

Sistema para el control remoto de un robot mediante un navegador de Internet (II)

Se presenta a continuación la segunda parte del artículo "Sistema para el control remoto de un robot mediante un navegador de Internet". En el número anterior se presentó la descripción general del robot y el sistema de comunicaciones que permite controlar el robot desde cualquier navegador de internet utilizando como pasarela el servidor central del Instituto de Investigación Tecnológica. En este número se muestra la interfaz gráfica desarrollada en Java que está accesible en modo de demostración en <http://www.iit.upco.es/palacios/robot>



Rafael Palacios Hielscher

Ingeniero Industrial del I.C.A.I. Es profesor del departamento de Electrónica y Automática e Investigador del área de sistemas inteligentes del IIT, en la Escuela Técnica Superior de Ingeniería (I.C.A.I.).

Doctor Ingeniero Industrial por la Universidad Pontificia Comillas en 1998. Áreas de trabajo: sistemas de monitorización y diagnóstico, análisis de señal y tratamiento digital de imágenes.

Álvaro Sánchez Miralles

Ingeniero del I.C.A.I. especialidad electrónica. Pertenece al área de Sistemas Inteligentes del Instituto de Investigación Tecnológica de la UPCO. Áreas de trabajo: Robótica móvil, Comunicaciones y Control wireless, Sistemas de Diagnóstico y Modelado de sistemas no lineales.



La interfaz gráfica

La característica fundamental de una interfaz gráfica y, aún más, si está abierta a Internet debe ser la facilidad de su uso. No es necesario que haya un manual de usuario porque el objetivo de la WWW es ofrecer un ambiente agradable donde se puede navegar sin conocimientos previos. Se ha tratado de diseñar una interfaz muy sencilla de controlar con botones y barra deslizadora para que el usuario pueda utilizarla desde la primera conexión a la página *web* de control. La ventaja de programar en Java es que da la posibilidad de utilizar componentes y clases ya programados. Así se puede diseñar *applets* más rápidamente y desarrollar entornos estandarizados.

El robot cuenta con varios circuitos de actuación para poder moverse, encender luces y emitir sonidos. El sistema de tracción está formado por dos motores independientes que pueden moverse hacia adelante o hacia atrás. Las luces son de dos tipos, una luz de potencia variable y un conjunto de 8 LEDs que se pueden encender de manera independiente. Los sonidos se generan con un pequeño altavoz controlado por un circuito que permite obtener distintas tonalidades.

La página *web* permite al usuario controlar directamente el robot, ver Figura 6. El bloque superior del *applet* sirve para controlar la tarjeta de ocho LEDs. Hay ocho botones para

acceder a los ocho LEDs de forma independiente.

Para controlar la tarjeta del generador de sonido existen unos botones que permiten tocar siete sonidos diferentes y un botón de apagado del generador de sonido.

“Se puede acceder a la página web de control del robot (en modo demostración) <http://www.iit.upco.es/~palacios/robot>”

Para demostrar las posibilidades de este tipo de acción remota se han añadido dos series de animaciones de LEDs y de sonidos. Si se hace un clic sobre un botón de animación, se produce una pequeña secuencia de LEDs o de sonidos, por ejemplo un bucle led1 led3 led5 led7 led2 led4 led6 led8... o una pequeña melodía. Es decir, cada animación es una se-

cuencia de actuaciones que se genera automáticamente. Estas animaciones sirven para comprobar la velocidad de comunicación y de ejecución de actuaciones del robot.

La luz a potencia variable es controlada por medio de

una barra deslizadora que permite producir cinco niveles de luminosidad de la bombilla. También hay un botón para apagar directamente la bombilla.

El movimiento del robot se controla mediante nueve botones de dirección. Cada botón pone en marcha los motores de una manera determinada para

poder avanzar, girar o avanzar girando. Las acciones que permiten ejecutar estos se muestran en la tabla 1.

El último objetivo es la visualización de las medidas de los sensores. Como se ha visto antes, la técnica utilizada para la visualización de los valores es un componente Java que permite leer el contenido de un archivo del servidor. El contenido de este archivo se muestra en unas áreas de texto en la pantalla y se actualiza cada 5 segundos. En el ejemplo de la figura 6 se muestra la fecha y la hora del PC del robot como simulación de las medidas de los sensores.

