

# VII Encuentro de Investigación •Alberto Magno•

Implementación de tecnologías opensource, con interfaz de comunicación para la simulación de movimientos del brazo robótico IRB 120 ABB de la facultad de ingeniería mecatrónica

## Información del semillero o grupo de investigación

### GRUPO DE INVESTIGACIÓN EN APLICACIONES MECATRÓNICAS (ASIMOV)

Misión: El Grupo de Investigación en Aplicaciones Mecatrónicas, GRAM, asume como misión realizar investigación científica y de desarrollo tecnológico en las líneas de Investigación en Control y Automatización y en Diseño Mecatrónico con el propósito de generar proyectos orientados a mejorar la productividad y eficiencia del sector empresarial de la región y del país. Visión: El Grupo de Investigación en Aplicaciones Mecatrónicas, GRAM, espera consolidarse en 2027 como un referente de alta calidad a nivel nacional en investigación y desarrollo tecnológico en las líneas de control y automatización y diseño mecatrónico al brindar asesoría científica y técnica de carácter integral, para que las comunidades y sectores económicos mejoren la productividad, la competitividad y la calidad de vida en el marco del desarrollo sostenible.



## Problema a resolver

El brazo robótico IRB 120 ABB de la facultad de ingeniería mecatrónica de la Universidad Santo Tomas, se encuentra inhabilitado para el manejo de este mediante software, debido al vencimiento de la licencia del programa original del brazo el cual es el Robotstudio. Para darle solución a esto se buscarán nuevas alternativas de conexión Opensource.

Para la realización de pruebas con el brazo se requiere un espacio en el cual no tenga ningún tipo de interferencia en sus movimientos, por esta razón se adaptará una zona de trabajo. Se espera encontrar un software Opensource compatible con el IRB120 para dar uso a sus funcionalidades, con el propósito de que los estudiantes de la facultad de ingeniería mecatrónica puedan complementar sus conocimientos en el área de robótica.



## Resultados esperados

Una vez escogido el software Opensource que mejor se adapta a nuestros requerimientos de comunicación, se ha realizado la instalación de un sistema operativo que nos permite trabajar con ROS, el cual es nuestro framework seleccionado. El sistema operativo es UBUNTU 16.04, este debido a que es compatible con ROS KINETIC, que es la versión más estable y es en la que se encuentra mayor documentación soportada por los desarrolladores de software y por la comunidad, para nuestro proyecto. Se instalo ROBOTSTUDIO en un computador con Windows, para familiarizarse con el sistema e interfaz de simulación con el fin de más adelante poder hacer pruebas de la comunicación entre nuestro pc con UBUNTU y el mencionado anteriormente, esto debido a que no se puede acceder al brazo robótico el cual se encuentra en la Universidad Santo Tomas Bucaramanga. Se realizo la respectiva instalación del framework ROS KINETIC, dentro del cual se instalaron diferentes dependencias necesarias para la comunicación, se descargaron algunos códigos fuente los cuales se les analizó la estructura para comprender como funcionaban y como estos envían datos al robot. Un computador contiene el programa Robotstudio previamente instalado y en otro se tiene Linux con Ubuntu 16.04 y ros el framework seleccionado obsérvese figura 1.

