



**Universidad Nacional Autónoma
de México**



Facultad de Ingeniería

División de Ingeniería Eléctrica

Laboratorio de Diseño Digital VLSI

Proyecto Final VLSI

“Máquina de Garra”

Profesor: M. I. Rafael Prieto Meléndez

Grupo: 3

Alumno: Iñaky Ordiales Caballero

Semestre: 2022 - 2

Fecha de entrega: 3 de junio del 2022.

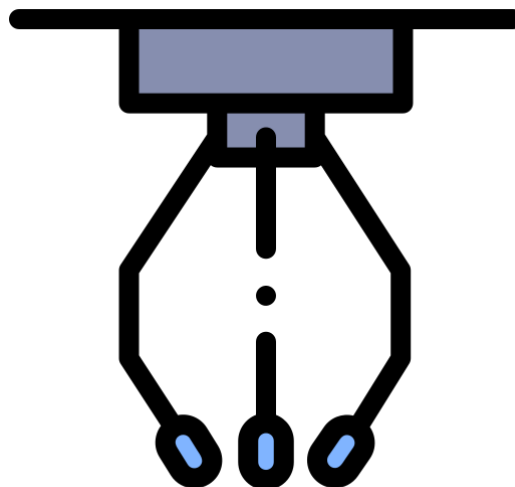
Proyecto Final: “Máquina de Garra”

Introducción



Durante el curso de diseño digital VLSI nos enfocamos principalmente en dos aspectos. Primero en el lado teórico estuvimos trabajando con el diseño de soluciones de problemas o sistemas mediante los diagramas de Mealy y Moore, Cartas ASM, direccionamiento y trayectoria de memoria, y tablas de transición. A través de las especificaciones que se nos dieran en un inicio nuestro trabajo era desarrollar el sistema de señales que nos diera la solución solicitada. Por el otro lado, en el laboratorio aprendimos no sólo de programación en lenguaje VHDL, sino que también la utilización de componentes externos a las tarjetas de desarrollo para ampliar nuestras posibilidades de soluciones. Aprendimos sobre la comunicación entre dispositivos, algunos protocolos de video, sensores y motores, etc.

En este proyecto final, el objetivo era el poder unir todos estos conocimientos (tanto teóricos como prácticos) y demostrarlos a través del diseño de un sistema más complejo. En un principio se me dificultó el pensar en qué desarrollar, pero finalmente me vino a la cabeza la idea de las máquinas de garra. En este trabajo final se muestra el desarrollo de dicho sistema desde lo conceptual hasta el prototipo con la finalidad de reflejar el aprendizaje de la asignatura.



Descripción (Diseño conceptual)

Planteamiento de la aplicación a desarrollar

Todos nosotros hemos visto alguna vez las máquinas electrónicas para jugar y poderte ganar un premio. Existen establecimientos completos destinados a estos dispositivos electrónicos llamados arcades. Dentro de estas máquinas de premios, posiblemente la más común y llamativa es la máquina de garra. Esta consiste en una garra manipulada por el usuario que después de ser colocada en una posición, o de que haya pasado determinado tiempo, baja y se cierra con la posibilidad de atrapar algo entre sus brazos. Después vuelve a subir y se mueve hasta una posición donde se vuelve a abrir dejando caer el premio del usuario que éste podrá recolectar. Acaba nuevamente en la posición inicial, lista para un nuevo juego.



Descripción detallada del proyecto y actividades a desarrollar

El proyecto en sí sería elaborar una maqueta funcional de una máquina de garra. El planteamiento de operación sería el siguiente:

El usuario insertaría una moneda y un sensor que simule el recolector de monedas de una máquina real (no valida denominación ni validez de la moneda, sólo toma en cuenta que una ficha pasó por ahí), pondrá el cronometro en 30 (segundos) e indicará que el usuario puede jugar. Al apretar el botón de inicio, el cronometro irá retrocediendo y mediante 4 botones (arriba, abajo, derecha e izquierda) el usuario podrá posicionar, con dos ejes de libertad, la garra sobre alguno de los premios. Cuando el usuario presione un botón o en su defecto se termine el cronometro la garra dejará de moverse y comenzará a bajar. Una vez llegando a una altura establecida, se cerrará y volverá a subir. Después se desplazará a la posición inicial donde se abrirá liberando el posible premio y dejándolo caer a una salida física. Habrá otro sensor que se active cuando caiga un premio para indicarle al usuario que ha ganado algo. Se podrá volver a jugar indefinidamente insertando fichas/monedas. Además, unos leds adornarán el juego, recorriéndolos una luz cuando no se esté jugando y apagándose conforme el cronómetro disminuye cuando el juego está activo.

Los elementos que se utilizarán serían dos displays de 7 segmentos para el cronometro, 3 motores a pasos (dos para el movimiento y otro para la altura de la garra), 1 servomotor (abrir y cerrar la garra), 6 botones (4 movimiento, uno inicio, uno listo para agarrar), 10 leds de adorno y 2 sensores (monedas y ganador). Algunos podrían usarse en la tarjeta (displays y leds) por la limitación de pines y los demás externos.

Desarrollo (Diseño lógico/funcional)

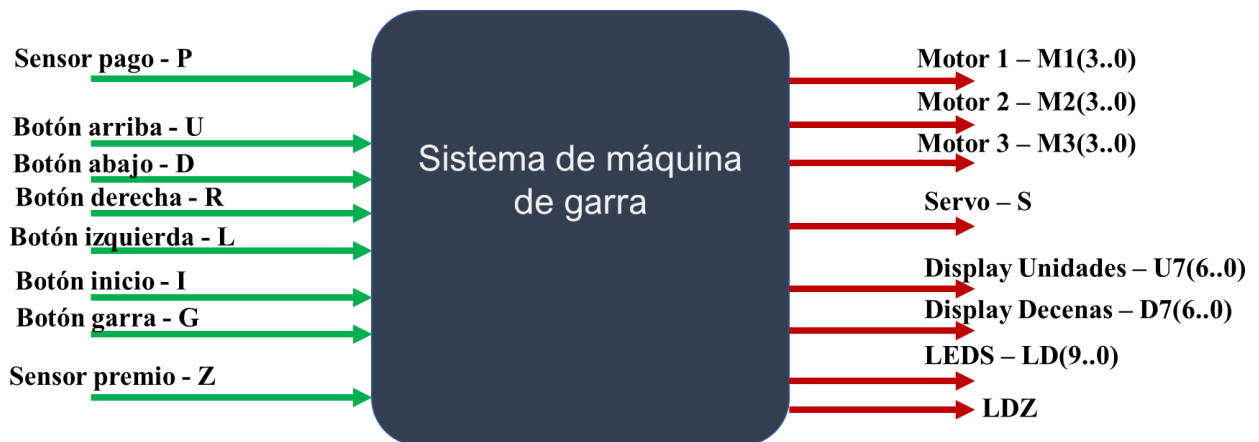
Entradas del sistema:

- botón inicio
- botón capturar
- botón arribar
- botón abajo
- botón derecha
- botón izquierda
- sensor pago
- sensor premio.

