

KIT DEPECABOT: PLATAFORMA EDUCATIVA DE HARDWARE Y SOFTWARE ABIERTO PARA LA INICIACIÓN A LA ROBÓTICA MÓVIL.

MARCELO SALAZAR ARCUCCI y JULIO PASTOR MENDOZA

Departamento de Electrónica.

Universidad de Alcalá. España.

{marcelo.salazar, pastor} @depeca.uah.es

Este documento describe una plataforma robótica desarrollada para iniciar a los alumnos de ingeniería y formación profesional en la robótica móvil, motivarlos en sus estudios, cubrir sus inquietudes personales y poner en práctica sus conocimientos adquiridos.

Palabras clave: robot, microcontrolador, hardware, software, taller, kit

1. Introducción

La utilización de la robótica móvil en ingeniería como elemento educativo ha demostrado ser una forma interesante y divertida de poner en práctica conocimientos que se estudian en diversas asignaturas a lo largo de la carrera; con este fin, en el Departamento de Electrónica de la Universidad de Alcalá se lleva promoviendo el diseño de robots por sus alumnos desde hace más de 10 años [1].

Con el fin de introducir a alumnos universitarios, no universitarios y aficionados en la robótica móvil, en los últimos años se han desarrollado diversas plataformas [3] que se han utilizado en varios talleres de iniciación. En este artículo se presenta la última plataforma desarrollada que se le ha dado como nombre “Depecabot” (figura 1) que fue desarrollado inicialmente en el Departamento de Electrónica [2] de la Universidad de Alcalá en el año 2008 por alumnos, ex-alumnos y profesores y que siguen actualmente en desarrollo nuevas bibliotecas de funciones de software abierto. El objetivo inicial era disponer de un robusto y económico sistema móvil programable, que dispusiera de lo mínimo necesario para poder realizar un taller de iniciación a la robótica móvil al alcance de los estudiantes.

El sistema se ha utilizado en varios talleres de iniciación a la robótica dentro de las actividades que se realizan todos años en la Semana de la Robótica de la Universidad de Alcalá [1] en el que se conoce como “Taller DepecaBot”, de veinte horas de duración, en el que se introduce a los alumnos en los principales componentes del robot (sensores, actuadores, sistema de control) y se les enseña las diferentes estrategias de programación de los mismos. En el taller se utiliza el Kit DepecaBot y se ofrece a los alumnos una importante documentación para el montaje y puesta en marcha del mismo.

En el artículo se presenta en primer lugar los objetivos del Kit continuando con un resumen de las diferentes plataformas educativas desarrolladas en años anteriores. A continuación se hace una descripción del Kit DepecaBot, descomponiéndolo en tres grandes partes: estructura mecánica, electrónica y software (bibliotecas de funciones junto con el entorno de programación). Seguidamente se describe toda la documentación facilitada al alumno para montar el robot, las tarjetas e iniciarse con la programación del mismo. Finalmente se explica en qué consiste el Taller DepecaBot y otras aplicaciones docentes en las que se ha utilizado y podría utilizarse.

2. Antecedentes

El “Proyecto Depecabot” surge como la evolución de varios proyectos educativos previos que comenzaron con un Kit de Robot Velocista desarrollado en 2002 para realizar un taller en la Campus Party [10] basado en la estructura de un coche teledirigido y utilizando en un procesador de la familia PIC18 de Microchip [4]. De este proyecto salió la tarjeta que se denominó AlcaPIC.

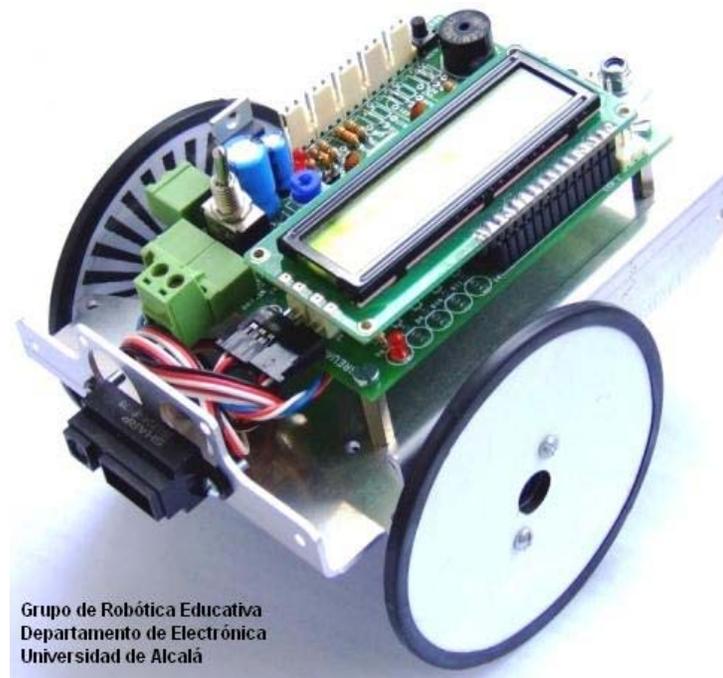


Figura 1. Robot DepecaBot

En 2003 se desarrolló con el mismo fin otro Kit de Robot velocista basado esta vez en varias tarjetas: una con el procesador basado en el MC51 y la etapa de potencia (AlcaI2C), otra con un array de sensores reflexivos (AlcaCNY) y una tercera con hardware de adquisición de señales analógicas y digitales que sirve para conectar diversos sensores.

