

# ALCABOT 2002

## ROBOT: VIGILANTE GRUPO: RBZ-VIGILANTE

Raúl Moya González

Luarme23@hotmail.com

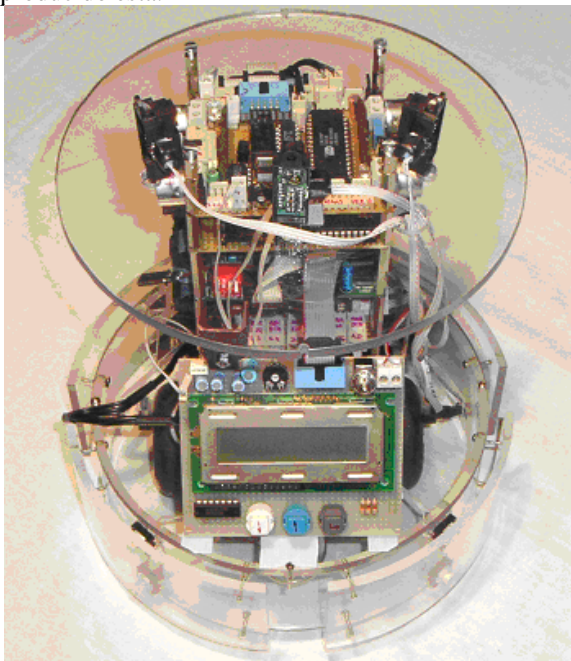
### Resumen

Este robot compete en la PRUEBA LIBRE.

VIGILANTE es un robot móvil aplicado a la vigilancia de recintos cerrados, con un interfaz de comunicación GSM, tanto para realizar llamadas en caso de emergencia, como para recibir comandos desde la central de alarmas.

El robot puede seguir varias rutas marcadas en el suelo, distinguiendo en cual se encuentra, o bien desplazarse en modo libre, sin colisionar, con capacidad para volver a enlazar con las rutas marcadas.

En el caso de que se detecte un intruso, envía un mensaje corto mediante su módulo GSM a la central de alarmas, o bien, podría realizar una llamada telefónica a la central indicando con voz que se ha producido la alarma y donde se ha producido esta.



### 1. Características

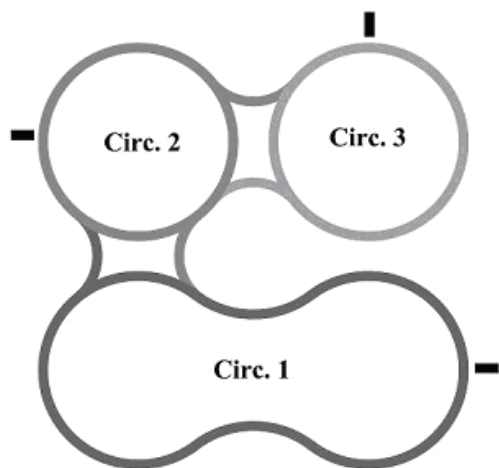
- Microcontrolador Motorola 68HC11A1
- Ampliación memoria externa RAM no volátil del 32KBytes
- Ampliación de cuatro puertos más
- Modulo de voz de 20 segundos
- LCD de 2 líneas y 16 caracteres cada una

- Teclado de tres pulsadores
- Dos baterías de 7,2v y 1900mA. Una para los motores y otra para todo el hardware.
- Dos reguladores de tensión de bajo dropout de 5v, uno por batería.
- Medida de tensión de las baterías por el micro.
- Teléfono móvil NOKIA 5110, para su comunicación con el exterior.
- Detector de tonos multifrecuencia (DTMF) para el teléfono.
- Estructura realizada toda en metacrilato
- Medidas del robot son: 19cm de diámetro y 24cm de altura
- El peso del robot completo con baterías y teléfono, es de 2,5 kilos.
- Dos motores de corriente continua con caja reductora
- Sensores:
  - Dos ultrasonidos para la detección de intrusos
  - Cuatro bumpers para la detección de colisiones
  - Tres GP2D12 para la medida de distancias
  - Dos encoders uno por rueda
  - Dos CNY70 en modo analógico para la detección de líneas de distintos colores en el suelo
  - Dos CNY70 en modo digital para la detección de la línea negra en el suelo

### 2. Funciones y capacidades

Al encender el robot, mediante el display y el teclado configuramos el modo de funcionamiento del robot. También se nos indica el estado de las baterías, y debemos de tomar muestras de las líneas de los distintos circuitos que va a recorrer. Para esto situamos los CNY70 analógicos sobre las líneas de los circuitos y tomamos el valor.