Salidas del sistema:

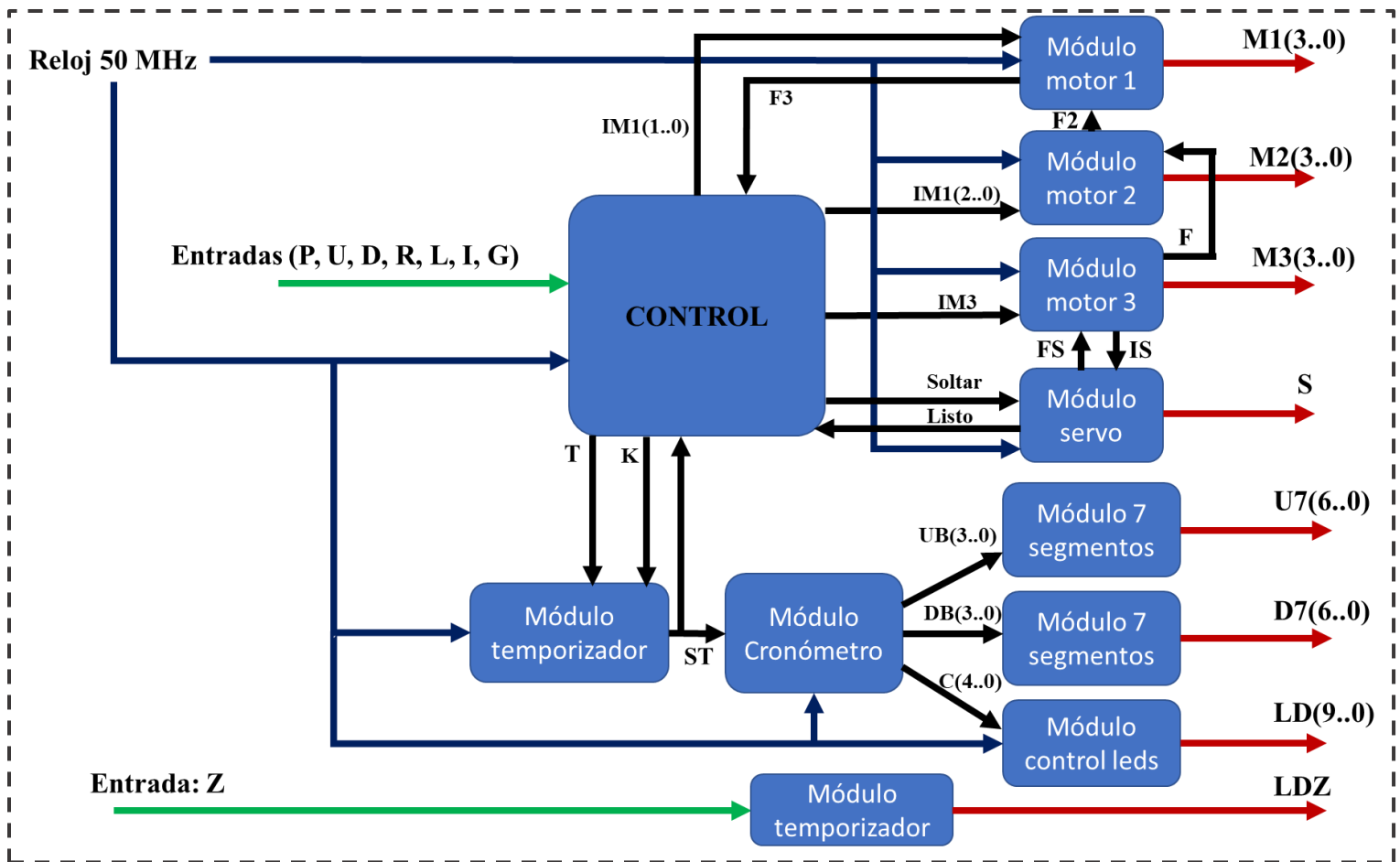
- motor 1 (eje x)
- motor 2 (eje y)
- motor 3 (eje z)
- servo (garra)
- display 7 segmentos decenas
- display 7 segmentos unidades
- leds x11 (10 adorno + 1 aviso premio)

Diagrama global entradas y salidas del sistema.



El sistema de máquina de garra a su vez se compone por varios módulos que realizarán el funcionamiento completo. Las entradas irán conectadas a un módulo central que se encargará de distribuir las tareas con base a esto. A su vez cada módulo se podrá subdividir por dentro. Aquí estamos realizando un diseño con la metodología **top-down**, también conocida como divide y vencerás.

Diagrama modular del sistema.



Breve descripción del diagrama modular:

El bloque más importante y central de sistema es el de control, ya que a pesar de que también hay comunicación entre los otros bloques, el de control es el responsable de la coordinación y algunas restricciones del sistema. A este módulo entran todas las señales de entrada externas al sistema, exceptuando la del premio, y además la señal de reloj y tres señales para controlar el flujo. A su vez saca una señal de dos bits al motor 1 y 2 para controlar su movimiento y dirección. Una señal al tercer motor para iniciar el proceso de recolección, el motor 3 se comunica con el servomotor y cuando cierra y regresa hasta arriba la garra le avisa al motor 2 para posicionarse en la posición de dejar caer el premio. El motor 2 le avisa al uno para hacer lo mismo y el 1 le avisa al control que está listo. El control le indica al servo para abrir la garra y este le responde cuando lo haga, con esta señal se volvería al estado de inicio sin movimiento ni entradas. Las otras relaciones son con un temporizador que se puede arrancar y apagar, su señal resultante se conecta al

