

Índice

1. Motivación

La motivación de este proyecto está dada por una de las aplicaciones que puede desprenderse de la rama de la Robótica, que investiga la manera óptima de hacer interactuar a un conjunto de robots, en un ambiente determinado, para cumplir un objetivo común, que hubiese sido dificultoso de resolver en forma individual por cada uno de los integrantes del conjunto. O bien, por la naturaleza del problema, no fuese útil disponer de un solo robot, sino que es inherentemente necesario disponer de una *Red* de individuos.

Esta aplicación a la que se hizo referencia al principio de este texto, es aquella vinculada al uso de una red de robots en un área de desastre, o de difícil acceso para los humanos. En la cuál estos robots pudieran buscar explosivos, objetos específicos, o incluso buscar por personas en situación riesgosa, que requieran de ayuda, ejemplo, en caso de haber sido víctimas de algún fenómeno natural, o de otro tipo.

Planteado el problema, se podría pensar en un conjunto de individuos (robots), cada uno de los cuales no estuviera preparado para resolver cualquier tipo de problema, sino que estuvieran dotados de sistemas que les permitieran reconocer un área, mapearla, tomar datos varios relacionados con el entorno, y eventualmente transmitirlos a un grupo humano, u otro grupo de robots, que al recorrer la misma zona, ya estarían advertidos de morfología del lugar, e incluso de ciertos peligros puntuales, de los que tuvieran que mantenerse alejados.

2. Objetivo

El objetivo de este trabajo es la fabricación de un robot capaz de salir de un laberinto.

3. Descripción general

Robot:

Para llevar a cabo este proyecto se fabricó un robot. El mismo está dotado de un microcontrolador ATmega16, dos motores con encoders, sensores de distancia por ultrasonido y sensores ópticos para detectar colores. La función de los sensores de distancia por ultrasonido es la medición de la distancia a cada pared del laberinto. La función de los sensores de colores, es detectar si el piso es blanco o negro, y de esta forma saber cuando se ingresó a una nueva celda o se encontró la salida del laberinto.

Laberinto:

El laberinto fue fabricado como módulos encastrables con unidad mínima modelizable como una matriz perteneciente a $\mathbb{R}^{3 \times 3}$, donde las paredes internas del mismo, definen *celdas* que se pueden interpretar como posiciones dentro de la matriz.

El mismo se contruyó con paredes color blanco y piso color negro, dividiendo cada celda por una línea blanca, porque sobre este color es óptimo el rebote de luz infrarroja para los sensores del piso, en el cual las marcas blancas sobre fondo negro dan un gran contraste, lo que se traduce en un marcado cambio de nivel en la señal de salida, y facilita la detección.

Como se mencionó subyacentemente en párrafos previos, cada celda se marcó con líneas blancas dispuesta en forma de cuadrado. Estas líneas sirven al robot para “saber” que al llegar a una nueva celda. El hecho de que las líneas formaran un cuadrado, es porque el robot podría venir de varias direcciones; es decir, al doblar para tomar un camino alternativo, atravesarían las celdas en sentido transversal al sentido de avance original.

4. Algoritmo

El algoritmo implementado para dotar a los robots de “inteligencia”, se basa en un sistema básico para la resolución de laberintos, que consiste en girar siempre para el mismo lado (derecha o izquierda). De esta forma siempre se encontrará la salida, aunque puede que sea no por el camino óptimo.

Resumen de funcionamiento:

5. Listado de componentes

Robot principal:

Referencia	Descripción	Cantidad	Precio unitario	Total
R11, R9, R10	1.5k	3		
R19, R12, R13, R14, R18	100	5		
C3	100n	1		
C2, C1	100u	2		
C6,C7,C8	10u	3		
RV2, R1, R2, R3, R4, R5, R6, RV1	10k	8		
D1, D2, D3, D4, D6, D7, D8,D9	1N4148	8		
C5	22u	1		
R15, R16,R17	330	3		
R8	390	1		
R7	4.7K	1		
C4	470n	1		
Q3, Q1, Q2	BC548	3		
IC1	C8051F330D	1		
U4, U3	CNY70	2		
P1	CON. BAT	1		
P7	CONN_1	1		
D10	IR1	1		
D11	IR2	1		
D12	IR3	1		
K1	IRU101533	1		
k4, k2, k3	ISU60	3		
U2	L298D	1		
D5	LED_1	1		
U1	LM2940	1		
P3	MOTOR_L	1		
P4	MOTOR_R	1		
P2	PROG	1		
SW1	RE-TRANS	1		
P5	RF_EMIT	1		
P8	SENS PLACA	1		
P9	SENS uC	1		
SW2	SW_PUSH	1		
P10	SWITCH_GRAL	1		
P6	Tx/Rx	1		

Robot secundario:

- Placa experimental FR2 de 10mc x 10cm (1)
- Microprocesador SiLabs 8051F330 (1)
- Puente H dual L298 (1)
- Regulador de tensión de 5V de baja caída LM2940 (1)
- Regulador de tensión de 3.3V de baja caída IRU101533 (1)
- Emisor IR y Fotorreceptor (en un solo encapsulado) CNY70 (2)
- Conectores Molex de dos vías (3)
- Conectores de cable plano de 10 hilos (1)
- LED color amarillo (1)
- Resistencias 100Ω (3)
- Resistencias 330Ω (3)
- Resistencias $1k\Omega$ (3)
- Resistencias 220Ω (1)
- Resistencias $4,7k\Omega$ (1)
- Resistencias $10k\Omega$ (6)
- Capacitores $10\mu\text{F}$ (5)
- Capacitores $100\mu\text{F}$ (2)
- Capacitores $100n\text{F}$ (1)
- Transistores BC548 (3)
- Diodos 1N4148 (8)
- Pulsador (1)
- Módulo receptor de RF a 418MHz (1)

6. Herramientas de software

- Keil μ Vision 4 (v4.02): IDE para desarrollo del firmware.
- [KiCAD \(Build: 20100406 SVN-R2508\)-final: Diseño de esquemáticos y ruteo de placas](#)
- CuteCom (v0.22.0): Captura de datos por puerto serie (S.O. Linux - Ubuntu)
- \LaTeX (compilado con MikTeX v2.7):Lenguaje para paginado de este informe
- TeXnicCenter (v1.0): Editor de textos para \LaTeX

7. Bibliografía

- [I.Scott Mackenzie, "The 8051 Microcontroller",Prentice Hall, 1995](#)
- [Datasheet del Microcontrolador 8051F330](#)
- [M. Daniela López De Luise, "Inteligencia Artificial y Redes Neuronales", EnterpriseWare SRL, Junio de 2001](#)

8. Apéndice

Esquemáticos:

- [Esquematico versión pdf](#)

Links útiles:

- [Wikipedia - Resumen de los algoritmos más conocidos para resolución de laberintos](#)
- [Página dedicada a competencias de *Micromouse* \(Resolvedores de laberintos\)](#)
- [Referencia a un gran número de sensores diferentes](#)
- [Página dedicada a la robótica en general](#)

Diagrama en bloques (*)

Diagrama de flujo (*)

Código fuente (*)

*Ver páginas adjuntas a este informe.