

# Autito

(Esquemáticos)

Programa Principal → Autito

Subprograma - 1 → Componente → Ruedas

Subprograma - 2 → Componente → Velocímetro

Programa - principal → Componente → Control

Cyclone IV E

FPGA → EP4CE22F17C6

1 clock → 50 MHz

PLL's

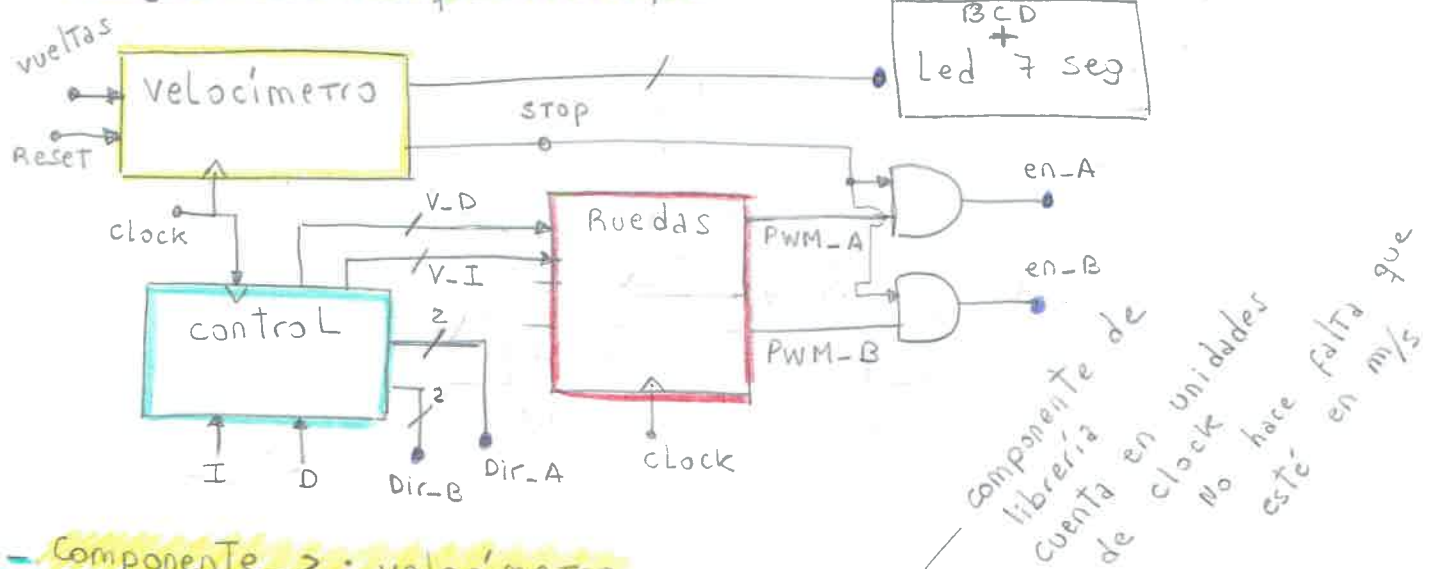
Multiplicadores

memoria RAM

In/OUT ~ 80

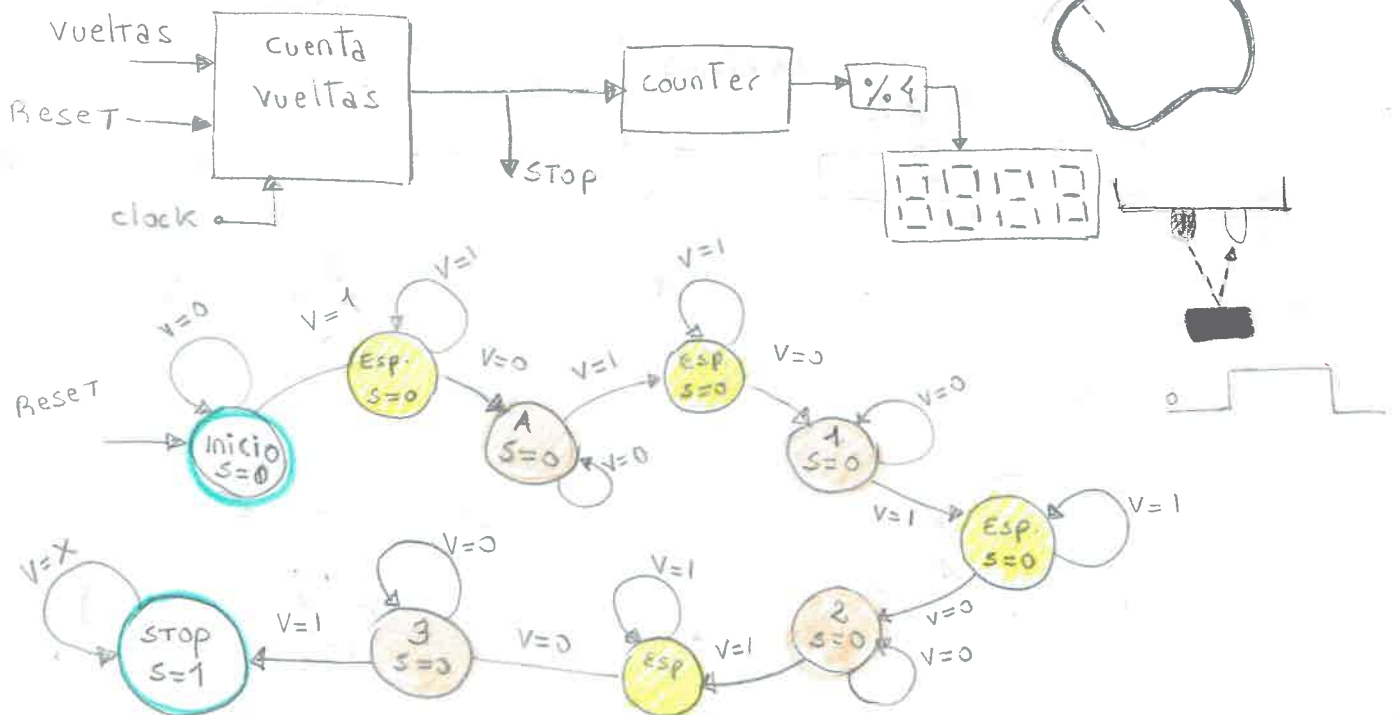
Voy a trabajar con diagramas en bloques.

