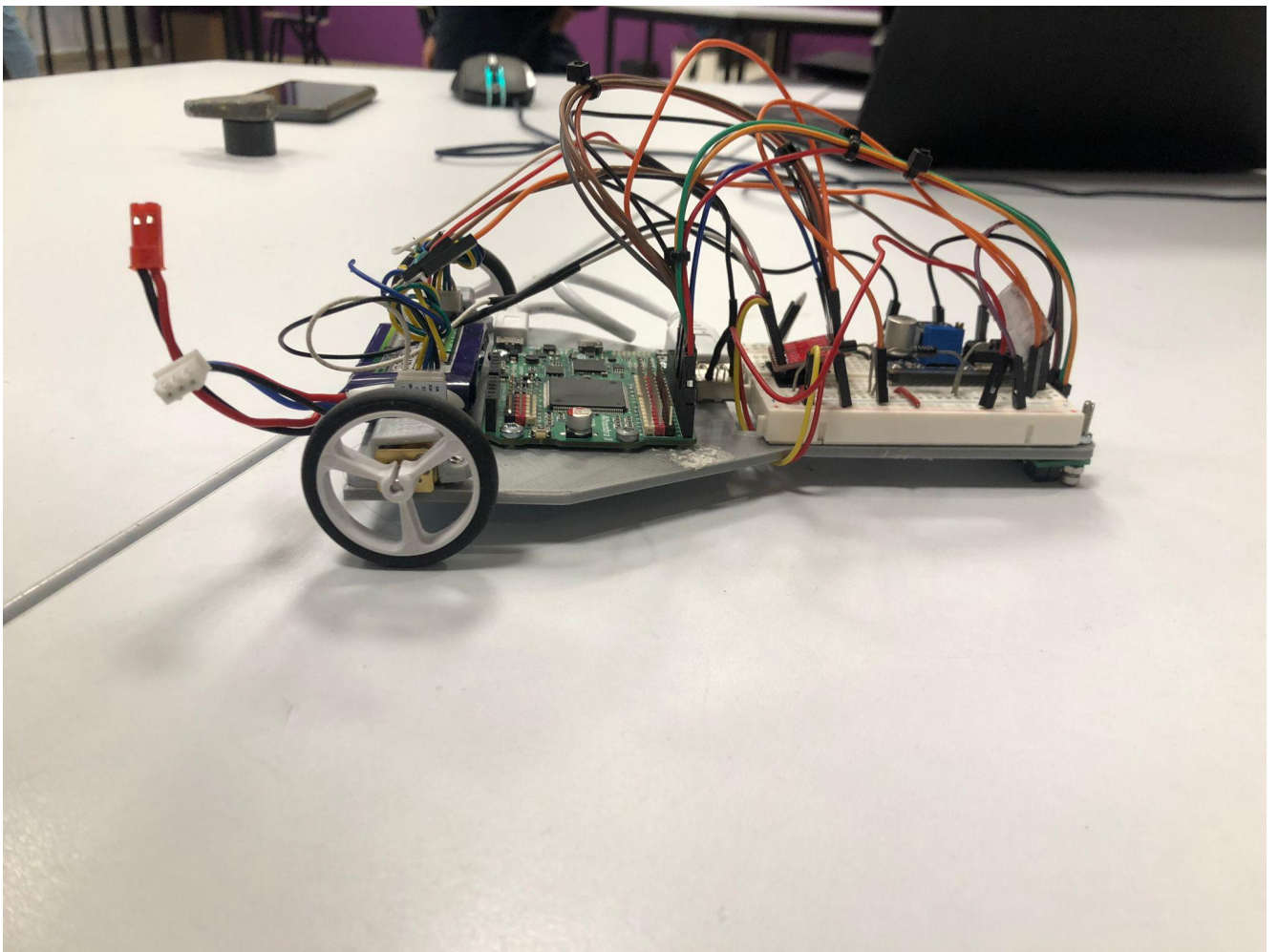


GUÍA DE MONTAJE

coche sigue-líneas



Creadores: Julen A., Iker G., Ander P.

