculations of the kinematic fundamentals of an IRB-120 robot of the ABB Company. One of the methods consists in creating a TCP/IP connection between RobotStudio and MatLab [1]. Another method is to use a physical IRB-120 to compare the results of a physical robot with the ones given by the simulation. The last method is using a homemade algorithm in MatLab designed to obtain the forward kinematic and the inverse kinematic of an IRB-120 robot to have an extra comparison.

Graphic Users Interface (GUI), Forward Kinematic, Inverse Kinematic. Kinematic Model. Algorithm.

Citación: MENDOZA-OLLERVIDES, Rosendo, ORTIZ-SIMÓN, José Luis, AGUILERA-HERNÁNDEZ, Martha, CRUZ-HERNÁNDEZ, NICOLÁS y ROJO-VELÁZQUEZ, Gustavo Emilio. Comunicación Robotstudio-Matlab mediante protocolo TCP/IP en un robot IRB-120 de 6 grados de libertad para uso en actividades didácticas. Revista de Innovación Sistemática 2018. 2-8:23-29

*Correspondencia al Autor (Correo Electrónico: rosendo_2208@hotmail.com,)

† Investigador contribuyendo como primer autor.

Introducción

La robótica es un campo que ha tenido una evolución vertiginosa en los últimos años, por lo cual se buscan nuevas formas de mejorar el uso y control de los mismos.

La interfaz gráfica de usuario (GUI) es la forma principal que se usa para interactuar con las computadoras. Windows, Mac OS y Linux tienen interfaces basadas en GUI. La GUI reemplazó a la interfaz de línea de comandos porque interactuar con una computadora usando un puntero de ratón tiene varias ventajas sobre otros comandos escritos.

Una interfaz gráfica de usuario es intuitiva. Una persona que no haya usado nunca antes una computadora podrá realizar una navegación básica con una GUI después de recibir unas breves instrucciones.

Podemos encontrar un ejemplo muy útil de cómo controlar un robot mediante una GUI en [1]. Este tipo de aplicaciones de una GUI para controlar un robot demuestran que aprender a manejar y familiarizarse con un robot es una de las maneras más prácticas de manejar.

Materiales y métodos

Se dividió el proyecto en dos partes; la primera en conocer el funcionamiento del software RobotStudio y MatLab. La segunda parte fue en comprobar que los resultados obtenidos por el software RobotStudio concuerden con los cálculos hechos para las comprobaciones.

Se profundizó en aprender el uso del software RobotStudio para controlar el robot y conocer sus valores de ejes, de posición, de rotación, etc.



Figura 1 Estación de simulación del robot IRB-120

Se realizó una simulación del robot (Figura 1) que se manejó para mover y comprobar sus valores. En la figura 2 se aprecia los parámetros del robot; valores de los ejes (valores de cada motor en grados), valores de la posición y rotación (respecto al efector final del robot).

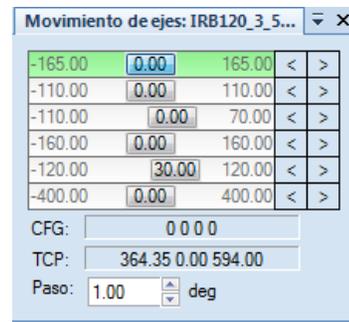


Figura 2 Valores de ejes, posición (del efector final) y configuración del robot

El parámetro CFG (Figura 2), el representa la “Configuración” del robot, esto significa el cómo él robot llegó a un punto deseado, puesto que puede llegar a un mismo punto de diferentes formas. La posición son los tres valores en el parámetro TCP (Tool Central Point, que significa Punto Central de la Herramienta que es el efector final del robot) que representa los valores de una distancia en X, en Y y en Z (en milímetros) respectivamente.



Figura 3 Valores de la posición, rotación y configuración de robot

En la figura tres podemos observar los valores de X, Y y Z de la posición del efector final, también la rotación en X, Y y Z del mismo. En “Cfg” se observa la configuración de la cual hablamos anteriormente. La lista despegable es para seleccionar el punto de referencia sobre el que se trabajará. Por último, el parámetro “Paso” es para seleccionar cuantos milímetros se moverá el robot o cuantos grados girará el robot.

Se desarrolló un programa que permite una interfaz gráfica en MatLab para poder crear una comunicación con el software RobotStudio mediante el protocolo TCP/IP.