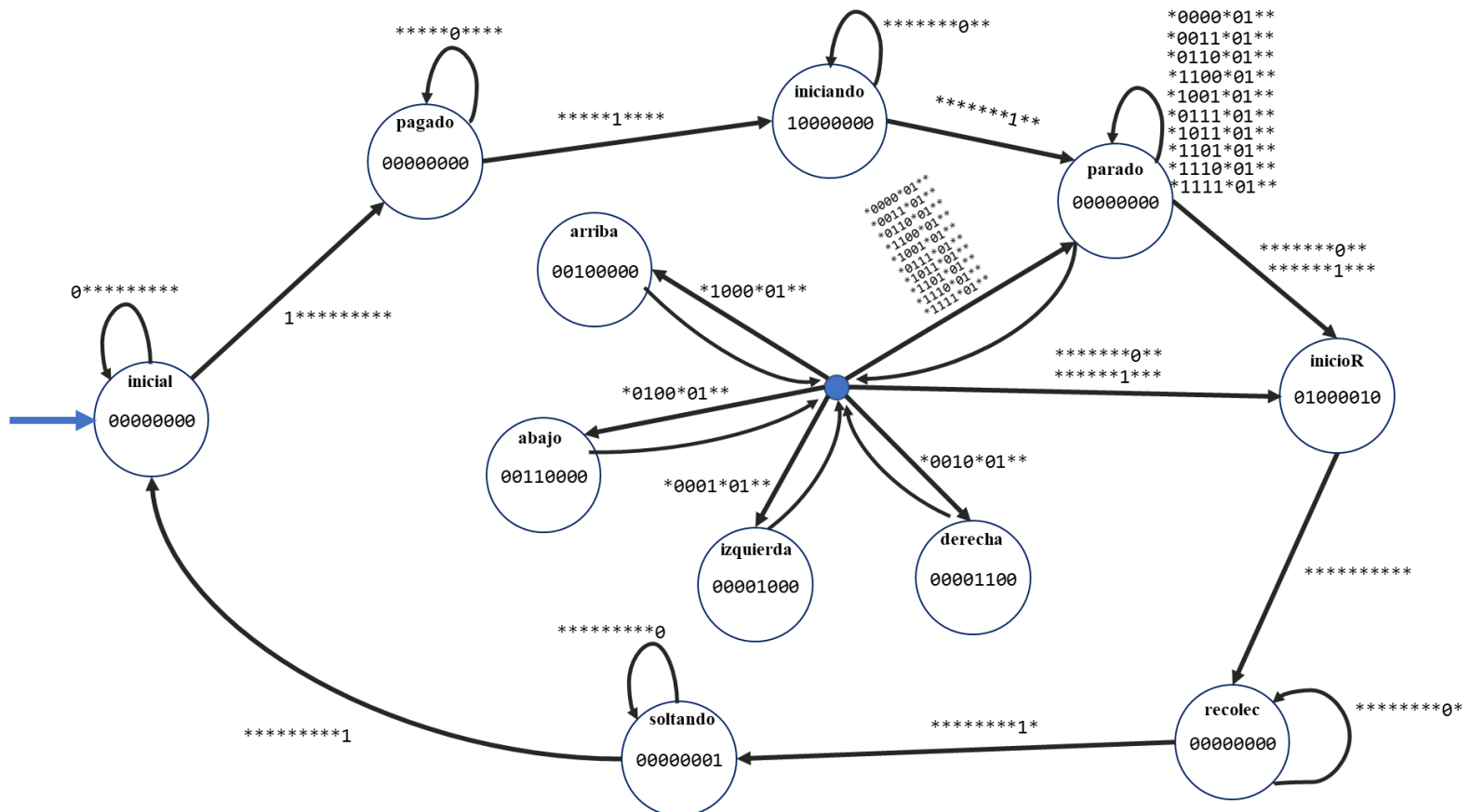
control y al cronómetro para realizar la cuenta del tiempo de juego y el cronómetro controla las unidades, decenas y cuenta total que se le pasan a los módulos de 7 segmentos y de leds.

A continuación, se desarrollará individualmente cada módulo.

Módulo de control.

Entradas: P,U,D,R,L,I,G,St,F3,Listo

Salidas: T,K,IM1(0),IM1(1),IM2(0),IM2(1),IM3,Soltar



En el centro se usó un círculo como centro de conexión para no tener que poner flechas que juntan a todos esos estados. Esto para mantener cierta limpieza en el diagrama y que no se viera un nodo de conectores entre los estados de movimiento de la garra (arriba, abajo, derecha, izquierda o parado). Se saldrá de ellos en cuanto se acabe el tiempo o se presione el botón de garra.

Explicación del diagrama de control:

En palabras el funcionamiento del módulo de control sería el siguiente:

En un inicio mientras no se active el sensor de pagado se mantiene en un estado **inicial** sin salidas, cuando se detecta una señal en alto del pago se pasa al estado **pagado** que espera a que el usuario apriete el botón de inicio (señal I). Una vez se inicia el juego se pasa al estado **iniciando** donde se le manda una señal al temporizador de 30 segundos y en cuanto se enciende la respuesta del temporizador (ST) se pasa a un estado activo de juego llamado **parado**. Aquí dependiendo de si sólo se aprieta uno de los botones de movimiento se le mandará al motor correspondiente una señal para indicar que se debe de mover. En caso de que se apague la señal del temporizador (ST) o que el usuario pique el botón de atrapar, se pasa al estado **inicioR** (inicio Recolección), este estado mata un posible temporizador activo y activa la señal IM3 para indicarle al motor 3 que debe empezar el proceso de recolección y pasa al estado **recolec**.

*(Este proceso de recolección se lleva a cabo mediante comunicación entre los módulos **motor1**, **motor2**, **motor3**, **servo**. Cuando el motor 3 reciba la señal de recolección bajará la garra y dará una señal al servo. El servo cerrará la garra y devolverá una señal al motor3 para que suba de nuevo la garra. Una vez el motor3 acaba de subir la garra manda una señal al motor 2 para que regrese a la posición inicial del eje Y. Ya en la posición inicial el motor 2 indica al motor 1 que haga lo mismo y cuando este termina manda una señal en alto al control (señal F3).)*

Cuando el bloque de control recibe en alto la señal F3 pasa al estado **soltando** el cual manda una señal al servo para que abra la garra y el servo le devuelve una señal indicando que ya soltó el objeto (señal Listo) para regresar al estado inicial del sistema.

Tabla de transiciones: Tendríamos la siguiente con flip flops D:

Nombre Edo.	Estado binario	Entradas	Edo. siguiente	Salidas	Señales Actv.
inicial	0000	0*****	0000	00000000	0000
		1*****	0001		0001
pagado	0001	*****0****	0001	00000000	0001
		*****1****	0010		0010
iniciando	0010	*****0**	0010	10000000	0010
		*****1**	0011		0011
parado	0011	*0000*01**	0011	00000000	0011
		*0011*01**			
		*0110*01**			
		*1100*01**			
		*1001*01**			
		*0111*01**			
		*1011*01**			
		*1101*01**			
		*1110*01**			
		*1111*01**			
		*****0**	1000		1000
		*****1***			
		*1000*01**	0100		0100
		*0100*01**	0101		0101
		*0001*01**	0110		0110
		*0010*01**	0111		0111
arriba	0100	*0000*01**	0011	00100000	0011
		*0011*01**			
		*0110*01**			
		*1100*01**			
		*1001*01**			
		*0111*01**			
		*1011*01**			

		*1101*01**				
		*1110*01**				
		*1111*01**				
		*****0**	1000		1000	
		*****1**				
		*1000*01**	0100		0100	
		*0100*01**	0101		0101	
		*0001*01**	0110		0110	
		*0010*01**	0111		0111	
abajo	0101	*0000*01**	0011	00110000	0011	
		*0011*01**				
		*0110*01**				
		*1100*01**				
		*1001*01**				
		*0111*01**				
		*1011*01**				
		*1101*01**				
		*1110*01**				
		*1111*01**				
		*****0**	1000		1000	
		*****1**				
		*1000*01**	0100		0100	
		*0100*01**	0101		0101	
		*0001*01**	0110		0110	
		*0010*01**	0111		0111	
izquierda	0110	*0000*01**	0011	00001000	0011	
		*0011*01**				
		*0110*01**				
		*1100*01**				
		*1001*01**				
		*0111*01**				

		*1011*01**			
		*1101*01**			
		*1110*01**			
		*1111*01**			
		*****0**	1000		1000
		*****1***			
		*1000*01**	0100		0100
		*0100*01**	0101		0101
		*0001*01**	0110		0110
		*0010*01**	0111		0111
derecha	0111	*0000*01**	0011	00001100	0011
		*0011*01**			
		*0110*01**			
		*1100*01**			
		*1001*01**			
		*0111*01**			
		*1011*01**			
		*1101*01**			
		*1110*01**			
		*1111*01**			
		*****0**	1000		1000
		*****1***			
		*1000*01**	0100		0100
		*0100*01**	0101		0101
		*0001*01**	0110		0110
		*0010*01**	0111		0111
inicioR	1000	*****	1001	01000010	1001
recolec	1001	*****0*	1001	00000000	1001
		*****1*	1010		1010
soltando	1010	*****0	1010	00000001	1010
		*****1	0000		0000

