

“Siéntete un Cyborg II”

Sistema demostrativo del funcionamiento de un Robot

Francisco Javier Vera, Julio Pastor Mendoza, Esther Samper Domeque

Departamento de Electrónica

Universidad de Alcalá

Alcalá de Henares (Madrid) - España

fjvera04@gmail.com, pastor@depeca.uah.es, esamper_18@hotmail.com

Abstract— Este documento presenta un sistema de monitorización y telecontrol de una plataforma móvil robótica, con posibilidad de comunicación vía RS232 o conexión inalámbrica mediante Bluetooth.

Se han diseñado tres aplicaciones o GUIs (Graphical user interfaces) con interfaz de usuario amigable y accesible con el fin de que cualquier persona pueda interactuar, en tiempo real y de forma remota, a través de ellas con los diferentes tipos de robot móviles que permite controlar. Para ello se pone a disposición del usuario un volante USB con pedales de aceleración y freno, desde los cuales se le permite pleno control sobre los parámetros de movilidad del robot. Para este desarrollo software se ha trabajado en lenguaje de programación JAVA fundamentalmente por ser un lenguaje multiplataforma.

Este sistema es una herramienta para acercar el mundo de la robótica a personas con diferentes niveles de conocimiento, permitiendo explicar de forma sencilla los diferentes elementos que forman un robot, así como su funcionamiento y su interacción con el entorno. El sistema también puede ser utilizado como herramienta didáctica para prácticas de asignaturas donde se estudien Sistemas Empotrados, Java y Robótica Básica.

Keywords – robótica, monitorización, telecontrol, Bluetooth, RS232, Netbeans, java, J2SE, ARCOS

I. INTRODUCCIÓN

El sistema presentado pretende ser una herramienta de acercamiento al mundo de la robótica con el que poder enseñar el funcionamiento de un robot. El objetivo principal para el que se diseñó fue para ser utilizado en exhibiciones, ferias de divulgación y en actividades de promoción de la ciencia, la tecnología y la ingeniería entre los más jóvenes como demostrador didáctico.

En concreto, este diseño permite experimentar cómo funciona el sistema sensorial de un robot para captar información del entorno, y cómo, en base a esa información, el robot toma decisiones que, en este caso, se traducen en movimiento. Los usuarios ven en una pantalla lo que percibe el robot por sus sensores, toman decisiones de movimiento y las ejecutan moviendo un volante siendo éstas transmitidas al robot como órdenes de movimiento. El usuario pone, por tanto, la inteligencia del robot.

El proyecto Cyborg II es la evolución del proyecto Cyborg I [1] [2] presentado con éxito en diversas ferias de promoción de la robótica y la tecnología, introduciendo un interfaz gráfico basado en Java que facilita su uso en diversas plataformas diferentes y ampliando sus funcionalidades permitiendo a los participantes telecontrolar tres tipos de robots mediante un volante con pedales de freno y acelerador, con ayuda de una comunicación Bluetooth para la transmisión de datos.

Mediante esta actividad práctica el usuario interactúa con el robot y de una forma lúdica se capta su atención, para posteriormente captar su interés por conocer el sistema y al principal protagonista, el robot.

Otra aplicación del sistema puede ser su utilización como herramienta docente en asignaturas en las que se enseñe a programar microcontroladores, programación en java o teoría de control. Los alumnos de forma práctica, amena y divertida podrán realizar diferentes aplicaciones y aprender sobre distintas áreas de conocimiento.

Finalmente hay que añadir que este sistema puede ser de gran utilidad también desde el punto de vista técnico, ya que puede utilizarse para facilitar el proceso de perfeccionamiento de la programación de un robot. En muchas ocasiones resulta complicado averiguar la causa de un determinado comportamiento no deseado de un robot, pero gracias a este sistema, que ofrece una monitorización y un telecontrol continuo y en tiempo real, se podría realizar una completa depuración de su funcionamiento de una forma extremadamente sencilla e intuitiva que permitiría estudiar y corregir los problemas.