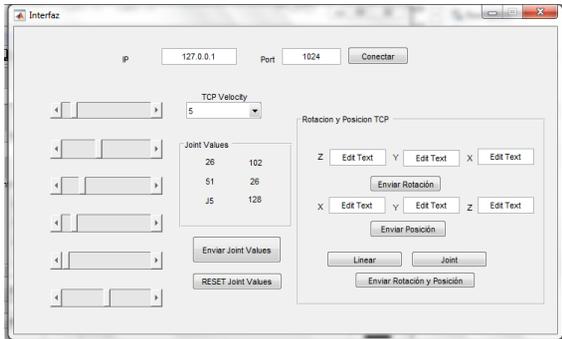


Figura 4 Ejemplo de una Interfaz gráfica en MatLab

Dentro de las funciones de los dos softwares, hay una que permite establecer una comunicación mediante el protocolo TCP/IP, el cual permite enviar datagramas del servidor al cliente y viceversa con una garantía de que los paquetes (datagramas) serán enviados y recibidos sin perder información. Esta es la razón por la que se opta usar este protocolo en lugar del UDP/IP, ya que, aunque sea más rápido para el envío de datos, no se asegura la integridad del datagrama.

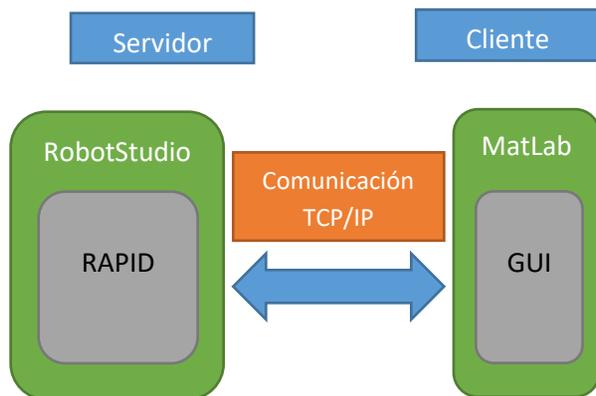


Figura 5 Esquema básico de la comunicación TCP/IP

Análisis Cinemático

Cinematica Directa: En el caso de un robot, consiste en calcular en qué posición está el efector final de nuestro robot y cuanto y hacia donde rotó (giró), conociendo el valor de cada eje (cuantos grados giró un motor y/o cuanto entró o salió un pistón). Para esto se usa la convención de Denavit Hartenberg (DH) [2] para obtener una matriz con los valores deseados a conseguir.

Cada robot tendrá una matriz propia y diferente de acuerdo a cuantos eslabones tenga y de su posición inicial.

Cinematica Inversa: Se busca conseguir el valor que tiene cada eje (valores de los motores o pistones) conociendo la posición y rotación del efector final.

Comprobaciones: Se dan valores de eslabón para comprobar la cinemática directa y se da una posición en el espacio con una rotación para comprobar la cinemática inversa. Se usó la simulación, el robot físico y el algoritmo para comparar los resultados.

Se comparó los resultados dados por las tres plataformas; se usó la cinemática directa para darle los valores de ejes igual a cero para comparar la posición inicial del robot.

-IRB-120

Se comprobó la posición del robot con valores de articulación igual a cero. La posición del robot fue la se observa en la figura 6.



Figura 6 Robot IRB-120 en posición con valores articulares cero

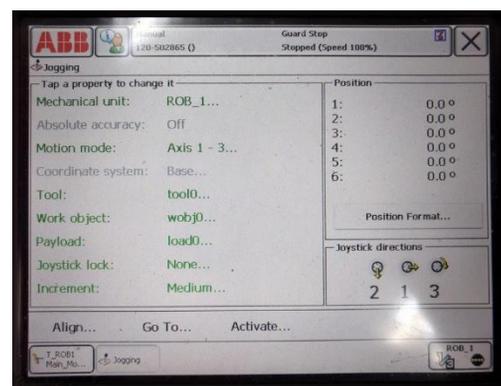


Figura 7 Valores reales de las uniones del IRB-120

En el cuadro derecho de la figura 7 se puede apreciar los valores reales que posee cada eslabón respecto a la posición del robot mostrada en las figuras 6. A su vez, en la figura 8 se puede observar en el lado derecho, los valores de la posición y rotación del efector final del robot con respecto a su origen (La base del IRB-120).