En la siguiente figura podemos ver, un ejemplo de tres circuitos con distinta tonalidad de gris:



Se puede ver como los circuitos tienen distintas tonalidades de gris para que los infrarrojos puedan diferenciarlos.

El robot tiene varios modos de funcionamiento:

- **Modo rastreador:** sigue la línea de los tres circuitos. En concreto lo que hace es dar dos vueltas al circuito 1, luego se pasa al circuito 3 y da dos vueltas a este, luego pasa al circuito 2 y da dos vueltas, y de nuevo pasa al circuito 1 para seguir con la misma secuencia. Las tres líneas negras que se ven en los circuitos son para que cuando el robot pase por ellas, se pare durante unos 3 segundos a vigilar, pues el robot cuando está paseando lleva los detectores de movimiento desactivados, solo los activa cuando se para, ya que en movimiento detectaría a los objetos moverse y saltaría la alarma sin haber ningún intruso. Las marcas negras se pondrían en sitios conflictivos, como son puertas y ventanas, con lo cual habría que poner varias por circuito, pero para la demostración se ha puesto una sola por circuito para que no sea tan lento.
- **Modo circuito 1:** cuando se activa este modo independientemente de en qué circuito se encuentre se va a este directamente y se queda continuamente en él.
- **Modo circuito 2:** igual que el anterior pero se queda en el circuito 2.
- **Modo circuito 3:** igual pero quedándose en el circuito 3.
- **Modo libre:** en este modo ignora las líneas del suelo y que pasea libremente por las habitaciones sin colisionar. Para ello emplea bumpers para la detección de choque, y tres GP2D12 para la medida de distancias a los objetos. Para la detección de intrusos lo que hace es pasearse durante 7s y pararse 3s a vigilar, y así continuamente.
- **Modo manual:** este modo sólo funciona cuando estamos conectados al robot telefónicamente. Y consiste en que mediante las teclas del teléfono lo movemos hacia delante, hacia atrás, derecha e izquierda, como si fuese un mando. Este modo solo es útil, si estamos viendo al robot, para ello tendría

que haber cámaras de vigilancia en las distintas habitaciones.

Cuando el robot detecta un movimiento mediante sus sensores de ultrasonidos, se activa la alarma, y lo que hace es enviar un SMS al móvil que le tengamos programado. El mensaje que envía es "ALARMA".

También el robot en todo momento, puede detectar una llamada entrante, lo que hace es descolgar y decir el siguiente mensaje mediante su modulo de voz: "Introduzca código". A continuación el usuario mediante el teclado del teléfono introduce el código, que es 45. En caso de haber introducido correctamente el código el robot reproducirá: "Código correcto", a continuación dirá en el modo de funcionamiento en que se encuentra y si es el de rastreador indicará en que circuito. Si el código introducido es incorrecto dirá: "Código incorrecto" y de nuevo pedirá que tecleemos el código.

Una vez que el robot ha dicho en que modo se encuentra, el usuario mediante el teclado podrá cambiar de modo, activar/desactivar el envío de mensaje, y finalizar. La correspondencia de teclas es:

- 1 → modo circuito 1
- 2 → modo circuito 2
- 3 → modo circuito 3
- 4 → modo libre
- 5 → modo rastreador
- 6 → modo manual
- 7 → activación del envío de mensajes al móvil
- 8 → desactivación de envío de mensajes al móvil
- 9 → Fin. Término de la conexión. El robot cuelga su móvil.
- Si pulsamos cualquier otra tecla dirá: "Incorrecto"

En caso de que no pulsemos el 9 para terminar la conexión, el robot colgará, al permanecer más de un minuto sin recibir ninguna nueva orden.

En el modo manual, la correspondencia de teclas en la siguiente:

- 2 → avanza hacia delante
- 8 → avanza hacia atrás
- 4 → avanza hacia la izquierda
- 6 → avanza hacia la derecha
- 9 → salir de modo manual
- Cualquier otra tecla → parado

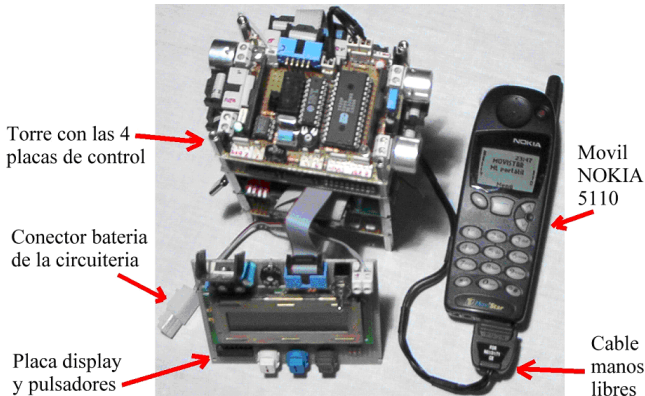
En el caso de estar en el modo manual y se corte la comunicación, el robot volverá al anterior modo en el que estaba funcionando y colgará el teléfono, trascurrido un minuto sin haber recibido ninguna orden.