En 2005 se desarrolló el “Kit Speed Runner” (ver figura 2) basado también en una plataforma que utiliza como estructura la de un coche de radio control con tres tarjetas: una con el procesador y su lógica asociada (AlcaDsPIC), una segunda de potencia con varios puentes en H (AlcaDRIVER) y una tercera que implementa un array inteligente de sensores reflexivos (AlcaCNYII). Estas tarjetas se utilizaron en dos proyectos: el KIT Speed Runner, que implementa un robot velocista, y el Kit Cyborg que forma un robot con una estructura de tracción diferencial.

El kit formado por las tres tarjetas (AlcaDsPIC, AlcaDRIVER y AlcaCNYII) tiene mucha potencia y muchas posibles aplicaciones al disponer de dos procesadores (un dsPIC de 16 bits para el control y un PIC en el sensor inteligente) y gran cantidad de puertos de entrada salida. Sin embargo tenía dos problemas: las tarjetas estaban diseñadas en montaje superficial lo que complica su montaje en pequeñas cantidades y el coste era elevado para ser utilizado como herramienta de iniciación a la robótica. Por este motivo se inicia el “Proyecto Depecabot” que se explica a más adelante.

3. Objetivos del “Proyecto Depecabot”

El objetivo principal del sistema que se presenta en el artículo, Kit Depecabot, era disponer de un kit de bajo coste que sirviera para que las personas interesadas pudieran iniciarse en la robótica móvil. Para cumplir este objetivo el kit tenía que ser simple, barato y fácil de usar. En la figura 3 se observa todo el material mecánico y electrónico que contiene el kit. El hardware, además de permitir la funcionalidad básica que se le suele pedir a un micro-robot, tenía que ir acompañado de potentes bibliotecas de funciones que hicieran fácil su utilización así como de multitud de ejemplos y una documentación explicativa.

Como el objetivo es difundir la afición por la robótica, toda la documentación esta disponible en Internet con acceso público bajo licencia de Creative Commons [11].

En este artículo se presenta la estructura, hardware y software del proyecto Depecabot. Información ampliada de estos proyectos puede encontrarse en la página Web [3].

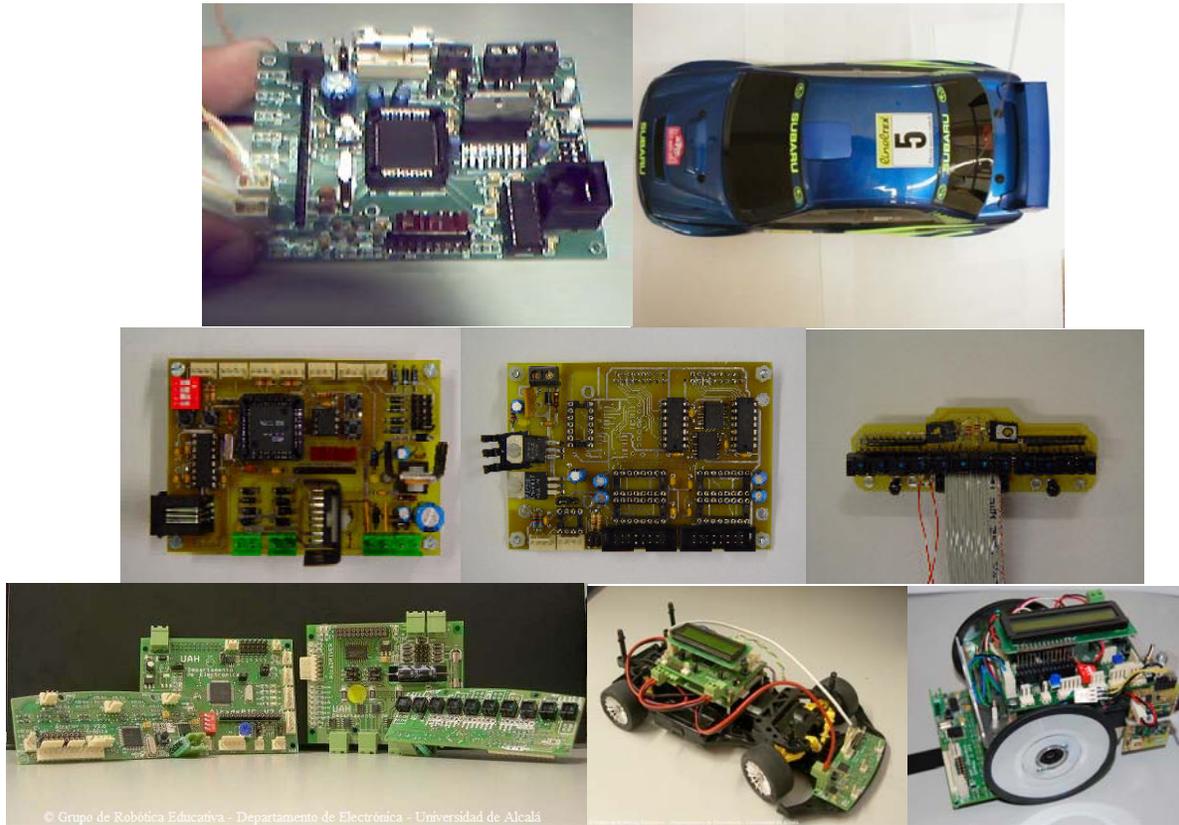


Figura 2. Arriba: Tarjeta AlcaPIC y estructura de coche de radio control: Medio: tarjetas AlcaI2C, Expansión I2C y AlcaCNY. Abajo: Tarjetas del proyecto Speed Runner (AlcaCNY, AlcaDsPIC y AlcaDRIVER) y robots Speed Runner y Cyborg que usan estas tarjetas.