0. Introducción:

Este documento es una guía que se puede utilizar para llevar a cabo el diseño, montaje y programación de un coche sigue-líneas. El objetivo de esta guía es el no necesitar ningún tipo de respaldo adicional para llevar a cabo el montaje del coche, con este propósito, esta guía contiene la siguiente información:

- Objetivo: En este punto, vamos a definir el proyecto que vamos a realizar.
- Lista de elementos: En este punto, vamos a listar todos los elementos que vamos a necesitar para poder realizar el montaje, así como añadir información necesaria de cada uno de ellos.
- Pasos a seguir: En este punto, vamos a empezar con la “guía” para el diseño, montaje y programación del coche.

Una vez sabemos el contenido del archivo, vamos a empezar con un índice, para luego comenzar con la guía.

0.1. Índice

0. Introducción:	1
0.1. Índice	1
1. Objetivo:	2
2. Lista de elementos:	2
2.1. Motores de CC:	2
2.2. Batería:	3
2.3. Sensores:	3
2.4. Alhambra II:	3
2.5. Driver:	3
2.6. Diodos:	4
2.7. DC/DC:	4
2.8. Cable USB:	4
2.9. Cables de conexión:	4
2.10. Chasis:	4
3. Pasos a seguir:	5
3.1. Diseño y esquemas:	5
3.2. Montaje del prototipo:	6
	1

1. Objetivo:

El objetivo de esta guía es llevar a cabo el diseño, montaje y programación de un coche sigue líneas. Para poder empezar con la guía, lo primero que haremos será un listado de los elementos que vamos a necesitar, para, una vez tengamos la lista de elementos, poder empezar con el diseño del coche.

Lo primero que tenemos que hacer es saber cómo funcionan estos coches: este tipo de proyectos consisten en hacer girar dos ruedas, propulsadas por dos motores, en diferentes direcciones dependiendo de un circuito que el coche detectará gracias a unos sensores colocados en la parte de delante del vehículo.

Para poder hacer el control de las ruedas dependiendo de los sensores se pueden usar diferentes tipos de elementos, nosotros hemos optado por usar un “FPGA board”, en nuestro caso será el Alhambra II.

Además de la Alhambra II, vamos a necesitar un “driver” para poder controlar correctamente los motores.

El vehículo necesitará alimentación, para llevar a cabo esta función vamos a usar una batería, la cual estará directamente en el coche.

Ahora que ya sabemos qué es lo que vamos a necesitar, vamos a hacer un listado con todos los elementos, ofreciendo algo de información y especificando el uso que le vamos a dar.

2. Lista de elementos:

Ahora vamos a realizar el listado de elementos que vamos a necesitar:

- 2 motores CC.
- 1 batería.
- 4 sensores.
- 1 Alhambra II.
- 1 driver para los motores.
- 3 diodos.
- 1 DC/DC.

- 1 cable tipo USB.
- Cables de conexión.
- 1 placa board.
- 1 chasis.

Estos van a ser los elementos de electrónica que vamos a necesitar, además de estos elementos, vamos a necesitar un ordenador para llevar a cabo la programación.

2.1. Motores de CC:

Los motores que vamos a utilizar son dos motores de corriente continua (CC) los cuales necesitan una alimentación de 6V para poder funcionar correctamente, podemos alimentarlos a más voltaje, pero no sería lo óptimo.

2.2. Batería:

La batería que vamos a utilizar es una batería que nos ofrece 7,4 - 11 V, nosotros usaremos los 7,4 V. Siempre tendremos en cuenta el voltaje que nos da la batería, tanto para hacer los cálculos como para evitar que se estropee. En nuestro caso la batería nos da un voltaje de 8,14 V, tendremos el voltaje en cuenta a la hora de hacer el resto del circuito.

2.3. Sensores:

Necesitamos sensores “infrarrojos”, los cuales nos permitirán saber si la superficie es clara o oscura, esto nos permitirá seguir el circuito.

Los sensores que vamos a utilizar nosotros son los CNY70, estos sensores tienen un led de infrarrojos y un fototransistor. Estos sensores funcionan enviando haces de luz desde el led infrarrojo y recibiendo los en el fototransistor.