Los motores a pasos se controlan por un módulo FSM que se rige por las siguientes tablas:

Estado Presente		Condiciones			Estado Sigiente
Estado 0	SM0	F	H	UD	
		0	0	0	SM0
		0	0	1	SM0
	Salidas	0	1	0	SM0
		0	1	1	SM0
		1	0	0	SM0
	M3 M2 M1 M0	1	0	1	SM0
	0 0 0 0	1	1	0	SM0
		1	1	1	SM0

Estado 1	SM1	F	H	UD	Estado Sigiente
		0	0	0	SM7
		0	0	1	SM3
	Salidas	0	1	0	SM8
		0	1	1	SM2
		1	0	0	SM8
	M3 M2 M1 M0	1	0	1	SM2
	1 0 0 0	1	1	0	SM4
		1	1	1	SM1

Estado 2	SM2	F	H	UD	Estado Sigiente
		0	0	0	SM7
		0	0	1	SM1
	Salidas	0	1	0	SM8
		0	1	1	SM4
		1	0	0	SM1
	M3 M2 M1 M0	1	0	1	SM3
	1 0 0 0	1	1	0	SM4
		1	1	1	SM9

Estado 3	SM3	F	H	UD	Estado Sigiente
		0	0	0	SM1
		0	0	1	SM5
	Salidas	0	1	0	SM8
		0	1	1	SM2
		1	0	0	SM2
	M3 M2 M1 M0	1	0	1	SM4
	0 1 0 0	1	1	0	SM4
		1	1	1	SM9

Estado Presente		Condiciones			Estado Sigiente
Estado 4	SM4	F	H	UD	
		0	0	0	SM7
		0	0	1	SM1
	Salidas	0	1	0	SM2
		0	1	1	SM6
		1	0	0	SM3
	M3 M2 M1 M0	1	0	1	SM5
	0 1 1 0	1	1	0	SM10
		1	1	1	SM9

Estado 5	SM5	F	H	UD	Estado Sigiente
		0	0	0	SM3
		0	0	1	SM7
	Salidas	0	1	0	SM8
		0	1	1	SM2
		1	0	0	SM6
	M3 M2 M1 M0	1	0	1	SM4
	0 0 1 0	1	1	0	SM4
		1	1	1	SM9

Estado 6	SM6	F	H	UD	Estado Sigiente
		0	0	0	SM7
		0	0	1	SM1
	Salidas	0	1	0	SM4
		0	1	1	SM8
		1	0	0	SM5
	M3 M2 M1 M0	1	0	1	SM7
	0 0 1 1	1	1	0	SM4
		1	1	1	SM9

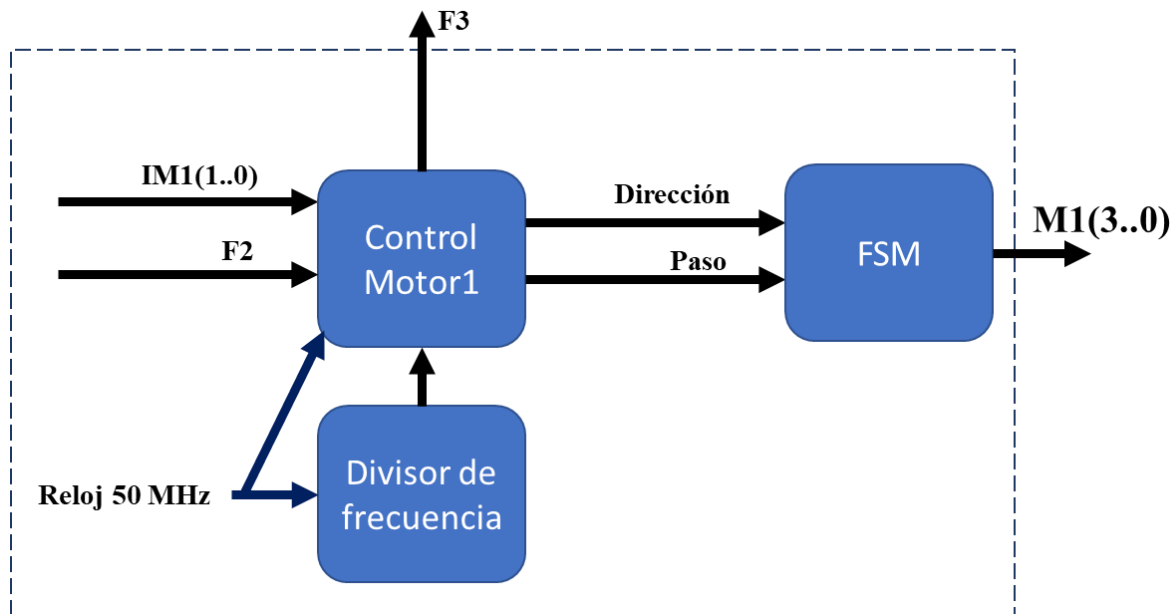
Estado 7	SM7	F	H	UD	Estado Sigiente
		0	0	0	SM5
		0	0	1	SM1
	Salidas	0	1	0	SM8
		0	1	1	SM2
		1	0	0	SM6
	M3 M2 M1 M0	1	0	1	SM8
	0 0 0 1	1	1	0	SM4
		1	1	1	SM9

Estado Presente		Condiciones			
Estado 8	SM8	F	H	UD	Estado Sigiente
		0	0	0	SM7
		0	0	1	SM1
		0	1	0	SM6
	Salidas	0	1	1	SM2
		1	0	0	SM7
	M3 M2 M1 M0	1	0	1	SM1
	1 0 0 1	1	1	0	SM9
		1	1	1	SM10
Estado 9	SM9	F	H	UD	Estado Sigiente
		0	0	0	SM7
		0	0	1	SM1
		0	1	0	SM8
	Salidas	0	1	1	SM2
		1	0	0	SM8
	M3 M2 M1 M0	1	0	1	SM1
	1 0 1 0	1	1	0	SM4
		1	1	1	SM8
Estado 10	SM10	F	H	UD	Estado Sigiente
		0	0	0	SM7
		0	0	1	SM1
		0	1	0	SM8
	Salidas	0	1	1	SM2
		1	0	0	SM8
	M3 M2 M1 M0	1	0	1	SM1
	0 1 0 1	1	1	0	SM8
		1	1	1	SM4

Ahora podemos pasar a describir los demás módulos del sistema que lo utilizan.