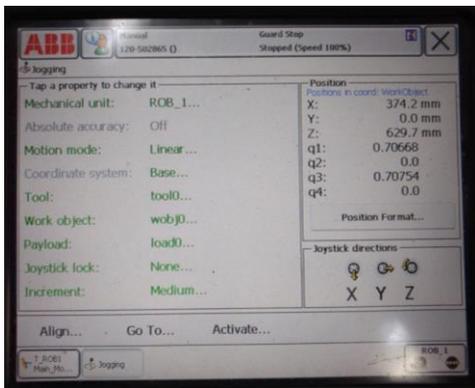


Figura 8 Valores reales de la posición y rotación del IRB-120

En el software RobotStudio se simuló un modelo IRB-120 para comparar los resultados dados por el modelo físico y el algoritmo realizado en MatLab.

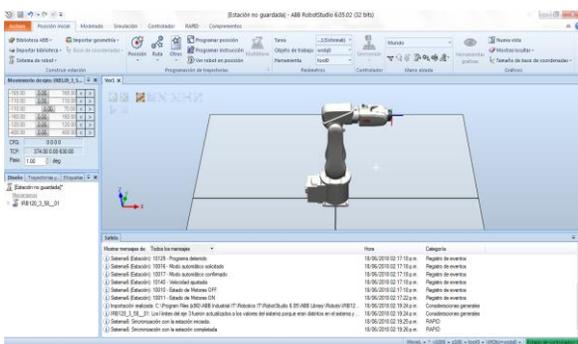


Figura 9 Simulación de robot IRB-120 con valores articulares cero

En la figura 9 se observa que la posición inicial del robot IRB-120 coincide con la dada por la simulación. En la figura 10 se observan los valores de ejes de la simulación y su posición y rotación.

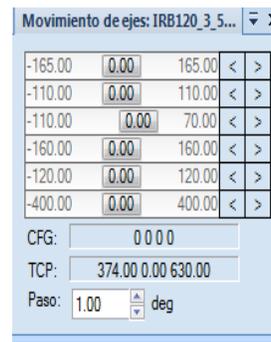


Figura 10 Valores de los ejes, de posición y rotación del robot simulado

Se elaboró un algoritmo en MatLab que permite calcular la cinemática directa e inversa de un modelo IRB-120, se introduce los parámetros (medidas del robot) y dependiendo el procedimiento a realizar, los valores articulares o los valores de la posición y rotación.

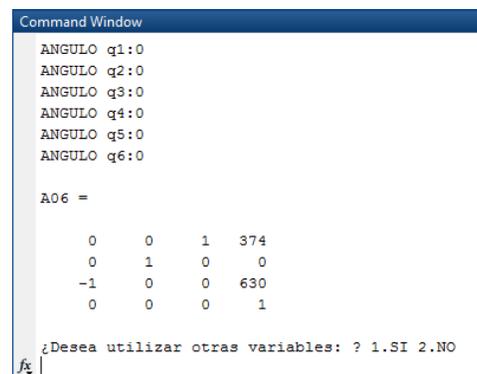


Figura 11 Comprobación de posición inicial en algoritmo de MatLab

En la figura 11 se observa que al darle los valores de 0 a cada eslabón (los valores de la posición inicial), nos arrojó la matriz homogénea de nuestro robot, la cual muestra la matriz rotación y el vector posición. En el vector posición podemos observar que obtuvo una posición en X igual a 374, una posición en Y de 0 y una posición en Z de 630.

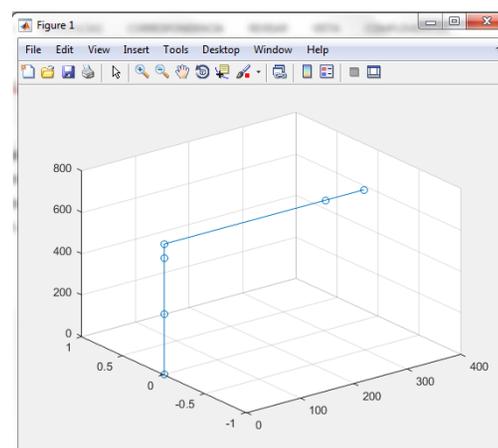


Figura 12 Grafica hecha con los datos del algoritmo

En la figura 12 se observa una gráfica realizada en base al algoritmo creado en MatLab para comprobar los movimientos del robot.

Resultados

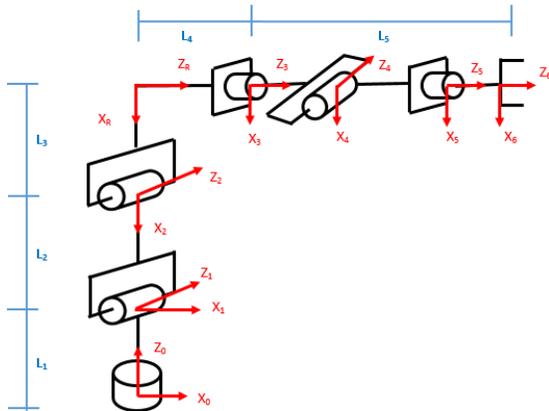


Figura 13 Representación de robot IRB-120, convención Denavit-Hartenberg

En la Fig. 13 se puede observar el robot IRB-120 cuando: \$q_1 = q_2 = q_3 = q_4 = q_5 = q_6 = 0\$, de la cual se obtiene la siguiente tabla de parámetros DH:

Tabla 1. Tabla de parámetros del robot, convención D-H.				
\$i\$	\$\theta_i\$	\$d_i\$	\$a_i\$	\$\alpha_i\$
1	\$q_1\$	\$L_1\$	0	\$-90^\circ\$
2	\$q_2 + 90^\circ\$	0	\$-L_2\$	\$0^\circ\$
3	\$q_3\$	0	\$-L_3\$	\$90^\circ\$
R	0	\$L_4\$	0	\$0^\circ\$
4	\$q_4\$	0	0	\$-90^\circ\$
5	\$q_5\$	0	0	\$90^\circ\$
6	\$q_6\$	\$L_5\$	0	\$0^\circ\$

Tabla 1

Se realizó el cálculo de las matrices de rotación usando la siguiente ecuación:

$${}^{i-1}T_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Las matrices quedaron de la siguiente manera:

$${}^0_1T = \begin{bmatrix} C_1 & 0 & -S_1 & 0 \\ S_1 & 0 & C_1 & 0 \\ 0 & -1 & 0 & L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^1_2T = \begin{bmatrix} C_{(q_2+90^\circ)} & -S_{(q_2+90^\circ)} & 0 & -L_2 C_{(q_2+90^\circ)} \\ S_{(q_2+90^\circ)} & C_{(q_2+90^\circ)} & 0 & -L_2 S_{(q_2+90^\circ)} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2_3T = \begin{bmatrix} C_3 & 0 & S_3 & -L_3 C_3 \\ S_3 & 0 & -C_3 & -L_3 S_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^3_4T = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4_5T = \begin{bmatrix} C_5 & 0 & S_5 & 0 \\ S_5 & 0 & -C_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^5_6T = \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 1 & L_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} {}^3_R T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Multiplicamos las matrices para obtener la matriz \${}^0_6T\$.

$${}^0_3T = {}^0_1T * {}^1_2T * {}^2_3T$$

$${}^0_RT = {}^0_3T * {}^3_RT$$

$${}^0_6T = {}^0_RT * {}^3_4T * {}^4_5T * {}^5_6T$$

La matriz \${}^0_6T\$ quedó de la siguiente manera; donde

$${}^0_6T = \begin{bmatrix} 0 & 0 & 1 & 374 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 630 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

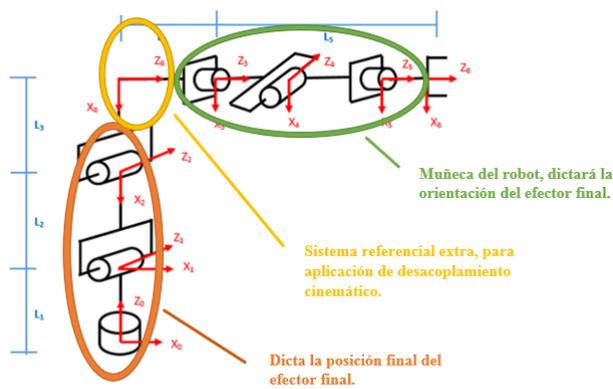


Figura 13 Desacoplamiento del robot, y ubicación de su “muñeca”