Los bumpers siempre tienen preferencia sobre cualquier modo, si estando en modo manual el robot choca, este retrocederá, independientemente de que nosotros le estemos pulsando hacia delante. En modo seguidor de línea ocurre lo mismo, si se coloca un objeto que esté sobre la

línea el robot al chocar retrocede o gira dependiendo del bumper pulsado.

### 3. Recursos hardware

El hardware del robot está compuesto por cinco placas, cuatro de las cuales están superpuestas formando una torre, para así ahorrar espacio. Las conexiones entre placas se realizan mediante cable plano de diez hilos. A continuación se muestra una foto con todo el hardware del robot:



La placa del display no está en la torre para poder tener un mejor acceso a los pulsadores y a la visualización del display. Todas las placas están hechas en wrapping, menos la del microcontrolador que es de la casa Microbótica, realizada en placa de circuito impreso.

Los sensores que se pueden ver en la imagen son los dos de ultrasonidos, pero también hay sensores de distancia los GP2D12 que son por infrarrojos, los CNY70 que son de infrarrojos y se usan para la detección de colores en el suelo y como encoders, y también hay bumpers para la detección de colisiones.

Se puede apreciar como hay tres placas de la torre, en las que se ve el cable rojo de wrapping y que la placa del microcontrolador tiene un acabado profesional y es la única que no está hecha a modo de prototipo. También se ve el cable plano de diez hilos con el que se unen las placas. La medida de todas las placas es de 64 por 82mm.

### 4. Estudio del protocolo de móviles de Nokia

Esta es la parte principal y más novedosa de este proyecto, también la que más complicada ha resultado de realizar, esto ha sido debido, a que las marcas más comerciales de móviles (Nokia, Ericsson, Alcatel, Philips,...), no tienen a disposición pública el protocolo de comunicación de sus móviles con el exterior a través de su pinout, con lo cual se ha tenido que aplicar ingeniería inversa para su hallazgo.

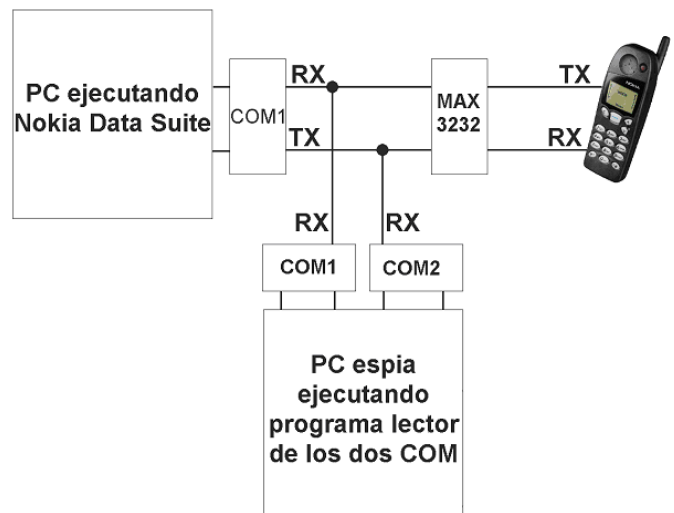
Se ha elegido un modelo de Nokia, debido a que esta casa dispone de un software para la comunicación de sus móviles con el PC. Con este software podemos realizar llamadas y enviar mensajes con el móvil desde nuestro PC. Así que aplicaremos **ingeniería inversa**, para espíar el

protocolo utilizado para enviar un mensaje y realizar una llamada, que es lo que a nosotros nos interesa.

Dentro de los modelos de Nokia se ha seleccionado el 5110 debido a su módico precio en el mercado de segunda mano (unos 20 €), también porque es completamente compatible con el software de Nokia de comunicación con el PC, por encontrarse información detallada de su pinout y de la implementación del cable de conexión con el ordenador. También sería compatible para este proyecto cualquier modelo de Nokia que en su pinout tuviese un TX y un RX (como las series 51xx y 61xx), estas patillas ya solo las tienen los modelos antiguos, ya que los modernos realizan la comunicación mediante un puerto de infrarrojos.

