



Técnicas y Dispositivos Digitales II Sistemas Embebidos	DEPTO. DE ELECTRÓNICA Y COMPUTACIÓN ÁREA DIGITALES - PLAN 2003
PROYECTO	<div style="display: flex; justify-content: space-around; align-items: center;">   </div>

COMPETENCIA DE ROBOTS EN LABERINTOS

OBJETIVO

El objetivo de este proyecto es que los alumnos apliquen y profundicen sus conocimientos en diseño digital mediante la implementación de un sistema de control eficiente en una FPGA, que permita a un robot autónomo navegar y salir de un laberinto desconocido en el menor tiempo posible. El proyecto se centra en el uso de lógica combinatorial, bloques embebidos, máquinas de estado finito e IP cores, programados en un lenguaje de descripción de hardware.

El sistema deberá permitir al robot tomar decisiones en tiempo real para identificar y reaccionar a obstáculos, desplazarse en línea recta sin desviaciones y sin colisiones, y ajustar su trayectoria de forma autónoma para evitar paredes y otras barreras hasta alcanzar la salida. Además, el robot deberá implementar un algoritmo que optimice el recorrido, minimizando el tiempo y la distancia recorrida dentro del laberinto.

METODOLOGÍA DE TRABAJO

Los estudiantes trabajarán en equipos multidisciplinarios, compuestos por alumnos de las carreras de Ingeniería Electrónica e Ingeniería en Computación. Cada equipo participará en una competencia, en la cual se premiará al equipo que logre que su robot complete el laberinto en el menor tiempo posible, incentivando el desarrollo de soluciones eficientes y creativas.

El sistema completo se dividirá en varios subsistemas, tales como el control de motores, sensores de detección de obstáculos y paredes, algoritmos de navegación y la lógica de control central. Cada equipo será responsable de asignar internamente las tareas y la implementación de estos subsistemas, de acuerdo con las habilidades y fortalezas de sus miembros, promoviendo tanto la especialización como la colaboración. Los grupos deberán gestionar sus proyectos utilizando Git, lo que les permitirá trabajar de manera colaborativa, realizar un seguimiento de los cambios y mantener un historial claro de las versiones del proyecto. Para simular la dinámica de trabajo actualmente utilizada en entornos profesionales.

OBJETIVOS DE APRENDIZAJE

- Diseñar sistemas digitales utilizando máquinas de estado.
- Desarrollar sistemas digitales sobre plataformas de diseño digital.
- Desarrollar habilidades de escritura de reportes de resultados e informes técnicos.
- Desarrollar habilidades para el desarrollo de criterios y evaluación en la solución de problemas.
(criterios de diseño, toma de decisiones, justificación y otras prácticas relacionadas)

1° PARTE: DISEÑO DEL SISTEMA

Presentación del diseño planteado en borrador. No requiere un formato especial, pero sí una descripción detallada de la solución implementada. Debe ser redactada de forma clara y con esquemas que ayuden a la explicación. No es necesario que sea en formato digital.

2° PARTE: IMPLEMENTACIÓN

La implementación se realiza sobre una placa de desarrollo FPGA. Los alumnos desarrollarán cada parte del sistema completo, realizarán simulaciones y realizarán pruebas en la FPGA.

3° PARTE: PRESENTACIÓN Y COMPETENCIA

El equipo deberá pegar el link del repositorio de Github donde este el circuito implementado en el siguiente link: https://docs.google.com/spreadsheets/d/1XUmhbFGSmgneNx_I5bad_plaMs7D-hDb9RR11urEhGk/edit?usp=sharing

El grupo debe realizar una breve presentación en la que TODOS los integrantes deben exponer el circuito desarrollado y se les harán preguntas. Luego se realizará una competencia entre los equipos, ganará el equipo que logre salir del laberinto pasando por la mínima cantidad de casilleros.

FECHA DE PRESENTACIÓN ⇒ 25 de Noviembre.

Designación de Grupos.

EQUIPO 1: 6 personas	EQUIPO 2: 6 personas	EQUIPO 3: 6 personas	EQUIPO 4: 5 personas
Manuel Perez Cabrera Lemuel Cotado Aga Joaquin Garré Sebastián Loyza Augusto Loza Benjamin	Julián Álvarez José Manuel Salgueiro Santiago Solari Sab Almeida Ramiro Larrieu Lacoste Ian Mozo Demian	Jorge Turina Gustavo Beltrán Amilcar Alcaraz Fontanals Manuel Meneghini Juan Iampietro Facundo	Máximo De Lellis Matias Estrada Starita Lucas Mosler Santiago Rabini Franco

* Alumnos de Electrónica.