II. DESCRIPCIÓN GENERAL DEL SISTEMA

Una visión general del sistema, se puede observar en la Figura 1. , en ella se muestra un esquema de la estructura general y los elementos principales del sistema, pudiéndose observar los dos robots sobre los que van a trabajar las aplicaciones software: el robot denominado “Cyborg”, desarrollado por el Departamento de Electrónica de la Universidad de Alcalá y justamente debajo el robot “Pioneer”, de la compañía MobileRobots [3] ampliamente utilizado en proyectos de investigación. En la figura también se muestra que el sistema incluye una interfaz de usuario gráfica que se ejecuta en un ordenador al que se conecta un volante y unos

pedales de acelerador y freno. La comunicación entre el ordenador y los robots puede realizarse mediante un cable serie o utilizando Bluetooth.



Figura 1. Estructura general del sistema.

En este sistema los robots no son autónomos sino que envían la lectura de sus sensores al ordenador que ejecuta la interfaz gráfica la cual presenta la medida gráficamente. Un usuario, con la información de la pantalla, actúa sobre el volante y los pedales y telecontrola el movimiento del robot.

Por tanto, en este sistema la inteligencia del control está en el operador cumpliendo así su principal objetivo que es que los usuarios sean conscientes de las acciones que suelen realizar los robots de forma autónoma.

La comunicación entre el ordenador de visualización y el robot se realiza utilizando el protocolo ARCOS desarrollado por la empresa MobileRobots como interfaz entre el sistema empotrado que controla el hardware del robot y un ordenador que ejecuta las acciones de alto nivel. En el robot Cyborg se ha emulado el protocolo ARCOS [4] para mantener la compatibilidad con los robots Pioneer.

III. PLATAFORMA ROBÓTICA MÓVIL

El robot desarrollado por la Universidad de Alcalá y utilizado, como robot principal en este sistema, es el robot denominado "Cyborg", representado en la Figura 2.

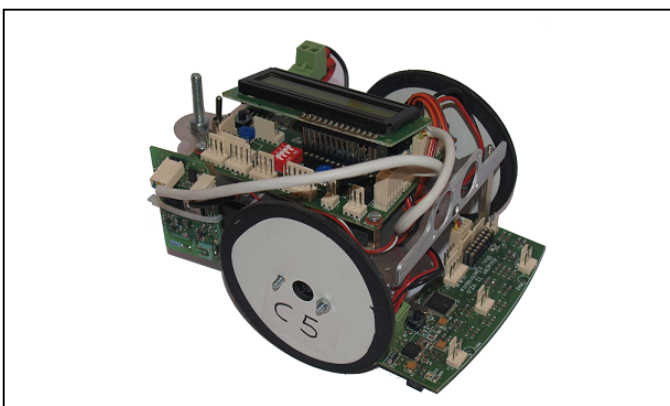


Figura 2. Robot Cyborg

Consiste en una pequeña plataforma robótica móvil de bajo coste de tamaño 16 x 11,5 x 9,5 cm, montada sobre una

estructura de aluminio, Figura 3., de 1,5 mm de espesor cortada por láser y plegada, diseñada a medida para poder fijar los distintos componentes que forman el robot.

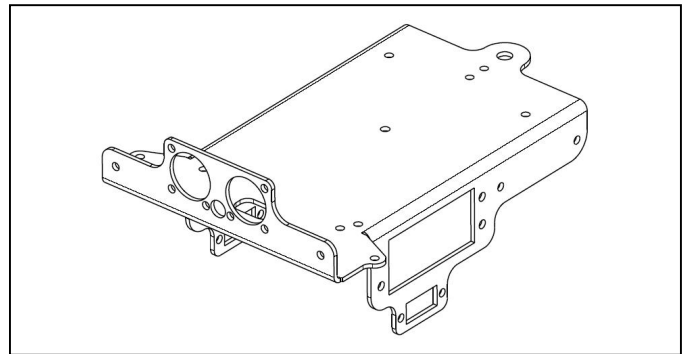


Figura 3. Estructura de aluminio

"Cyborg" es un robot de tracción diferencial con dos ruedas, movidas por un servomotor trucado y situadas diametralmente opuestas en un eje perpendicular a la dirección del robot. Además tiene un apoyo en la parte trasera para mantener la horizontal en todo momento y evitar cabeceos al cambiar de dirección.