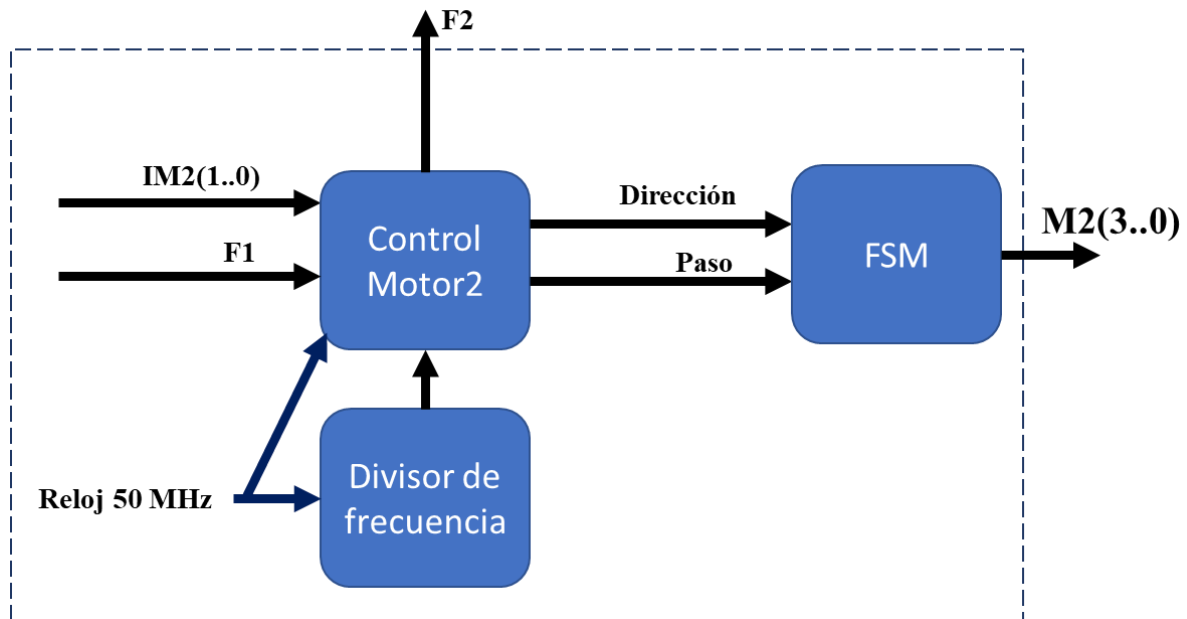
Módulo motor 1.

El módulo del motor 1 se compone a su vez de algunos componentes: Un divisor de frecuencia para establecer el paso, y el módulo de motor a pasos revisado en clase. Dependiendo de las entradas IM1 se le indica que empiece el movimiento y la dirección. Dentro del control de motor se encarga de verificar los límites y establecer la frecuencia de regreso en cuanto se activa la señal F2.



Módulo motor 2.

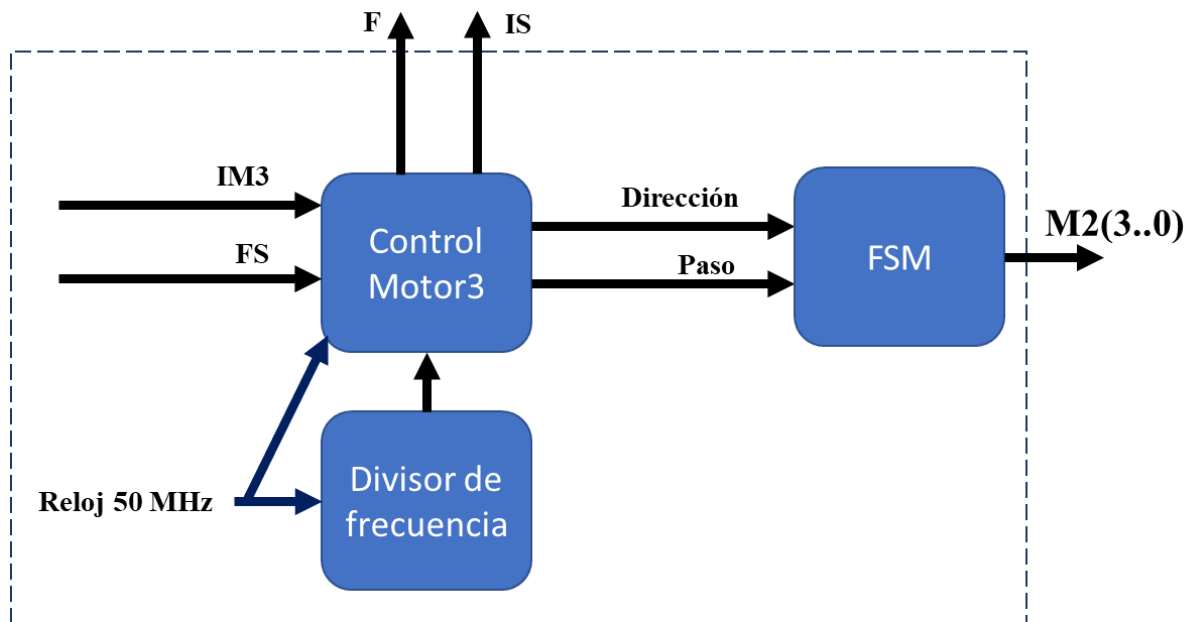
El módulo del motor 2 se compone a su vez de algunos componentes: Un divisor de frecuencia para establecer el paso, y el módulo de motor a pasos revisado en clase. Dependiendo de las entradas IM1 se le indica que empiece el movimiento y la dirección. Dentro del control de motor se encarga de verificar los límites y establecer la frecuencia de regreso en cuanto se activa la señal F1.



Módulo motor 3.

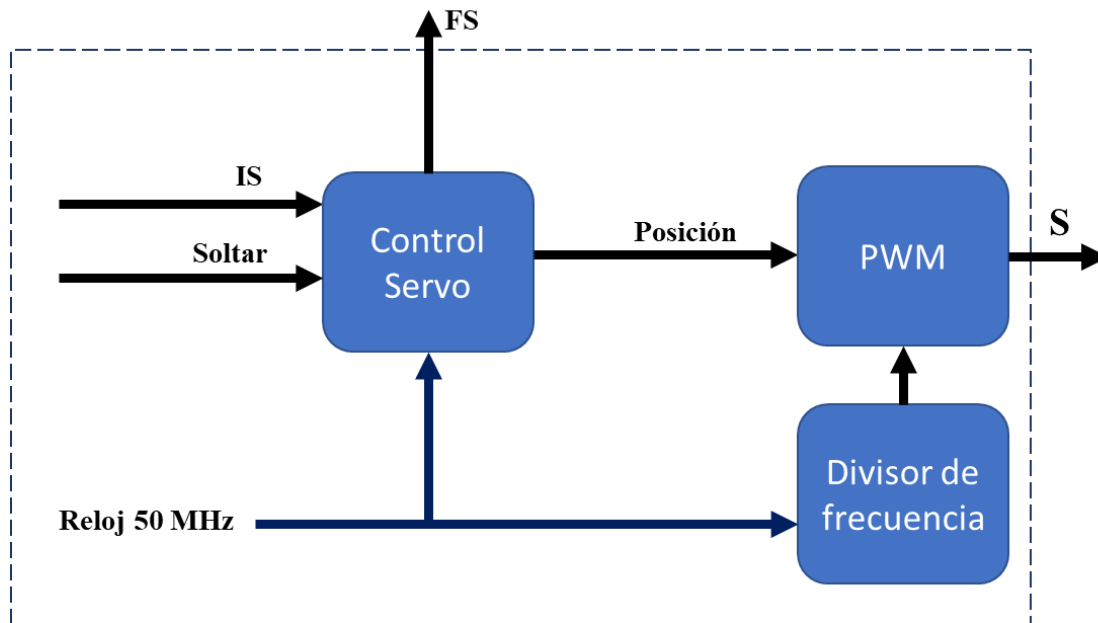
El módulo del motor 3 se compone a su vez de algunos componentes: Un divisor de frecuencia para establecer el paso, y el módulo de motor a pasos revisado en clase.

