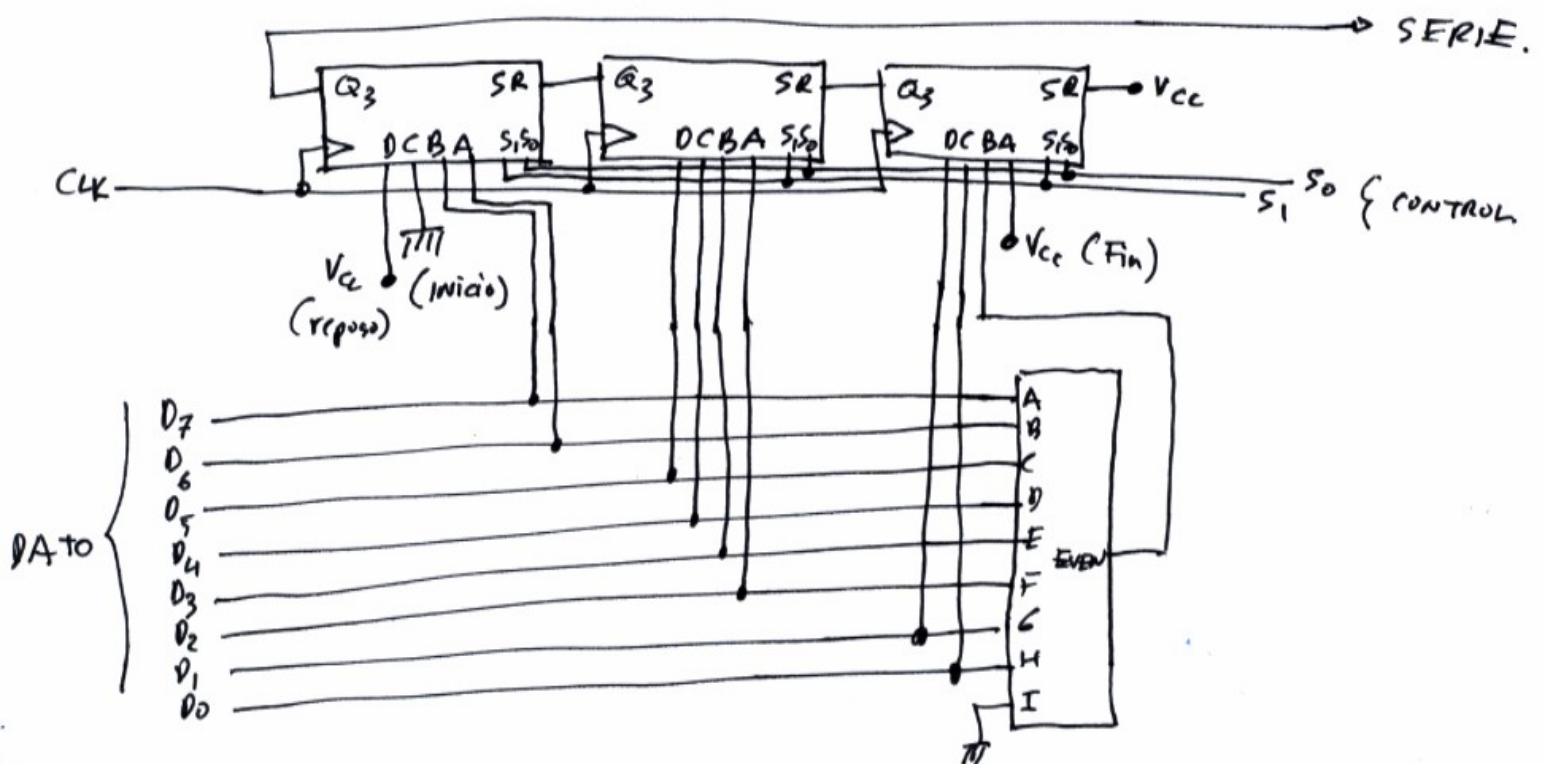


1) Por lo que dice el enunciado del problema vemos que tenemos que aceptar un dato en paralelo y transmitirlo en serie, para lo que necesitaremos un registro de desplazamiento con entrada serie y salida paralelo, además debemos añadirle el bit de inicio y el de parada (que se puede hacer sin problema con el mismo registro de desplazamiento) y un bit de paridad, para lo que necesitaremos un generador de paridad. Vamos a ver que dispositivos comerciales podemos usar:

74194 como registro de desplazamiento. Es de 4 bits y necesitamos 11 (8 de datos, inicio, FIN, PARIDAD), así que necesitaremos 3.

74280 como generador de paridad

La arquitectura quedará así:



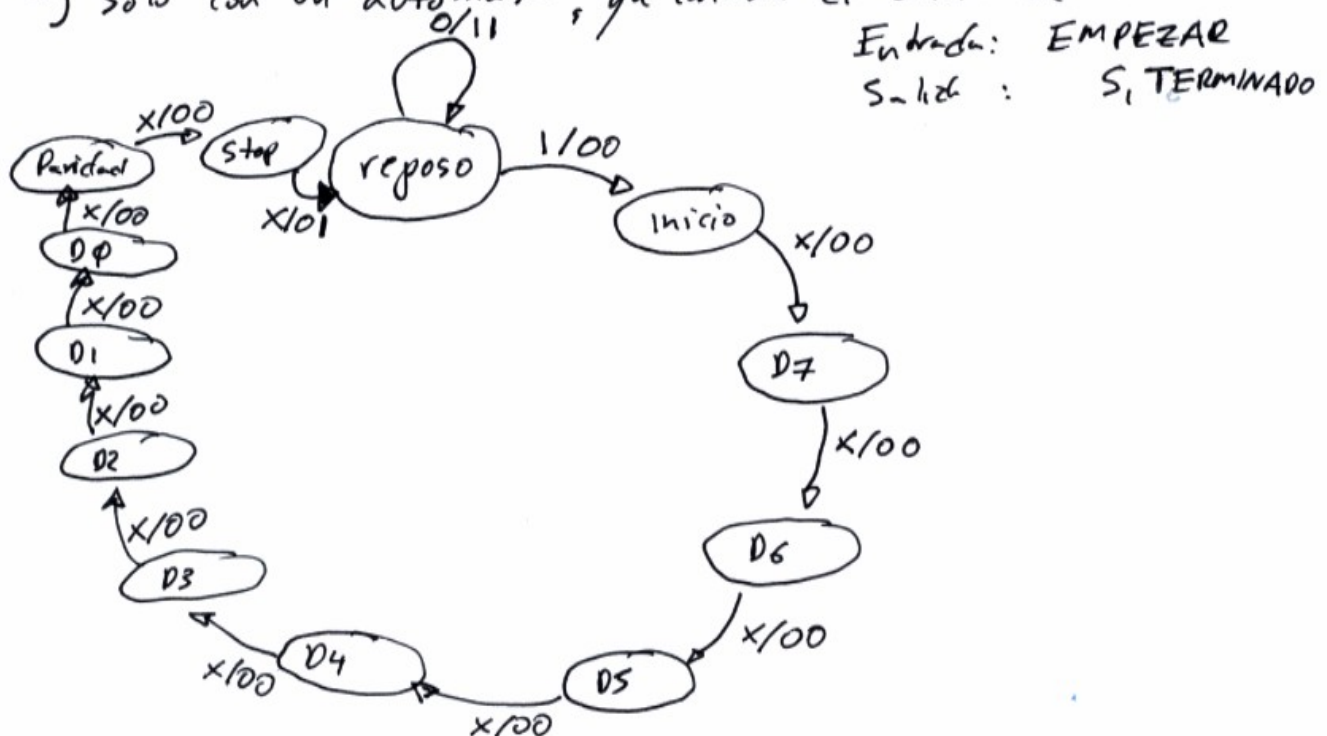
Con S_1, S_0 se controla si se carga el dato, se desplaza (para sacarlo en serie) o se mantiene al final el 1 (línea en reposo). Veamos como será necesario generar esas señales de control:

- En reposo (antes de que se active EMPEZAR) podemos activar $S_1, S_0 = 11$ de forma que se active la carga paralelo de forma continua. De esa forma por la salida serie estará saliendo siempre un 1, que es lo que carga D.
- Al activarse EMPEZAR activaremos $S_1, S_0 = 01$ de forma que se desplace a la izquierda, para que vayan saliendo los datos (11 en total) (a partir del próximo flanco de reloj).
- Una vez transmitidos los 11 bits se vuelve a reposo.

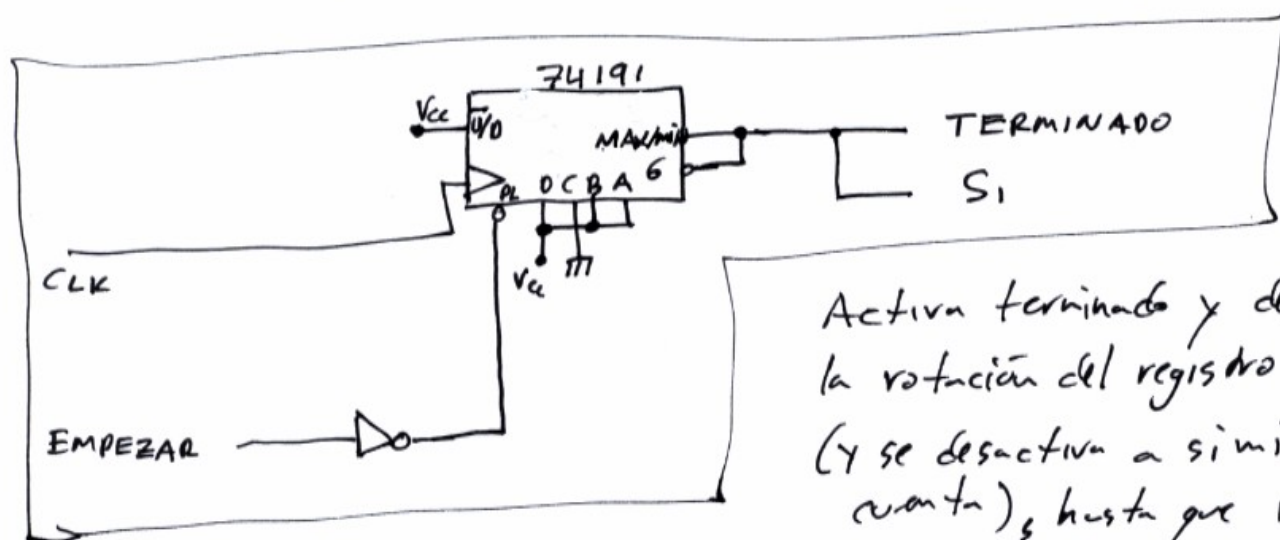
Vemos que $S_0 = 1$ siempre.

El sistema de control podrá diseñarse de dos formas

a) Sólo con un autómata, que controle el envío de los 11 bit.



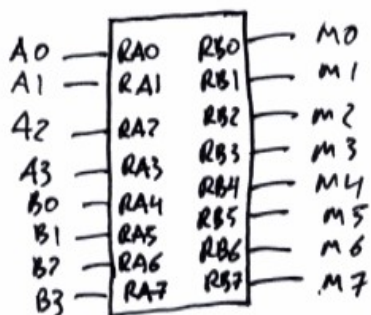
b) Se podría incluir un contador que llevara la cuenta de los 11 bits, por ejemplo, con un 74191: se inicializa a 11 cargando en paralelo con el pulso EMPEZAR (empezar debe volver a nivel bajo). En el siguiente pulso de reloj empiezan a contar hacia abajo. Al llegar a 0



recibir otro pulso en EMPEZAR.