En cuanto a la electrónica, el robot está formado por cuatro tarjetas diseñadas en el Departamento de Electrónica de la universidad de Alcalá, visibles en la Figura 4. .

- La tarjeta AlcaDspic es parte central del sistema de control, cuyo nombre se debe a que el procesador central es un microcontrolador de la familia dsPIC de Microchip.
- La tarjeta AlcaCnyII que posee diez sensores reflexivos y que envía la información captada por estos mediante I2C a la tarjeta principal.
- La tarjeta AlcaBluetooth que dota al robot de conectividad Bluetooth, contiene una antena y el módulo Bluetooth comercial GS-BT2416C1.AT1 fabricado por ST Microelectronics, y se conecta a una de las UARTs del microcontrolador de la tarjeta principal, por lo que el intercambio de datos entre estas dos tarjetas se realiza mediante una comunicación serie.
- Por último, se ha desarrollado una tarjeta para control de ocho sensores de ultrasonidos con conexión I2C. Los sensores utilizados han sido los SRF02, que disponen de nivel de tensión TTL y capacidad de medida de 15 a 600 cm, además de poder conectar hasta 16 sensores en el mismo bus I2C cambiando la dirección interna de cada uno de ellos para que no se produzcan duplicidades. Utiliza un único traductor tanto para enviar la ráfaga ultrasónica como para recibir el eco. Esta placa se encuentra en fase de ensayo y se ha verificado su funcionamiento con éxito en la aplicación software que permite al robot "laberinto" la detección de obstáculos alrededor de sí mismo, permitiendo su uso en este tipo de entornos.

El control de los dos servomotores que mueven las ruedas se realiza utilizando señales PWM generadas por el microcontrolador central. Todas las tarjetas que forman el robot, así como los servomotores, están alimentados mediante una batería de Litio-Polímero de 7,4V.

En la Figura 5. se puede observar el esquema electrónico descrito anteriormente.

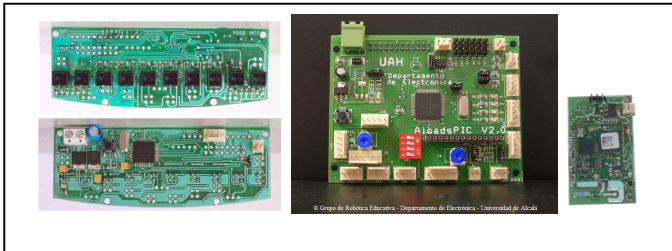


Figura 4. Tarjetas electrónicas. De izquierda a derecha: AlcaCny II, AlcaDspic, y AlcaBluetooth.

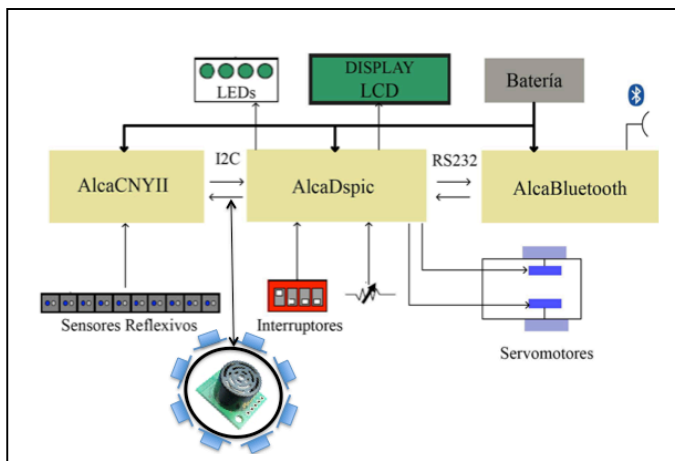


Figura 5. Esquema Hardware del robot "Cyborg".

IV. COMUNICACIÓN

En este proyecto, la comunicación que se establece entre el robot y la aplicación de usuario se puede realizar a través de una conexión RS232 o Bluetooth, usándose RS232 para la parte de desarrollo en laboratorio y Bluetooth para pruebas finales y demostraciones ya que el alcance y la velocidad que ofrece esta tecnología son suficientes para el correcto funcionamiento de esta aplicación, siendo su coste y su consumo considerablemente inferior al de la tecnología WiFi.