Cuando se activa la señal IM3 esta le indica al control que la garra debe bajar, una vez abajo se le avisa al servomotor con la señal IS y se espera la respuesta mediante FS para volver a subir la garra.



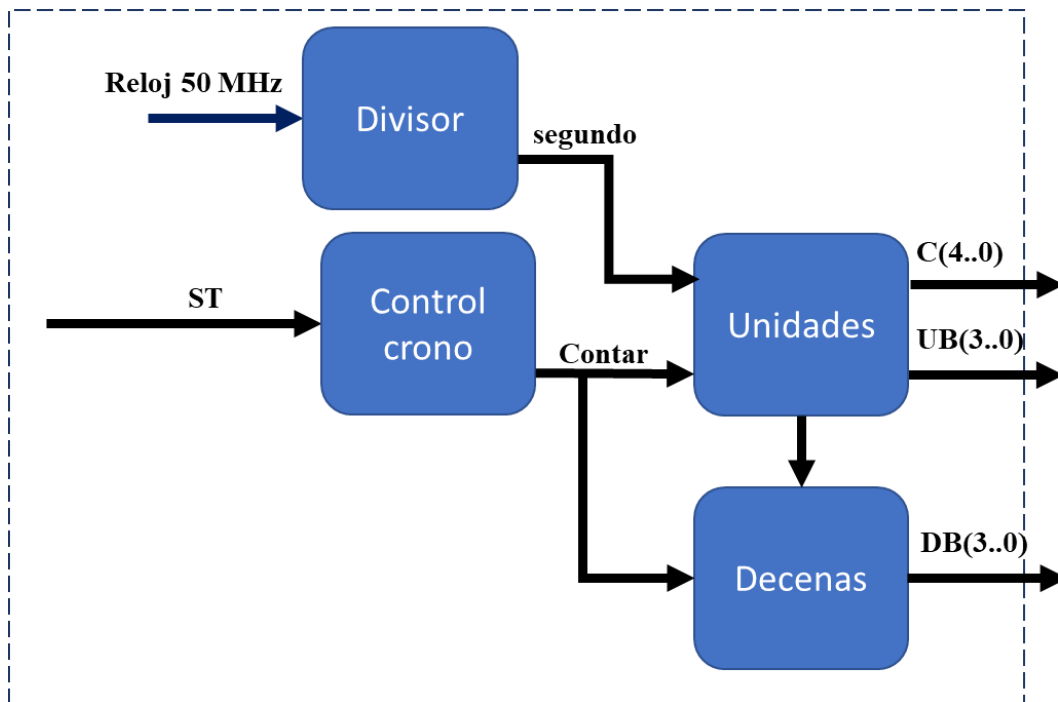
Módulo servomotor de modelismo.

El módulo del servomotor es bastante simple dependiendo de las señales de entrada se da la señal para que el servomotor se abra o se cierre. Si se activa la señal IS se manda la posición de cerrado al PWM y si se activa la señal de Soltar se manda la de abierto.



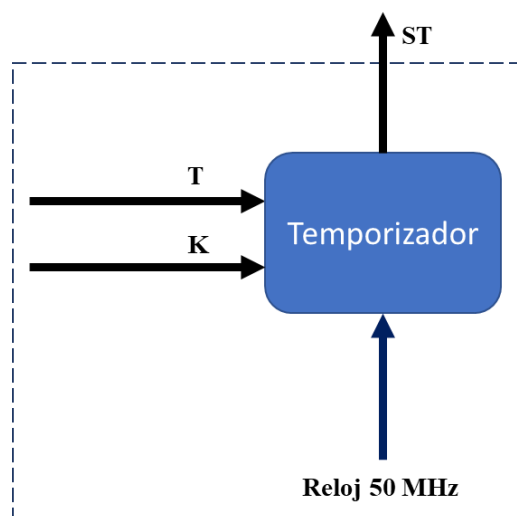
Módulo cronometro.

El módulo de cronometro es un cronometro que al recibir el cambio de estado bajo a estado alto de ST reinicia una cuenta regresiva de 30 a 0 cada segundo. Cuando termina la cuenta, mientras no se reciba otro flanco de subida se mantendrá en 0.



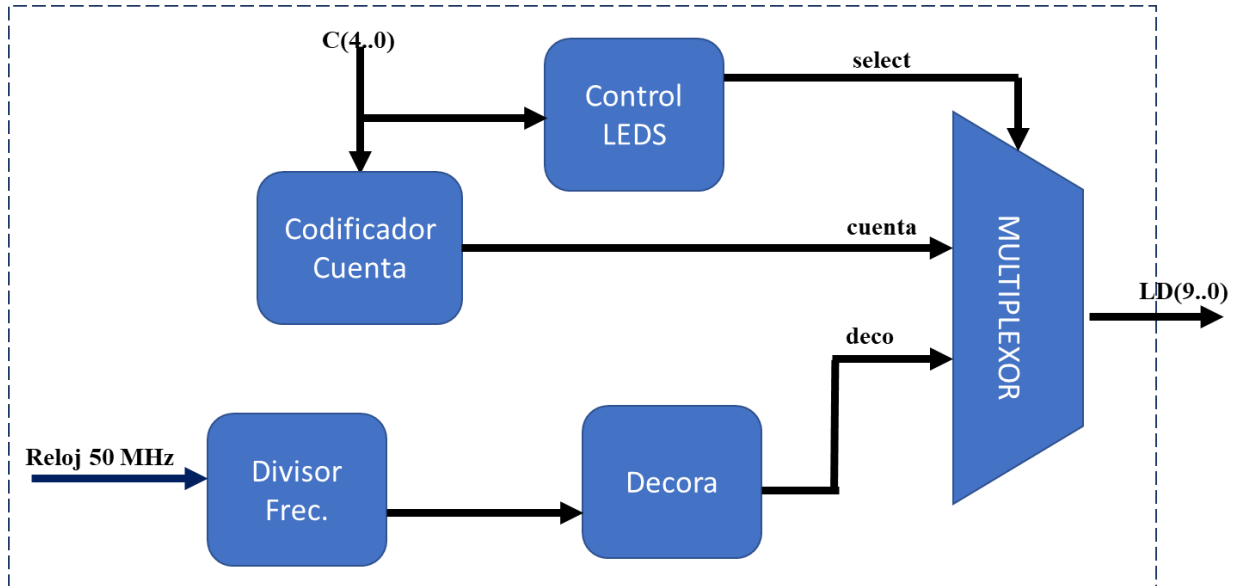
Módulo temporizador.

El módulo temporizador es el revisado en clase con la modificación de que se puede apagar en cualquier instante con una señal K (Kill).



Módulo control leds.

El módulo de control de leds depende del estado de la cuenta para la salida. Si la cuenta es 0 activa el módulo decora (decoración) y se prenden y apagan los leds de un lado a otro, si la cuenta es cualquier otro valor activa la el codificador cuenta para apagar leds conforme avanza el tiempo.