Figura 3. Material mecánico y electrónico incluido en Kit DepecaBot

4. Estructura Mecánica

La estructura del robot es de tipo diferencial, con dos ruedas delanteras que realizan la tracción y un tercer punto de apoyo (similar a un tornillo con cabeza esférica) situado en la parte trasera cuya función es facilitar deslizamiento.

Todos los módulos mecánicos se sujetan al chasis de aluminio (ver figura 4) de 1.5 mm de espesor que se ha enviado a cortar por láser y plegar. Dicho chasis se ha ido mejorando a lo largo de los últimos años para ser muy versátil en cuanto a los elementos a sujetar (motores, sensores, electrónica, cableado, etc.), a la vez ser muy robusto mediante la aplicación de pliegues en sitios estratégicos para reducir su deformación, es ligero ya que se ha fabricado en aluminio con varios agujeros, y finalmente económico gracias a la búsqueda constante de proveedores dedicados a esta actividad.

La estructura de aluminio tiene la posibilidad de incorporar dos servo-motores de radio control (FUTABA 3003 o equivalente) como elementos motrices. Este tipo de motores, además de ser de bajo coste, se caracterizan por ser fáciles de adquirir en muchas tiendas de juguetes o comercios de venta por Internet, y es relativamente sencillo modificar su funcionamiento interno para permitir el movimiento continuo de su eje de salida.

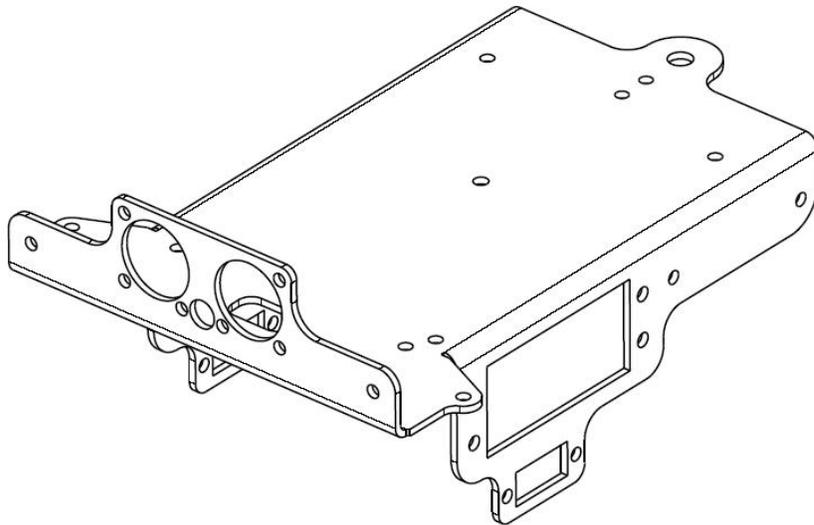


Figura 4. Chasis de aluminio utilizado en el robot

La rueda elegida es de 8 cm de diámetro y 3 mm de espesor realizada en material plástico. Estas medidas son similares a los discos compactos de reducido tamaño que se pueden encontrar en cualquier tienda de informática. La sujeción de la rueda al motor se realiza mediante una pieza en forma de cruz suministrada por el fabricante de motores. Para evitar el deslizamiento de las ruedas en el suelo, se ha elegido rodear a las mismas de una goma económica equivalente a las que se pueden encontrar en las cámaras de las ruedas de una bicicleta.

Además de los motores, la estructura permite sujetar varios tipos de sensores. Haciendo uso de tornillos de métrica 3 mm, se puede colocar módulos comerciales de infrarrojo (GP2D) [7] y ultrasonidos (SRF08) [8] para detectar distancia a un objeto cercano. También es posible incorporar económicos sensores reflexivos (CNY70) [9] en las zonas de las ruedas para dotar al sistema de la capacidad de medir la velocidad de las mismas a modo de “encoders”. La parte superior del chasis dispone de agujeros adicionales para permitir sujetar la tarjeta de control y pasar cables de la parte superior a la inferior, facilitando la interconexión eléctrica de los sistemas.

5. Electrónica

La electrónica del robot está formada principalmente por una tarjeta de control de hardware abierto llamada miniAlcadsPIC (figura 5), elementos sensoriales y una batería que suministra la energía necesaria para alimentar al mismo.

La tarjeta de control miniAlcadsPIC está basada en el procesador dsPIC de Microchip [4] y ha sido desarrollada principalmente para el control de robots móviles autónomos, aunque puede utilizarse para otro tipo de aplicaciones. La tarjeta miniAlcadsPIC es la evolución de otras tarjetas desarrolladas en el Departamento de Electrónica con el principal objetivo de abaratar costes. Dicha tarjeta integra la mayoría de los componentes básicos necesarios:

- Microcontrolador dsPIC30F4012 de Microchip [4] (con 48 Kbytes de memoria de programa, 2 Kbytes de memoria de datos, capacidad de generar señales PWM, adquirir señales analógicas, comunicarse por puerto serie y bus I2C, entre otras cosas).
- Doble puente en H de potencia L293D, que permite suministrar hasta 1 amperio de corriente a dos motores.
- Pantalla de cristal líquido alfanumérica de dos filas para facilitar la depuración de programas y permitir la comunicación con el usuario.
- Cuatro pulsadores destinados a permitir al usuario interactuar con el robot, configurar parámetros y controlar el flujo del programa.
- Zumbador piezoeléctrico que permite generar tonos con el fin de conocer la evolución del programa cargado en el robot.
- Interfaz con sensores para permitir el acondicionamiento y protección de las señales eléctricas que provienen del exterior la tarjeta.
- Interfaz de comunicación con el exterior. Mediante el bus I2C se facilita la comunicación con dispositivos electrónicos inteligentes. A través del puerto serie y un cable de adaptación a USB se permite la carga de programas y visualización de la evolución del mismo.