## Diagrama en bloque Principal (Archivo principal)

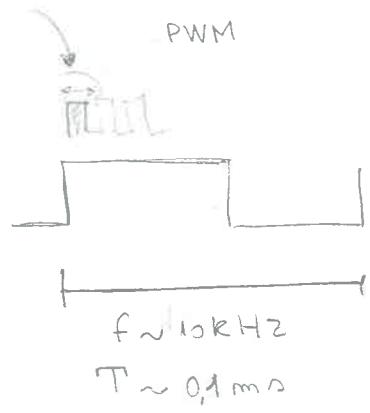
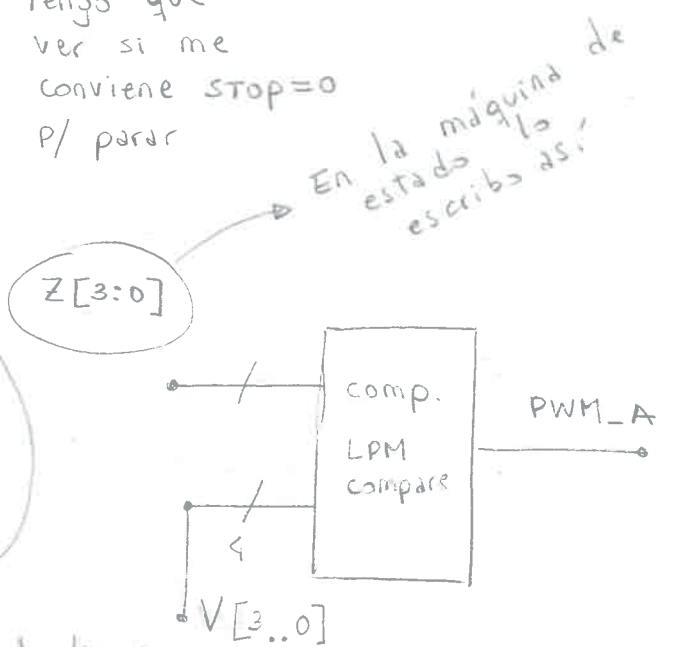
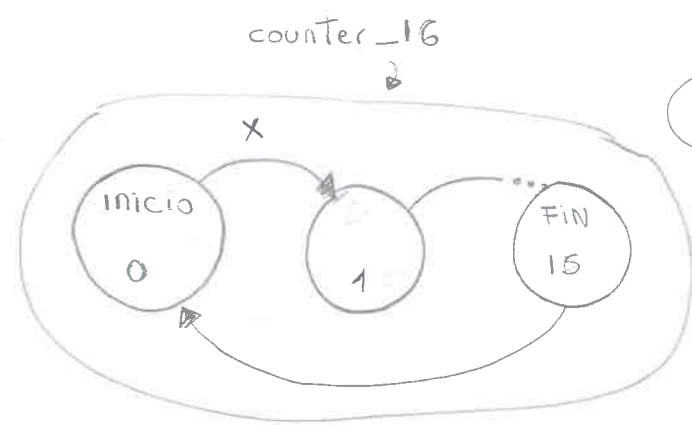
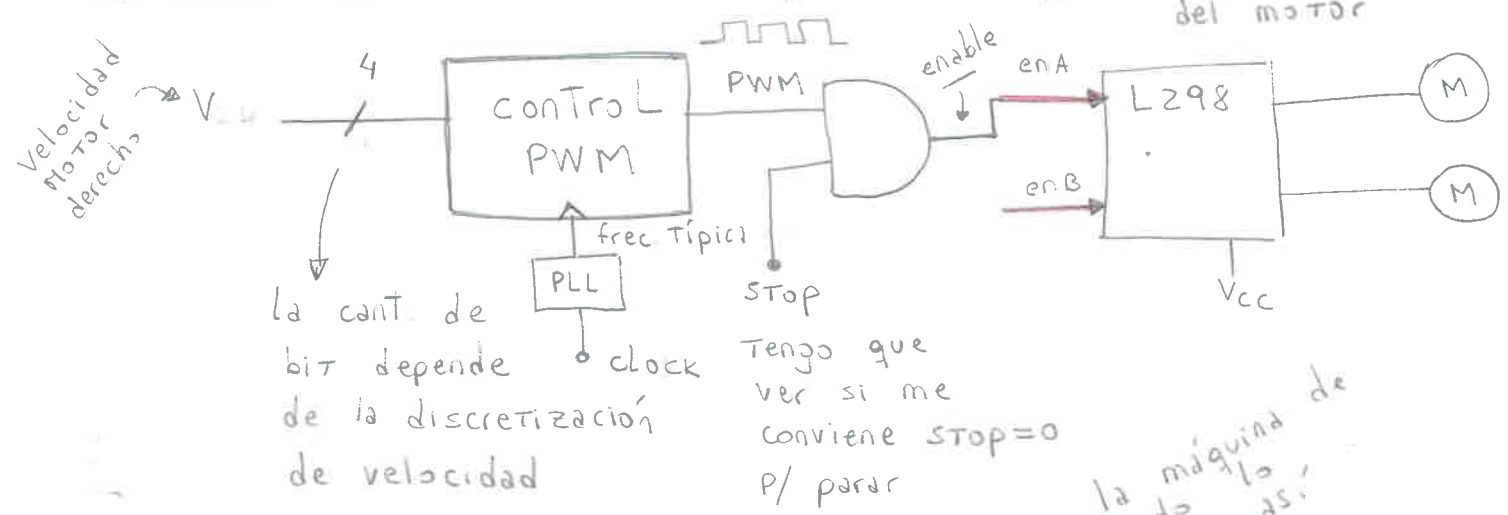