Conclusiones

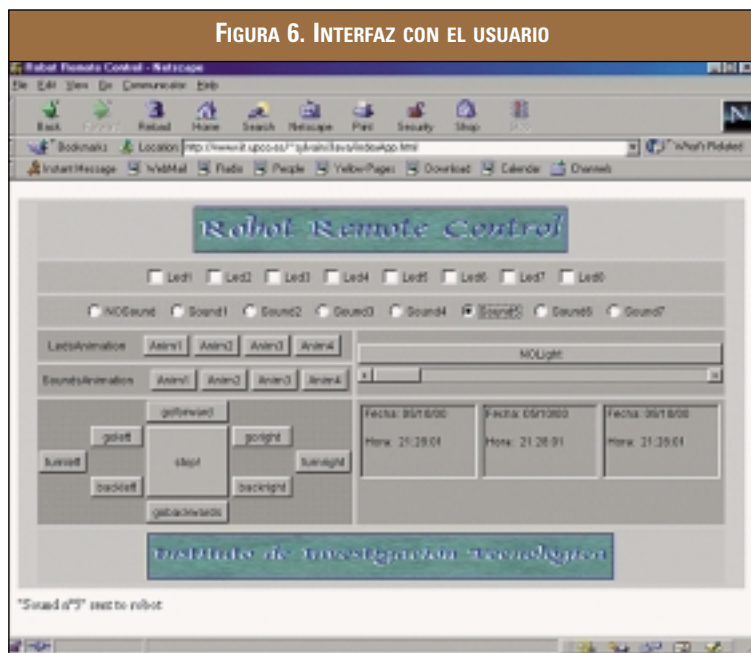
Se ha desarrollado un sistema de control de un robot orientado a Internet para que pueda ser utilizado desde cualquier lugar y desde cualquier ordenador. Para ello se ha diseñado una arquitectura basada en un servidor web como pasarela a Internet, que permite manejar volúmenes grandes de información, ya que el tráfico con el robot debe ser mínimo. Se ha programado una página web intuitiva, de manera que pueda ser utilizada por cualquier persona.

Al aprovechar las capacidades gráficas, de comunicaciones y de programación que ofrecen los navegadores de Internet actuales, la programa-

TABLA 1

Acción	Motor izquierdo	Motor derecho
Adelante	↑	↑
Adelante derecha	↑	X
Adelante izquierda	X	↑
Atrás	↓	↓
Atrás derecha	↓	X
Atrás izquierda	X	↓
Gira derecha	↑	↓
Gira izquierda	↓	↑
Para	X	X

FIGURA 6. INTERFAZ CON EL USUARIO



ción del sistema de control ha resultado muy sencilla. Además hay que destacar la gran flexibilidad de utilización, ya que se puede usar cualquier tipo de ordenador para controlar el robot. Por último comentar que resulta muy sencillo ampliar el sistema con nuevos sensores o actuadores ya que los cambios en los *scripts* son


mínimos y añadir nuevos botones a la página se puede hacer con cualquier editor HTML.

Como futuros desarrollos, un avance importante sería montar una cámara *web* sobre el robot. Como se ha comentado anteriormente, no es ningún problema enviar imágenes en tiempo real por la red. Mostrar

una imagen nueva cada dos segundos en el *applet* JAVA desarrollado es algo muy sencillo de programar. Sólo habría que tener en cuenta el paso por el servidor para no saturar la comunicación por radio.

Por otro lado, sería más cómodo para el usuario tener una respuesta visual de sus acciones. Lo que se podría llevar a cabo añadiendo dos *webcam*, una montada sobre el robot y otra para localizar el robot en su entorno próximo.

Agradecimientos

Los autores agradecen la colaboración de todos los alumnos que han participado en el desarrollo de los distintos módulos hardware y software del robot. Especialmente Sylvain Dubois durante el curso 1999-2000 por su colaboración en el apartado de comunicaciones. 

Bibliografía

- [1] Álvaro Sánchez Miralles, Rafael Palacios Hielscher. Diseño de un vehículo autoguiado controlado mediante un ordenador personal. UPCO-IIT 2000.
- [2] David Flanagan. JAVA in a nutshell. A Desktop Quick Reference. Ed. O'Reilly. 1997.
- [3] David Flanagan. JAVA Examples in a nutshell. A Tutorial Companion to Java in a Nutshell. Ed. O'Reilly. 1997.
- [4] Jean-Francois Macary, Cédric Nicola. Programación Java. Ed. Eyrolles. 1996
- [5] M. Felton CGI Internet Programming with C++ and C. Prentice Hall 1997
- [6] Shishir Gundavaran. CGI programming on the world wide web. O'Reilly 1996.
- [7] Jesús Bobadilla. HTML dinámico, ASP y JavaScript. Marcas comerciales RA-MA,1999
- [8] Jason Hunter, William Crawford. JAVA Servlet programming. O'Reilly 1998.
- [9] Arnold Robbins. Unix in a Nutshell: System V Edition. O'Reilly, 3rd Edition 1999.