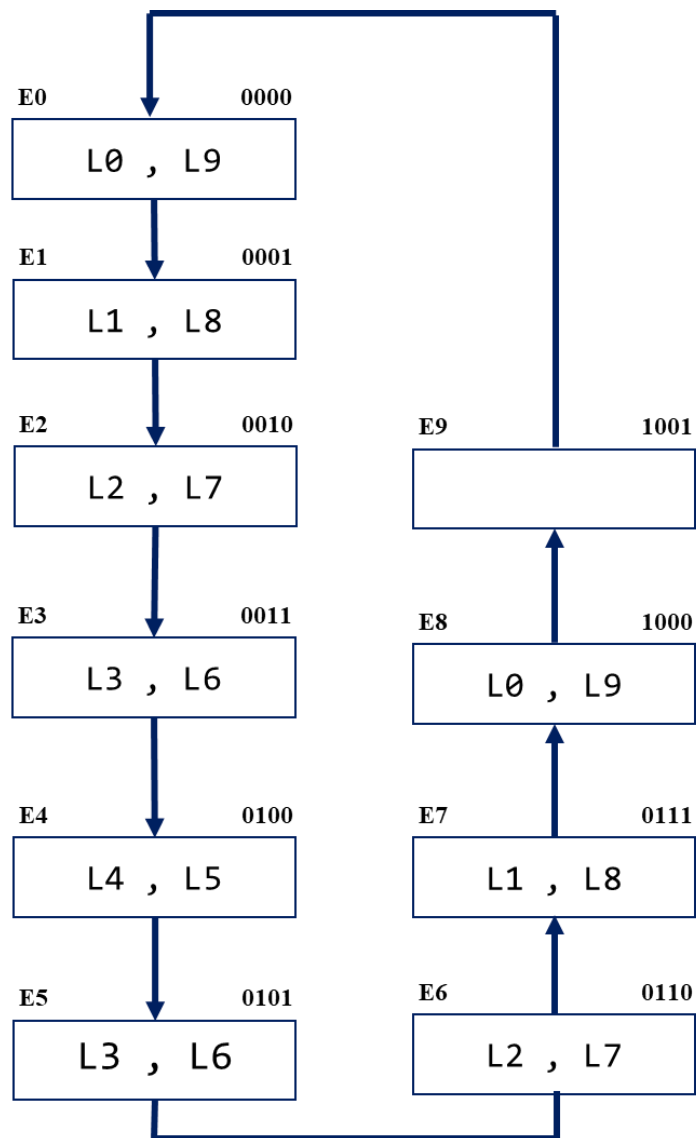
Control LEDS:

ENTRADAS C4, C3, C2, C1, C0	SALIDAS select
00000	1
1****	0
*1***	
1	
***1*	
****1	

Codificador Cuenta:

C	LD
C > 1 1101 = 24	1111111111
C > 1 1101 = 18	0011111111
C > 0 1100 = 12	0000111111
C > 0 0110 = 06	0000001111
C > 0 0000 = 00	0000000011

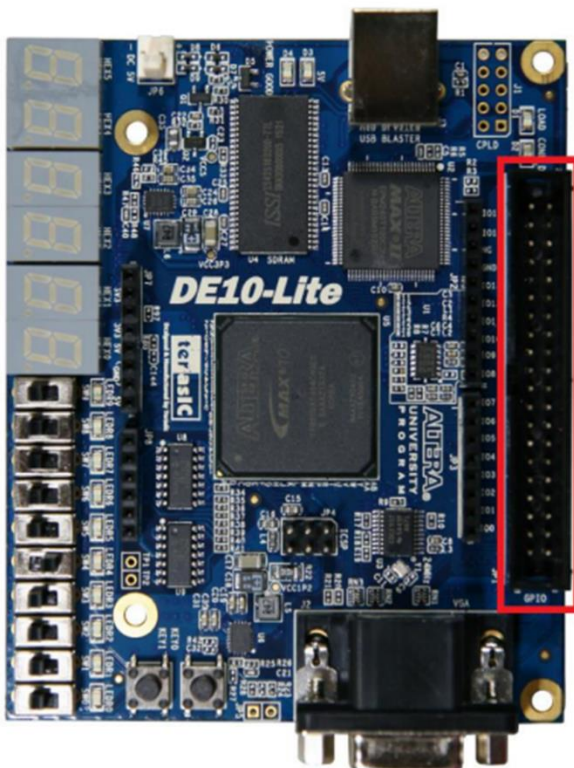
Decora:



Módulo 7 segmentos.

BCD	7 segmentos
0000	1000000
0001	1111001
0010	0100100
0011	0110000
0100	0011001
0101	0010010
0110	0000010
0111	1111000
1000	0000000
1001	0010000
otros	1000000

Asignación de pines de la tarjeta de desarrollo (DE10 – Lite)



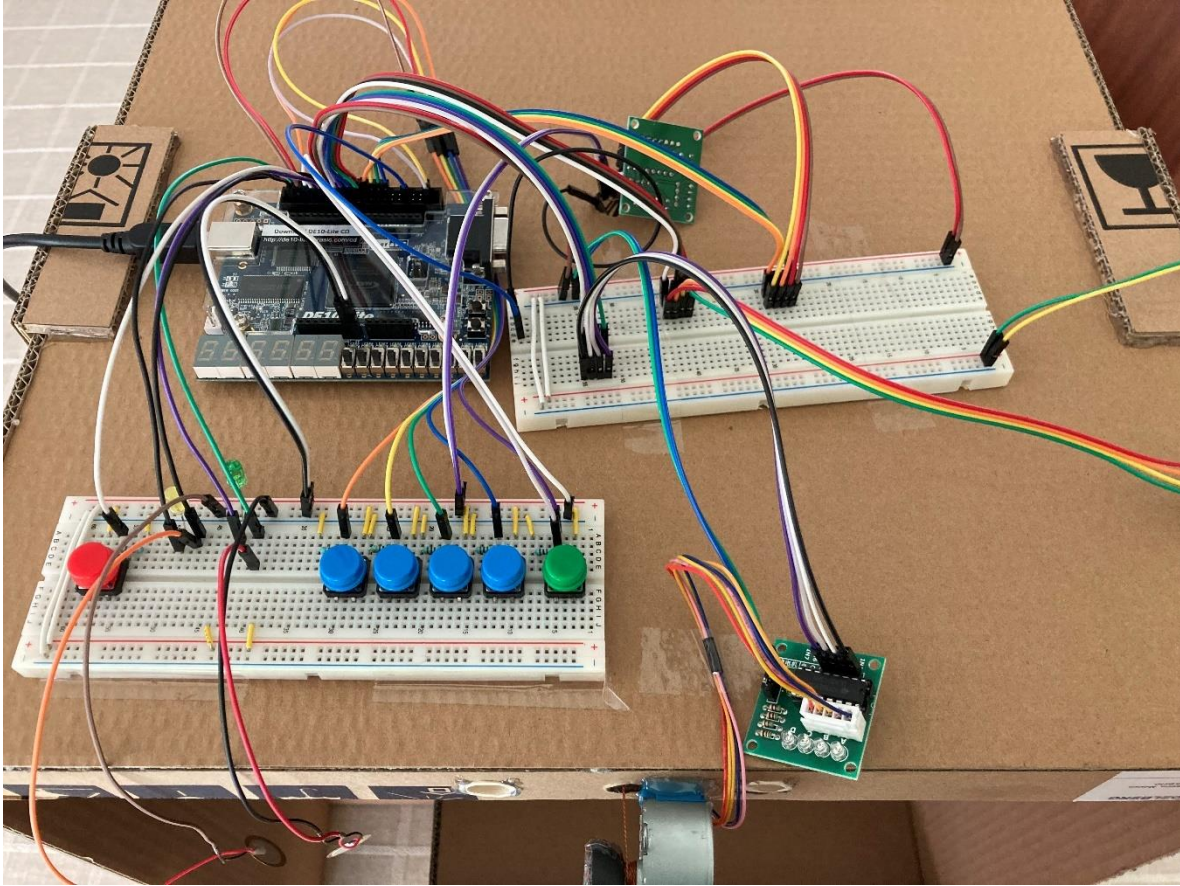
GPIO (JP1)	
PIN_V10	GPIO_[0] P 1
PIN_V9	GPIO_[2] U 3
PIN_V8	GPIO_[4] R 5
PIN_V7	GPIO_[6] I 7
PIN_W6	GPIO_[8] LDZ 9
5V	11
PIN_W5	GPIO_[10] 12
PIN_AA14	GPIO_[12] 14
PIN_W12	GPIO_[14] 16
PIN_AB12	GPIO_[16] 18
PIN_AB11	GPIO_[18] 20
PIN_AB10	GPIO_[20] 22
PIN_AA9	GPIO_[22] 24
PIN_AA8	GPIO_[24] 26
3.3V	28
PIN_AA7	GPIO_[26] 30
PIN_AA6	GPIO_[28] S 32
PIN_AA5	GPIO_[30] 34
PIN_AB3	GPIO_[32] 36
PIN_AB2	GPIO_[34] 38
	40
	GPIO_[35] PIN_AA2
	GPIO_[1] PIN_W10
	GPIO_[3] PIN_W9
	GPIO_[5] PIN_W8
	GPIO_[7] PIN_W7
	GPIO_[9] PIN_V5
	GPIO_[11] PIN_AA15
	GPIO_[13] PIN_W13
	GPIO_[15] PIN_AB13
	GPIO_[17] PIN_Y11
	GPIO_[19] PIN_W11
	GPIO_[21] PIN_AA10
	GPIO_[23] PIN_Y8
	GPIO_[25] PIN_Y7
	GPIO_[27] PIN_Y6
	GPIO_[29] PIN_Y5
	GPIO_[31] PIN_Y4
	GPIO_[33] PIN_Y3
	GPIO_[35] PIN_AA2