### CABLE ESPÍA:

El cable espía es él que hace posible que mediante un segundo PC, estemos observando la tramas que se envían entre el móvil y el PC que está ejecutando el Data Suite. A continuación se muestra el diagrama de conexión del cable espía:



El cable empleado para la conexión del Nokia 5110 y el PC, se utiliza como parte del cable espía. En el esquema al cable móvil-PC se le conectan dos cables justo antes del MAX3232, y estos nuevos cables se llevan al PC espía conectándolos uno al RX del COM1 y otro al RX del COM2, los dos se conectan a RX puesto que deseamos hacer la lectura de tanto lo que envía el Data Suite al móvil, como de lo que envía el móvil al Data Suite.

En el esquemático se ha omitido el poner la masa de todos los COM y el móvil, que tienen que interconectarse

### 5. Descripción del software

Toda la programación de Vigilante se ha realizado en ensamble. Aunque al disponer de 32KBytes de memoria, se podría haber hecho perfectamente en C. Pero como gran parte del programa, son interrupciones, y rutinas que acceden a los recursos internos del micro y los dispositivos externos que se le han añadido, se ha preferido hacer en ensamble por el control que este nos permite, ya que se

puede calcular cuantos ciclos de reloj ocupa una interrupción, y podemos ver con claridad si va a pisar a otra función. Además para el acceso a los recursos internos y externos del micro controlador, es más cómodo con ensamble. Solo para hacer bucles y rutinas complejas, con diversas posibilidades es mejor la programación en más alto nivel con C.

En la parte de programación de este proyecto, se han realizado las funciones para el control de todos los recursos que se le han puesto al robot. Y que todos estos recursos se puedan poner en funcionamiento en conjunto, sin perder el control de ninguno. Esta ha sido la parte más compleja de la programación, ya que, lo que se ha hecho ha sido ir programando cada una de las funcionalidades del robot por separado, pero a la hora de poner todas a la vez, han dado problemas, aunque por separado funcionaban perfectamente. Y se han tenido que depurar muchas veces, hasta conseguir, que cada función durase lo justo, para dar paso a la siguiente sin perder información. Por ejemplo cuando el robot está siguiendo la línea y recibe una llamada, este sigue rastreando, pero a la vez tiene que hablar con el interlocutor, y escuchar sus ordenes, esto resulta complicado para un micro tan pequeño que no puede realizar procesos en paralelo.

Una de las funciones que tiene más posibilidades de mejora en el robot, es la navegación en modo libre, ya que actualmente lo que hace es desplazarse evitando las colisiones, pero sin ningún objetivo concreto. En un futuro lo que se hará es aprovechar más los 32KBytes de memoria, para lo cual se dejarán todas las funciones que ya hay en ensamble, pero para la navegación se hará una función en C, que trabajará en forma borrosa, y se le podrán marcar trayectorias y rutas a seguir.

Motorola da para su micro el compilador de ensamble AS11, en los primeros pasos es el compilador que se utilizó, pero este tiene un gran inconveniente y es que no permite el poner INCLUDES, con lo cual toda la programación había que hacerla en el mismo fichero. Pero hace un año Javier de López desarrollo el RAS que es el mismo compilador que el AS11, pero que permite los INCLUDES, y tiene algunas otras ventajas. Esto ha facilitado mucho el desarrollo de un programa tan grande como este, ya que permite desglosar en varios ficheros el programa, y se hace mucho más fácil el control de todo. Hay que decir que el programa definitivo ocupa unos 3KBytes, aunque esto no pueda parecer mucho, para un programa realizado en ensamble si lo es, pues el programa puede tener unas 1200 líneas, haciendo una media de unos 2,5bytes por línea, ya que la mayoría de las intrusiones ocupan 2 o 3 bytes. ( $3KB/2,5B = 1200$ ).

Para la carga de los programas desde el PC al microcontrolador, se esta utilizando el CTLOAD, que también es de Javier de López, y funciona bajo Windows.

## 6. Conclusiones

El robot está listo para realizar su función de vigilar, cumpliendo todas las expectativas. Pero se le pueden añadir mejoras para que sea más robusto y pueda cumplir su función con más eficacia.

Lo más complicado y novedoso de la realización del proyecto ha sido:

- El envío de un SMS con un teléfono móvil estándar, sin información de su protocolo de comunicación, para lo cual se ha aplicado ingeniería inversa. En el mercado hay módulos GSM para este fin, pero se descarto por su elevado costo.
- Que el robot siga líneas de distintos colores, ya que no se han encontrado sensores de colores en el mercado y se ha tenido que trabajar con un sensor pensado para la detección de solo el color negro.

