

Plataforma robótica de bajo coste basada en la arquitectura software Player/Stage y en el hardware de La Fonera

Guillermo Asín, Julio Pastor

Departamento de Electrónica, Universidad de Alcalá

Alcalá de Henares, España

guillermo.asin.prieto@gmail.com, pastor@depeca.uah.es

Resumen— Se presenta en este trabajo una propuesta de bajo coste para el desarrollo de un laboratorio remoto, ofreciendo la posibilidad de proporcionar aprendizaje en el campo de la robótica a usuarios sin necesidad de permanecer en el laboratorio físicamente, o simplemente de tener que coger la plataforma robótica cada vez para su programación, así como se ofrece la posibilidad de que una misma plataforma sea utilizada por un grupo de usuarios por turnos para la prueba de sus algoritmos. El sistema está basado en un router Wi-Fi al que se le ha actualizado el firmware introduciendo el sistema operativo Linux y al que se ha conectado una tarjeta de control que gestiona el hardware de un robot móvil. En el procesador se ha introducido, además del nuevo sistema operativo, el entorno Player Server muy utilizado en plataformas de robótica comercial en investigación.

Keywords- robot, laboratorio remoto, bajo coste, La Fonera, Player/Stage

I. INTRODUCCIÓN

En la Universidad de Alcalá se trabaja desde hace varios años en investigación con los robots de Pioneer [1] de MobileRobots [2] en conjunción con el software de desarrollo y simulación Player/Stage [3][4][7], que además de presentar una interfaz estándar de comunicación con el robot, incorpora un simulador (Stage). Este software se ejecuta en un PC adosado al robot con sistema operativo Linux. Pioneer pertenece a una familia de robots móviles, orientados a ser plataformas robóticas para investigación con un coste relativamente elevado no estando al alcance de todas las personas que quieran utilizarlos.

El gran uso de esta plataforma robótica por la comunidad científica la ha convertido en una plataforma que se utiliza para comparar prestaciones de algoritmos de posicionamiento y control y existen diferentes aplicaciones de software abierto o comerciales que han desarrollado drivers para hacerlos compatibles. Un ejemplo de esto es la plataforma Player/Stage.

Player es un servidor en red, que permite controlar los dispositivos de un robot y obtener información de sus sensores. Funciona como una capa de abstracción del hardware, es decir, permite operar con multitud de robots (incluso otras máquinas, si están soportadas en su extensa biblioteca de drivers) sin

preocuparse del hardware específico de los mismos. La arquitectura utilizada está basada en sockets TCP/IP, de tal manera que por una parte se tiene el propio servidor de Player corriendo en la máquina conectada al robot, y por otra se tiene el cliente que se comunica con el mismo para la obtención y/o envío de datos de y hacia el robot, para su operación. Para comunicarse como cliente con el servidor Player, pueden usarse las aplicaciones que a tal efecto es posible instalar en el momento de instalación de las bibliotecas y del servidor; así como las propias bibliotecas en diversos lenguajes: C, C++, Java, Python, etc.

Stage es un simulador de robots móviles con visualización en 2'5D (2'5 dimensiones), es decir, en un modo de presentación que podría denominarse pseudo-3D: es un 2D, pero con altura, dándole una tercera dimensión ficticia, en la que no hay formas, que se usa en conjunción con Player. Permite la simulación de los programas clientes realizados por el usuario, antes de ser probados en hardware real, lo cual facilita y agiliza el trabajo. Es tremendamente personalizable, permitiendo realizar modelos del robot que va a utilizarse, si es que no existen ya, incluyendo no sólo modelos de los sensores, sino distancias reales a escala, obstáculos móviles, fijos...

Para utilizar la plataforma Player/Stage con los robots Pioneer, se suele añadir al robot un PC empotrado ejecutando el Player sobre Linux conectado al robot mediante un cable serie intercambiando información utilizando el protocolo *Advanced Robotics Control and Operations* (ARCOS) [1] establecido por MobileRobots en sus sistemas.