## Componente - 2: velocímetro

in: vueltas (sensor costado)  
Reset (pulsador)



**Componente 1: Puentes**



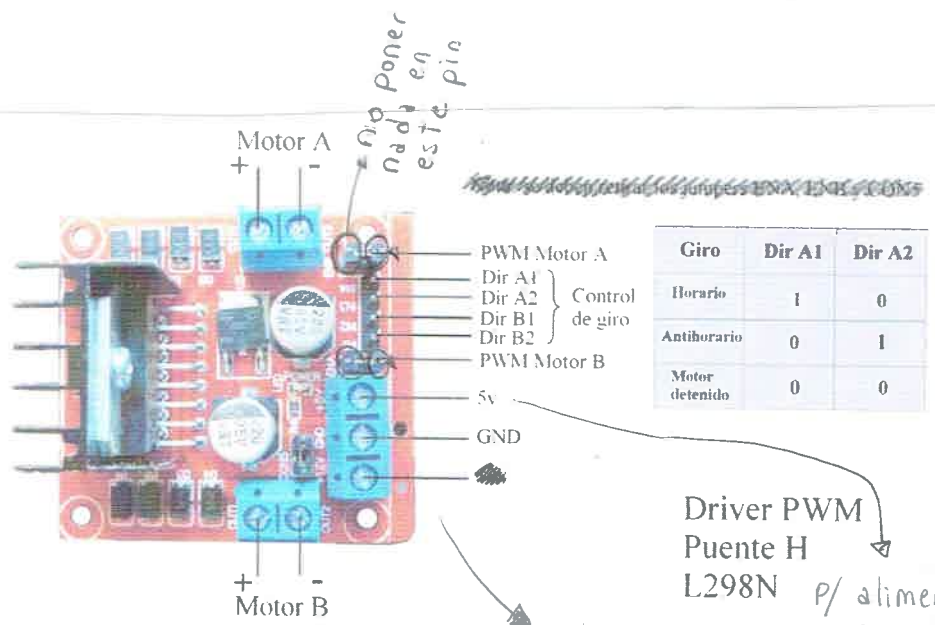
yo puse 4 bit  $\Rightarrow$  16 valores

$f_{\text{clock}} = 16 \cdot f_{\text{conmutación}}$

frec. Típica conmutación = 25 kHz PWM

Hay que ver cuál es la velocidad cruce

V-D P/ que el auto ande balanceado



Giro	Dir A1	Dir A2
Horario	1	0
Antihorario	0	1
Motor detenido	0	0

P/ este caso saco Jumper y alimento compuertas con FP6A (mejorar ruido)