La comunicación inalámbrica mediante Bluetooth se realiza utilizando el perfil de Bluetooth SPP (SerialPort Profile), basado en los estándares de telecomunicaciones europeos ETSI 07,10 y en el protocolo de comunicación RFCOMM, el cual realiza la emulación de un cable serie, sobre el protocolo L2CAP encargado de reestructurar de forma correcta los datos a enviar, facilitando el desarrollo de la aplicación puesto que permite implementar la comunicación como si se utilizase un cable físico.

Este desarrollo utiliza la API BlueCove de Java que maneja comunicaciones a través de dispositivos Bluetooth. Fue, Originalmente desarrollado por Intel y actualmente es mantenido por voluntarios. Utiliza la biblioteca de funciones JSR-82 para J2SE. Esta biblioteca hace de software intermedio entre la aplicación en java y el SO del ordenador de visualización, el cual interactúa con el Bluetooth externo conectado por USB.

Para poder llevar a cabo esta comunicación, tanto el ordenador en el que se ejecuta la aplicación de usuario, como la plataforma robótica móvil con la que se desea conectar, deben poseer la tecnología Bluetooth necesaria. Si el ordenador no posee capacidad de conectividad Bluetooth se utiliza un adaptador externo USB-Bluetooth, y en cuanto al robot, se utiliza la tarjeta AlcaBluetooth comentada anteriormente. Esta tarjeta, en concreto el módulo que contiene, permite realizar la comunicación Bluetooth utilizando el perfil de puerto serie (SPP), conectándolo directamente a la UART del microcontrolador, de esta forma el programador se abstrae de la comunicación Bluetooth y sólo debe pensar en usar el puerto serie del microcontrolador.

Con respecto a la comunicación entre el software y el periférico USB, volante y pedales, se ha usado la biblioteca de funciones de Centralnexus para manejadores de eventos desde el puerto USB. Esta API, al igual que la de Bluecove, hace de software intermedio entre las clases creadas en la aplicación en java y el SO del ordenador de visualización.

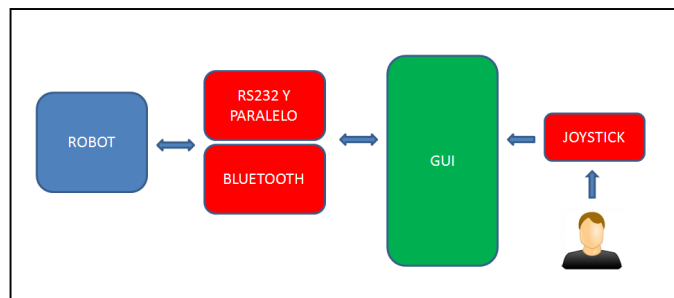


Figura 6. Esquema de funcionamiento.

Finalmente, hay que señalar que este intercambio de información entre los diferentes robots y las aplicaciones de telecontrol java, siguen las especificaciones del protocolo estándar de comunicación de datos ARCOS, protocolo utilizado por el software del mismo nombre, ARCOS (Advanced Robotics Control Operating System), software cliente/servidor de bajo nivel que hace de interfaz para el control del robot en las plataformas de MobileRobots. ARCOS define un protocolo para la comunicación mediante paquetes entre cliente y servidor. La plataforma robótica es el servidor, que espera recibir órdenes de la aplicación de monitorización y telecontrol, que actúa como cliente. Las tramas que envía el servidor tienen la función de informar al cliente del estado del robot: velocidad, alimentación, lectura de los sensores, valores en las entradas,... Sin embargo, las tramas enviadas por el cliente contienen órdenes para configurar, controlar y manejar el robot, así como para solicitar el envío de determinada información.

Este protocolo define una estructura de trama que se representa en la Figura 7. Como se puede observar, estará formada por dos bytes de cabecera, un byte de cuenta, los datos y dos bytes de checksum. La trama no puede ser mayor de 207 bytes, por lo que el campo datos no puede contener mas de 202 bytes.