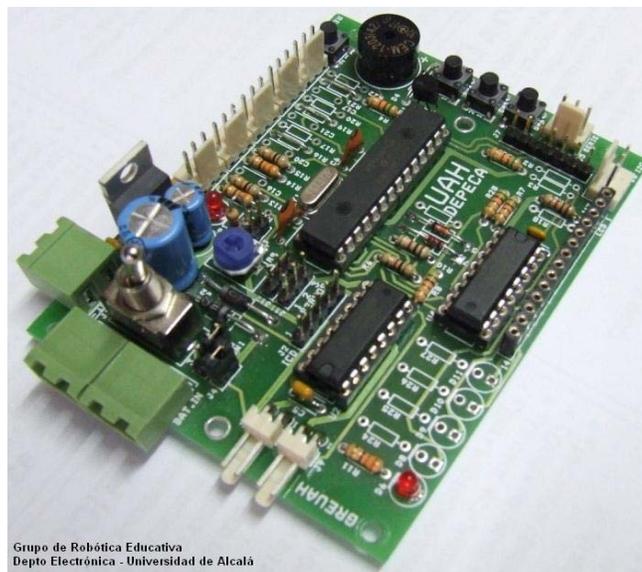


Figura 5. Tarjeta de control miniAlcadsPIC

Como elementos sensoriales del robot, se incluyen en el kit sensores GP2D y CNY70 (figura 6). Los primeros ofrecen una tensión de salida analógica variable inversamente proporcional a la distancia del objeto a detectar. El CNY70 es un sensor reflexivo compuesto por un emisor infrarrojo y un fototransistor; su uso está enfocado principalmente a aplicaciones de seguimiento de línea y detección de movimiento de las ruedas utilizando marcas en las mismas.

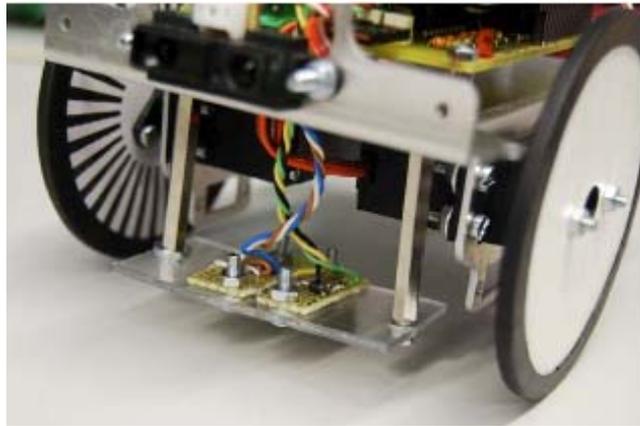


Figura 6. Sensor GP2D12 en la parte superior y sensores CNY70 en la inferior

La batería de alimentación que se suministra con el robot es de Litio-Polímetro de alta eficiencia energética utilizada comúnmente en aplicaciones de radio control. La misma se sujeta al robot con cinta adhesiva de doble cara o con bridas de sujeción. Se entrega también un cargador de este tipo de baterías que se conecta a la red eléctrica

6. Bibliotecas de funciones

Se han creado unas bibliotecas de código abierto desarrolladas para el robot DepecaBot y la tarjeta miniAlcadsPIC. Estas bibliotecas se continúan manteniendo por una gran comunidad de alumnos, ex-alumnos y profesores con el siguiente fin:

- Disponer de unas bibliotecas sencillas de utilizar para los principiantes. Por otro lado se busca que sean muy potentes y flexibles para los aficionados.
- Facilitar el aprendizaje de los alumnos en el área de software embebido mediante la programación de robots móviles.
- Optimizar los recursos de un microcontrolador.
- Crear una comunidad de desarrolladores y usuarios con intereses en la robótica móvil y el desarrollo de aplicaciones embebidas.

Para cumplir estos objetivos se ha propuesto separar las bibliotecas en diferentes capas o niveles de abstracción. En la figura 7 se presentan las diferentes bibliotecas de funciones desarrolladas por el Grupo de Robótica Educativa [3] para la realización de varios proyectos educativos.

Para la realización del Kit DepecaBot se utiliza la biblioteca de bajo nivel (UAHdsPIClib) que soporta el procesador dsPIC30F4012, la biblioteca de medio nivel (miniAlcadsPIClib) que es compatible con la tarjeta miniAlcadsPIC y por último la biblioteca de alto nivel (DepecaBotlib) utilizada para controlar el robot DepecaBot.

