

12º CONGRESSO IBEROAMERICANO DE ENGENHARIA MECANICA

Guayaquil, 10 a 13 de Novembro de 2015

DESARROLLO E IMPLEMENTACIÓN DE UN CONTROLADOR BASADO EN TECNOLOGÍA CNC PARA EL ROBOT ASEA IRB6-S2 USANDO LINUXCNC

Toquica Arenas, J. S.*, Alvares, A. J.º

*ºUniversidad de Brasilia, Campus Universitario Darcy Ribeiro, Brasilia DF, Brasil.

*e-mail: jstoquica@gmail.com

Palabras claves:

RESUMEN

Este trabajo presenta una propuesta para el Retrofitting de un brazo robótico ASEA IRB6-S2 compuesto de 5 grados de libertad (DOF), que actualmente no está en funcionamiento. Contribuyendo de esta forma a la consolidación de una arquitectura abierta de control universal para manipuladores robóticos, independiente de los sistemas desarrollados por los fabricantes. Mediante la técnica Retrofitting el manipulador volverá nuevamente a ser funcional y podrá ser integrado con un Célula de Manufactura Flexible (FMC). El análisis del estado de la estructura mecánica del robot y la actualización tecnológica del sistema de control y potencia son actividades fundamentales de la metodología propuesta. La integración del sistema mecánico con el sistema de control actualizado para posible el desplazamiento deseado del robot a través de los pulsos generados por el controlador desarrollado. El control propuesto se caracteriza por tener aspectos de una arquitectura abierta que permite mejoras futuras mediante cualquier tipo de modificación. Las ecuaciones homogéneas que modelan la cinemática directa e inversa del robot estarán presentes en el controlador desarrollado en Linux, denominado LinuxCNC lo que proporcionara flexibilidad, y validara la arquitectura propuesta mediante las características de software OpenSource.

PALABRAS CLAVE: Retrofitting, Robot ASEA IRB6-S6, FMC, LinuxCNC, Cinemática.

INTRODUCCIÓN

El presente artículo presenta la propuesta de una arquitectura abierta cuyo objetivo fundamental facilitara la integración de un robot Asea IRB6-S2 actualmente fuera de operación, a una Célula de Manufactura Flexible (FMC) del laboratorio Graco de la UnB, mediante la aplicación de una metodología conocida en ingeniería mecánica como Retrofitting. Dicha metodología propone la verificación de estado actual de los componentes propios del robot tales como sensores, motores, fuente de potencia, accionamientos mecánicos, drives de potencia etc., para establecer la utilización de los componentes mecánicos y electrónicos funcionales y la integración con tecnologías modernas. Debido a que el Retrofitting se aplica en robots antiguos es recomendable usar simplemente las piezas del sistema mecánico [1].

El Retrofitting se caracteriza por volver funcionales los robots industriales que fueron considerados obsoletos, teniendo en cuenta que la estructura mecánica ha cambiado poco en la evolución tecnológica de este tipo de robots. Las principales diferencias están relacionadas con los accionamientos de las articulaciones y en el sistema de controle, pudiendo aprovechar de esta forma la estructura del sistema mecánico [2].

Por tanto el presente trabajo pretende contribuir con información relevante y actualizada en el intento de consolidar una arquitectura estándar para el control de robots industriales, debido a la falta de arquitecturas abiertas para los controladores de los manipuladores. Actualmente, existen protocolos de comunicación, lenguajes de programación o estructuras de control corporativas, las cuales no permite el desarrollo de mejoras tecnológicas, lo que conduce a los clientes a la compra de nuevos sistemas con los proveedores originales [3].

La característica fundamental en el presente trabajo es el control del movimiento del IRB6 con la integración de LinuxCNC, el cual se caracteriza por ser un sistema OpenSource para el desarrollo de sistemas de control de accionamientos compuestos en cada uno de los grados de libertad (DOF) propios de un robot [4]. Permitiendo de esta forma el control en tiempo real de la transferencia de información al sistema de control, que junto con la definición de la cinemática del manipulador proporcionara el movimiento del robot.

Los primeros robots del mundo fueron producidos en la década del 70 necesitando hoy mantenimiento o reemplazados por nuevos, para que las empresas continúen con los procesos productivos. Por tanto, las grandes compañías no tienen problema alguno para comprar nuevos manipuladores, sin embargo las pequeñas y medianas empresas no están en condiciones de acceder a dichas tecnologías por los altos costos relacionados. De este modo, el Retrofitting contribuyo desde los años 90 para que las empresas que no pudieran adquirir manipuladores actualizaran los robots antiguos mediante la integración con tecnología actual. Justificando así que los cambios en aproximadamente 40 años de inserción de estos sistemas, solo está sujeta a la mejoras de los sensores de las articulaciones, por consiguiente la estructura mecánica continua vigente [5].

Un proceso de Retrofitting para un robot industrial IRB2000 fue generado describiendo la construcción de un nuevo banco de control que permite alojar una arquitectura abierta de control independiente de los algoritmos de control y sistemas de potencia suministrados por el fabricante [6]. El sistema de accionamiento del robot fue conectado a través de una red CAN-Bus a una interface HMI alojada en un computador y desarrollado en Simulink/Matlab. Este desarrollo hace parte del software con módulos de control personalizados que consideran las características del robot, como su región de trabajo, la limitación de velocidad de las articulaciones, parámetros de la cinemática etc. Una de las características adicionales realizadas, fue el control de la orientación del “puño” del robot, mediante un dispositivo Wiimote (Control de Nintendo Wii). La herramienta permite un control de movimiento preciso del efector final del manipulador, con una arquitectura flexible, y con la característica de red distribuida permite integrar otros robots o mecanismos relacionados.

Una aplicación adicional en la implementación de Retrofitting fue desarrollada en ambiente académico motivado por el aumento de los costos de adquisición de manipuladores, ya que los fabricantes generan protocolos e interfaces de comunicación cerradas para las estructuras lógicas de los robots, dificultando la integración con nuevas aplicaciones [2]. De esta forma se presentó una propuesta de arquitectura abierta específicamente para el manipulador Asea IRB6-S2 desarrollado originalmente para un robot antropomórfico. Dicha arquitectura propone un módulo de procesamiento distribuido para el control de los robots. El sistema tiene dos buses de campo, uno para el control en tiempo real de los parámetros del robot y otro para la adquisición de los datos de supervisión, con conexiones tipo CAN y Ethernet, respectivamente.

Un software diseñado para una arquitectura abierta fue desarrollado en Java y C++, con un computador con sistema operativo Linux y una variante para el procesamiento en tiempo real RTAI. Las etapas para el Retrofitting generado se basaron en la identificación y verificación del sistema mecánico y actualización del sistema electrónico. Adicionalmente, el Retrofitting del trabajo tuvo como resultado adicional la comparación en el uso de los sensores de posición y velocidad originales del robot en relación a encoders digitales, brindando mejor precisión y facilitando el proceso de integración tecnológica con la segunda tecnología enunciada [6].

El Retrofitting para el Robot IRB6-S2 de Graco/GIAI/UnB volverá a estar nuevamente operacional, permitiendo ser integrado en la cadena productiva a nivel académico en una FMC de mecanizado, con la función de manipular y posicionar piezas brutas y acabadas.

Con un robot de 5 grados de libertad se realizó el mecanizado de piezas de material de baja densidad que permitió la operación a través de una arquitectura de control abierta basada en LinuxCNC, que cuenta con un sistema de simulación y verificación que se ejecuta cuando inicia el ciclo de mecanizado [4]. El trabajo presenta un modelo de robots reconfigurables que permite la incorporación de sistemas CAD/CAM para generar piezas basadas en la operación de fresas verticales. Una aplicación diferente es presentada a lo largo del presente trabajo, debido a que posible conocer características de una arquitectura abierta que permite la reutilización de robots obsoletos para que estén nuevamente funcionales y sean incorporados en una línea de producción.

Un nuevo modelo de arquitectura abierta de control para robots bajo el estándar ISO 7498-1 (Capas OSI) fue propuesto e implementado en sistemas embebidos, en el cual la programación de tareas de control fue realizada en alto nivel y después decodificado para procesadores DSC, usados en el control de movimiento de un robot de 6 DOF. La capacidad modular de esta propuesta permite la integración con otros sistemas o la cooperación con otros manipuladores, debido a la definición de 5 capas del sistema OSI. Lo anterior facilitara integrar nuevas funcionalidades y mejorar la estructura de las capas propuestas validando los sistemas abiertos de control [7].

Con la definición de una arquitectura abierta a través de propuestas similares al presente trabajo, es posible convertir los robots IRB6-S2 en funcionales, mediante la disposición de una arquitectura y software integrados con la cinemática propia del robot con una solución tipo OpenSource. De igual forma, es posible facilitar una metodología genérica de Retrofitting de robots industriales basada en tecnología de Comando Numérico.