Activa terminado y desactiva la rotación del registro (y se desactiva a su vez la cuenta), hasta que vuelva a

2) El esquema del circuito será el siguiente:



Como estoy utilizando como entrada RA5 no debo tener activado el pin MCLR.

Como estoy usando como entradas RA6 y RA7 tendré que configurar el PIC para usar reloj interno

El programa en pseudocódigo será:

INICIO

CONFIGURAR PORTA ENTRADA DIGITAL

CONFIGURAR PORTB SALIDA DIGITAL

REPETIR

A = PARTE BAJA DE PORTA

B = PARTE ALTA DE PORTA

CONTADOR = B
 RESULTADO = 0
 REPETIR

RESULTADO = RESULTADO + A

CONTADOR = CONTADOR - 1

HASTA QUE RESULTADO == 0

PORTB = RESULTADO

SIEMPRE

FIN

El desarrollo del programa en ensamblador será el siguiente:

INCLUDE "PIC16F88.H"

--CONFIG CONFIG1, _MCLR_OFF & _INTRC_IO & _LVP_OFF

DATO A EQU 0x20

DATO B EQU 0x21

CONTADOR EQU 0x22

RESULTADO EQU 0x23

ORG 0

GOTO INICIO

ORG 4

INICIO:

BSF STATUS, RPO ; banco 1

CLRF ANSEL ; todos los pines digitales

CLRF TRISB ; PORTB salida

MOVLW 0xFF

MOVWF TRISA ; PORTA entrada.

BCF STATUS, RPO ; banco 0.

REPETIR: MOVLW 0x0F ; máscara para leer B

ANDWF PORTA, W ; lee el puerto A eliminando la
 ; parte baja.

MOVWF DATOB

SWAPF DATOB ; cambia el dato a la parte baja.

MOVLW 0xF0 ; máscara para leer A

ANDWF PORTA, W ; lee el PORTA eliminando la

MOVWF DATOA ; parte alta

MOVFW DATOB

MOVWF CONTADOR ; CONTADOR = B

CLRF RESULTADO

REPETIR2:

MOVFW DATOA

ADDWF RESULTADO ; RESULTADO = RESULTADO + A

DECFSZ CONTADOR ; CONTADOR = CONTADOR - 1

GOTO REPETIR2 ; REPITE HASTA QUE
; CONTADOR == 0

MOVFW RESULTADO

MOVWF PORTB ; PORTB = RESULTADO

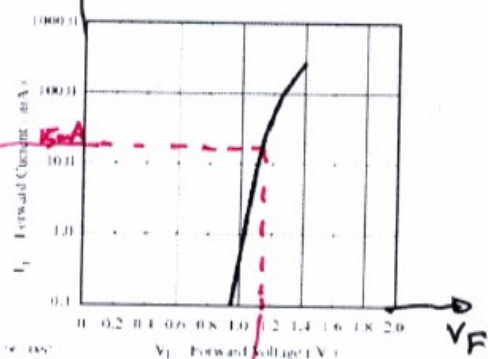
GOTO REPETIR

END.