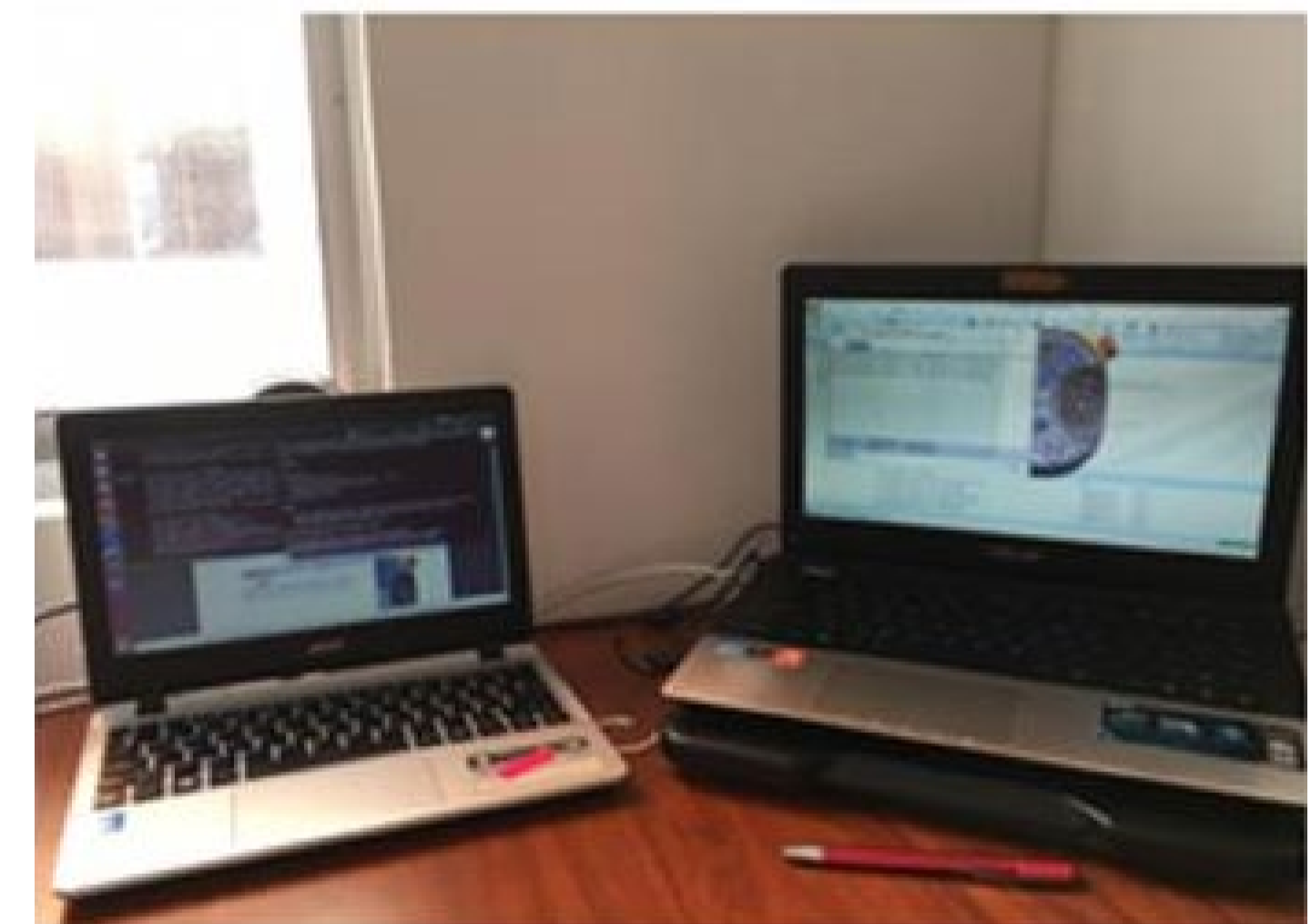


Figura 1. Computadores con software a utilizar

## Metodología empleada

Recuperar la funcionalidad del brazo robótico IRB 120 ABB de la facultad de ingeniería mecatrónica, mediante la implementación de un software opensource que contenga una interfaz gráfica que permita realizar la programación de rutinas de cinemática directa, para complementar las asignaturas de robótica de la facultad.

1. Realizar un comparativo de las tecnologías de hardware y software opensource existentes que se adapten a las especificaciones y requerimientos de comunicación y control del brazo robótico IRB 120 ABB.

1.1 Adquirir información de las especificaciones y requerimientos de comunicación y control del brazo robótico IRB 120 ABB.

1.2 Indagar sobre los softwares que pueden operar bajo los parámetros de comunicación y control del brazo robótico.

1.3 Analizar las características propias de los softwares, con las que estos operan, para tener presente bajo que estructuras o lenguaje trabajan estos.

2. Adaptar el habitáculo o área de trabajo del brazo robótico, para evitar lesiones a los estudiantes y prevenir daños a los componentes electro-mecánicos del brazo, para así poder realizar las prácticas de una forma segura.

2.1 Identificar los componentes necesarios para operar el brazo de manera óptima según el fin planteado (equipos, cables, alimentación, demás) adapten a las especificaciones

2.2 Obtener un aproximado de las dimensiones del área de operaciones en el cual se pueda ejecutar el brazo para realizar las respectivas delimitaciones de seguridad con el fin de evitar posibles accidentes.

2.3 Designar una zona acorde a los componentes que se requieren para operar el brazo y a las dimensiones del área de operación obtenidas.

3. Implementar la tecnología opensource que presente mejores características para crear una interfaz de comunicación con el brazo robótico IRB 120 ABB, verificando su funcionamiento mediante el posicionamiento del mismo utilizando cinemática directa.

3.1 Analizar el principio de comunicación del brazo robótico.

3.2 Adquirir información sobre el manejo y funcionamiento de la parte gráfica del software seleccionado para con esto plantear un diseño preliminar de la parte grafica con la que el usuario coaccionaría al momento de operar el brazo robótico.

3.3 Plantear un código preliminar con el que se operen los motores que conforman el brazo robótico, en este se tienen en cuenta las entradas de información, comunicación de los elementos y salida de datos (que sería el movimiento que debería de realizar el brazo).

## Principales referentes bibliográficos

- ✓ Cuadrado, J. M. (2017). Integración del brazo robot IRB120 en entorno ROS-MATLAB.
- ✓ Durán Gómez, M. A., & Arenas Ardila, N. D. Control de un brazo robótico industrial basado en procesamiento de imágenes para la manipulación de piezas.
- ✓ Wiki.ros.org. 2020. Abb\_Driver/Tutorials/Installserver - ROS Wiki. [online] Available at: <[http://wiki.ros.org/abb\\_driver/Tutorials/InstallServer](http://wiki.ros.org/abb_driver/Tutorials/InstallServer)>[Accessed 2 September 2020].
- ✓ Running the ROS Server. (2020). ros.org. [http://wiki.ros.org/abb\\_driver/Tutorials/RunServer](http://wiki.ros.org/abb_driver/Tutorials/RunServer).
- ✓ abb\_driver/Tutorials/RobotStudio - ROS Wiki. (2020). ros.org. [http://wiki.ros.org/abb\\_driver/Tutorials/RobotStudio](http://wiki.ros.org/abb_driver/Tutorials/RobotStudio)