- *Biblioteca de bajo nivel:* Las funciones diseñadas en este nivel están adaptadas a un microcontrolador concreto y permiten utilizar todos sus periféricos tales como pines de entrada-salida, temporizadores, puertos serie, puertos I2C, generadores de PWM y conversores analógico a digital, entre otras cosas. Se la ha llamado bajo nivel, ya que es la que está más cerca del nivel hardware. En éste caso decimos que trabajamos con la biblioteca a nivel de chip. Debido a que se han utilizado en diversos proyectos dos microcontroladores diferentes (dsPIC30F6010A y dsPIC30F4012) hay dos versiones de la biblioteca UAHdsPIClib correspondientes a cada uno de ellos. En la tabla [1] se muestra el listado de funciones de bajo nivel disponibles.

- Biblioteca de medio nivel:** Se describen como de medio nivel, o nivel de tarjeta, ya que se apoyan sobre el bajo nivel (descrito anteriormente) para dar funcionalidad a diferentes periféricos disponibles en una tarjeta electrónica concreta. Existen dos bibliotecas diferentes, cada una adaptada a las dos tarjetas disponibles (AlcadsPIClib y miniAlcadsPIClib). En el proyecto Depecabot se usa la tarjeta miniAlcadsPIC y por tanto la biblioteca miniAlcadsPIClib. Existen funciones para escribir en la pantalla alfanumérica, generar tonos con el buzzer, medir tensión de un puerto analógico y de la batería, leer información de los pulsadores y generar una tensión variable mediante un chip de potencia. En la tabla [2] se muestra el listado de funciones de medio nivel disponibles.
- Biblioteca de alto nivel:** Por último, este nivel permite al alumno trabajar con funciones orientadas al movimiento y percepción del entorno de un robot, como puede ser la petición de movimiento en una dirección, lectura de velocidad de las ruedas, lectura de sensores para detección de obstáculos y colores, etc. Para el correcto uso de ésta biblioteca se requiere disponer del nivel medio y bajo descrito anteriormente. Se han desarrollado tres bibliotecas de alto nivel, correspondientes a los tres proyectos desarrollados (Cyborlib, SpeedRunnerlib y DepecaBotlib). En la tabla [3] se muestra el listado de funciones de alto nivel disponibles.

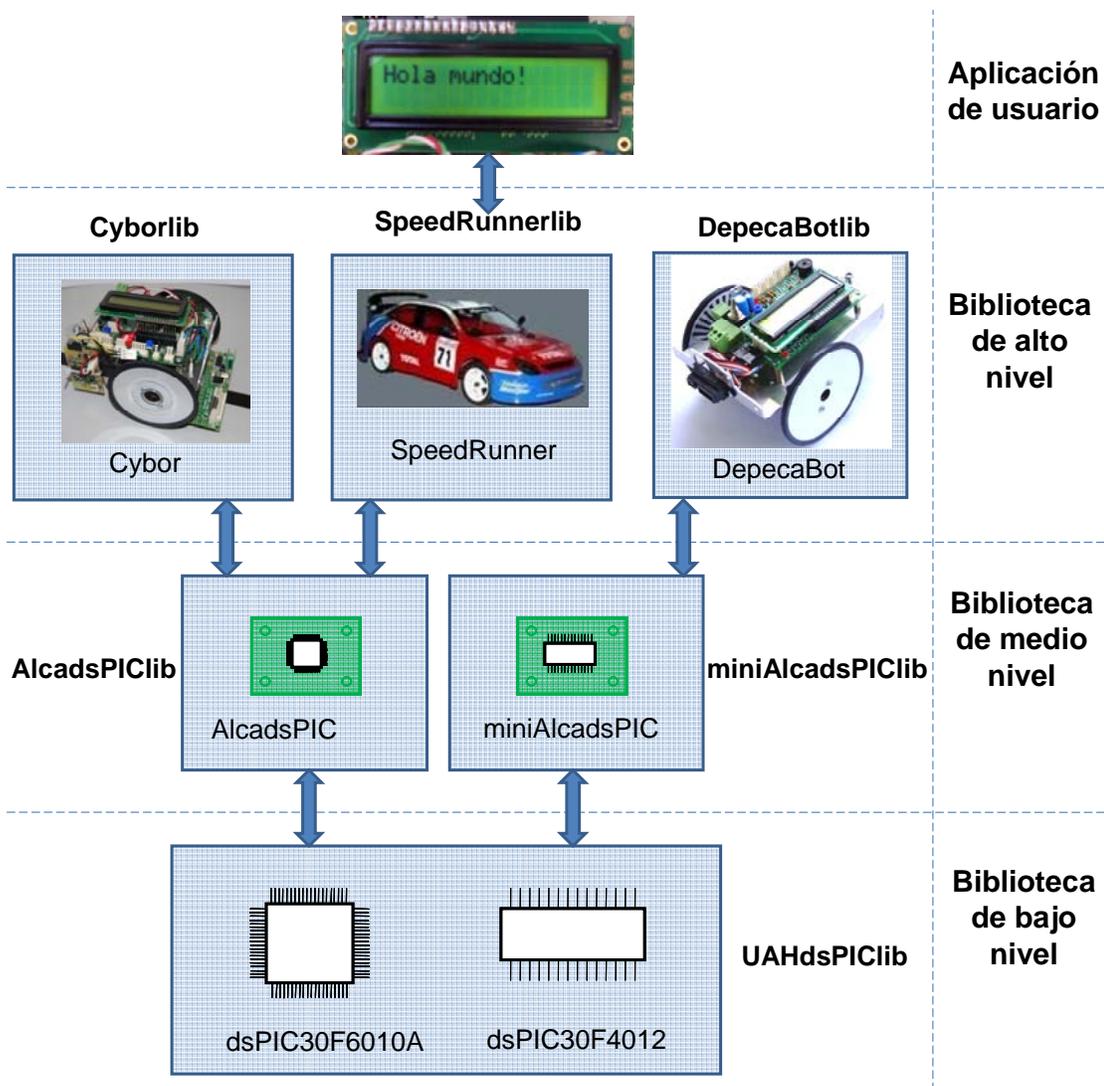


Figura 7. Detalle de las diferentes bibliotecas disponibles y la relación entre ellas