Los costos de inversión en Retrofitting pueden ser de hasta el 14% del valor actual de un nuevo robot con características similares en el mercado brasilero. De esta forma es posible aprovechar el uso de la estructura mecánica del robot Asea IRB6-S2, cambiando los componentes de control y potencia, debido a la falta de compatibilidad con los sistemas producidos actualmente [1]. Las etapas desarrolladas para ejecutar el Retrofitting y generar una arquitectura abierta son detalladas en la propuesta realizada por Bomfim [8]. En el trabajo realizado el software para la interface con el usuario y generación de pulsos hacia los drives de los motores son software licenciados compatibles con Windows (Matlab e Match 3). La selección de este software de desarrollo se seleccionó por la fácil instalación y uso del sistema de control, sin la inversión de recursos empresariales para capacitar los operarios en otros ambientes de desarrollo. Teniendo en cuenta que los software mencionados necesitan licencias activas y compatibles en el sitio de la aplicación de la metodología descrita.

El modelamiento cinemático y la simulación de un robot de 6 grados de libertad fue realizada aplicando las capacidades propias de un controlador tipo OpenSource como lo es LinuxCNC, permitiendo que el robot siguiera una determinada trayectoria generada mediante un archivo en código G [9]. Aunque el trabajo realizado no presenta implícitamente la adecuación bajo una arquitectura abierta de controle, al usar este tipo de software basados en lenguaje libre es posible determinar los resultados del trabajo desarrollado por Preez [9], dentro del contexto de arquitecturas abiertas de control para robots manipuladores.

Los trabajos citados anteriormente tiene dos características en común, la adopción de la metodología Retrofitting para contribuir y desarrollar un estándar con arquitectura abierta de control para manipuladores antiguos, así como el uso de la estructura mecánica para integrar con el sistema de control de movimiento del eslabón final del robot con componentes de control actuales. LinuxCNC ha desarrollado una plataforma para que los desarrolladores puedan publicar nuevas configuraciones y beneficieren a la comunidad relacionada con este controlador, de esta forma es posible introducir la configuración del robot ASEA IRB6-S2, permitiendo complementar la biblioteca de cinemáticas no triviales con una opción adicional de maquina tipo manipulador.

PROPUESTA DE ARQUITECTURA ABIERTA USANDO LINUXCNC

Arquitectura LinuxCNC

LinuxCNC es un proyecto desarrollado por la NIST (National Institute of Standards and Technology), que termino incentivando la creación de una comunidad de desarrolladores voluntarios, generando que el código fuente pudiera ser modificado desde junio del año 2000. El proyecto es un controlador con licencia tipo GNU/Linux, con capacidad de controlar maquinas CNC y sistemas con cinemáticas avanzadas [10].

Actualmente LinuxCNC se encuentra continuamente desarrollado por la comunidad de programadores con interés en difundir sus ventajas del uso de este software a través de la simplificación, organización y mejoramiento del rendimiento del sistema de acuerdo con los feedback de los usuarios [11].

La estructura lógica de alto nivel de LinuxCNC está compuesta por 4 módulos con tareas independientes, con una capa principal denominada HAL (Hardware Abstraction Layer), siendo una interface entre software y

hardware, que provee una transferencia en tiempo real desde y hacia LinuxCNC con el hardware de control o módulos lógicos de bajo nivel [12]. En la Figura 1 es presentada la arquitectura general del controlador.

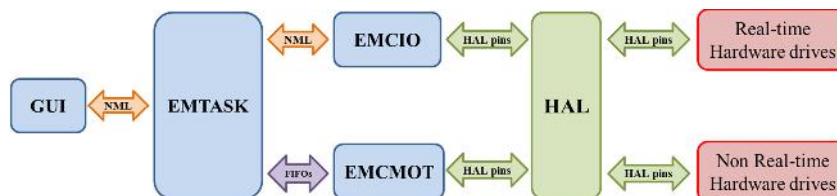


Fig. 1. Arquitectura lógica LinuxCNC

A continuación se presenta la definición de los módulos que integran la arquitectura lógica de LinuxCNC [13], y que será parte de la arquitectura propuesta: Motion Controller (EMCMOT), Discrete I/O Controller (EMCIO), Task coordinating module (EMCTASK), Text-based and graphical user interfaces (GUI).

LinuxCNC está en capacidad de ser un sistema confiable de control, de acuerdo a su estructura abierta para la administración de sistemas robóticos. Permite el control en tiempo real, interpretar programas hechos en código G, control de procesos en lazo abierto o cerrado, controle simultáneo de ejes o grados de libertad.

Actualmente las versiones disponibles de LinuxCNC tienen como ejemplos aplicaciones de máquinas CNC, y solo una configuración de manipulador tipo PUMA560, permitiendo añadir en la librería de ejemplos de este software una configuración adicional desarrollada desde el controlador. Generando mayor interés de los desarrolladores en hacer proyectos con robots tipo manipuladores como una alternativa tipo OpenSource.

Metodología Propuesta tipo OpenSource

La robótica industrial se inició rápidamente en Europa, con ABB y KUKA robotics, incorporando robots al mercado industrial en 1973. ABB (antes ASEA) desarrollo el IRB6-S2. Una empresa sueca compró los primeros dos manipuladores que la compañía comercializo en el año de 1974 [14]. Sera presentada a continuación, una propuesta de arquitectura abierta para un robot Asea IRB6-S2 compuesto de 5 DOF. El manipulador tiene más de 40 años de fabricación, y será integrado al estar nuevamente funcional a una célula de manufactura flexible (FMC), donde la función principal será el de transportar piezas brutas y terminadas, integrándose con diferentes tareas propias en el proceso de manufactura. La estructura del manipulador puede ser observada en la Figura 2.



Fig. 2. Robot IRB6-S2 en la FMC – UnB

La metodología propuesta volverá funcional al manipulador, y es presentada a través de fases generales en la Figura 3. De acuerdo al diagrama propuesto, es necesario para aprovechar la estructura mecánica del manipulador verificar el estado actual de las piezas que componen el robot, permitiendo evaluar si es necesario hacer mantenimiento a las partes o si hay que cambiar algunas piezas deterioradas por el uso prolongado en las tareas propias de un robot industrial [5].



Fig. 3. Metodología Retrofitting IRB-S2

En estructuras de robots antiguos no es viable reaprovechar todos los componentes, ya que existen componentes desgastados y obsoletos. Lo que genera consumo excesivo de energía y no existe confiabilidad para el trabajo futuro que va a tener el manipulador [8]. En el presente trabajo se opta por cambiar los componentes electrónicos y reutilizar únicamente la estructura mecánica del robot, según los componentes propuesto por Bomfim, et al. [8].

El sistema de control va a ser programado e implementado en LinuxCNC, mediante la estructura lógica presentada anteriormente, y por las ventajas como estabilidad y rendimiento [15]. LinuxCNC tiene la característica fundamental disponer el código fuente de manera libre lo que permite realizar cualquier modificación y posterior compilación.

Estructura de Control Propuesta

La propuesta para una arquitectura abierta de control para robots tiene una estructura en capas. La primera capa está compuesta por los componentes software, para el cálculo de las variables articulares, así como de la generación de señales hacia la capa hardware. Los componentes software son controlados por LinuxCNC. La segunda capa hardware se integra por la estructura mecánica del robot y los componentes de control especificados en el “Overhoulng-kit”. Gráficamente se puede apreciar la configuración propuesta en la Figura 4.

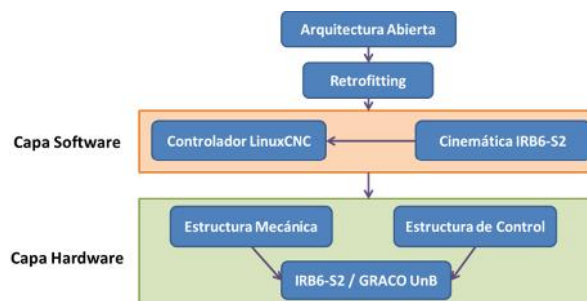


Fig. 4. Estructura de Control para Robot IRB6-S2

Para introducir la cinemática del robot en LinuxCNC es necesario de configuraciones particulares en el paquete específico de la distribución Linux usada, permitiendo la compilación del archivo con la cinemática del robot. Los pasos para integrar la cinemática del robot en el software de control están presentados en la Figura 5:

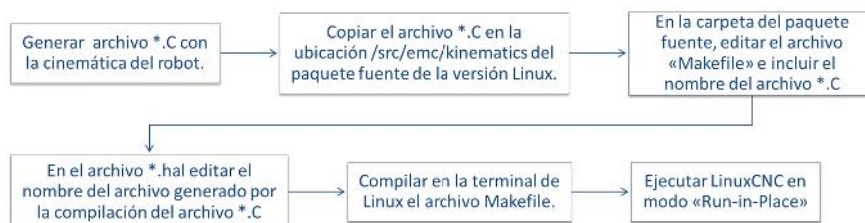


Fig. 5. Procedimiento inserción de la cinemática en LinuxCNC

Con el procedimiento anterior es posible alojar las ecuaciones homogéneas propias de la cinemática no trivial de cualquier robot tipo manipulador en LinuxCNC, permitiendo el control específico respecto a las características de fabricación de cada robot.

SISTEMA DE CONTROL USANDO LINUXCNC

Con el uso del controlador LinuxCNC es posible alojar en él la cinemática del robot como archivo ejecutable, lo cual permitirá a través del modelo virtual, la simulación de una trayectoria determinada generada en código G.

Cinemática del Robot IRB6-S2

Para un control eficiente de la posición y la orientación de la articulación final del manipulador, en el proceso de manufactura es necesaria la definición de un modelo cinemático que permita establecer una descripción matemática adecuada para los 5 grados de libertad del IRB6 [16]. Mediante los parámetros establecidos por algoritmo Denavit-Hartenberg (DH) para la relación entre dos eslabones contiguos es posible establecer los valores DH [17]. La definición de los sistemas de coordenadas se puede apreciar en la Figura 6.

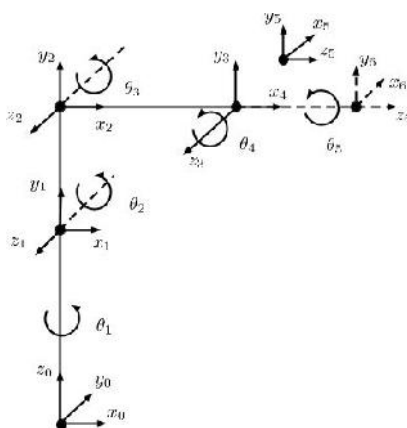


Fig. 6. Definición Sistema de Coordenadas IRB6-S2

Para una posición *home* como la representara en la Figura 6, se pueden inferir las variables DH, dispuestas en la tabla 1. Los sistemas de coordenadas 4 y 5 conviven en el mismo origen, los cuales forman el “puño” del robot:

Tabla 1. Parámetros DH del Robot IRB6-S2

Eslabón	α_i	$a_i[m]$	$d_i[m]$	θ_i
1	90	0	$\lambda_1 = 0.70$	θ_1
2	0	$l_2 = 0.45$	0	θ_2
3	0	$l_3 = 0.6$	0	θ_3
4	90	0	0	θ_4
5	0	0	$d_5 = 0.095$	θ_5

Para definir las variables articulares del robot en función de las coordenadas finales del efector final es necesario definir la cinemática inversa del mismo [18], obteniendo de esta forma las ecuaciones homogéneas que definen las variables articulares, las cuales representan el modelo fundamental para ser introducido en cualquier controlador. Luego de la definición de los parámetros DH, es posible definir el sistema de ecuaciones con las variables articulares del sistema [19], las cuales se pueden apreciar de la Ec. (1) a la (8):

$$\theta_1 = \arctan\left(\frac{-P_x}{P_y}\right) \quad (1)$$

$$\theta_3 = \arctan\left(\frac{S_3}{C_3}\right), S_3 = \frac{w_1^2 + w_2^2 - (l_2^2 + l_3^2)}{2l_2l_3}, C_3 = \sqrt{1 - S_3^2} \quad (2)$$

$$\theta_2 = \arctan\left(\frac{S_2}{C_2}\right), S_2 = \frac{w_2l_3C_3 - w_1(l_3S_3 + l_2)}{l_3^2C_3^2 + (l_3S_3 + l_2)^2}, C_2 = \frac{w_1l_3C_3 + w_2(l_3S_3 + l_2)}{l_3^2C_3^2 + (l_3S_3 + l_2)^2} \quad (3)$$

$$w_1 = -S_1p_x + C_1p_y + S_1a_x - S_1a_y, w_2 = p_z - l_1 - S_1a_z \quad (4)$$

$$}{}_5S_{34} = S_1S_2p_x - C_1S_2p_y + C_2p_z - }_1C_2 - l_2 - l_3S_3 \quad (5)$$

$$}{}_5C_{34} = -S_1C_2p_x + C_1C_2p_y + S_2p_z - }_1S_2 - l_3C_3 \quad (6)$$

$${}_{34} = \arctan\left(\frac{S_5S_{34}}{S_5C_{34}}\right), \quad \theta_4 = \theta_{34} - \theta_3 \quad (7)$$

$$\theta_5 = \arctan\left(\frac{S_5}{C_5}\right), \quad S_5 = C_1n_x + S_1n_y, \quad C_5 = C_1o_x + S_1o_y \quad (8)$$