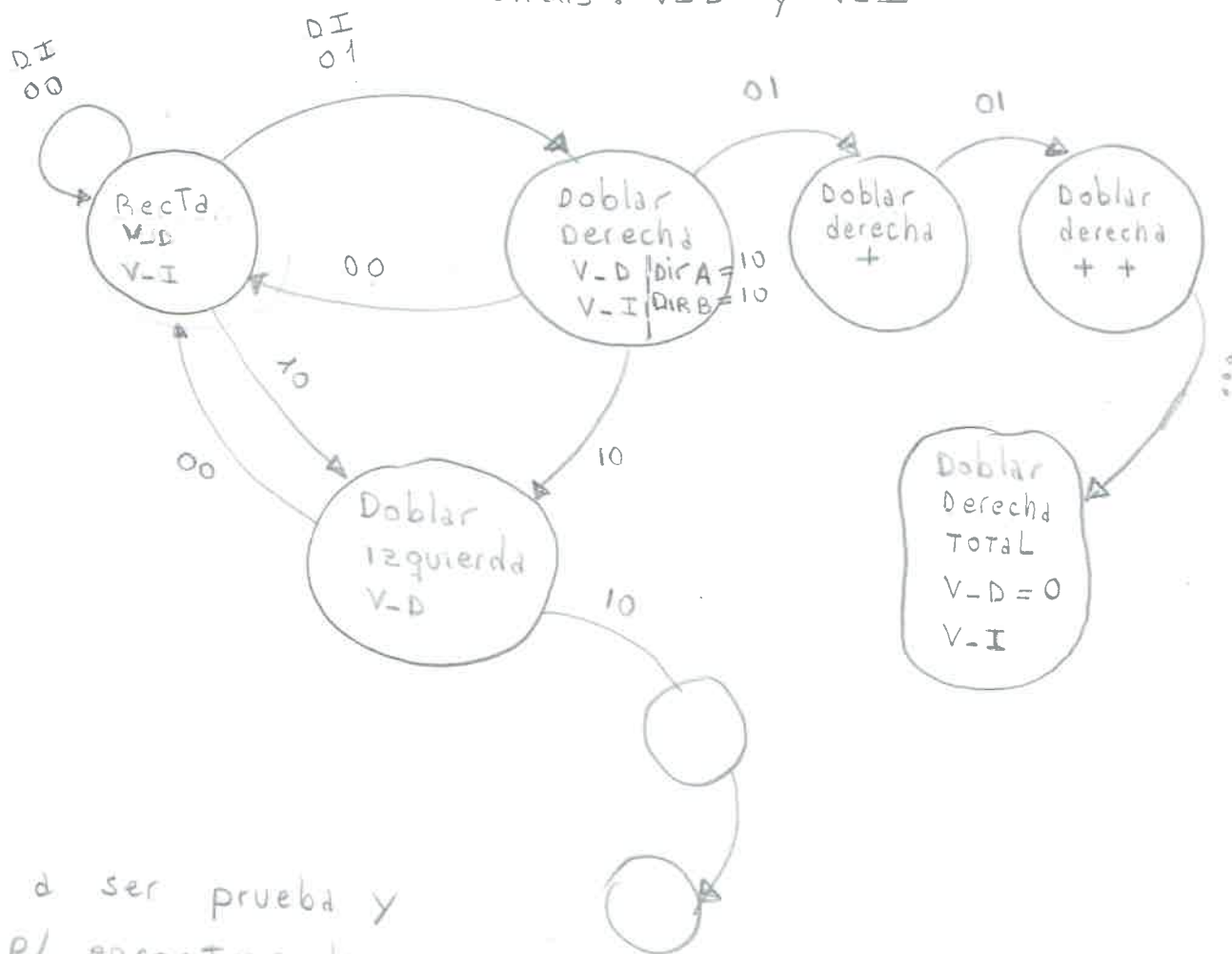
Alim. P/ los motores con Jumper puesto saca de ahí la alimentación de las compuertas internas

Driver PWM Puente H L298N P/ alimentar externamente compuertas internas del LM298. (Saco Jumper)

# - Componente 3 : control

in: I y D (sensores)  
salidas: V-D y V-I

(2)



Acá va a ser prueba y error p/ encontrar la mejor estrategia en cuanto a la velocidad de los motores

También se podría hacer marcha atrás si ambos son blancos  $\Rightarrow$  11  $\Rightarrow$  que de marcha atrás hasta que encuentre camino

Dir\_A = 10

Dir\_B = 10

se podría hacer que la rueda que debe bajar la velocidad vaya p/ el otro lado no se si va a ser mejor o peor.

sobre el sensor

I



D



Línea negra

Detecta Línea negra

⇒ cero

Detecta Línea  
blanca

⇒ uno