Tabla 1. Listado de funciones de bajo nivel

Convertor analógico-digital	void ADC_init (int channels) unsigned int ADC read (char channel)
Temporizador 1	void TIMER1_soft_set (unsigned char n, int time, char flags) void TIMER1_soft_reset (unsigned char n) unsigned char TIMER1_soft_quest_state (unsigned char n) void TIMER1_soft_init (void) void TIMER1_periodic_control (void)
Temporizador 2	void TIMER2_freq_init (void) void TIMER2_freq_set (unsigned int timer2_frecuency) void TIMER2_enable (unsigned char timer2_enable)
Puerto Serie	void UART1_init (char uart1_baud_config) char UART1_send_char (char uart1_tx_char) char UART1_send_string (char *uart1_tx_string) char UART1_receive_char (void) char UART1_ask_n_elements_buffer_rx (void)
Generador PWM	void PWM_init (unsigned char pwm_channel) void PWM_set_duty_cycle (unsigned char pwm_channel, int value) void PWM_enable (unsigned char pwm_channel) int PWM_get_direction (unsigned char pwm_channel)
Bus I2C	void I2C_master_init (void) char I2C_send_pkg (struct tpkg i2c_pkg, unsigned char priorit) void I2C_master_int_enable (unsigned char i2c_int_enable)

Tabla 2. Listado de funciones de medio nivel

Pulsadores	void BUTTON_init (void) void BUTTON_refresh (void) char BUTTON_status (unsigned char button_number)
Leds	void LEDS_init (unsigned char chip_id) void LEDS_set (char leds_number, unsigned char leds_status) void LEDS_all_off (void)
Pines entrada-salida	void INOUT_init (char channel, char mode) unsigned int INOUT_analog_read (char channel) void INOUT_digital_set (char channel, char level) char INOUT_digital_read (char channel) unsigned int INOUT_battery_read (void)
Control de motores	void MOTOR_init (unsigned char motor_channel) void MOTOR_enable (unsigned char motor_channel) int MOTOR_get_direction (unsigned char motor_channel) void MOTOR_set (unsigned char motor_channel, int motor_speed)
Pantalla alfanumérica LCD	void LCD_clear_buffer (void) void LCD_init (char chip_id) void LCD_send_character (char chr, char row, char column) char LCD_send_string (char *string, char row, char lcd_column)
Generador de sonidos	void BUZZER_init (void) void BUZZER_set_frecuency (int frecuency, int buzzer_duration) void BUZZER_enable (unsigned char buzzer_enable) void BUZZER_control (void)
Control de servomotores	void SERVO_MOTOR_init (unsigned char servo_motor_channel) void SERVO_MOTOR_enable (char channel, char enable) void SERVO_MOTOR_set (char channel, char servo_motor_pos)

Tabla 3. Listado de funciones de alto nivel

Control del movimiento	void MOVEMENT_motor_set (char channel, int speed) void MOVEMENT_disable (void) int MOVEMENT_get_motor_direction (unsigned char motor_channel) void MOVEMENT_init (void) void MOVEMENT_turn (int lineal_speed, int angular_speed) void MOVEMENT_straight (int speed, int error)
Sensor de distancia GP2D	unsigned int GP2D_read_mm (char channel)
Sensor reflexivo CNY70	unsigned char CNY70_analog_read (char channel) unsigned char CNY70_digital_read (char channel, int bound)
Encoder de las ruedas	void ENCODER_reset (char channel) int ENCODER_read (char channel) void ENCODER_control (void) void ENCODER_init_cny70_bound (int bound_ch1, int bound_ch2)
Robot en general	void DEPECABOT_init (void) void HIGH_LEVEL_periodic_control (void) void BATTERY_level_control (void)

La licencia con la que se suministran las bibliotecas de funciones es Creative Commons [11]. Los usuarios pueden copiar y distribuir estas bibliotecas, así como modificarlas para un uso personal. Únicamente se exige que se realice un reconocimiento de los autores, que no se obtenga beneficio comercial y si se comparten sea bajo el mismo tipo de licencia.

Por encima de estas bibliotecas está el programa de usuario que es el que realizaría la persona que programaría el robot. Dependiendo de la formación de esta persona o de los objetivos del curso o taller se puede limitar a utilizar las funciones de alto nivel o utilizar, modificar y añadir funciones de medio y bajo nivel.

7. Entorno de Programación

El entorno de programación utilizado es el MPLAB suministrado por Microchip junto con el compilador C30 con licencia de estudiante ambos con carácter gratuito. El compilador C30 utiliza el conocido GCC como compilador por lo que también podría utilizarse por línea de comando o con un entorno que utilice este compilador.

El MPLAB es un software profesional utilizado ampliamente en la industria de desarrollo de software embebido. Con éste entorno de programación los alumnos aprenden y adquieren experiencia en una herramienta que puede ayudarles en su futuro profesional.