II. DESCRIPCIÓN GENERAL DEL SISTEMA

En el proyecto que se presenta en este artículo se ha desarrollado una plataforma robótica de bajo coste con el robot Cyborg/DepecaBot del Departamento de Electrónica de la Universidad de Alcalá que implementa el protocolo ARCOS como elemento de comunicación con el exterior. Como plataforma de comunicación y control de alto nivel se ha utilizado el entorno Player/Stage implementado, en vez de sobre un PC empotrado de altas prestaciones, sobre un punto de acceso Wi-Fi "La Fonera" [8] [9] con OpenWRT (distribución de Linux para sistemas empotrados, disponible para muchas plataformas, routers).

Además de utilizar Player para leer información de los sensores reflexivos del robot, se ha desarrollado un modelo para prueba y simulación de algoritmos del sistema en Stage.

Todo ello se completa con el desarrollo de una interfaz web que elimina la dependencia del sistema operativo del usuario, al ofrecer un entorno para la compilación y carga de programas en la plataforma robótica de manera remota. Esta interfaz permite la monitorización de los sensores reflexivos en un ejemplo de programa de usuario, así como el control de manera remota por medio de cursores en pantalla del movimiento del robot, todo ello monitorizado desde una cámara que presenta en tiempo real el movimiento de la plataforma.

La plataforma desarrollada se puede utilizar, además de para asignaturas relacionadas con la robótica, como un ejemplo de integración de sistemas operativos y aplicaciones de alto nivel en hardware específico como en este caso es un router o punto de acceso Wi-Fi, y su interconexión con aplicaciones de control mediante microcontroladores de bajo coste.

El objetivo de este trabajo es por tanto el de ofrecer la posibilidad de tener un entorno de desarrollo robótico independiente de la máquina y sistema operativo utilizados por el usuario, y favoreciendo una deslocalización, para su uso como laboratorio remoto.

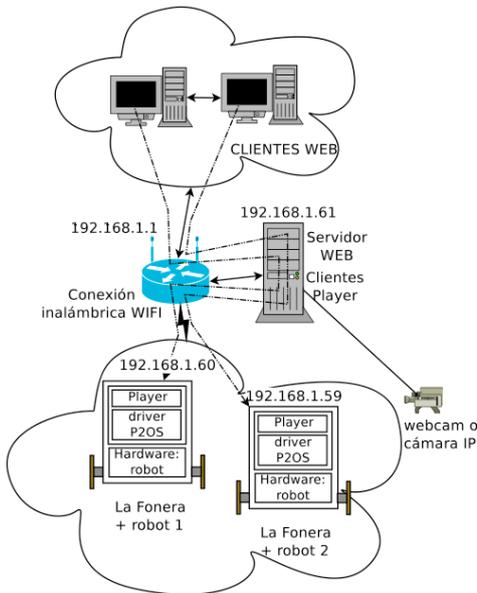


Figura 1. Diagrama general del sistema

III. ARQUITECTURA HARDWARE

A. Diagrama de bloques

En la Figura 1 se muestra un esquema general del sistema con varios robots, cada uno con un sistema Linux sobre el que corre el Player Server, un router que gestiona las comunicaciones Wi-Fi, los ordenadores remotos que ejecutan clientes HTML donde permite controlar parámetros del robot y cargar nuevos programas y un ordenador principal que ejecuta un servidor web que sirve a los clientes remotos y que presenta

las aplicaciones necesarias para recibir de los clientes el código a ejecutar y lo carga en los robots. El sistema permite tanto la posibilidad de la simulación como de la prueba sobre hardware real. El objetivo como se ha comentado es el de realizar un laboratorio remoto, en el que pueda practicarse la programación de robots con lenguaje de alto nivel a través de una interfaz que permita abstraerse del sistema operativo.

B. Plataforma Cyborg/DepecaBot

La plataforma Cyborg/DepecaBot (Figura 2) 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, con 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.

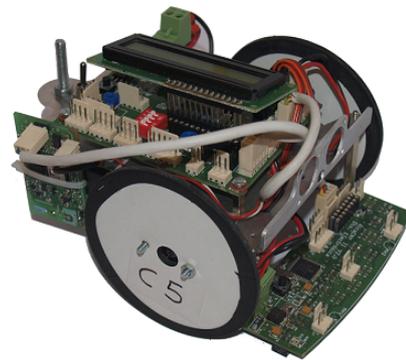


Figura 2. Robot Cyborg

En cuanto a la electrónica, el robot está formado por dos tarjetas diseñadas en el Departamento de Electrónica de la Universidad de Alcalá, visibles en la Figura 3.

- 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. [5][6]
- La tarjeta AlcaCnyII que posee diez sensores reflexivos y que envía la información captada por estos mediante I2C a la tarjeta principal.

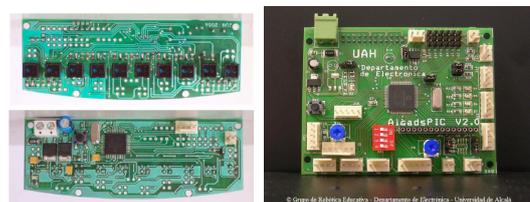


Figura 3. Tarjetas electrónicas. De izquierda a derecha: AlcaCny II y AlcaDspic.

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.

C. Punto de acceso Wi-Fi “La Fonera”

La Fonera tiene su origen en la empresa FON con la idea de compartir Wi-Fi y hacer una red social. FON es una iniciativa empresarial de Martín Varsavsky surgida en el año 2005 con el objetivo de fundar una comunidad Wi-Fi global, que permite a sus usuarios la conexión gratuita a los puntos de acceso de otros usuarios, repartidos por todo el mundo, a la vez que brinda acceso de pago a terceros mediante un sistema en el que las ganancias se reparten entre la compañía y el usuario que presta su conexión [9]. Para ello, comenzó proveyendo routers Linksys WRT54G/GL y Buffalo AirStation G54 con un firmware basado en OpenWRT modificado; en Octubre de 2006 empezó con la distribución de La Fonera, un “Access Point” con su hardware basado en los routers que comenzó distribuyendo, y por tanto compatible con el firmware.

Si bien hay varias revisiones y actualizaciones de La Fonera, a continuación se comentan las versiones 2100/2200, que han sido las utilizadas en este trabajo. Cabe mencionar que siendo la versión más básica de La Fonera, es probable que pueda ser aplicable sin cambios importantes a versiones más avanzadas (La Fonera +, La Fonera 2.0, La Fonera 2.0n).

La Fonera incorpora un firmware basado en Linux, concretamente, una modificación personalizada para FON de OpenWrt [10]. Existen dos distribuciones “libres” (no personalizadas) que son OpenWrt y DD-WRT [11]; en este proyecto se ha optado por usar OpenWrt por ser la más extendida y por tanto haber más documentación en la red. También incorpora un sistema muy sencillo para la compilación de firmwares personalizados. La versión elegida es la “Kamikaze”, concretamente la versión 8.09.2 por ser la última versión existente en el momento en que se comenzó el trabajo.

El Linux de La Fonera es un Linux normal, pero sin entorno gráfico. Por ello, para poder acceder a él, se ha de hacer a través de un terminal de comandos, y para hacerlo, pueden utilizarse dos canales: vía puerto serie, vía red. El acceso a través del puerto serie requiere únicamente un programa de terminal (como por ejemplo “putty”), y una conexión a la UART de La Fonera, con la única limitación a tener en cuenta de que la conexión deberá tener la correspondiente conversión de niveles de tensión si fuese necesario (la UART de La Fonera utiliza niveles TTL). Por su parte, el acceso a través de red, bien puede realizarse por Ethernet (con cable de red cruzado), bien por Wi-Fi.

El espacio disponible en La Fonera es de 8 Megabytes, aunque el firmware original de la misma sólo permite el uso de 4 MB. La razón probablemente sea que debido a que hay algunas de la versión 2100 que sólo disponen de 4 MB, es más cómodo tratarlas a todas como la versión más limitada. Es por ello que se parte de la premisa de que sólo hay disponibles 4 MB de ROM para dar soporte a Las Foneras con sólo 4 MB y no 8MB.

D. Arquitectura Hardware del robot

La plataforma completa está formada por la base del robot Cyborg/DepecaBot unido a La Fonera por una comunicación serie. En la Figura 4 se muestra la apariencia final del robot, en la Figura 5 el diagrama de bloques del sistema completo y en la Figura 6 el detalle de la unión entre La Fonera y la tarjeta de control del robot.



Figura 4. Plataforma completa

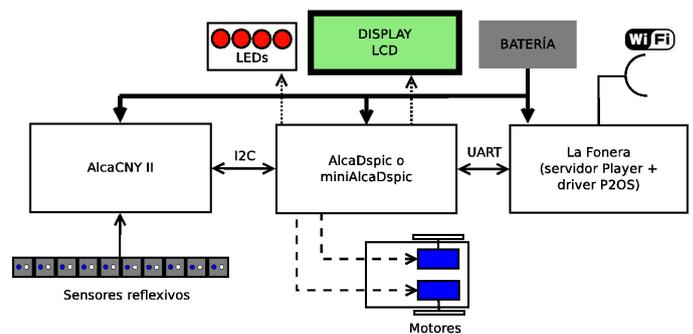


Figura 5. Diagrama de bloques de la Arquitectura Hardware

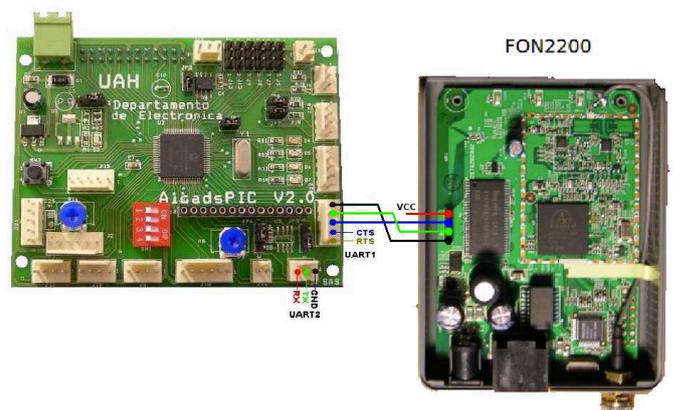


Figura 6. Detalle de la interconexión AlcaDspic –La Fonera

IV. SIMULACIÓN CON STAGE

En el proyecto, además de instalar e implementar el Player en la plataforma robótica Cyborg/DepecaBot/La Fonera, se implementó una simulación de la misma utilizando el simulador Stage simulando un robot rastreador. Para ello se implementó un modelo de robot "dpkbot", así como las definiciones de "suelo" y "reborde". En el archivo .world se definen características tanto para los elementos de la simulación, como para la propia ventana que alberga la simulación.

Cada uno de los diez sensores reflexivos del robot siguelíneas se simula con el modelo de Stage de un sensor de distancia y el suelo se simula por una superficie con relieve donde todo aquello que no es blanco se le da una altura mínima. Midiendo la información de los sensores de distancia se detecta si están encima de una línea negra o no en función de la distancia que miden.

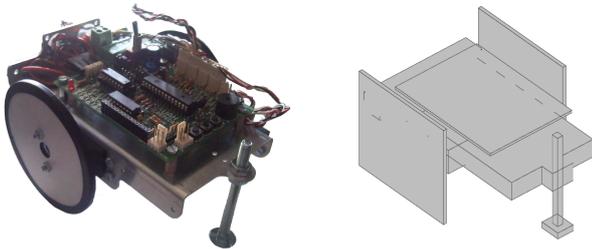


Figura 7. Robot real y modelo en el Stage

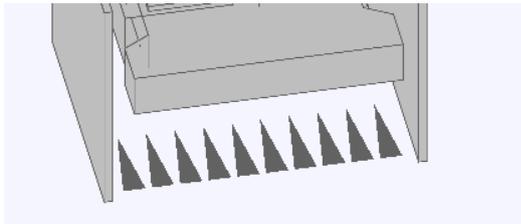


Figura 8. Detalle del modelo de Stage de los sensores del robot

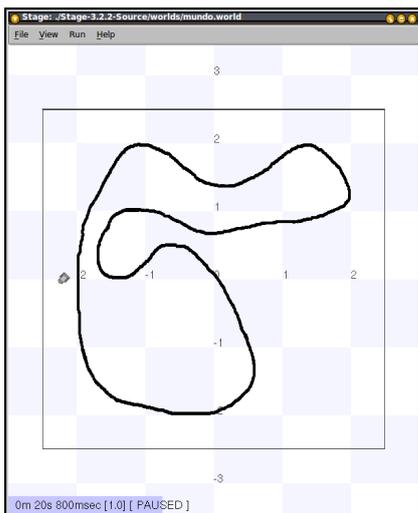


Figura 9. Entorno de simulación

En la Figura 7 se muestra el robot real y su modelo en Stage. En la Figura 8 se aprecia el detalle de la implementación de los sensores en el modelo y en la Figura 9 se presenta un ejemplo de pista por donde el robot se va a mover y que se obtiene de una imagen en blanco y negro.

V. ARQUITECTURA DEL SISTEMA COMPLETO

El sistema se compone, como puede observarse en el diagrama de bloques de la Figura 10 de las siguientes partes:

- Clientes web, desde los que los usuarios, a través del navegador de Internet, suben y compilan sus programas clientes.
- Router con conexión inalámbrica.
- PC con sistema operativo Linux, que hace las veces de servidor web, y de cliente Player.
- Webcam o cámara IP para el visionado de la plataforma robótica en tiempo real.
- Plataformas robóticas en sí, compuestas por la conjunción de La Fonera y Cyborg conectados por puerto serie.

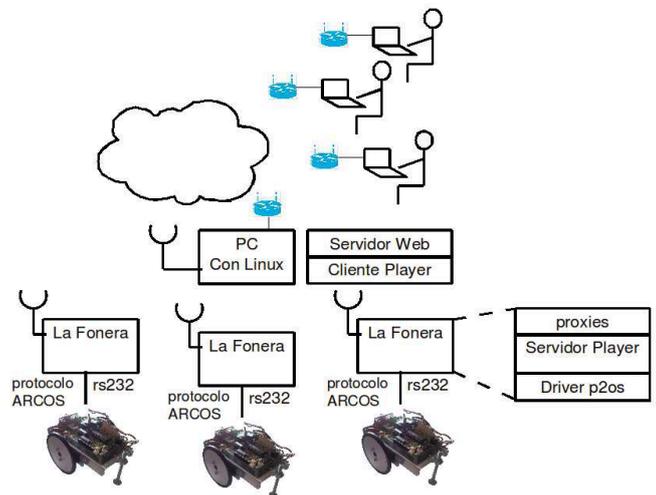


Figura 10. Arquitectura del sistema completo

La Fonera incorpora una distribución de Linux orientada a dispositivos empotrados, OpenWRT y sobre este Linux, corre el servidor de Player, utilizando el driver p2os que implementa el protocolo ARCOS sobre RS232 que utiliza para intercambiar información con el hardware del robot (plataforma Cyborg/DepecaBot).

VI. EJEMPLO DE APLICACIÓN

A continuación se presentan un ejemplo de aplicación del sistema consistente en la realización del código de un robot siguelíneas que es primeramente simulado con el Stage y posteriormente ejecutado en el robot real. Debido a que el robot Cyborg/DepecaBot con los diez sensores de infrarrojos, emula el comportamiento de un robot Pioneer con sus sensores de ultrasonidos, el mismo código se puede implementar, con pequeñas modificaciones, para que un robot Pioneer siga a un objeto que tiene delante.

A. Estructura del programa del robot siguelíneas

En la Figura 11 se muestra el diagrama de bloques de una posible implementación del programa del siguelíneas como cliente del Player Server que se ejecuta en La Fonera. Los bloques de inicio y final son comunes a todas las aplicaciones siendo necesario cambiar sólo el bucle principal.

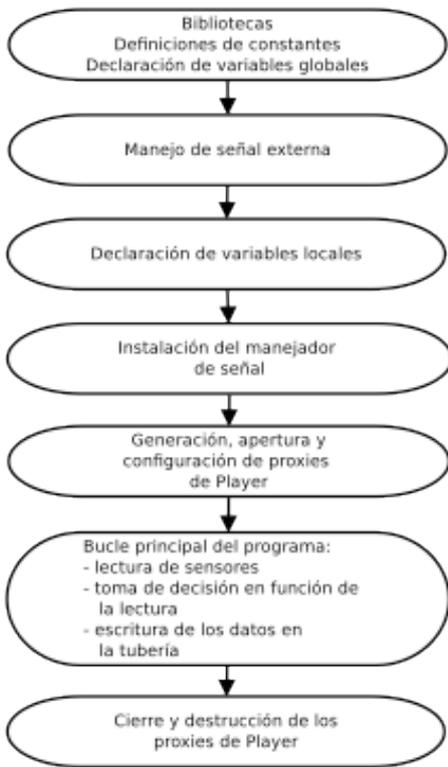


Figura 11. Esquema del programa de un siguelíneas con Player

B. Entorno de carga y monitorización remota

El acceso remoto a los robots se realiza mediante una página web en la que en primer lugar se elige el robot con el que se quiere conectar y posteriormente solicita al usuario que suba y compile el código del programa a ejecutar en el robot (Figura 12).

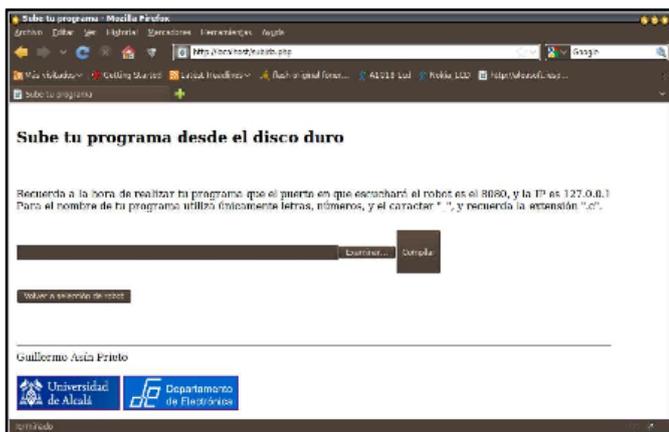


Figura 12. Pantalla de subida del código

El programa es compilado en el ordenador remoto e informa al usuario de los posibles fallos que haya podido haber en la compilación (Figura 13).

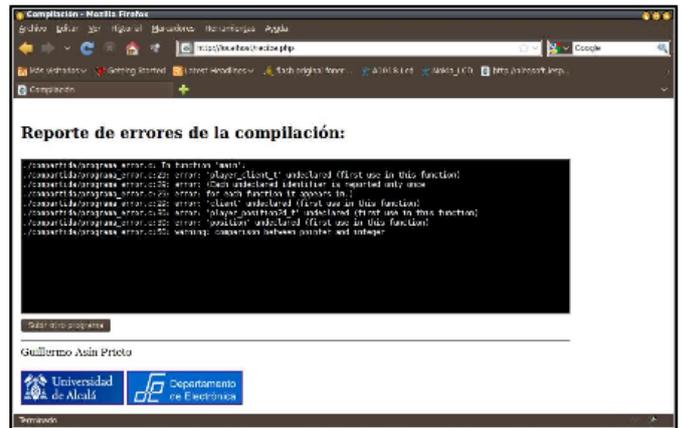


Figura 13. Pantalla que muestra los resultados de compilación

El funcionamiento del programa puede verse utilizando una ventana de simulación basada en Stage o se puede cargar y ejecutar en el robot y monitorizar su funcionamiento utilizando una cámara IP o webcam (Figura 14).