Las ecuaciones presentadas anteriormente expresan la relación entre los ángulos de cada eslabón y las dimensiones del manipulador. Definidas dichas variables articulares el robot en función de la posición final de la última articulación, es posible generar movimiento deseado según los requerimientos atribuidos al robot. Permitiendo además de conocer el modelo matemático del manipulador, poder tener control del desplazamiento angular de los motores de cada uno de los eslabones, permitiendo el posicionamiento final del robot.

Simulación de la Cinemática IRB6-S2 en LinuxCNC

Con las ecuaciones homogéneas vinculadas al controlador LinuxCNC, resultantes de la cinemática del manipulador, es posible generar una simulación del movimiento del elemento final del robot que permita verificar el desplazamiento correcto en una trayectoria definida en un simulador disponible en el controlador. En la Figura 7 se puede apreciar el modelamiento del robot, de igual forma se aprecia la trayectoria predefinida que tendrá que seguir el robot permitiendo comprobar los valores generados de las variables articulares del sistema.

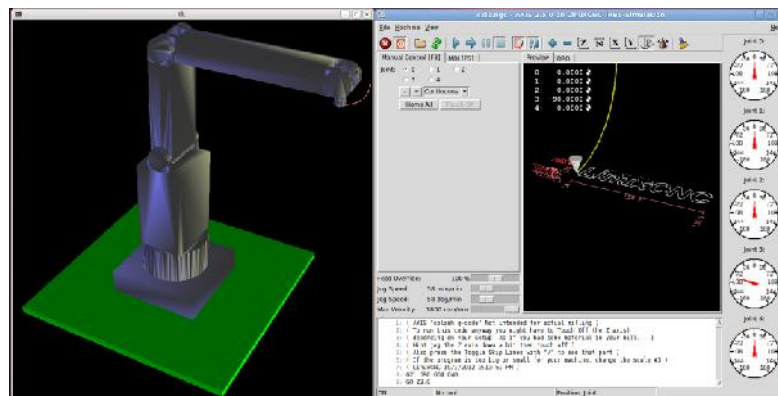


Fig. 7. Posición Home IRB6-S2

Después de finalizada la simulación del programa ejemplo realizada en código G, se aprecia en la Figura 8 como el robot siguió de manera correcta las instrucciones generadas y consiguió realizar la simulación de forma satisfactoria, permitiendo validar la cinemática del sistema, así como la integración entre la simulación del robot junto con el archivo compilado bajo LinuxCNC.

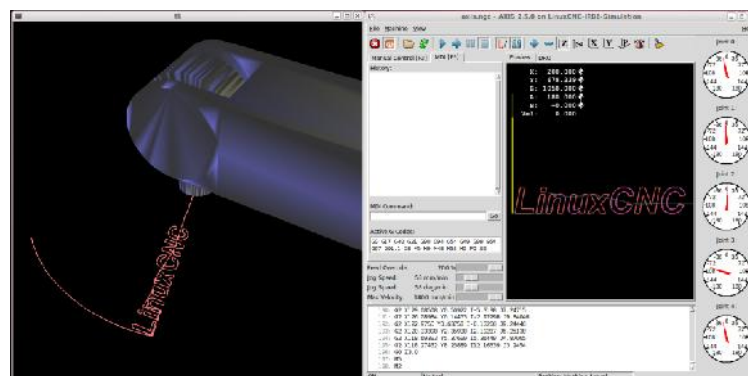


Fig. 8. Simulación IRB-S2 Finalizada

Es posible generar trayectorias específicas para el posicionamiento de piezas brutas y acabadas en LinuxCNC con instrucciones basadas en código G, permitiendo de esta forma una integración efectiva con la FMC para el posicionamiento de piezas brutas y terminadas.

Integración con la FMC

La función principal del robot IRB6-S2 en una FMC es el posicionamiento de piezas brutas y acabadas en los en la FMC del laboratorio Graco (UnB). El manipulador presente en la FMC está acompañado de equipos dispuestos en diferentes estaciones de trabajo, como una unidad de procesamiento (centro de mecanizado), unidad de inspección (micrómetro laser), unidad de almacenamiento de piezas (pallet), un vehículo tipo AGV, así como sistema de control (Unidad de Gestión) [20]. A continuación se presentara en la Figura 9 el modelamiento funcional de las tareas propias del robot dentro de una FMC, usando el diagrama IDEF0, permitiendo verificar las funciones propias del sistema de manipulación y transporte de piezas.

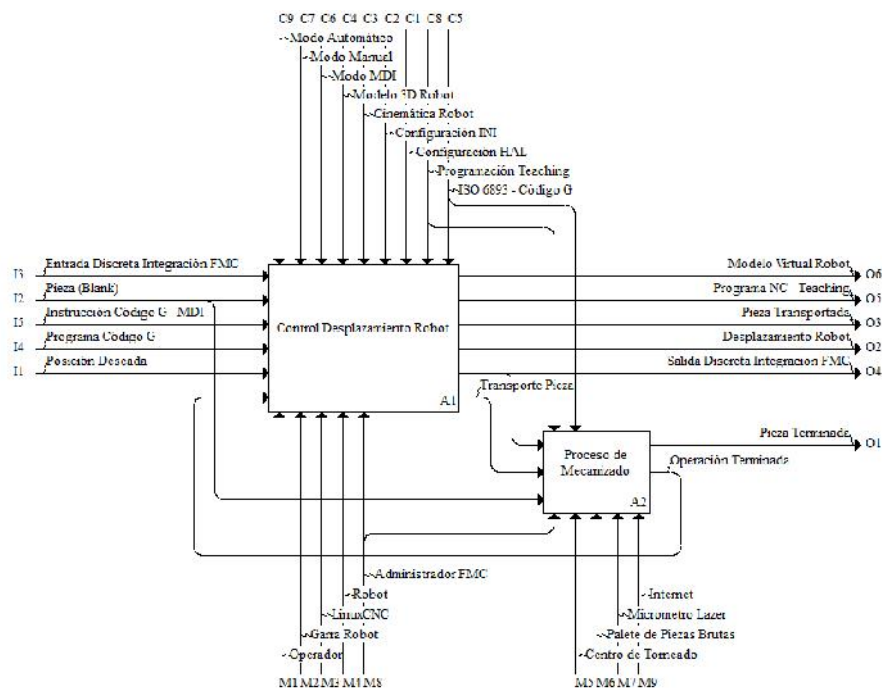


Fig. 9. Modelamiento IDEF0 – Sistema de Posicionamiento Robótico - FMC

Con el modelamiento en este tipo de diagramas es posible conocer las distintas variables que afectan un proceso determinado y como es su relación entre un proceso específico. De esta forma también es posible detallar de manera funcional las actividades que dependen del robot, específicamente la relación con las demás estaciones de trabajo presente en la FMC.

CONCLUSIONES

Fue presentada una metodología abierta para el Retrofitting de manipuladores robóticos que fueron fabricados hace más de 40 años, permitiendo la utilización de la estructura mecánica del robot y cambiando el sistema de potencia y control por tecnologías actuales.

El Retrofitting constituye una alternativa para la utilización de los mecanismos de los distintos sistemas que integran un robot fuera de operación, disminuyendo los costos en la implementación de esta metodología. Permitiendo el desarrollo de estrategias de controle, independiente del sistema original de fabricación.

LinuxCNC es usado como herramienta para el cálculo de las variables articulares mediante la cinemática del robot, así como el control del manipulador, generando flexibilidad a la arquitectura propuesta, por las características tipo OpenSource, así como la reducción de los costos asociados a la implementación dicha arquitectura. El uso de este controlador garantiza un comportamiento satisfactorio de acuerdo a los resultados plasmados en el desplazamiento del robot a lo largo de la trayectoria propuesta en la simulación en LinuxCNC.

Luego de la fase de modelamiento y validación de la cinemática con la simulación presentada, se inicia la etapa de sustitución del hardware asociado a los accionamientos del robot, como motores, encoders, drives de control con PID, fuente de potencia y la interface con LinuxCNC mediante puerto paralelo.

La modernización de este robot permitirá el desarrollo de nuevos proyectos que garanticen la consolidación del trabajo realizado con una arquitectura abierta, la cual soportara modificaciones para las nuevas tecnologías desarrolladas en robótica. Como ejemplo un robot fabricado en la década de los 70, que retornó a operación a través del Retrofitting.

REFERENCIAS

- [1] M. H. S. Bomfim, R. A. Gontijo, A. Q. Bracarense, and E. J. Lima II, "Overhauling of a ASEA robot IR6 with open architecture," *International Conference on Trust, Security and Privacy in Computing and Communications*, vol. XI, pp. 482-489, June 2012.
- [2] Walter Lages, Renato V. Bayan, and A. Q. Bracarense, "Arquitetura Aberta para Retrofitting de Robôs," UFRGS, São Paulo, 2003.
- [3] J.L. González, E. Baeyens, and F. Gayubo, "Desarrollo de un Controlador Abierto para un Robot Industrial Tipo SCARA," *RIAI*, vol. 1, no. 1, pp. 44-49, 2004.
- [4] Dragan Milutinovic, "Reconfigurable robotic machining system controlled and programmed in a machine tool manner," *The International Journal of Advanced Manufacturing Technology*, vol. 53, no. 9, pp. 1217-1229, April 2011.
- [5] Walter Lages and Alexandre Bracarense, "Robot Retrofitting: A Perspective to Small and Medium Size Enterprises," *Austrian-Brazilian Automation Day*, vol. III, pp. 1-11, 2003.
- [6] Eduardo Lima II et al., "Sensing For Retrofitting Of An Industrial Robot," *Information Control Problems in Manufacturing*, vol. I, pp. 545-550, April 2004.
- [7] A. Oliveira, E. De Pieri, and U. Moreno, "An Open-architecture Robot Controller applied to Interaction Tasks," *Advances in Robot Manipulators*, pp. 99-112, April 2010.
- [8] Marcelo Bomfim, Alexandre Bracarense, Fagner Coelho, Eduardo Lima, and Roberto Gontijo, "A Low Cost Methodology Applied To Remanufacturing Of Robotic Manipulators," *Congresso Brasileiro de Automática*, vol. XX, pp. 1506-1513, September 2014.
- [9] Rudy Preez, "3D 6-DOF Serial Armn Robot - Kinematics and Implementation in LinuxCNC," ASM, Pretoria, 2014.
- [10] Miguel Gutierrez, *Desenvolvimento de uma Fresadora CNC Aderente á Norma STEP-NC Baseado no Controlador de Máquina Avançado (EMC2)*. Brasília, Brasil: UnB, 2013.
- [11] LinuxCNC. (2014) [Online]. http://www.linuxcnc.org/docs/2.5/html/common/Getting_EMC.html
- [12] Tomislav Staroveški, Danko Brezak, Toma Udiljak, and Dubravko Majeti, "Implementation of a Linux-based CNC Open Control System," *International Scientific Conference on Production Engineering*, vol. XII, pp. 209-216, June 2009.
- [13] LinuxCNC. (2007, December) [Online]. http://wiki.linuxcnc.org/cgi-bin/wiki.pl?EMC_Components
- [14] Saleem Hashmi, *Comprehensive Materials Processing*, 1st ed. Netherlands, Amsterdam: Elsevier Science, 2014.
- [15] Milos Glavonjic, Dragan Milutinovic, Sasa Zivanovic, Zoran Dimic, and Vladimir Kvrjic, "Desktop 3-axis parallel kinematic milling machine," *International Journal of Advanced Manufacturing Technology*, vol. 46, no. 1, pp. 51-60, January 2010.
- [16] Bruno Siciliano, *Robotics: Modelling, Planning and Control*, 1st ed. London, England: Springer, 2009.
- [17] Antonio Barrientos, *Fundamentos de Robótica*, Segunda ed. Aravaca, España: McGraw-Hill, 2007.
- [18] Marcelo Bomfim, *Remanufatura de Manipuladores Robóticos Industriais Utilizando Arquitetura Aberta*. Belo Horizonte, Brasil: UFMG, 2013.
- [19] T Szkodny, "Forward and inverse kinematics of IRb-6 manipulator," *Mechanism and machine theory*, vol. 30, no. 7, pp. 1039-1056, 1995.
- [20] Evandro Teixeira, "Desenvolvimento da unidade de gerenciamento de uma célula flexível de manufatura integrada a um sistema CAD/CAPP/CAM," Brasília, 2006.