8. Documentación

La documentación es una de las tareas más importantes de todo proyecto y muchas veces por falta de tiempo se deja para último momento. En el Grupo de Robótica Educativa nos hemos concienciado a documentar continuamente todo el trabajo realizado y de esta manera facilitar el proceso de aprendizaje de los alumnos.

La documentación recopilada y generada en relación con el Proyecto Depecabot es la siguiente:

- *Documentación del código y las bibliotecas de funciones.* La documentación del código fuente se ha realizado utilizando el sistema de documentación automática de código Doxygen [5]. La documentación generada está disponible a través de Internet en formato HTML. Para obtener las bibliotecas los alumnos pueden descargarse las mismas a través de la página Web del curso ó acceder a través de un servidor de Subversion [6] destinado al control de versiones.
- *Ejemplos de utilización de las bibliotecas de funciones,* también documentados con Doxygen.
- *Listado de componentes del Kit Depecabot.*
- *Guía de montaje de la tarjeta miniAlcadsPIC.* Un manual que describe paso a paso y con fotos el procedimiento para realizar el montaje y soldadura de la tarjeta electrónica miniAlcadsPIC. Dicho manual describe aspectos para iniciarse en la soldadura de una tarjeta electrónica, secuencia de colocación y soldadura de cada uno de los componentes de inserción de la misma etc.
- *Guía de montaje mecánico del DepecaBot.* Un manual que describe paso a paso y con fotos el procedimiento para montar los diferentes elementos mecánicos del robot así como el trucaje del servomotor para permitir que la rueda gire constantemente.
- *Manual de Programación,* donde se explica la forma de utilizar las herramientas de programación, desde su descarga e instalación hasta los pasos a seguir para la creación de un proyecto software (incorporación de bibliotecas, definición del procesador, creación de un programa en lenguaje C y su posterior compilación) y el procedimiento cargar un programa en el robot.

- *Transparencias de introducción a la robótica móvil.* Material utilizado en el curso para introducir a los alumnos en los elementos básicos de un robot móvil y explicar el funcionamiento de los sensores que componen el Kit.
- *Transparencias explicativas de las funciones más importantes de las bibliotecas de funciones.*
- *Transparencias de explicación de fundamentos de robótica móvil,* como el control de motores con PWM y algoritmos de control clásico con ejemplos

Toda la documentación de transparencias y manuales se deja a disposición del alumno a través de la página Web del Grupo de Robótica Educativa [3].

9. El Taller Depecabot

El Taller Depecabot tiene como objetivo iniciar a la gente en la robótica y en la electrónica. Está diseñado para que pueda participar cualquier persona con el único condicionante de que sepa programar en C (o tenga bastante experiencia en otro lenguaje similar).

En el taller se explica el funcionamiento de la tarjeta de control y de los sensores, se enseña cómo instalar y manejar las herramientas de desarrollo (compilador, depurador y cargador) y se realizan pequeños programas de ejemplo que ayudan a familiarizarse con los diferentes sistemas del robot.

Cuando ya se controla la programación básica del robot, se propone al alumno trabajar en la programación del seguimiento de una línea, seguir paredes, detectar y evitar obstáculos entre otras cosas. Como práctica avanzada se trabaja en el algoritmo para salir de un laberinto de la forma más rápida posible.

El taller esta formado por sesiones teóricas y sesiones prácticas supervisadas por varios profesores del taller. Cada grupo monta, programa y pone a punto su robot. En las anteriores dos ediciones del Taller se ha planteado con una duración de 20 horas dividido en 5 sesiones de 4 horas por la tarde (entre las 16 y las 20hs). En la Tabla 4 puede verse la planificación del taller.

Tabla 4. Planificación del Taller Depecabot

<p>Día 1:</p> <ul style="list-style-type: none"> ● Presentación ● Explicación de los elementos del KIT ● Trabajo personal tutorizado de montaje miniAlcaDspic (soldadura) ● Carga del programa de prueba ● Trabajo personal tutorizado de montaje DepecaBot (mecánica) <p>Día 2:</p> <ul style="list-style-type: none"> ● Explicación de las herramientas de desarrollo. ● Explicación de las bibliotecas de funciones de alto nivel. ● Trabajo personal tutorizado de programación básica <p>Día 3:</p> <ul style="list-style-type: none"> ● Se plantea un reto de competición entre los participantes. ● Trabajo personal tutorizado en programación 	<p>Día 4:</p> <ul style="list-style-type: none"> ● Explicación de los subsistemas internos del dsPIC ● Explicación de las bibliotecas de funciones de medio y bajo nivel ● Trabajo personal tutorizado de programación <p>Día 5:</p> <ul style="list-style-type: none"> ● Trabajo personal tutorizado de programación ● Se realiza la competición entre los participantes intentando analizar el porqué de los diferentes comportamientos de los robots. ● Clausura
--	---

El reto propuesto en la edición de 2008 fue salir de un laberinto. El reto del 2009 fue una especie de “Triatlón” en el que había que seguir una línea, detectar paredes y evitar obstáculos.