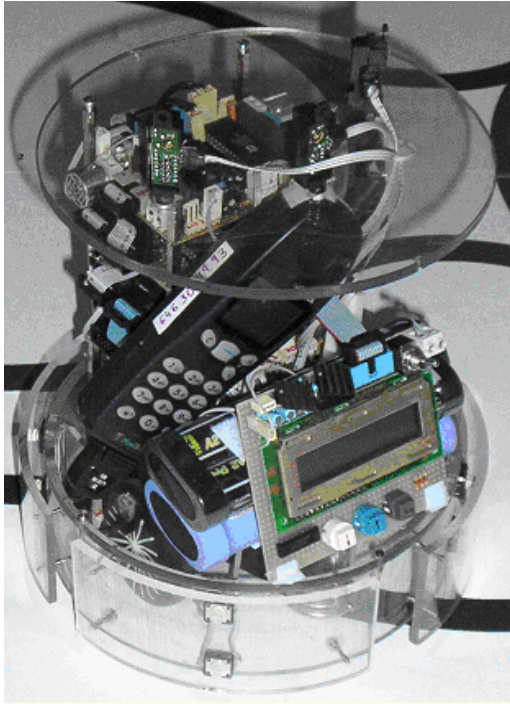
Considero que es un proyecto novedoso, pues estas funciones que incorpora Vigilante no se suelen ver. Los robot que he encontrado que tengan funciones similares son: los robot de fútbol que mediante códigos de colores son capaces de distinguir a los jugadores; y en cuanto al móvil, en Japón han sacado un robot con un modulo GSM, al cual puede llamar el propietario para indicarle tareas al robot.

En cuanto a otras de sus funciones:

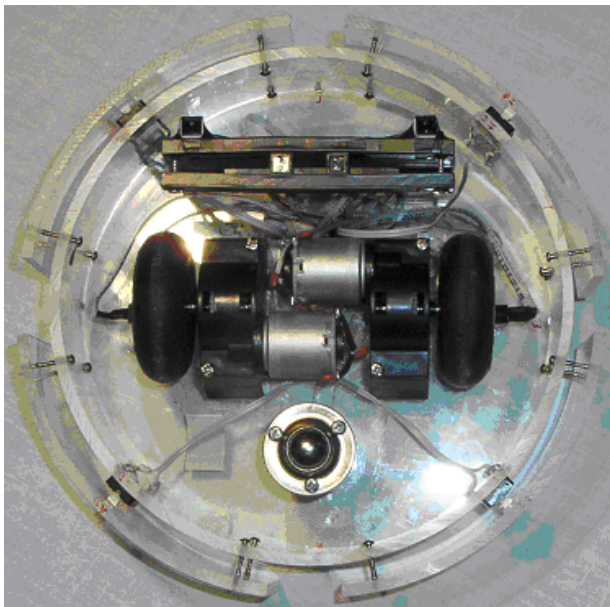
- El módulo de voz y el DTMF, han sido sencillos de implementar pues son C.I. que no necesitan de prácticamente nada externo, con lo que su puesta en funcionamiento es sencilla.
- Los GP2D12 y el detector de movimiento mediante ultrasonidos, son sensores que ya vienen con lógica adjunta y te dan una salida ya tratada, así que solo hay que conectarlos y ya podemos tomar muestras.
- Los CNY70 en modo digital han funcionado perfectamente desde el principio. Pero como ya he comentado, han dado problemas cuando se han puesto en modo analógico para la distinción de colores, ya que no están pensados para este fin.
- El L293 para el control de los motores, es casi un estándar para la mayoría de los microbot, por lo que hay mucha información para su puesta en marcha.
- La placa de ampliación de memoria y de puertos ha sido compleja de hacer pues lleva muchos integrados con gran cantidad de pines, además la información para su realización es escasa.
- Poner el display ha sido sencillo, pues hay mucha información e incluso hay programas ya realizados para su gestión.

Las placas aunque son prototipos hechos en wrapping, son muy robustas pues los componentes van soldados al cable de wrapping. La estructura hecha con metacrilato es bastante sólida y las piezas no van pegadas sino atornilladas, con lo cual en caso de romperse alguna es simple de sustituir.

El robot terminado ha quedado muy robusto, y no da ningún problema en su funcionamiento.



*Vista trasera de Vigilante*



*Vista inferior de Vigilante*