Figura 14. Monitorización en tiempo real del funcionamiento del robot

VII. TRABAJOS FUTUROS

A continuación se exponen algunas ideas y propuestas de trabajos futuros para complementar o ampliar éste:

A. Implementación de otros sensores en la interfaz web

Pueden incluirse en la interfaz la monitorización de más sensores del mismo tipo, u otros de los que se provea al robot como pueden ser encoders, sensores de ultrasonidos, ...

B. Sistemas completamente autónomos

Implementación del servidor web dentro de la propia La Fonera, ampliando sus capacidades de almacenamiento por medio de una modificación para el uso de tarjetas SD.

C. Desarrollo de un driver específico

Se ha modificado el driver p2os existente en el repositorio de Player, pero puede implementarse un driver específico que tenga en cuenta las peculiaridades del hardware de Cyborg.

D. Utilización de hardware alternativo

El desarrollo ha sido pensado utilizando un sistema operativo como es Linux disponible para muchas plataformas hardware, y Player, que abstrae del hardware subyacente si está soportado. Se podría utilizar por tanto otros router o sistemas empujados que soporten Linux, y otros plataformas robóticas soportadas por Player.

VIII. CONCLUSIONES

Con el proyecto que se presenta se ha conseguido un sistema completo y funcional para la realización de un laboratorio remoto para el aprendizaje de la robótica a nivel de algorítmica, que además permite la independencia del hardware y software desde el que se trabaje siempre que se disponga de un dispositivo con acceso a internet, y navegador web.

Esta implementación por tanto facilita el aprendizaje de la robótica sin necesidad de laboratorios presenciales.

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.

REFERENCIAS

- [1] MobileRobots Inc., Pioneer 3 Operations Manual, Pioneer 3 Operations Manual v.3 (ARCOS-based DX and AT).pdf, 2006.
- [2] RAIN. (2010) “Mobile Robots - Autonomous mobile robot cores, bases and accessories”. Comprobada el 7 de Noviembre de 2010. [Online]. Disponible: www.mobilerobots.com/
- [3] (2009, Diciembre) “The Origins of the Player/Stage Project”. Comprobada el 3 de Septiembre de 2010. [Online]. Disponible: playerstage.sourceforge.net/wiki/PlayerHistory
- [4] (2010, Junio) “Player Project”. Comprobada el 3 de Septiembre de 2010. [Online]. Disponible: [http://en.wikipedia.org/wiki/Player Project](http://en.wikipedia.org/wiki/Player_Project)
- [5] “Tarjeta miniAlcadsPIC”. Comprobada el 3 de Septiembre de 2010. [Online]. Disponible: [geiser.depeca.uah.es/moodle robot/course/view.php?id=22](http://geiser.depeca.uah.es/moodle/robot/course/view.php?id=22)
- [6] “Tarjeta AlcadsPIC”. Comprobada el 3 de Septiembre de 2010. [Online]. Disponible: [geiser.depeca.uah.es/moodle robot/course/view.php?id=4](http://geiser.depeca.uah.es/moodle/robot/course/view.php?id=4)
- [7] O. Colomina y M. Ángel Cazorla. (2007, Noviembre) “Introducción a Player/Stage”. Comprobada el 3 de Septiembre de 2010. [Online]. Disponible: rua.ua.es/dspace/bitstream/10045/11427/1/intro-ps.pdf
- [8] (2010, Agosto) “La Fonera”. Comprobada el 3 de Septiembre de 2010. [Online]. Disponible: es.wikipedia.org/wiki/Fonera
- [9] (2010, Julio) “FON”. Comprobada el 3 de Septiembre de 2010. [Online]. Disponible: es.wikipedia.org/wiki/FON
- [10] “OpenWrt - Wireless Freedom”. Comprobada el 3 de Septiembre de 2010. [Online]. Disponible: www.openwrt.org/
- [11] “dd-wrt.com”. Comprobada el 3 de Septiembre de 2010. [Online]. Disponible: www.dd-wrt.com/