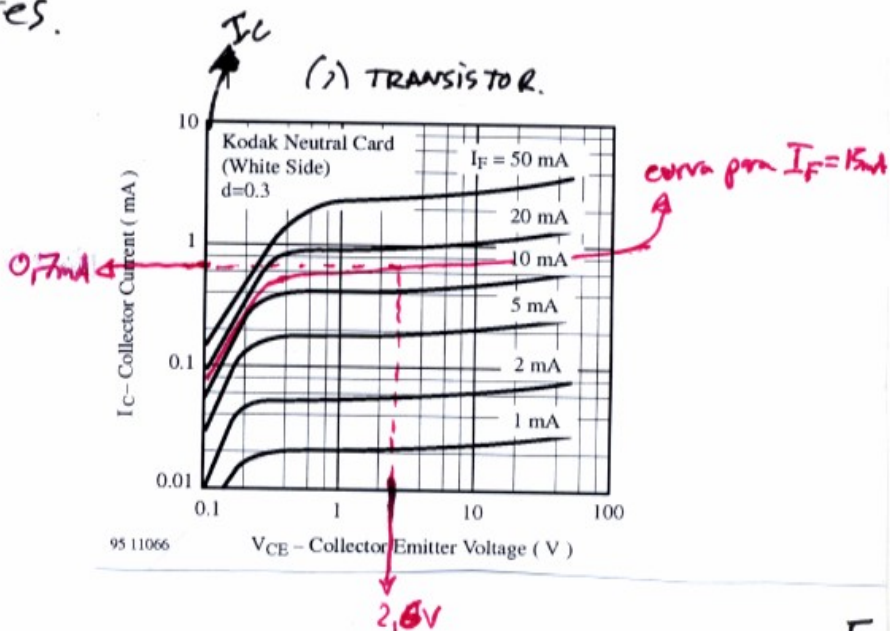
El programa lee de forma continua A y B, haciendo una máscara AND del puerto A con 00001111 para leer A y con 11110000 para leer B, y hace la multiplicación a base de sumar B veces A para sacarlo después por el puerto.

3) • Circuito de los sensores.

I_F (1) DIODO



(2) TRANSISTOR.



El diodo se polarizará según la curva 1. Para una corriente en directo de 15mA vemos que la tensión debe ser de 1,1V. Por lo que en la resistencia R_D habrá una caída de tensión $V_{RD} = V_{CC} - 1,1V$. Si $V_{CC} = 5V \Rightarrow V_{RD} = 3,9V$

$$\gamma \quad R_D = \frac{V_{RD}}{15mA} = 260 \Omega$$

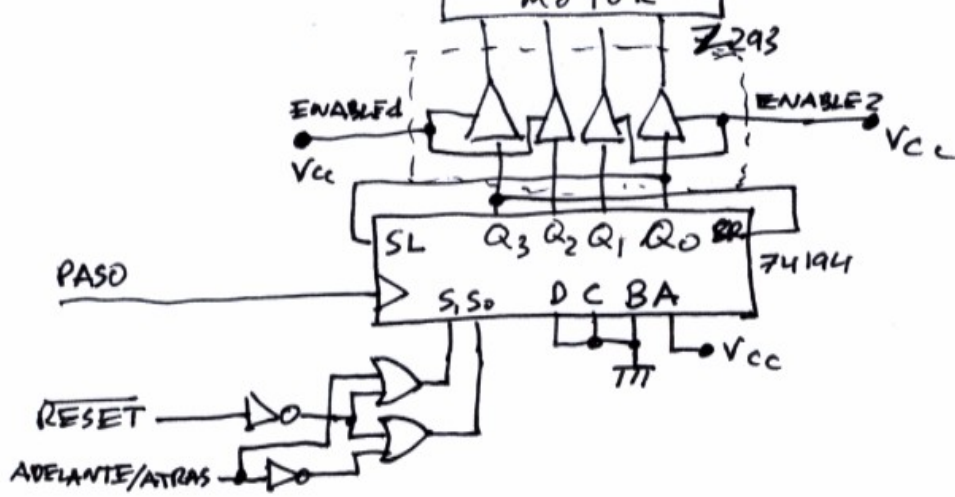
Como necesitamos un 0 para negro (el transistor no conduce) y un 1 para blanco (el transistor conduce). Tendremos que usar el montaje a). El transistor se polariza según la curva 2, para $I_F = 15mA$. Si queremos que V_{SALIDA} sea un 1 cuando el transistor conduce será necesario que $V_{SALIDA} > ?$
 $\Rightarrow V_{CE} < 5 - 2,4V \Rightarrow V_{CE} < 2,6V \Rightarrow$ de las curvas se deduce $I_C < 0,7mA$.