Se usó la figura 13 y se desarrolló un procedimiento en funciones de tangente para obtener los valores de q_4, q_5 y q_6 .

$${}^3T_6 = \begin{bmatrix} C_4C_5C_6 - S_4S_6 & -C_4C_5S_6 - S_4C_6 & C_4S_5 & L_5C_4S_5 \\ S_4C_5C_6 + C_4S_6 & -S_4C_5S_6 + C_4C_6 & S_4S_5 & L_5S_4S_5 \\ -S_5C_6 & S_5S_6 & C_5 & L_5C_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

Modelo Cinemático Inverso

$$q_4 = tg^{-1} \left(\frac{r_{23}}{r_{13}} \right)$$

$$q_5 = tg^{-1} \left(\frac{\sqrt{1 - r_{33}^2}}{r_{33}} \right)$$

$$q_6 = tg^{-1} \left(\frac{-r_{32}}{r_{13}} \right)$$

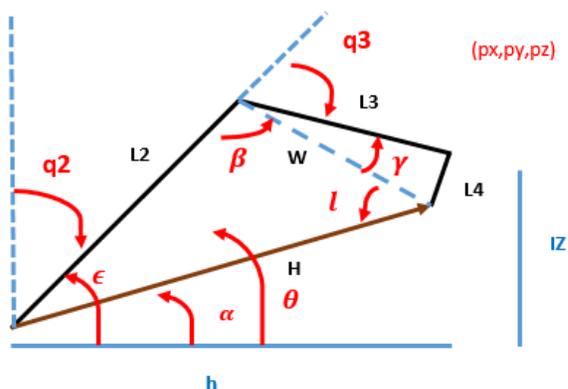


Figura 14 Configuración codo arriba del robot

Se utilizaron los parámetros de la figura 14 para poder realizar el despeje de q_1, q_2 y q_3 , los cuales quedan de la siguiente manera:

$$q_1 = atan \left(\frac{py}{px} \right)$$

$$q_2 = 90^\circ - \epsilon$$

$$q_3 = 180^\circ - \beta - \gamma$$

Se utilizaron estas ecuaciones obtenidas de los métodos de cinemática directa e inversa sustituyendo los valores de las q 's o los valores de posición y rotación. Se realizaron 10 pruebas (5 de cada método) y en todas se llegó al mismo resultado, comparando los resultados dados sustituyendo valores de las ecuaciones con los valores dados por las diferentes plataformas utilizadas.

Discusión

Una de las ventajas de este proyecto es que utiliza una interfaz gráfica sencilla de usar y entender. Permite a los alumnos un fácil entendimiento de los movimientos básicos del robot y sus limitaciones.

Las desventajas que presenta este proyecto es que no tiene todas las funciones que tiene el software RobotStudio y a veces hay problemas al momento de enviar los valores de MatLab a RobotStudio y hay que volver a establecer la conexión.

Referencias

GUTIÉRREZ CORBACHO, Azahara (2014), *Trabajo Fin de Grado: “Desarrollo de una interfaz para el control del robot IRB120 desde Matlab”*. Universidad de Alcalá.

J. CRAIG, John (2006), *Robótica*, 3ª edición, México, Prentice Hall, 310 páginas.

Mark W. Spong, Seth Hutchinson, and M. Vidyasagar (1989), *Robot Dynamics and Control*, John Wiley & Sons Inc, 352 páginas.

Matos Agudo, B. (2017). Modelado, programación y simulación del Robot IRB 120 de ABB con Robotstudio. (Trabajo Fin de Máster Inédito). Universidad de Sevilla, Sevilla.

Adrian-Florin NICOLESCU¹, Florentin-Marian ILIE , Tudor-George ALEXANDRU. "FORWARD AND INVERSE KINEMATICS STUDY OF INDUSTRIAL ROBOTS TAKING INTO ACCOUNT CONSTRUCTIVE AND FUNCTIONAL PARAMETER'S MODELING", *Proceedings in Manufacturing Systems*, num. 4, volumen 10, 2015, pp. 157-164.

Manual de referencia técnica. Instrucciones, funciones y tipos de datos de RAPID.