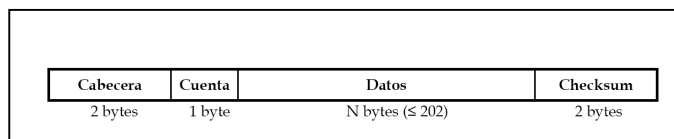


Figura 7. Esquema de tipo de trama.

Mediante una comunicación serie y utilizando este software se puede realizar el control de los motores, las lecturas de los sensores y otros procesos básicos sin necesidad de manejar directamente el hardware del robot y permitiendo la portabilidad a otros robots que tengan instalado este software. Además dicho control se puede realizar utilizando Player que es un servidor multihilo de código abierto para el control de robots móviles, que se comporta como una capa de abstracción hardware y que posee implementados un conjunto de drivers que permiten controlar muchos robots comerciales, entre ellos el driver p2os que se utiliza para controlar los robots de MobileRobots utilizando el protocolo de comunicación ARCOS. Por tanto, este protocolo se eligió porque no sólo permite controlar el intercambio de información, que es el objetivo principal que se deseaba conseguir, sino que también permitía ampliar la funcionalidad del sistema.

El software implementado para el robot “Cyborg” ofrece la posibilidad de controlar el robot utilizando Player y además, al ser un protocolo implementado en todas las plataformas de MobileRobots, la aplicación de monitorización y telecontrol diseñada podrá utilizarse también con estos robots. Esto es importante ya que las plataformas de MobileRobots se utilizan como plataformas estándar de investigación en muchas Universidades y Player es la herramienta software para el control de robots móviles más utilizada del mundo.

V. DESCRIPCIÓN DE LA ARQUITECTURA SOFTWARE

El Software que se ha desarrollado para este proyecto se divide en dos grandes bloques: la programación del robot “Cyborg”, concretamente de su microcontrolador principal ubicado en la tarjeta AlcaDspic, y la aplicación usuario de monitorización y telecontrol programada en JAVA.

A. Software del Robot Cyborg

El lenguaje que se ha utilizado para programar el robot “Cyborg” es el lenguaje C, el entorno de desarrollo MPLAB-IDE y el compilador MPLAB C30, proporcionados por Microchip para poder programar sus microcontroladores, en este caso el dspic30F6010A. Una descripción detallada de esta implementación se puede observar en trabajos anteriores [1].

B. Aplicación de usuario monitorización y telecontrol

La aplicación de usuario se ha desarrollado en lenguaje de programación JAVA por ser más fácil de actualizar y mantener que las versiones anteriores realizadas en LabView y por ser multiplataforma permitiendo ser ejecutado en sistemas operativos como Windows, Linux, Mac, etc... Actualmente se está trabajando en realizar las modificaciones necesarias para que también pueda ejecutarse en dispositivos móviles.

La función principal de la aplicación desarrollada es monitorizar y telecontrolar, de forma remota y en tiempo real, una plataforma móvil robótica, utilizando para ello una conexión Bluetooth y el protocolo de comunicación ARCOS, enviando las órdenes recogidas de un mando de control y visualizando la información recibida de los sensores y actuadores.

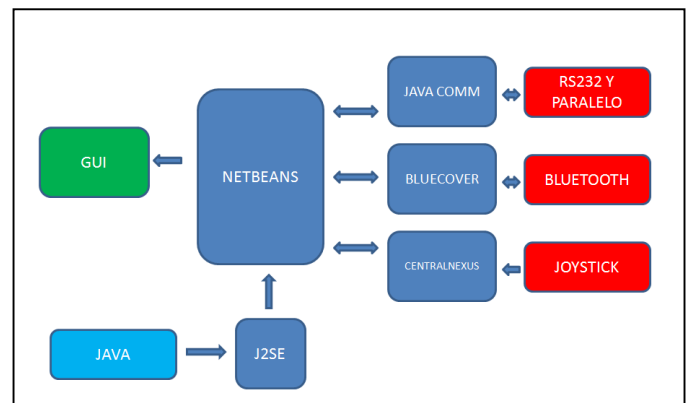


Figura 8. Diagrama general modular de la aplicación.

Como panel principal, visible en la Figura 9. , se ha desarrollado una pantalla en que se permite al usuario calibrar, tanto en velocidad lineal como en velocidad angular, el movimiento del robot. Con ello se corrigen errores en el movimiento inicial, aplicando dicha corrección a lo largo de todo el desplazamiento o conexión de la aplicación, pudiéndose corregir o modificar esos valores de calibración si así lo desea el usuario.

También permite seleccionar entre comunicación vía RS232, usada habitualmente para pruebas de laboratorio y verificación de la aplicación, y comunicación vía Bluetooth, usada como base principal de comunicación con los diferentes robots con fines educativos.

Dispone también de botones para seleccionar entre los diferentes tipos de robot a telecontrolar, robot siguelíneas, robot laberinto y robot de posicionamiento, lanzando la aplicación pertinente al robot seleccionado. Además, dispone de botones para poner en marcha la aplicación y cierre de la misma.

1) Telemonitorización de Robot Rastreador

Al trabajar con un robot siguelíneas, se lanza el panel representado por la Figura 10. En este panel se encuentra como punto central la representación de los 10 sensores de la tarjeta “AlcaCny”, para los cuales se ha diseñado un sistema de “recuerdo” para que el usuario pueda observar, sin necesidad

de tener en visión directa al robot, el desplazamiento que desarrolla sobre el terreno, aplicando una línea en forma de circuito.



Figura 9. Panel principal de inicio de aplicación.

También dispone de una representación del volante, barra de velocidad positiva/negativa y pedal de freno/aceleración, con el fin de que el usuario y los espectadores puedan observar las acciones que se ejercen desde los periféricos hacia el software y éste sobre el robot.

Por otro lado se le ha dotado de iconos para la medición del nivel de batería, para la monitorización de una posible pérdida de comunicación por pérdida de carga, así como botones de puesta en marcha, parada y salida de la aplicación.

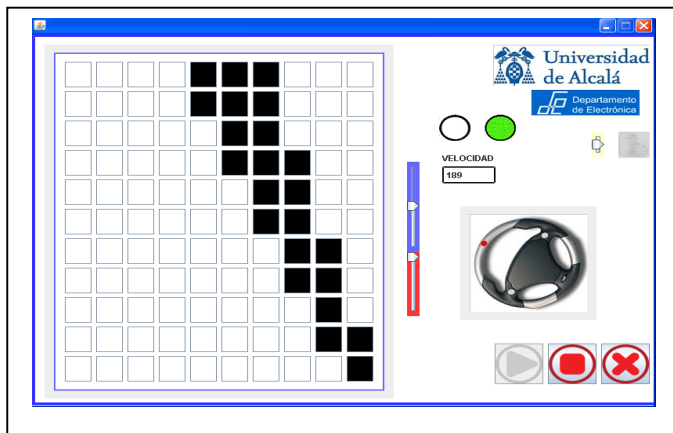


Figura 10. Panel robot siguelineas.

2) *Telemonitorización de Robot de Laberinto*

Para el robot laberinto, se debe incorporar al robot “Cyborg” la tarjeta con 8 sensores de ultrasonidos y la aplicación de monitorización permite visualizar gráficamente las medidas de cada uno de ellos como se puede observar en la Figura 11. El operador deberá intentar conducir el robot por un laberinto con la información visualizada en la pantalla.

El panel se ha desarrollado con el fin de poder usar un número de sensores entre 1 y 8, siendo configurable a través de

la selección de los 8 radio-button representados bajo la etiqueta “sensores”.

El resto de los controles y señales son iguales a los del panel del robot siguelineas. Pudiéndose observar, la representación del volante, barra de velocidad positiva/negativa y pedal de freno/aceleración, como iconos para la medición de la batería.

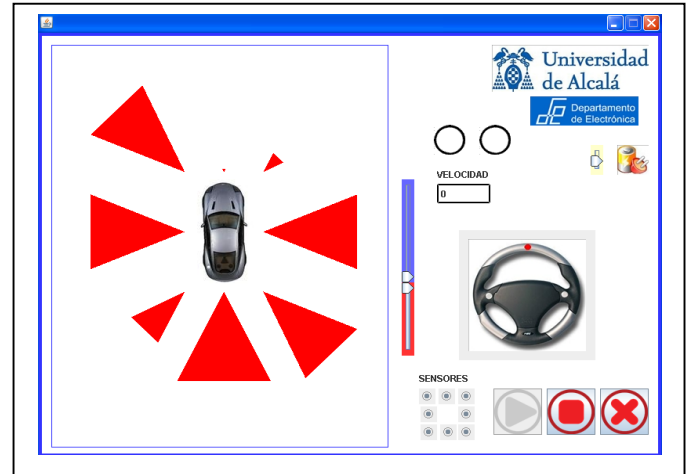


Figura 11. Panel robot laberinto.

3) *Telemonitorización de Robot de Posicionamiento*

Para el robot de posicionamiento, primeramente hay que cargar la imagen que va a hacer las funciones del mapa a recorrer o circuito deseado con ayuda de un mensaje Pop-Up donde se inserta el nombre del mapa a cargar en formato imagen, como representa la Figura 12. Se debe representar este mismo mapa de forma real y proporcional para poder estudiar las posibles derivas de posicionamiento y comportamiento tanto del software como del hardware.

Una vez cargado el mapa y lanzada la aplicación, el operador debe intentar que el robot siga un camino representado en la pantalla del ordenador de telecontrol y que será el mismo que el representado en el suelo del área de presentación. Debido a los errores de odometría en poco tiempo se observan diferencias entre la información monitorizada en la pantalla (donde el robot cree que está) y la posición real del robot (Figura 13.). Mostrar este fenómeno es el objetivo de esta aplicación.

La aplicación dispone de un panel informativo en color verde claro en el cual se le indica la posición en el eje y, eje x y del ángulo de giro que toma el robot en tiempo real con el fin de poder explicar las posibles derivas y funcionamientos del robot “Laberinto”.

El resto de los controles y señales son idénticos a los anteriormente explicados en los anteriores robots.

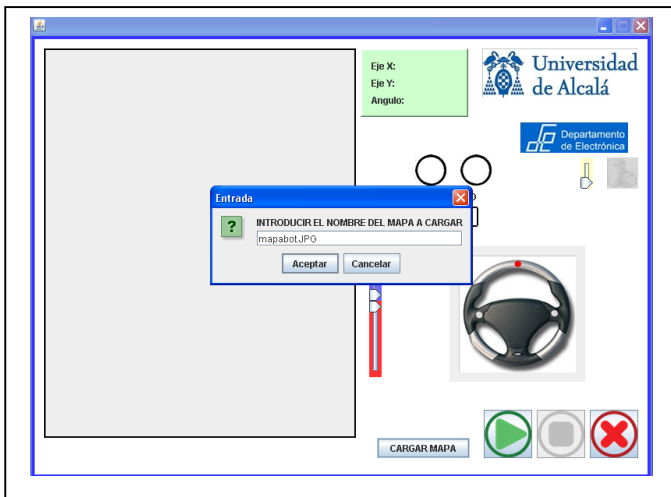


Figura 12. Panel robot posicionamiento. Carga de mapa.

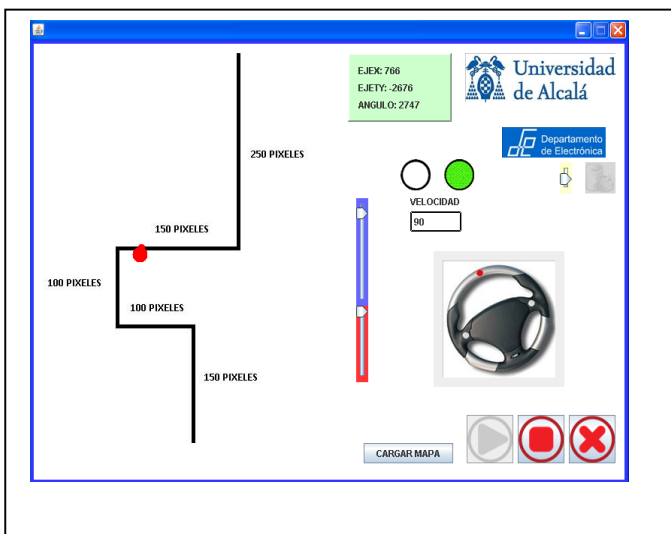


Figura 13. Panel robot posicionamiento. Control de posición.

La aplicación de posicionamiento se realiza con el robot Pioneer debido a que éste tiene un buen sistema de encoders que ayuda a demostrar cómo un robot puede conocer su posición durante un determinado periodo de tiempo y cómo le influyen factores como el deslizamiento de las ruedas en el suelo o el desplazamiento lateral del robot. Con el robot Cyborg se observa que el robot se pierde mucho antes debido a la baja calidad de sus encoders.

VI. UTILIDAD COMO HERRAMIENTA DE APRENDIZAJE

Claramente el sistema Cyborg sirve para que, personas de todas las edades, comprendan lo que es un sensor y tenga idea de lo que percibe un robot en los diferentes escenarios que se han desarrollado en estas aplicaciones software.

Estos mismos escenarios son fácilmente reproducibles en talleres o laboratorios de iniciación a la robótica como apoyo en la explicación de conceptos básicos. En esta línea, en el Departamento de Electrónica de la Universidad de Alcalá tiene

dos asignaturas donde se muestra y que son Introducción al Diseño de Microrrobots Móviles y Robótica para Todos.

También se puede utilizar en asignaturas de especialización en robótica para mostrar fundamentalmente el comportamiento de la odometría en el posicionamiento de robots móviles.

El uso del lenguaje de programación JAVA posibilita el uso en diferentes plataformas y la labor de adaptación de la aplicación a esas plataformas puede ser objeto de prácticas de asignaturas de sistemas empujados.

Por último, al estar muy definido el protocolo de comunicación entre el ordenador y el robot, da pie a ser utilizado en asignaturas de diseño electrónico de sistemas empujados animando a los alumnos a realizar sus propios robots móviles que cumplan con mismo interfaz de comunicación y por tanto, que puedan ser telecontrolados desde la aplicación.

VII. CONCLUSIÓN Y TRABAJOS FUTUROS

En los cursos, talleres y exhibiciones en los que se ha utilizado este material ha demostrado atraer el interés de todos los públicos con lo que se cumple uno de los objetivos de su desarrollo. La presentación sencilla de conceptos básicos de robótica también ha demostrado ser muy útil en asignaturas de robótica.

El paso de la programación gráfica de LabView a JAVA ha sido un gran avance al darle mucha más flexibilidad para adaptarse a diferentes plataformas y presenta una plataforma abierta para desarrollar futuros desarrollos.

En la plataforma Cyborg II se han introducido las funcionalidades de laberinto y posicionamiento donde se puede estudiar el comportamiento de los sensores de ultrasonidos y las ventajas e inconvenientes de su uso así como todos los conceptos relacionados con el posicionamiento por odometría.

En el futuro se trabajará en un mejor integración con dispositivos móviles, en el desarrollo de una nueva versión de robots móviles de bajo coste y en sistemas demostradores multirrobots.

AGRADECIMIENTOS

El proyecto ha sido posible por el apoyo del Departamento de Electrónica de la Universidad de Alcalá y por la cofinanciación de la Fundación Española para la Ciencia y la Tecnología (FECYT) – Ministerio de Ciencia e Innovación dentro de los programas de Ayudas para el Programa de Cultura Científica y de la Innovación 2009 y 2011.

REFERENCIAS

- [1] E. Samper Domeque, J. Pastor Mendoza, "‘‘Sientete un Cyborg’’: Sistema demostrativo del funcionamiento de un robot’’. Resúmenes de trabajos del IX Congreso TAAE 2010 (ISBN: 978-84-96737-69-3). Abril 2010. Madrid.
- [2] E. Samper Domeque, Proyecto Fin de Carrera "Sistema de monitorización y telecontrol de robots móviles con fines educativos". Universidad de Alcalá (2009).
- [3] Pioneer Robot. Mobile Robots Company. <http://www.mobilerobots.com/ResearchRobots/P3AT.aspx>
- [4] MobileRobots, "Pioneer 3 Operations Manual," (2007).