* Alumnos de Computación

DESCRIPCIÓN DEL PROYECTO:

Dadas las características deseadas, el trabajo se divide en 4 sub-proyectos:

- **Subsistema_1** ⇒ Desplazamiento en línea recta

El autito debe ser capaz de avanzar en línea recta, para ello cuenta con los sensores laterales. Se deberá tomar la señal de estos sensores (entradas analógicas conectadas al ADC de la placa), y en caso de que el valor entregado por los sensores sea menor a cierto umbral accionar sobre las ruedas a través del motor para alejar al autito de la pared u obstáculo.

- **Subsistema_2** ⇒ Control de giro 90° / 180°

El autito debe ser capaz de realizar giros de 90° y 180° cuando detecte un muro en frente, para ello se deberá emplear el sensor de frente, y cuando detecte el muro accionar las ruedas a través de los motores y con un contador lograr que gire el ángulo deseado.

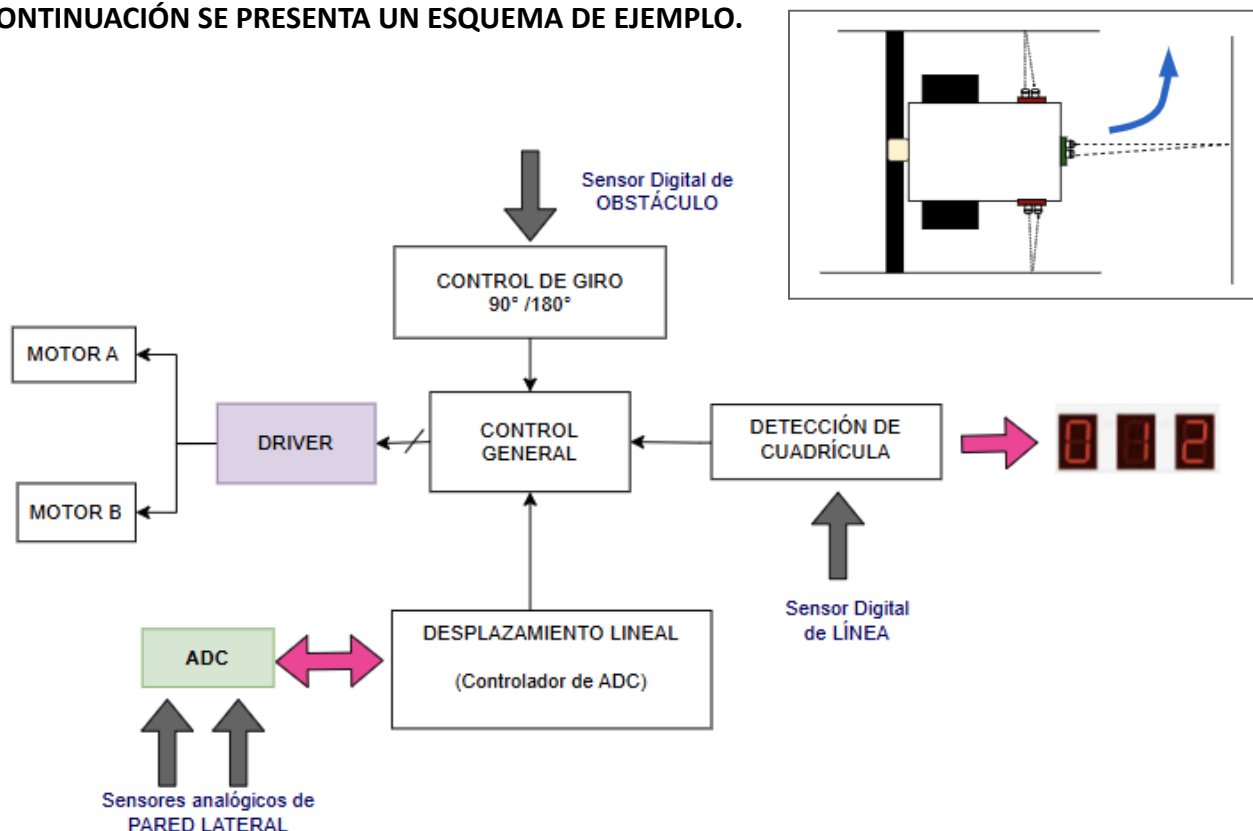
- **Subsistema_3** ⇒ Detección de cuadrícula

El autito debe ser capaz de “saber” en qué cuadrícula se encuentra. Para ello debe ser capaz de detectar cuando el sensor de línea (conectado debajo del autito) detecta la línea e incrementar un contador. El valor de este contador deberá mostrarse en los leds disponibles en la placa.

- **Subsistema_4** ⇒ Control General. Esto incluye Lógica de decisión.

El autito debe ser capaz de salir del laberinto, para ello se debe decidir qué estrategia emplear. Por ejemplo cada vez que detecte un muro girar aleatoriamente (esto no asegura que alguna vez salga del laberinto). Existen otros algoritmos, como el algoritmo de llenado (recomendado) <https://youtu.be/shlWNkn10oc>

A CONTINUACIÓN SE PRESENTA UN ESQUEMA DE EJEMPLO.



ESPECIFICACIONES TÉCNICAS

Se realizará sobre una placa de desarrollo FPGA DE0-Nano

(https://catedra.ing.unlp.edu.ar/electrotecnia/islyd/hoja_de_datos/DE0_Nano_User_manual.v1.1.pdf o <https://www.ti.com/lit/ug/tidu737/tidu737.pdf>) y se utilizará la herramienta de diseño Quartus II, producida por Altera, para el análisis y la síntesis de diseños realizados en HDL.

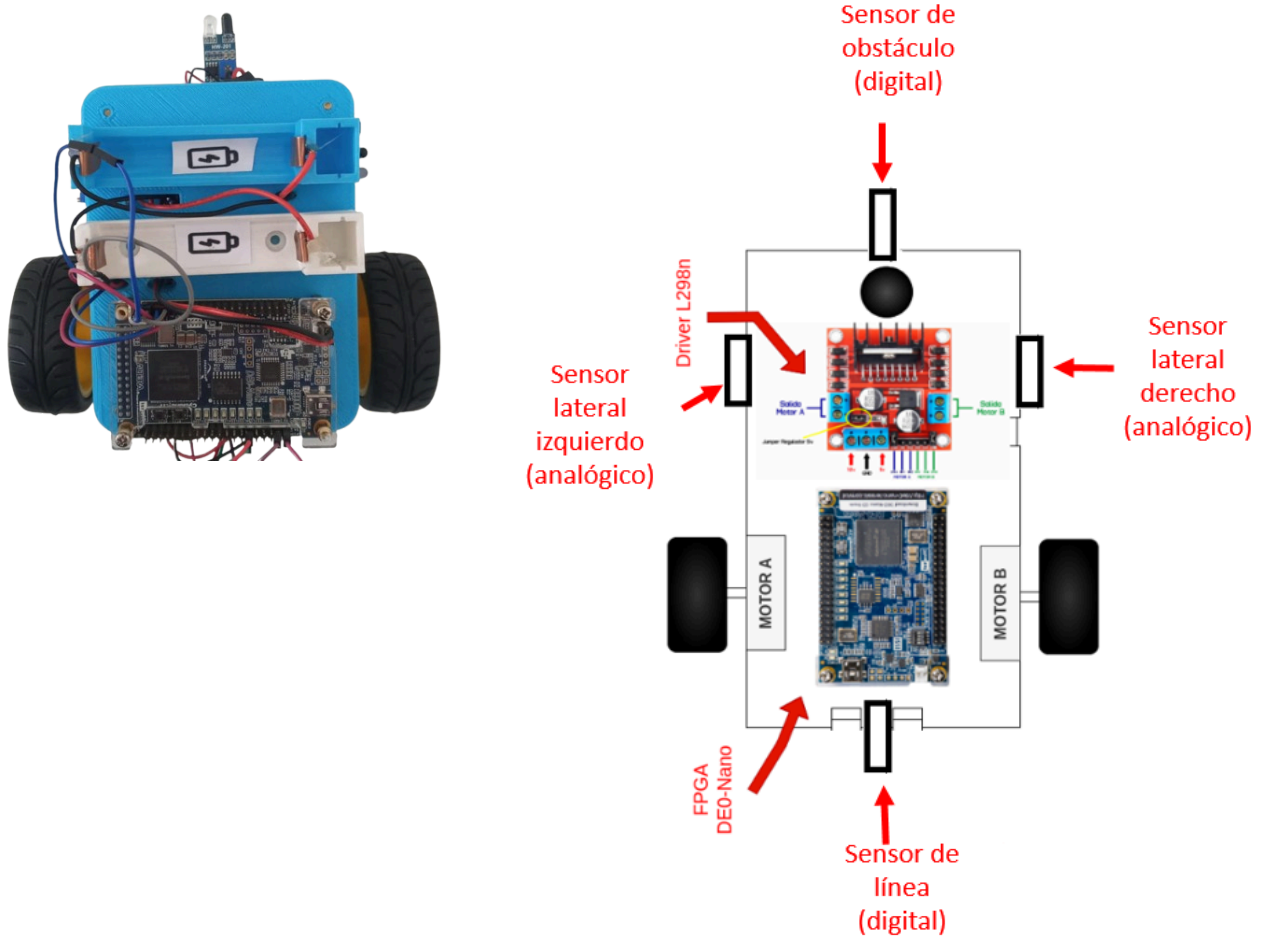


Fig. 1 Esquema del Robot

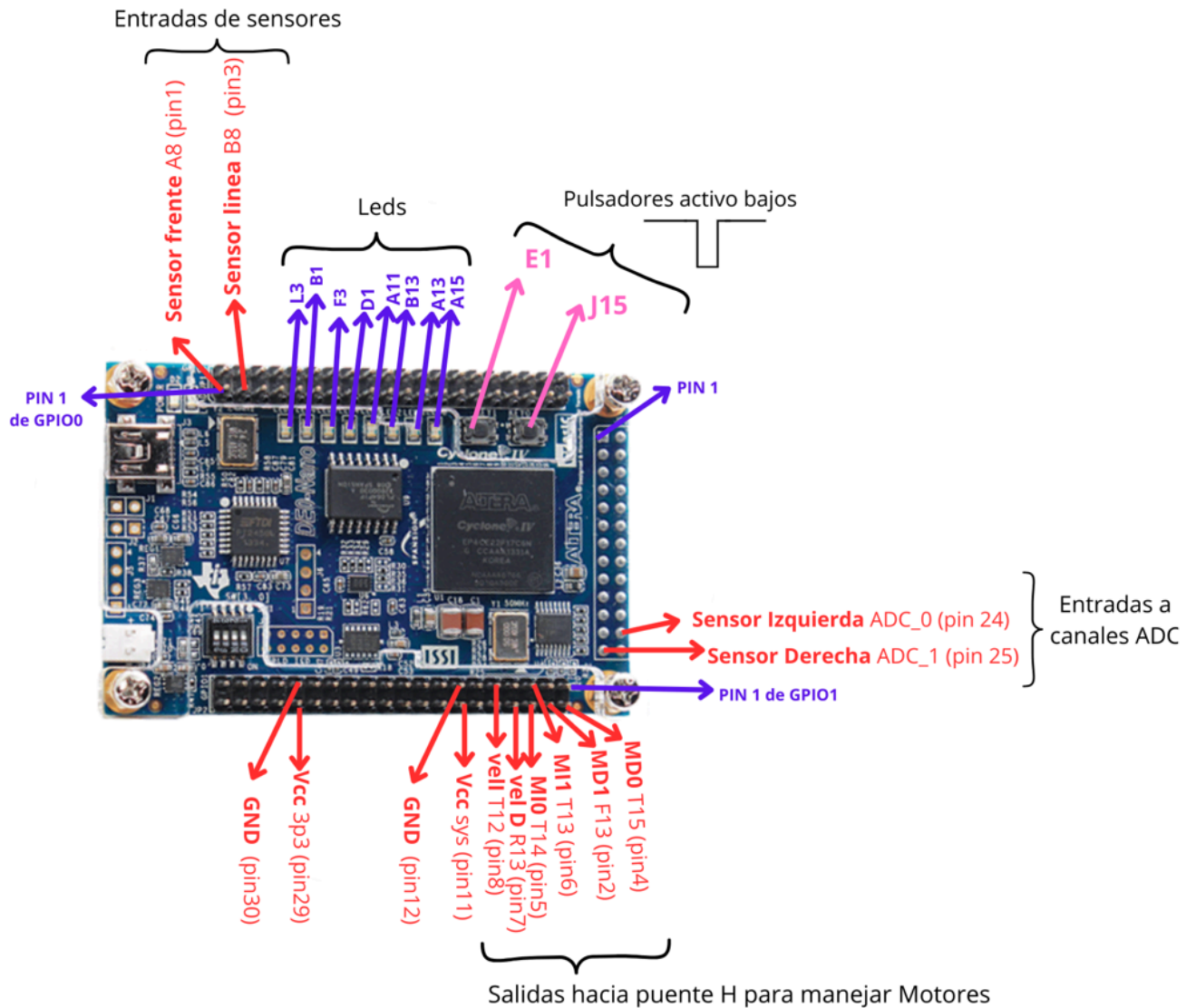


Fig. 2 Conexiones de la FPGA

Motores		Pin Planner
Cada motor se maneja con 3 líneas de entrada. VER TABLA 1		
velD	Velocidad del motor derecho	R13
MD1	Motor derecho	F13
MD0	Motor derecho	T15
velI	Velocidad del motor izquierdo	T12
MI1	Motor izquierdo	T13
MI0	Motor izquierdo	T14

vel	M1	M0	
Velocidad del motor, con 5V máxima velocidad, sino se le debe generar una señal PWM con frecuencia hasta 40kHz y según el ciclo de trabajo varía la velocidad.	1	0	Avanza
	0	1	Retrocede
	0	0	Frena
	1	1	Frena

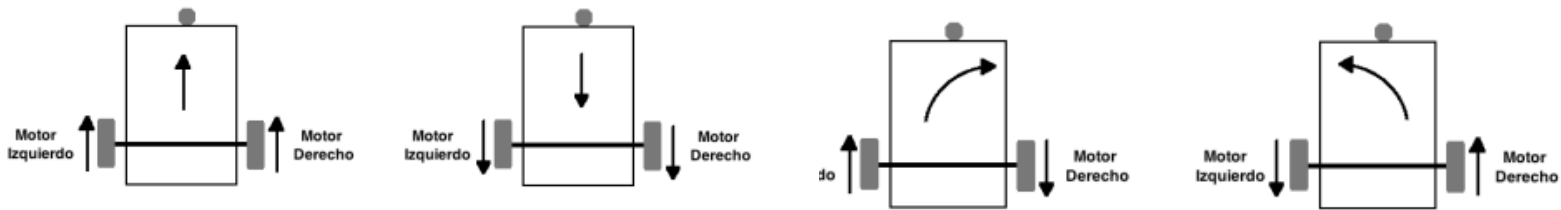


Fig. 3 Comandos motores

Sensores laterales					
Entradas analógicas					
Sensor Derecha	Entrada analógica que toma el canal 1 del ADC de la FPGA, 12 bits. Si no detecta obstáculo entrega aprox el valor 4000, detecta pared entrega aprox 400				
Sensor Izquierda	Entrada analógica que toma el canal 0 del ADC de la FPGA, 12 bits. Si no detecta obstáculo entrega aprox el valor 4000, detecta pared entrega aprox 400				

Ver cómo emplear el ADC de la FPGA:

<https://docs.google.com/document/d/1hhlUyFFCvPkUnULyCUsoKFMRIp0JCCRJ5ONtz2cIEPI/edit?usp=sharing>



Fig. 4 Sensor lateral derecho.

Sensor Línea					
Activo bajo					
Sensor línea	Sensor digital que entrega un 1 cuando detecta blanco y un 0 cuando detecta negro				
En el Pin planner del Quartus va B8					

Sensor Frente					
Activo alto					
Sensor frente	Sensor digital que entrega un 1 cuando detecta un obstáculo, sino entrega un 0.				
En el Pin planner del Quartus va A8					

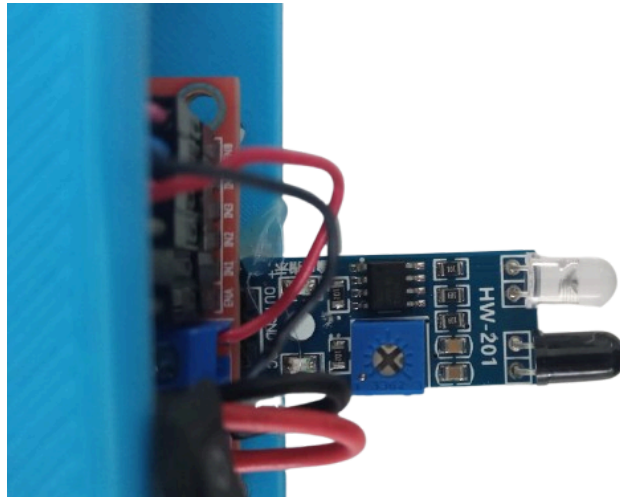


Fig.5 Sensor frente (detecta obstáculo).