En la última edición del taller (2009) han participado un total de 28 personas entre los que había estudiantes de ingeniería, estudiantes de ciclos formativos y bachillerato e incluso profesores de

secundaria y de ciclos formativos. En las tablas 5 y 6 se muestran los resultados de una encuesta que se pasó a los participantes del taller. En las primeras preguntas (Tabla 5) se muestra la mediana de las respuestas correspondientes (1 – Muy en desacuerdo, 2 – En desacuerdo, 3 – Normal, 4 – De acuerdo y 5 - Muy de acuerdo). En el segundo bloque de preguntas (Tabla 6) se indica el porcentaje de respuestas positivas.

Tabla 5. Encuesta de opinión I

	Mediana
En número de profesores me ha parecido adecuado	4
Los profesores me han ayudado cuando lo he necesitado	5
Los profesores han demostrado dominar la materia	4
El trato ha sido correcto	5
La atención ha sido personalizada	5
Voy a seguir utilizando el KIT tras el taller	4
El KIT me puede ser útil para hacer otro robot	4,5
El KIT me ha parecido completo	4
El KIT ha sido fácil de montar	4,5

Tabla 6. Encuesta de opinión II

	% Afirmativo
El Taller ha cumplido con mis expectativas	100%
El Taller me ha ayudado a aclarar conocimientos sobre mis estudios	92%
El taller me ha ayudado a enfocar mis estudios en el futuro	50%
Ya sabía programar en C antes de llegar al Taller	92%
He aprendido a utilizar las funciones de las bibliotecas de alto nivel	100%
Me ha parecido sencilla la utilización de las bibliotecas de alto nivel	100%
La documentación sobre el KIT ha sido clara	67%
La documentación sobre el KIT ha sido completa	75%
He tenido todos los recursos necesarios para poder realizar el taller	92%
He trabajado mucho para poner en marcha el robot	50%
El Taller me ha llevado más tiempo del que pensaba	50%
Me gustaría haberle dedicado al taller menos tiempo	17%
Me ha compensado el esfuerzo realizado respecto a la experiencia vivida durante el taller	92%
Me ha compensado el esfuerzo realizado respecto a los resultados obtenidos	75%

En las tablas anteriores se puede comprobar que el 92% de los participantes están satisfechos con la experiencia y que el 17% no querían haberle dedicado más tiempo. Eso teniendo en cuenta que el 50% cree haber trabajado mucho para poner en marcha el robot. Todos dicen que el taller ha cumplido con sus expectativas.

Es importante indicar que el 92% dicen que el taller le ha ayudado a aclarar conocimientos sobre lo que han estudiado y el 50% les ha ayudado a enfocar sus estudios en el futuro.

En la figura 8 se pueden encontrar un par de instantáneas del taller.



Figura 8. Alumnos trabajando en el Taller DepecaBot

10. Otras aplicaciones docentes

Además del Taller Depecabot, la plataforma diseñada ha sido utilizada como punto de partida del hardware en varios proyectos fin de carrera y en la realización de prácticas de una asignatura de libre elección denominada “Introducción al Diseño de Microrrobots Móviles”[2]. Se está trabajando en la plataforma para desarrollar nuevos talleres más avanzados.

11. Conclusiones

A lo largo del curso académico muchos alumnos optan por comprar el Kit DepecaBot para desarrollar de manera personal sus inquietudes, realizar experimentos y en varias ocasiones realizar su proyecto de fin de carrera en áreas relacionadas con la robótica móvil.

Los resultados presentados en la encuesta de opinión realizada a los participantes del Taller Depecabot demuestran que están satisfechos, que les ha ayudado a clarificar algunas ideas respecto a sus estudios y se lo han pasado bien a la vez que aprenden.

La disponibilidad en Internet de todas las funciones de código y los esquemas de las tarjetas está ayudando a crear comunidad y que poco a poco el sistema vaya siendo más utilizado.

Toda la información relacionada con el Kit de Robot DepecaBot y sus desarrolladores se encuentra en su página Web [3].

12. Agradecimientos

Se agradece la colaboración del Departamento de Electrónica de la Universidad de Alcalá por el apoyo que siempre ha mostrado en las actividades de promoción de la Ciencia y la Tecnología. También se agradece a Esther Samper, Javier Rodríguez, Ana Vera, Marcelino Camacho y Borja Vizcaíno por su colaboración en la preparación del material para el taller.

Las actividades se han organizado con la colaboración y financiación de la Fundación Española para la Ciencia y la Tecnología – Ministerio de Ciencia e Innovación dentro de la Convocatoria de ayudas para el programa de cultura científica y de la innovación 2009.

Referencias

- [1] Semana de la Robótica de la Universidad de Alcalá: www.alcabot.org
- [2] Departamento de Electrónica, Universidad de Alcalá: www.depeca.uah.es
- [3] Grupo de Robótica Educativa, Kit del Robot DepecaBot: www.roboticaeducativa.es
- [4] Microchip fabricante de microcontroladores: www.microchip.com
- [5] Doxygen, generador de documentación para C: www.doxygen.org
- [6] Subversion, software de sistema de control de versiones: <http://subversion.tigris.org>
- [7] Sensor de distancia infrarrojo GP2D de Sharp: <http://www.acroname.com/robotics/parts/SharpGP2D12-15.pdf>
- [8] Sensor de distancia ultrasonido SRF08 de Devantech: <http://www.robotstorehk.com/srf08tech.pdf>
- [9] Sensor reflexivo CNY70 de Vishay: www.vishay.com/docs/83751/cny70.pdf
- [10] Robótica en la Campus Party 2006 <http://www.depeca.uah.es/alcabot/campusparty2006/>
- [11] Creative Commons www.creativecommons.org