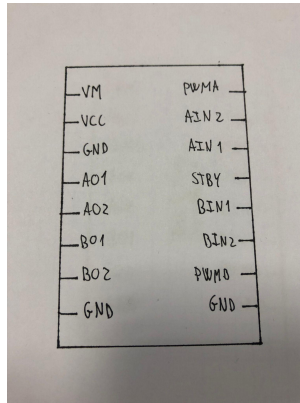
2.4. Alhambra II:

La Alhambra II va a ser el cerebro de nuestro coche, es un FPGA board el cual podemos programar usando el ordenador. El FPGA contiene, entre otras cosas, puertas lógicas, con lo que podremos programar las puertas lógicas para poder hacer que el coche funcione como queramos.

Para programar el FPGA nosotros vamos a utilizar el programa **Icestudio**, este programa es de libre acceso por lo que cualquiera puede descargarlo, así como todos los elementos de la Alhambra II.

2.5. Driver:

Este elemento nos va a permitir controlar los motores usando los pines del FPGA, este elemento sirve para realizar de una forma sencilla el control de uno o dos motores. Este elemento contiene dos etapas MOSFET con un conexionado en H. Nos permite controlar de forma independiente cada motor, dirección y velocidad de giro.



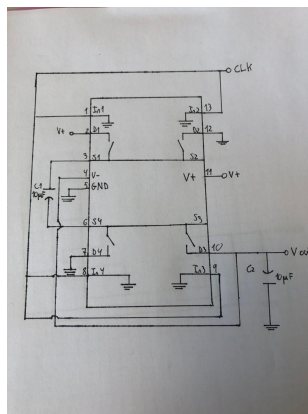
En este dibujo hemos representado las entradas y salidas del driver.

2.6. Diodos:

Estos diodos son elementos que vamos a utilizar para adecuar la tensión que nos da la batería a los diferentes elementos que tenemos en el coche.

2.7. DC/DC:

Junto a los diodos, vamos a utilizar un DC/DC para adecuar la tensión que vamos a llevar a cada elemento, en este caso usaremos el DC/DC para la alimentación de la Alhambra, dejando protegida la Alhambra.

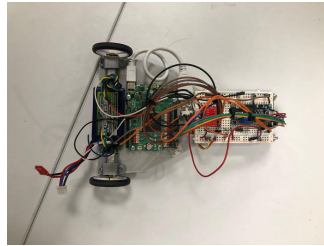


2.8. Cable USB:

Este cable es el que vamos a utilizar para comunicar la Alhambra con el ordenador así como para alimentar la Alhambra.

2.9. Cables de conexión:

Son los cables que vamos a utilizar para realizar las conexiones que nos hacen falta.



2.10. Chasis:

Este chasis está diseñado e impreso en una impresora 3D.



3. Pasos a seguir:

Lo primero que vamos a hacer es un listado de los pasos a seguir:

- Diseño del prototipo y esquemas.
- Montaje del prototipo.
- Programación.
- Pruebas.

Como ya sabemos qué material vamos a utilizar no es necesario buscarlo, y lo primero será diseñar y hacer los esquemas de las conexiones.

3.1. Diseño y esquemas:

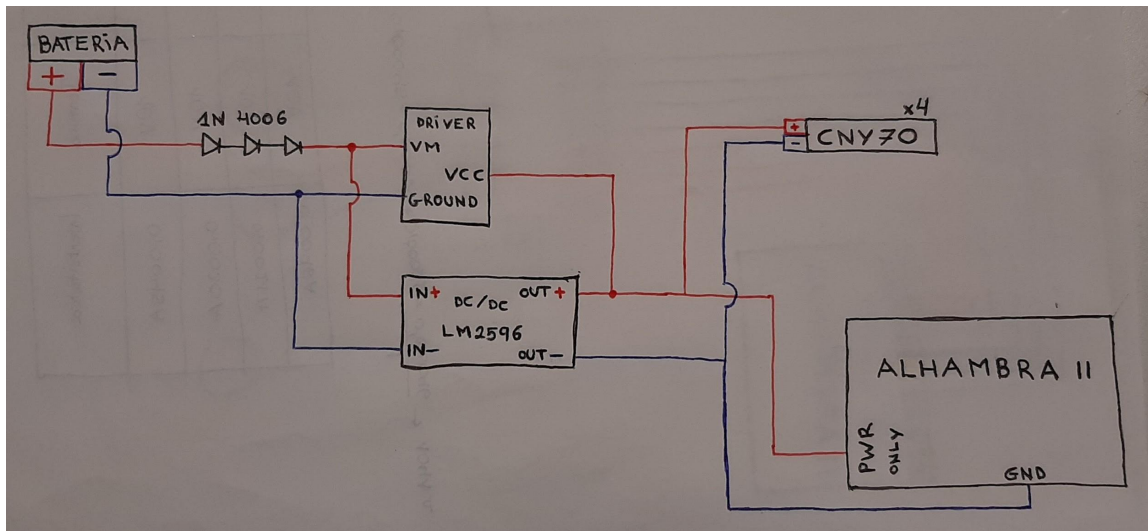
Lo primero que vamos a hacer es el diseño/esquema de la alimentación del coche. Para eso, lo primero que vamos a mirar es la tensión de alimentación de cada uno de los componentes:

- Motores: 6V
- Batería (salida): 7.4/11V
- Sensores: 5V
- Alhambra: 5V
- Driver: VM 6V (alimentación de los motores), VCC 5V (alimentación del driver)

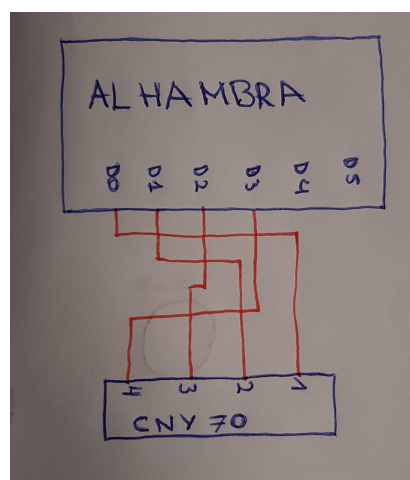
El resto de elementos no necesitan alimentación.

Sabiendo la tensión de alimentación de cada componente vamos a usar los diodos para bajar la tensión de la batería hasta alrededor de los 6V, para esto usamos 3 diodos en serie donde cada uno tiene una caída de tensión de 0.7V, haciendo que la caída de tensión total sea de 2.1V. Luego usamos el DC/DC para dar una tensión de salida de 5V estables.

Con esto el esquema de alimentación quedaría algo así:

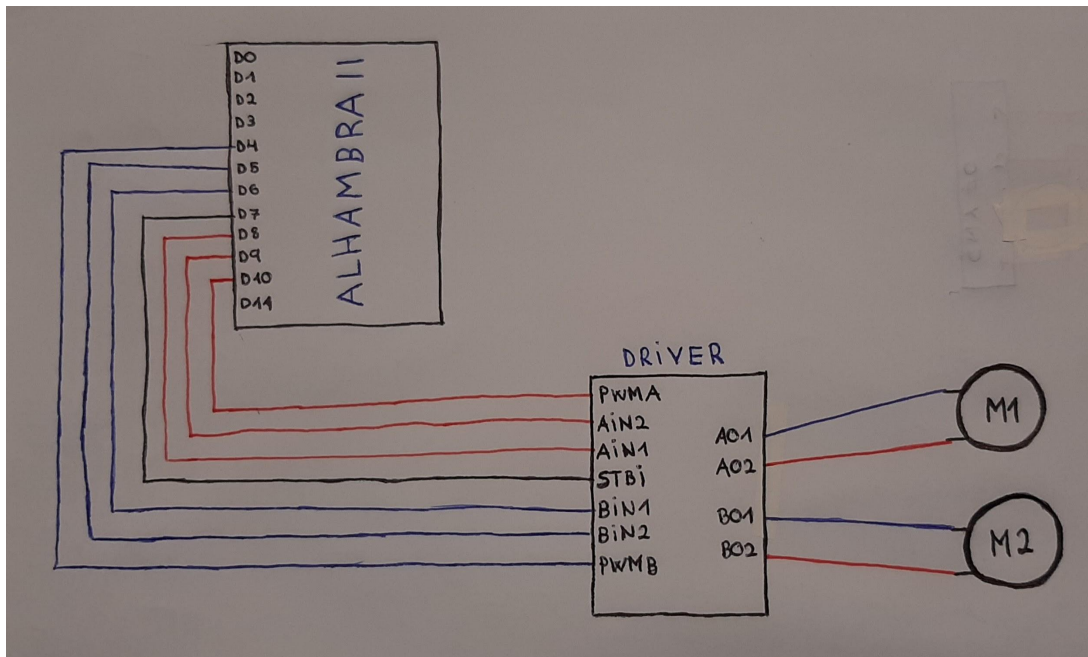


Ahora vamos a hacer el esquema de conexión de los sensores:



Los pines de la Alhambra los hemos elegido nosotros, no es necesario usar los mismos.

Para terminar, el esquema que nos queda es el de conexión de los motores:



Al igual que antes los pines no tienen porqué ser los que hemos puesto nosotros, pero tenemos que tener en cuenta para qué sirve cada uno de ellos, dado que luego los tenemos que tener en cuenta a la hora de programarlos.

3.2. Montaje del prototipo:

En este paso lo único que hay que hacer es seguir los esquemas que hemos hecho en el paso anterior, en nuestro caso el coche ha quedado de la siguiente manera:

3.3. Programación:

Para la programación, lo primero que vamos a hacer es la tabla de la verdad de nuestro coche, para eso tenemos que identificar las entradas y salidas que necesitamos, en nuestro caso hemos identificado 4 entradas y 6 salidas:

- Entradas:
 - Los cuatro sensores CNY70.

- Salidas:
 - A1 y A2, los dos sentidos del motor 1.
 - B1 y B2, los dos sentidos del motor 2.
 - PWMA y PWMB, la velocidad de giro de ambos motores.

La tabla de la verdad que hemos usado es la siguiente:

S1	S2	S3	S4	A1	A2	PWMA	B1	B2	PWMB
0	0	0	0	1	0	1	0	1	1
0	0	0	1	0	0	0	0	1	1
0	0	1	0	0	1	1	0	1	1
0	0	1	1	0	1	1	0	1	1
0	1	0	0	1	0	1	1	0	1
0	1	0	1	0	0	0	0	1	1
0	1	1	0	1	0	1	0	1	1
0	1	1	1	0	1	1	0	1	1
1	0	0	0	1	0	1	0	0	0
1	0	0	1	1	0	1	0	1	1
1	0	1	0	1	0	1	0	0	0
1	0	1	1	0	1	1	0	1	1
1	1	0	0	1	0	1	1	0	1
1	1	0	1	1	0	1	1	0	1
1	1	1	0	1	0	1	1	0	1
1	1	1	1	0	0	0	0	0	0

Donde los sensores están situados en el mismo orden que en el chasis, A1 representa al motor 1 avanzando y A2 retrocediendo, B1 representa al motor 2 retrocediendo y B2 avanzando, y PWMA y PWMB representan a los motores encendidos o apagados

Ahora usando el método de karnaugh vamos a sacar directamente la función simplificada de cada una de las salidas:

A1:

3 4 \ 1 2	00	01	11	10
00	1	1	1	1
01	0	0	1	1
11	0	0	0	0
10	0	1	1	1

$$A1 = (-3 \times -4) + (1 \times -3) + (2 \times -4) + (1 \times -4)$$

A2:

3 4 \ 1 2	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	1	0	1
10	1	0	0	0

$$A2 = (-1 \times 3 \times 4) + (-2 \times 3 \times 4) + (-1 \times -2 \times 3)$$

PWMA:

3 4 \ 1 2	00	01	11	10
00	1	1	1	1
01	0	0	1	1
11	1	1	0	1
10	1	1	1	1

$$PWMA = (1 + 3 + -4) \times (-1 + -2 + -3 + -4)$$

B1:

3 4 \ 1 2	00	01	11	10
00	0	1	1	0
01	0	0	1	0
11	0	0	0	0
10	0	0	1	0

$$B1 = (2 \times -3 \times -4) + (1 \times 2 \times -3) + (1 \times 2 \times -4)$$

B2:

3 4 \ 1 2	00	01	11	10
00	1	0	0	0
01	1	1	0	1
11	1	1	0	1
10	1	1	0	0

$$B2 = (-2 + 3 + 4) \times (-1 + 4) \times (-1 + -2)$$

PWMB:

3 4 \ 1 2	00	01	11	10
00	1	1	1	0
01	1	1	1	1
11	1	1	0	1
10	1	1	1	0

$$PWMB = (-1 + -2 + -3 + -4) \times (-1 + 2 + 4)$$

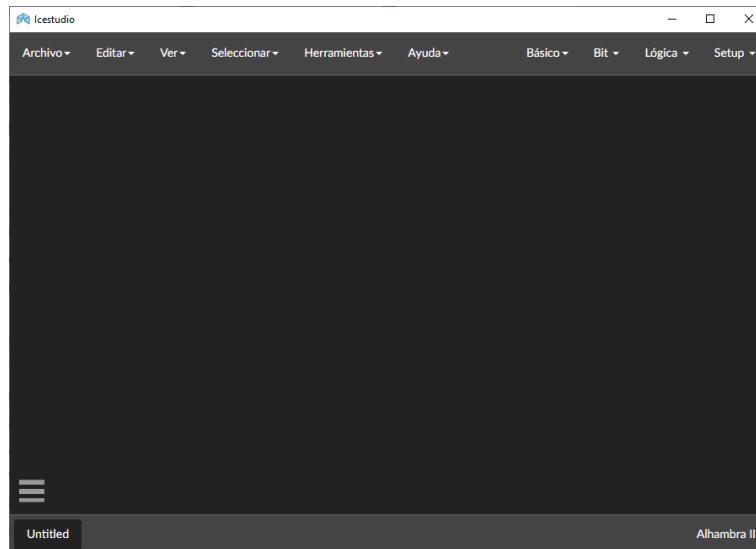
Ahora nos toca convertir estas funciones a circuitos lógicos, para eso vamos a utilizar el programa **lcestudio**.

3.4. Icestudio:

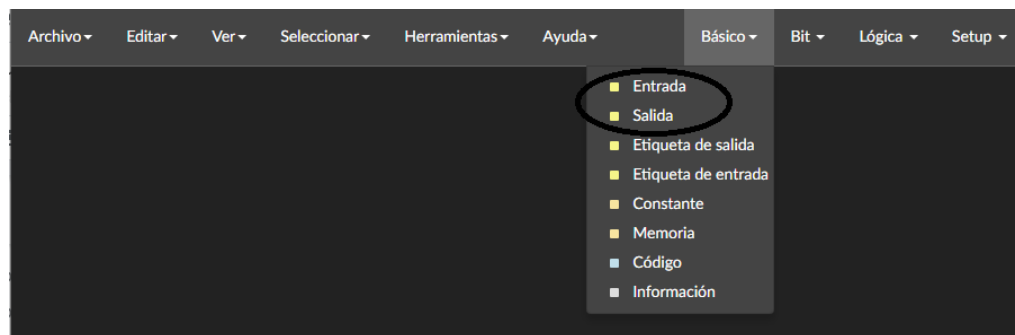
Como ejemplo vamos a usar la primera función que hemos conseguido:

$$A1 = (-3 \times -4) + (1 \times -3) + (2 \times -4) + (1 \times -4)$$

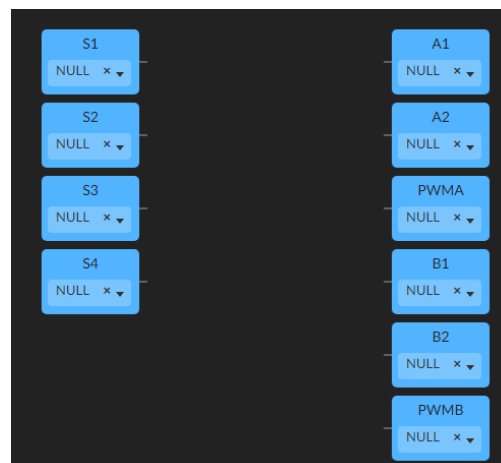
Al abrir el programa lo que vemos es una ventana como esta:



Ahora vamos a sacar las entradas y las salidas:

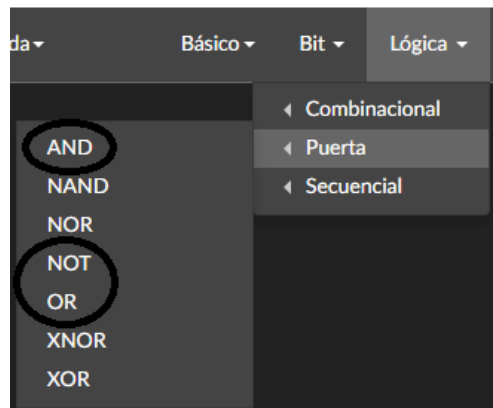


En nuestro caso las entradas y las salidas (todas) serían las siguientes:

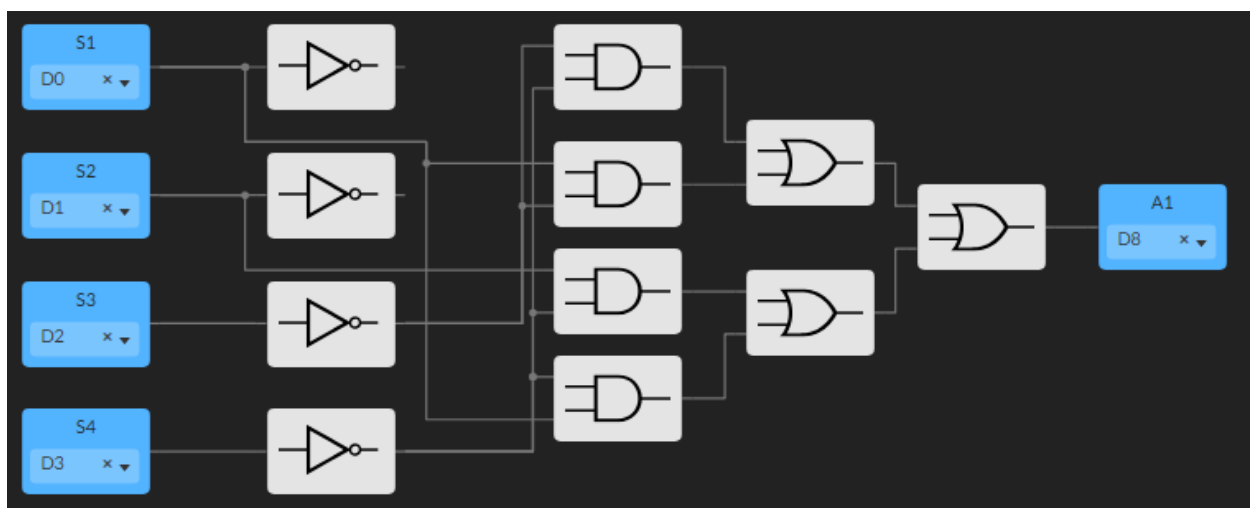


Ahora podemos elegir los pines en los que se encuentran cada una de las salidas/entradas, para esto solamente tenemos que clicar en la flecha que tenemos en cada uno de ellos.

Lo siguiente es ir sacando las puertas lógicas, nosotros vamos a usar tres tipos: NOT, AND y OR.



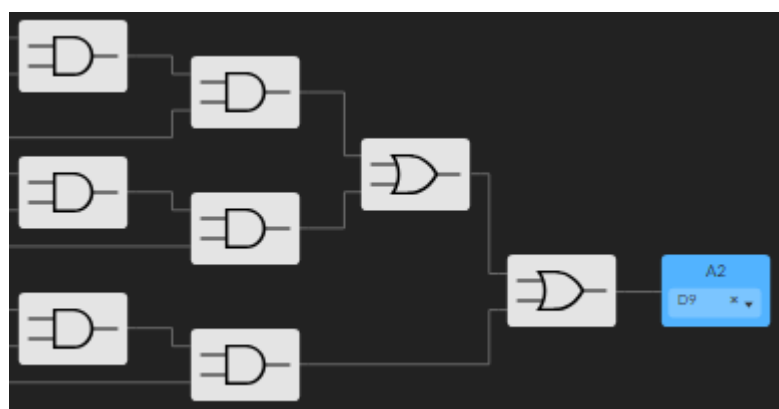
Teniendo en cuenta la función, y usando las puertas lógicas, el circuito lógico de la salida A1 sería el



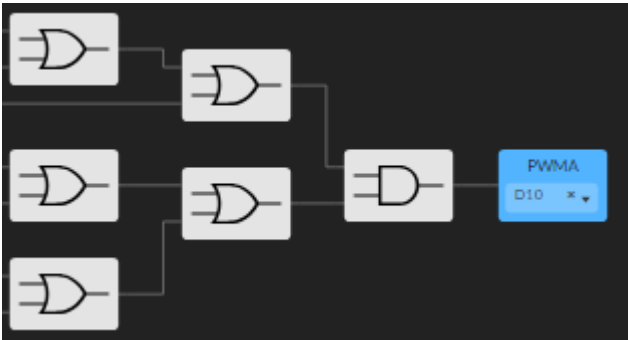
siguiente:

Siguiendo el ejemplo anterior, vamos a hacer el resto de las salidas:

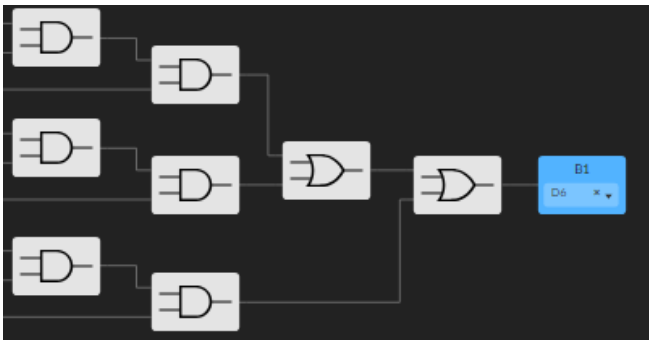
A2: $(-1 \times 3 \times 4) + (-2 \times 3 \times 4) + (-1 \times -2 \times 3)$



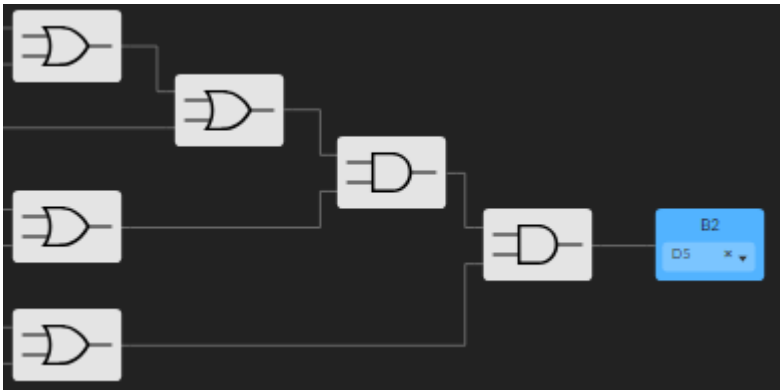
PWMA: $(1 + 3 + -4) \times (-1 + -2 + -3 + -4)$



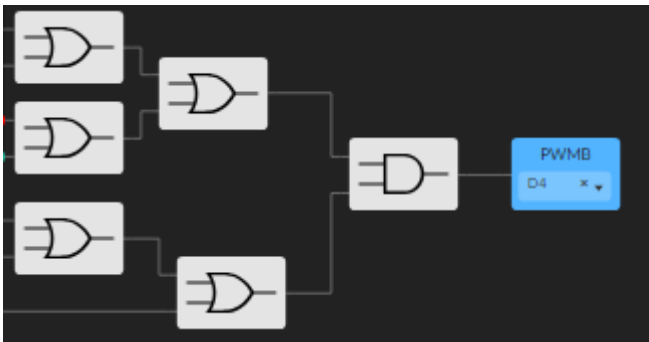
B1: $(2 \times -3 \times -4) + (1 \times 2 \times -3) + (1 \times 2 \times -4)$



B2: $(-2 + 3 + 4) \times (-1 + 4) \times (-1 + -2)$



PWMB: $(-1 + -2 + -3 + -4) \times (-1 + 2 + 4)$



El resultado debería ser un circuito lógico con todos los enseñados hasta ahora.

Junto con la guía hay un archivo que contiene apuntes para la iniciación en la electrónica y un vídeo con el resultado de nuestro montaje en el que se ve el funcionamiento del coche.