La resistencia límite, para $V_{CE} = 2,6V$, $I_C = 0,7mA$

$$\text{será} \quad R_T = \frac{5 - 2,6V}{0,7mA} = 3,43K. \quad (\text{valor máximo para } R_T)$$

• Circuito de los motores

Se quiere por un lado que el motor se pueda parar y por otro que pueda ir adelante/atrás. Esto se va a conseguir en cada motor ~~seguir~~ con un contador en anillo reversible que se puede hacer fácilmente con un 74194 así:



Función	S ₁	S ₀
Reset	1	1
Adelante	1	0
Atrás	0	1
Parado	0	0

Para generar S₁S₀ lo haremos a partir de la señal RESET (aunque no está en el diagrama de bloques siempre habrá un reset).

Y ADELANTE / ATRAS

Función	ENTRADAS		SALIDAS	
	ADELANTE/ATRAS	RESET	S ₁	S ₀
Reset	X	0	1	1
Adelante	1	1	1	0
Atras	0	1	0	1
Parado	/	/	/	/

→ no es necesario porque mientras no reciba pasos no se mueve.

de la tabla:

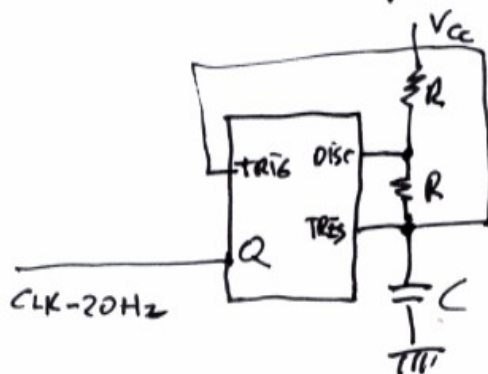
$$\begin{cases} S_1 = AD + \overline{RESET} \\ S_0 = \overline{AD} + \overline{RESET} \end{cases}$$

• Control del sistema.

Si añado a cada motor una señal de control que me permita hacer que gire o no, el control será combinatorial, quedaría así:



El reloj de 20Hz se puede generar con un 555 como astable.



$$T = 3R \cdot C \cdot 0,69 = 50ms.$$

$$R = 240 \Omega$$

El control será combinatorial según esta tabla:

Sensores	motor izdo		motor dcho	
	Función	Señales	Función	Señales
I C I C D D		AD-IZ M-IZ		AD-DC M-DC
0 1 1 0	avanza	1 1	avanza	1 1
1 1 0 0	avanza	1 1	para	X 0
1 0 0 0	avanza	1 1	retrocede	0 1
0 0 1 1	para	X 0	avanza	1 1
0 0 0 1	retrocede	0 1	avanza	1 1
0 0 0 0	retrocede	0 1	retrocede	0 1

Las señales AD-IZ, M-IZ, AD-DC y M-DC son función de I, C, I, C, D y se pueden simplificar así:

AD-IZ)

I C I / C D D	00	01	11	10
00	0	0	X	X
01	X	X	X	1
11	1	X	X	X
10	1	X	X	X

$$AD-IZ = I + CD$$

M-IZ)

I C I / C D D	00	01	11	10
00	1	1	0	X
01	X	X	X	1
11	1	X	X	X
10	1	X	X	X

$$M-IZ = \bar{C}_0 + \bar{D}$$

AD-0c

$I \ C_I \backslash C_D \ D$	00	01	11	10
00	0	1	1	X
01	X	X	X	1
11	X	X	X	X
10	0	X	X	X

$$AD-0c = D + C_D$$

M-0c

$I \ C_I \backslash C_D \ D$	00	01	11	10
00	1	1	1	X
01	X	X	X	1
11	0	X	X	X
10	1	X	X	X

$$M-0c = \overline{I} + \overline{C_I}$$

Por lo tanto el control queda:

