

## **Propuesta de un paquete integrado para el desarrollo de prototipos robóticos**

Ignacio Huitzil, Antonio Benítez, Aldo Hernandez, Oscar Flores y Alejandro Luna

I. Huitzil, A. Benítez, A. Hernandez, O. Flores y A. Luna  
Universidad Politécnica de Amozoc, Calle ampliación Luis Oropeza, N° 502 col. Ampliación San Andrés las Vegas  
72980, Amozoc Puebla  
ignacio.huitzil@upamozoc.edu.mx

M. Ramos., V.Aguilera., (eds.) .Ciencias de la Ingeniería y Tecnología, Handbook -©ECORFAN- Valle de Santiago,  
Guanajuato, 2014.

## Abstract

The development of robots or robotic prototypes is no easy task in this work is defined a way to work in a scheme involving the first three modules or software interface where the robot is simulated numerically or graphically, a communication module that manages the connection between physical logical part (software) and (hardware) and an execution module finalmente shares consisting of the dummy to be replicated to the real protipo. These three modules are applied to a project that uses a graphical interface Robot PUMA 560, a communication module is developed with the Arduino platform and tested with a minisumotype robot.

## 8 Introducción

Hoy en día en México, la robótica es un campo que se emplea en un número mayor en diversos sectores tales como el industrial, el de salud y educativo con la finalidad de tener una competitividad en la automatización de procesos para la generación de productos, en la calidad de vida de la sociedad e individuos capacitados. Así como, resolver problemas enfocados a la robótica y áreas vinculadas a ésta.

De tal forma, Oller (2001) define a la robótica como una ciencia encaminada a diseñar y desarrollar máquinas capaces de desempeñar tareas propias del ser humano en la ejecución de diversas actividades o trabajos, a través de la inteligencia artificial. Así también la Federación Internacional de Robótica (2014) tomando como referencia el ISO 8373 que un robot industrial es “un manipulador multipropósito controlado automáticamente, programable en tres o más ejes, el cual puede instalarse de manera estática o móvil para el uso en aplicaciones de automatización en la industria”. Es importante mencionar que la robótica se clasifica en cuatro generaciones de acuerdo con Ademir (2006): La primera generación llamada Manipuladores que son sistemas mecánicos multifuncionales con un sencillo sistema de control. La segunda generación titulada Robots de aprendizaje; donde repiten una secuencia de movimientos que ha sido ejecutada previamente por un operador humano. Tercera Generación nombrada Robots de control sensorizados. Los robots son programables, usan controladores o computadoras para analizar la información captada de su entorno mediante sensores y la cuarta generación o conocida como Robots inteligentes. Surgen a partir de mejores sistemas sensoriales, estrategias de control y análisis de información, capaces de comprender su entorno y realizar la planificación automática de sus tareas.

En esta investigación se ha retomado del trabajo de Benítez et al. (2011) en el cual presenta la simulación 3D de un brazo Robótico el modelo PUMA 560 y en su constitución presenta seis eslabones y seis grados de libertad (articulaciones de tipo rotatoria que están en función de un ángulo  $\theta$ , los primeros tres grados de libertad corresponden al brazo y los siguientes tres a efector final) y que resuelve la cinemática directa e inversa. La propuesta inicial es el uso de un esquema compuesto por tres módulos: interfaz gráfica de usuario, comunicación y ejecución de acciones, que interactúan entre sí para lograr una sinergia entre componentes para concluir en un paquete robótico funcional desde el software hasta llegar al hardware, por tal razón es incluida la interfaz gráfica de PUMA 560 y se describe detalladamente el proceso empenado el esquema modular.

En este artículo, se encuentra organizado de la siguiente manera en la primera sección se encuentra un estado del arte donde se presentan diversos trabajos que implica desarrollo de robots, en consecutivo la sección método se explica el esquema modular, posteriormente la sección de aplicación del esquema modular en el cual se pone en marcha el esquema propuesto.

Enseguida se presenta un apartado de pruebas con sus respectivos resultados, finalmente las conclusiones y el trabajo futuro.

### Estado del arte

En este apartado se presentan los diversos trabajos que involucran brazos robóticos y simulaciones, con las características más importantes. La finalidad es conocer el panorama y las propuestas de trabajos.

Las primeras propuestas presentadas por el Grupo de Automática Robótica y Visión Artificial (2010) que consiste de un brazo robótico físico modelo Scorbot ER-IX de cinco grados de libertad GL que posee un ambiente gráfico desarrollado bajo el lenguaje C++ del mismo modo (Trejo y Ríos 2005) hace empleo de este lenguaje y el desarrollo de un ambiente 3D. El trabajo de Grupo de Automática Robótica y Visión Artificial (2010) resalta por el empleo de un joystick para manipular un ambiente en 3D que permite programar movimientos, trayectorias donde posteriormente son ejecutadas en el robot físico. El código empleado en esta propuesta es abierto. Las herramientas matemáticas que se utilizan son: matrices de transformación homogéneas, los cuaternios y el algoritmo de Denavit-Hartenberg D-H.

Por otro lado, Blanco et al. (2004) desarrollaron Simurob un simulador de un robot antropomórfico IRB-1400 bajo la tecnología de Windows. Dicho robot es el empleado en la ETSII de la Universidad Politécnica de Cartagena (UPCT) para clases de prácticas. El software SimuRob fue desarrollado en Visual C++, y una de las ventajas que permite esta simulación es permitir a los alumnos aprender el manejo del robot y la programación de movimientos, el conocer la cinemática directa e inversa, la alineación de ejes y configuraciones singulares, entre otras. Un trabajo similar es definido por Carbonero y Vázquez (2004), con una simulación en 3D de un manipulador creada en Visual C++ y OpenGL. Cabe destacar que este trabajo ya cuenta con el robot físico y entre la simulación y el se realiza la replicación de movimientos en tiempo real (interacción medio de un teclado en 3D), el modelo del robot 3D se basa en un brazo robótico Puma 720, la librería que se utilizó para el desarrollo de dicho sistema fue Ecosimpro.

Un ambiente 3D de un robot tipo SCARA con 6 grados de libertad el cual fue programado en lenguaje C (lab Windows/CVI y OPEN-GL) propuesto por Barrera y Pérez (2005). En esta interfaz dan solución a la cinemática directa empleando D-H y realizan un análisis de la velocidad y fuerzas.

El trabajo Benítez et al. (2012), propone un manipulador físico de 4 GL de un kit Bioloid con servomotores Dynamixel AX-12 controlado mediante el lenguaje C++. También posee un simulador desarrollado en DevC++ utilizando el método D-H y el método geométrico para el problema cinemático directo e inverso. La cinemática directa solo fue probada en el prototipo y fueron detectados problemas con la arquitectura del kit Bioloid para la comunicación y programación.

Por otro lado, el trabajo Rodríguez et al. (2007) propone un ambiente 3D de un brazo robótico de 3 GL (visual C++ y OPEN-GL), los métodos para resolver la cinemática y dinámica son el método de Runge-kutta y D-H.

Finalmente, en el trabajo Martínrez et al. (2008) propone un robot físico el cual fue programado por coordenadas usando MATLAB, este brazo tendrá la característica de cargar hasta 4 Kilogramos. Para resolver la cinemática directa emplea la matriz de transformaciones y el método de D-H. El material utilizado para el diseño es aluminio, las herramientas utilizadas para el diseño del brazo fueron Solidworks y la tecnología CAT.

## 8.1 Método

En esta investigación se define un esquema modular para lograr prototipos robóticos que requieran de software (simulación numérica o gráfica) y hardware (manipulador) los cuales estén interconectados para lograr una sinergia entre estos dos elementos y finalmente obtener un paquete integrado.

A continuación se describen los tres módulos:

1. Interfaz Gráfica de Usuario IGU: Este módulo consiste en el empleo o desarrollo del software o programas que logren simular de forma numérica o gráfica problemas cinemáticos, dinámicos, leyes de control, generación de trayectorias o cuestiones robóticas, entre otras. Es importante considerar la morfología y características del robot a estudiar y simular. Para implementar la IGU existe una infinidad de software y lenguajes de programación libres y licenciados en los cuales se puede realizar la simulación requerida. Algunos ejemplos en los que se puede realizar estas simulaciones son en lenguajes estructurados C++, lenguajes orientados a objetos Java, Python, C#, .Net, y programas como Matlab, LabView, SolidWorks por nombrar algunos.
2. Comunicación: Este bloque, consiste en la interconexión entre la IGU y el robot físico para cumplir con el envío, recepción y procesamiento de datos entre ambas partes. La comunicación se puede realizar por diversos medios a través de una interfaz alámbrica USB, puerto serial, red ethernet o interfaz inalámbrica empleando Bluetooth, Wifi o radio frecuencia RF entre otros.
3. Ejecución de Acciones: Para este módulo es necesario la fabricación o la adquisición del robot que cumpla con las características de la IGU. El robot debe estar conectado o comunicado con el simulador en el que se cumplan uno de los dos aspectos que se mencionan enseguida: el primer que el usuario final ingrese datos o genere eventos sobre la simulación y éstas sean recibidas y replicadas por el robot (grabadas o en tiempo real) y un segundo aspecto cuando el robot percibe su entorno ya sea mediante sensores o el estado de los motores y otros datos que son útiles para la simulación que serán notificados de manera numérica o gráfica al usuario.

En la Figura. Se presenta el esquema con los tres módulos Interfaz Gráfica de Usuario, Comunicación y Ejecución de acciones. Cada uno se encuentra conectado en dos direcciones que indica el envío y recepción de datos necesarios para el funcionamiento de cada módulo.

Para lograr el tercer bloque es necesario que la IGU mande los datos al módulo de comunicación este los trate y sean enviados al robot para ser ejecutados, por otro lado el robot sense u obtenga una nueva configuración debe notificarle a la IGU mediante el módulo de comunicación.

**Figura 8** Representación del funcionamiento de un paquete software y hardware para protótipos robóticos



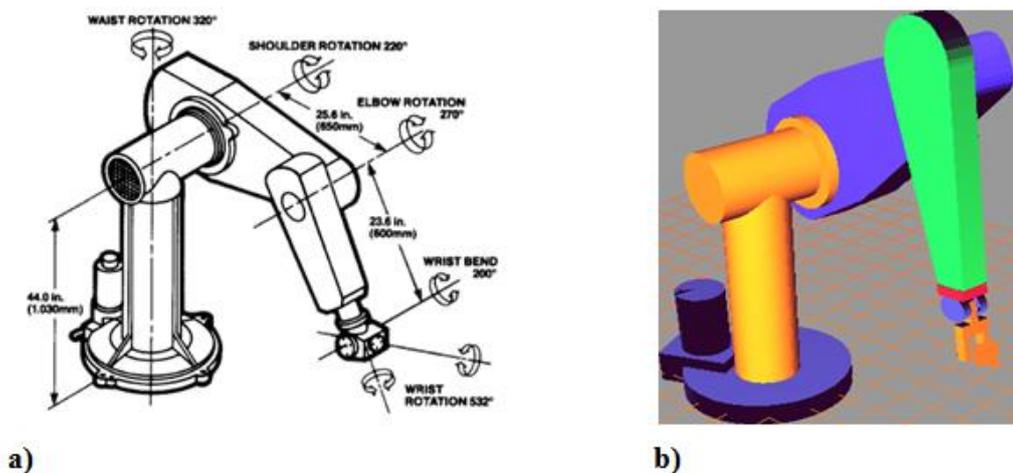
### Aplicación del esquema modular

En este apartado se describe la puesta en marcha del uso de los tres bloques para ellos se describe a continuación cada uno de ellos.

### Módulo Interfaz Gráfica de Usuario

Para esta investigación se emplea el trabajo de Benítez *et al.* (2011) el cual posee un ambiente 3D desarrollado en el lenguaje C++ y librerías de OpenGL graficando un manipulador PUMA 560. Éste consiste en una cadena cinemática que se define como una serie de elementos rígidos llamados eslabones (links) unidos mediante articulaciones (joints) que permiten un movimiento relativo entre dos eslabones consecutivos (Casillas (2000); Ademir (2006)). La constitución de robot PUMA 56 es de 6 grados de libertad, los tres primeros definen la posición y los tres últimos la orientación del efector final. En la Figura 2 se ilustra la morfología del robot y su modelo 3D.

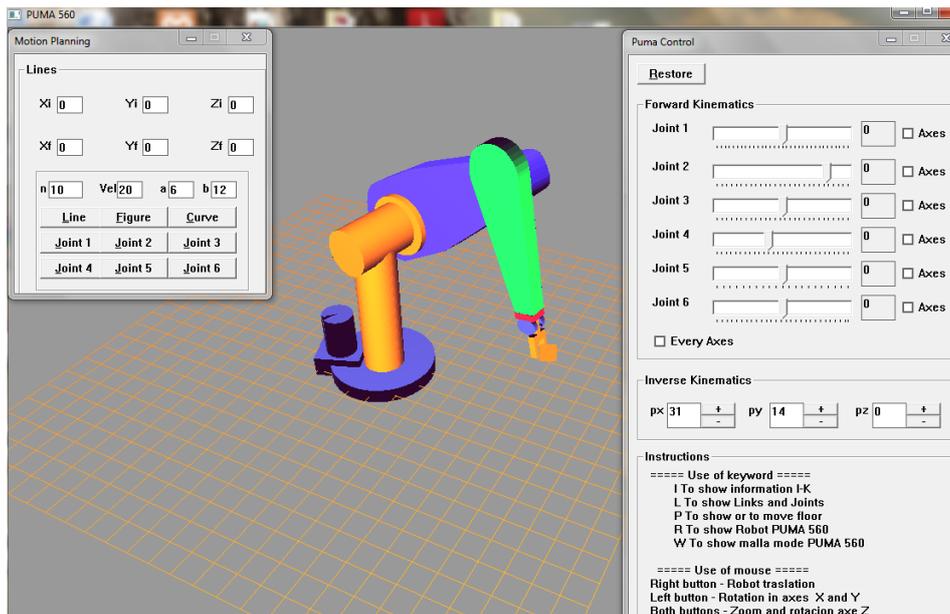
**Figura 8.1** Robot PUMA 560 , a) morfología y rangos de movimientos por grado de libertad y b) representación 3D de Benítez (2011)



Esta representación 3D resuelve los problemas cinemáticos tanto el directo como el inverso para ello debemos conocer a que se refiere cada uno de ellos. La cinemática directa consiste en determinar cuál es la posición y orientación del extremo final del robot con respecto a un sistema de coordenadas y la cinemática inversa determina el ángulo que debe adoptar cada una de las articulaciones con respecto a la posición y orientación del extremo final.

Para la solución de estos problemas Benítez et al. (2011) propone en la cinemática directa la multiplicación de matrices de transformación que tienen como entrada los ángulos de cada GL y en la cinemática inversa el uso del método geométrico que define una serie de ecuaciones y sus resultados son la entrada en la multiplicación matriciales. En la Figura 3 se presentan las soluciones cinemáticas, para la directa el usuario final manipula una barra de desplazamiento con los valores de los rangos de movimiento por articulación y en la inversa incrementa o decrementa la posición  $x, y, z$  del efector final. Otro apartado se encuentra en la esquina superior izquierda una ventana con trayectorias definidas y los rangos de movimientos del robot.

**Figura 8.2** Interfaz Gráfica de Usuario del robot PUMA 560 del trabajo Benítez (2012)



Es importante mencionar que esta IGU es flexible a nuestro proyecto además de permitir extender la simulación a otras cuestiones tales como la dinámica, leyes de control.

## Módulo comunicación

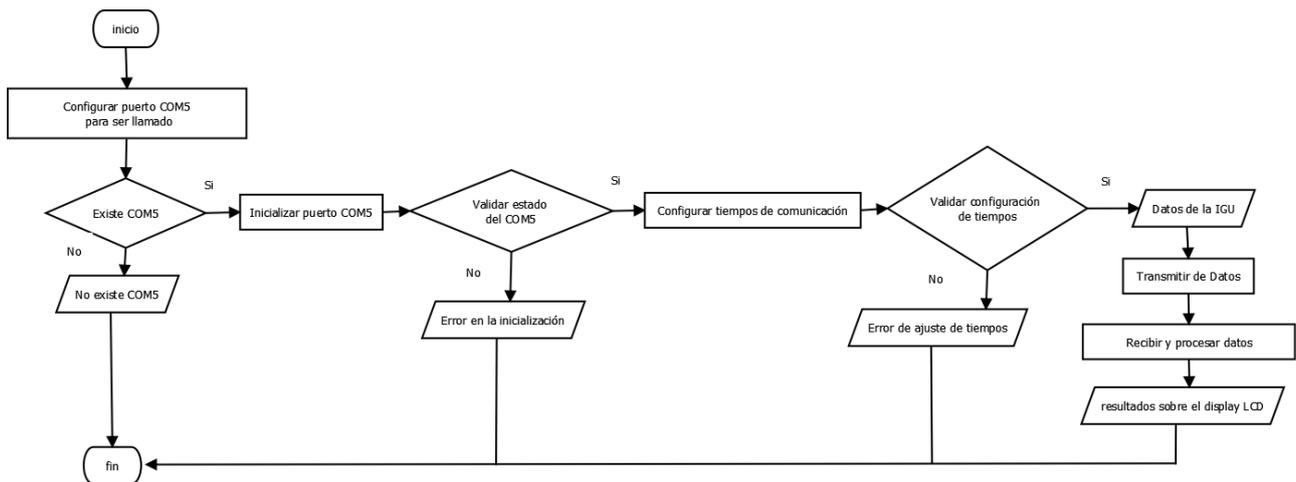
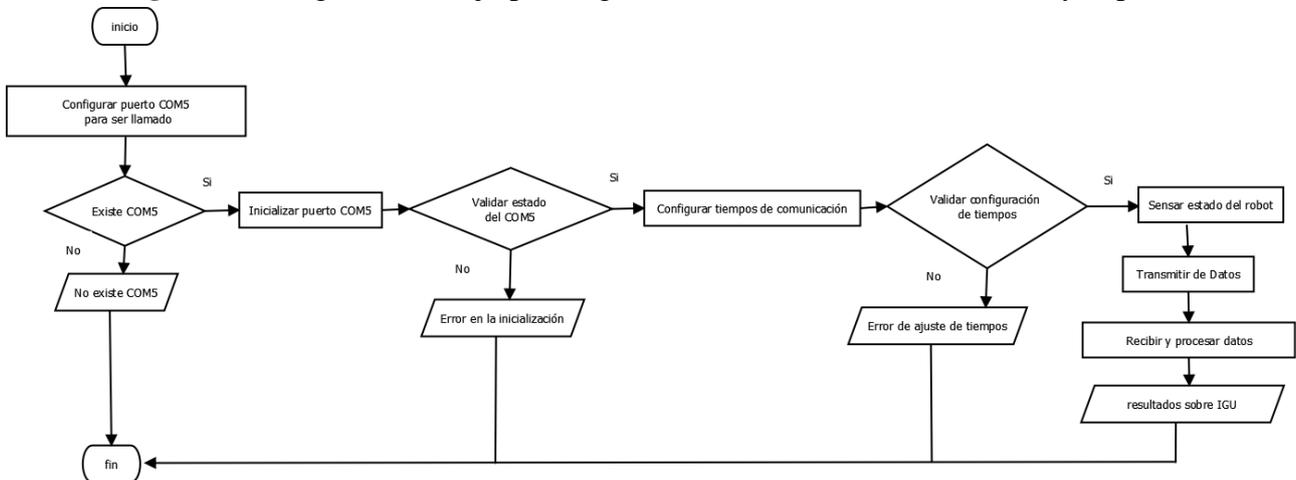
En un inicio la IGU PUMA 560 con la que trabajamos contaba ya con un módulo de comunicación del kit Biloid de tipo alámbrico mediante el puerto serial y USB pero se encontró una desventaja en cuanto a que solo trabaja con dispositivos propios de este kit limitando agregar nuevos aditamento como sensores y servomotores de características superiores. Ante esta desventaja se optó por trabajar en un nuevo módulo de comunicación el cual permita la heterogeneidad en el hardware del robot.

Para la comunicación entre la IGU y el robot físico se debe respetar el lenguaje de programación o el programa en el cual se desarrolla la simulación para este caso la simulación del ambiente 3D es desarrollada bajo el IDE de Dev C++ y la comunicación es síncrona ya que se requiere una respuesta en tiempo real. A continuación se mencionan los requerimientos de hardware y software con los cuales se trabajó el módulo de comunicación.

**Tabla 8** Materiales para el módulo de comunicación

Hardware	Software
<ul style="list-style-type: none"> <li>- Una Placa Arduino Mega y un chip atmega 2560</li> <li>- Display LDC</li> <li>- Clable USB</li> </ul>	<ul style="list-style-type: none"> <li>- Dev C++ (Versión 4.9.9.2)</li> <li>- Librerías para android y Dev C++</li> </ul>

El medio empleado para la comunicación entre la IGU y el robot es alámbrico utilizando el puerto serial COM5 USB conectado a la placa Arduino que funge como intermediario para conectar al robot. A grandes rasgos la comunicación consiste en la sincronía entre la IGU y Arduino a través del puerto serial, al estar listo el canal de comunicación, el módulo se encuentra en un estado de espera para la recepción o transmisión de datos permitiendo procesarlos para ser mostrados en un display.

**Figura 8.3** Diagrama de flujo para lograr la comunicación entre la IGU y la placa Arduino que muestra los datos recibidos y procesados sobre un display**Figura 8.4** Diagrama de flujo para lograr la comunicación entre la IGU y la placa

Arduino, el robot se encuentra sensando su ambiente y los datos son enviados a la IGU.

En el desarrollo de este módulo se tiene el reto de trabajar con la plataforma de Arduino que posee su propio software de desarrollo y el IDE de Dev C++ donde fue desarrollada la IGU. En ambos software el lenguaje de programación utilizado es C++ la diferencia reside en las librerías y en las instrucciones que cambian entre uno y otro, cabe destacar que el proceso de comunicación (envío y recepción de datos) se realizó en ambos software .

**Tabla 8.1** Lectura y escritura sobre el puerto serial en la plataforma Arduino y Dev C++

	Leer datos	Escribir o enviar datos
Arduino	<pre>if (Serial.available() &gt; 0) {   incomingByte = Serial.read(); }</pre>	<pre>Serial.write("PUMA 560");</pre>
Dev C++	<pre>ReadFile(hSerial, szBuff1, n, &amp;dwBytesRead, NULL);</pre>	<pre>WriteFile(hSerial, szBuff, 1, &amp;dwBytesRead, NULL);</pre>

### Módulo de ejecución de acciones

Para este apartado se optó por utilizar un kit de robot minisumo que tiene una placa Arduino, cuatro Micromotorreductores 50:1 y un sensor ultrasónico HC-SR04 en lugar de un robot manipulador de 6 GL el cual se encuentra en el proceso de fabricación por el área de Manufactura de la Universidad Politécnica de Amozoc. Con la finalidad de obtener resultados inmediatos este kit de robot móvil contó con una característica de cuatro motores que nos permite hacer la analogía de cuatro grados de libertad y un sensor que permite percibir el ambiente del robot.

Hacia el cumplimiento de los dos aspectos del módulo de ejecución de acciones 1) replicación y 2) percepción se realiza lo siguiente:

Para 1) una vez conectado robot a IGU, los motores de la cadena cinemática se enumeran por medio de un ID único. Por ejemplo, si el brazo a considera tiene 4 motores los identificadores son del motor 1 al 4. El motor tiene la capacidad de girar de forma horaria y anti horaria de acuerdo al tiempo. La comunicación desde la IGU con los motores se determinó mediante de una estructura tipo archivo json o archivo xml. Por ejemplo:

```
M1:{D: (f ó a),T:5},
M2:{D: (f ó a),T:5},
M3:{D: (f ó a),T:5},
M4:{D: (f ó a),T:5}
```

Donde:

M=Motor+ID, D=Dirección, f=movimiento horario, a=movimiento anti horario, T=Tiempo

De tal forma si mandamos a la placa arduino desde la IGU lo siguiente:

```
M1:{D: (F),T:15},
M2:{D: (A),T:10}
```

La placa arduino procesa estos datos y establece que el motor 1 tendrá una dirección horaria por 15 milisegundos y el motor 2 posee una dirección anti horario por 10 milisegundos. Para ello, el procesamiento de los datos que se emplea es el software de Arduino.

Para 2) el sensor ultrasónico de distancia permite detectar objetos a una distancia de 2cm a 4m y con un ángulo efectivo de 15° a partir de estos datos percibe los objetos que se encuentren en su ángulo de visión y la acción a ejecutar consiste en parar los cuatro motores.

## 8.2 Resultados y discusión

El esquema modular que define este proyecto permite trabajar por secciones en cuanto al desarrollo de un paquete que integra elementos tales como la interfaz de usuario, la interconexión robot con la interfaz y la interacción entre el software y el robot físico cabe mencionar que estos módulos son aplicables a cualquier tipo de proyecto con protótipos que pasen de la parte lógica de una computadora a lo real.

Los resultados obtenidos en relación al esquema modular son los siguientes:

1. El módulo de IGU empleado es parte de un trabajo previo el cual es flexible gracias a que se le pueden incorporar nuevas simulaciones tanto numéricas como gráficas en esta investigación.
2. En el módulo de comunicación es implementado en sobre una placa Arduino que comunica a la IGU y al robot. Se logró la comunicación con el IDE Dev C++ y el software de Arduino para lograr el envío, recepción y procesamiento de los datos ver Figura 6.
3. Para el módulo de replicación se obtuvo como resultado el envío de comandos para mover motores a partir de una terminal del IDE de Dev C++ ver Figura 6 inciso a), sobre el un kit minisumo (Ver Figura 6 inciso b) ) el cual pone en marcha los motores en relación al comando entrante y por su parte, el robot el sensor de cercanía de un objeto al detectarlo para los motores (ver Figura 6 inciso c) ).

## 8.3 Conclusiones y trabajo futuro

Iniciar proyectos que incurran al desarrollo de protótipos es necesario seguir un esquema de trabajo o metodología de desarrollo entorno a qué elementos son necesarios para obtener un resultado satisfactorio.

En este trabajo se establece un esquema modular que define tres módulos: el primer, una simulación hablando del lado del Software; segundo, la sincronización de la parte lógica a el mundo real como lo es un prototipo robótico y una tercera, en el que permite a partir de la comunicación ejecutar acciones que son introducidas por un usuario final o a partir del modelo físico comunicar su estado.

También, se concluye que este esquema es puesto en marcha logrando trabajar con los tres módulos.

En un paquete con la interfaz de usuario, la comunicación, la replicación y percepción del ambiente de un robot.

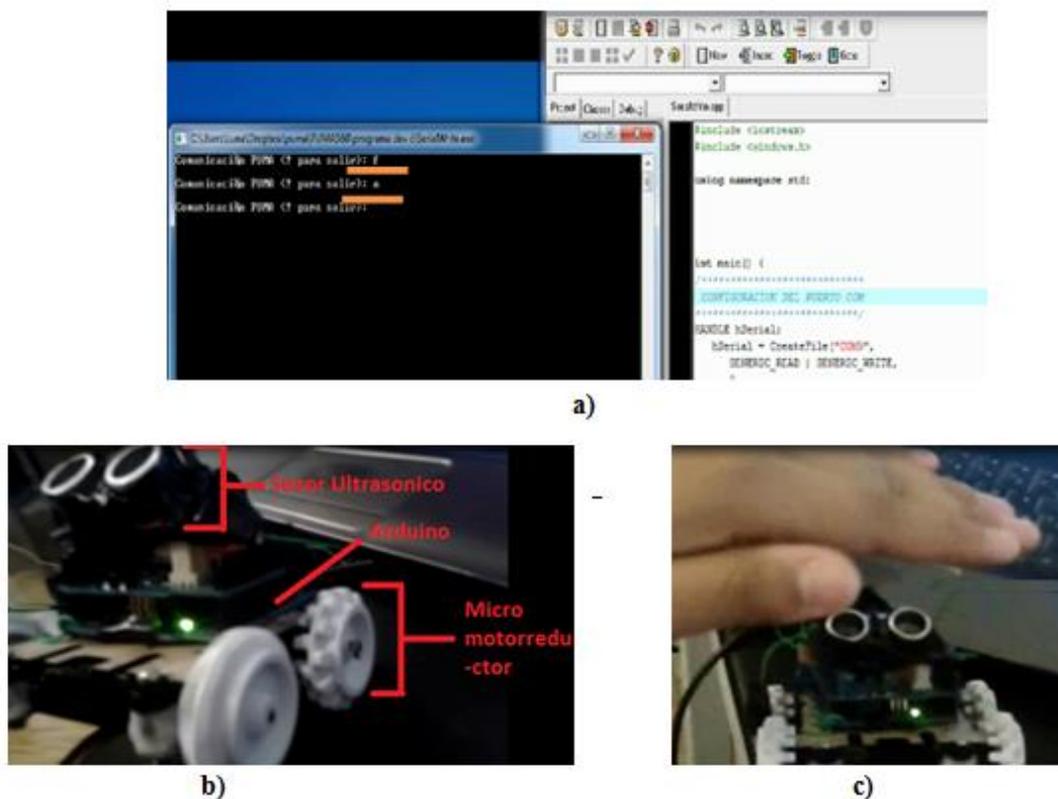
Enfatizando que el robot utilizado no cumple con la especificación de la interfaz y se encuentra en proceso de fabricación al finalizar se incorporará a este proyecto además de hacer la pruebas de comunicación y replicasiones pertinentes.

Con respecto al trabajo futuro se busca integrar a la interfaz nuevas simulaciones como la dinámica y leyes de control entre otras.

Con el objetivo de observar y analizar el comportamiento en el mundo real.

Asi también, se busca incorporar otros aditamentos tales como el uso de guantes, dispositivos móvileso kineck que permitan controlar tanto la simulación como el robot físico.

**Figura 8.5** Representación de los módulos de comunicación y ejecución de acciones, a) pruebas de comandos, b) robot empleado y puesta en marcha de los motores y c) detección de objetos y apagado de motores



## Referencias

Ollero, A. (2001). Robótica manipuladores y robot móviles. España: Marcombo.

IFR International Federation of Robotics (2014), Industrial robots, Consulta el 08 de octubre de 2013, <http://www.ifr.org/industrial-robots/>

Ademir, W. (2006). Cinemática directa aplicada a un actor digital. (Tesis de licenciatura). Universidad Iberoamericana Puebla, Puebla.

Benítez A., Huitzil I., Casiano-Ramos A., Medina Ma. A. y de la Calleja J. (2011). A 3D Simulation Environment for Kinematic Task of the PUMA 560 Robot. *Proceedings of the 21st International Conference on Electronics Communication and Computers (CONIELECOMP 2011)*, pp. 260-265

Grupo de Automática Robótica y Visión Artificial (2010). Control cinemático de un brazo robot: Robot Scorbot ER-IX, Consulta el 09 de septiembre de 2013, <http://www.aurova.ua.es/>.

Trejo, M y Ríos C. (2005). Hernández F., Vargas J. y Pedraza J., Software de Simulación de un Robot Manipulador Marca Mitsubishi RV-2AJ, 4° *Congreso Nacional de Mecatrónica 3-5*, Coahuila, México.

Blanco B., Marina, Batle F., Jorge J., Izquierdo C. y José Manuel. (2004) Simurob. Simulador del robot IRB -1400. En: *Jornadas de Automática ( 25º: Ciudad Real : 2004)*. XXV *Jornadas de Automática*. 5 p. ISBN 84-688-7460-4

Carbonero, F. y Vázquez F., (2001) Creación de Entornos 3D con Open-GL y Ecosimpro: Aplicaciones a la Robótica, *Jornada de usuarios EcosimPro*, Madrid España, pag. 8

Barrera E., Pérez W., (2005), Simulador de la Cinemática de un Robot Manipulador de Seis Grados de Libertad, 1° *Foro Académico de la DES de Ingenierías y Arquitectura de la Universidad Michoacana de San Nicolás de Hidalgo*. Morelia, Michoacán, México.

Benítez A., Huitzil I., Casiano-Ramos A., de la Calleja J. y Medina Ma. A. (2012), PUMA 560: Robot Prototipo with Graphic Simulation Environment. *Advances in Mechanical Engineering*. Vol 2. No. 1, pp. 17-24

Rodríguez, A., Ortega, M. & Leal, D. M. (2007). Seguimiento adaptativo de trayectorias con convergencia en tiempo finito de un robot Antropomórfico virtual de tres grados de libertad. (Tesis de licenciatura). Universidad Autónoma del estado de Hidalgo, Hidalgo.

Martínez A, Gloria M., Jáquez O., Rivera M. y Sandoval R. (2008), Diseño propio y Construcción de un Brazo Robótico de 5 GDL, *Riee&C, Revista de Ingeniería Eléctrica, Electrónica y Computación*, Vol. 4 No. 1

Casillas S. (2000). Robótica: Morfología y cinemática. Centro Universitario de Ciencias Exactas e Ingenierías. Universidad de Guadalajara. México. Consulta el 12 de octubre de 2013, <http://proton.ucting.udg.mx/materias/robotica>