Resultados y prototipo desarrollado

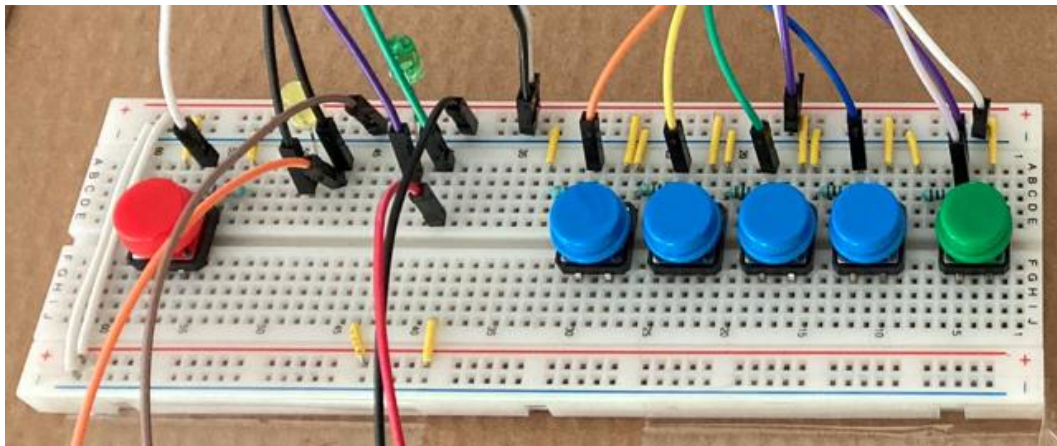
Para desarrollar el prototipo se tomaron ideas del diseño a partir de diversos videos revisados en los que se desarrollaban máquinas de garra mecánicas (con mangueras y jeringas para el movimiento). A pesar de que nosotros realizaríamos una funcionalidad electrónica, la estructura y diseño de la máquina sí nos eran de utilidad. Por esto se creó un prototipo de cartón reciclado, pegado con silicón caliente y con unos tubos de pvc reciclados para crear los ejes en la tapa donde se deslizaría nuestra garra. Llevo varias horas el cortar y pegar todas las piezas a medida, pensando en todo momento sobre cómo acomodar los motores y cables de la mejor manera posible. El resultado final fue el siguiente:



En la parte superior de la caja se hicieron todas las conexiones necesarias y se agregó el panel de control.



En la imagen anterior podemos apreciar la tarjeta de desarrollo con todas las conexiones y salidas de los pines lógicos GPIO. A partir de esta se alimentaban los tres motores a pasos que se utilizaron y el servomotor, así como también se recibían las señales de entrada de los botones que nos permiten interactuar con el sistema.



En la imagen anterior se pueden observar los botones con los que el usuario podría interactuar con el sistema y dos leds de indicaciones.

Funcionamiento e interacción

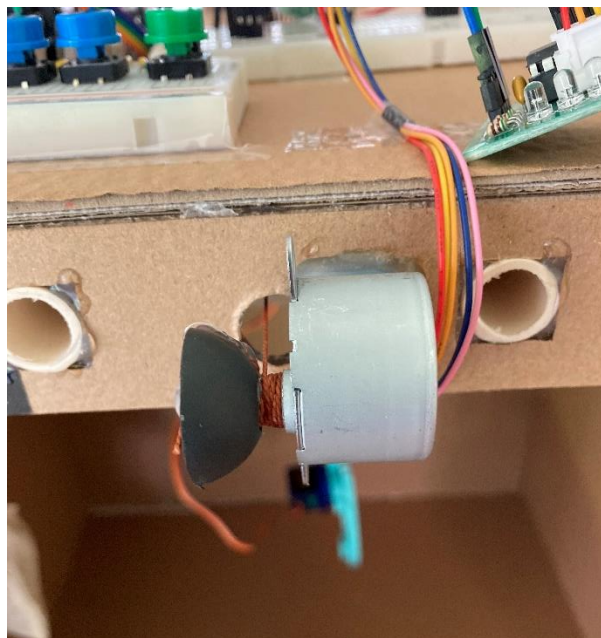
Al cargar el programa entraba en un estado de espera en el que se necesitaba una señal de activación que simulara el pago para jugar. Esta se realizaba mediante el movimiento de un piezoeléctrico que activaba el led amarillo de la tabla, indicando que ya se podía iniciar un juego. De la misma forma (con un piezoeléctrico) se simulaba la condición en la que se ganaba un premio (el premio caía encima del sensor y lo activaba) causando que el led verde se prendiera por 5 segundos.

Con el led amarillo activo al presionar el botón rojo se iniciaba el juego, en la tarjeta de desarrollo se veían en los displays la cuenta regresiva de 30 a 0 y una barra de leds que disminuía conforme al tiempo. En este momento ya se iniciaba el juego. Los botones azules nos permitían movernos de la siguiente forma:



Hasta que se acabara el tiempo o en su defecto se presionara el botón verde indicando que estamos listos para que baje la garra. En este punto la garra bajaba y se cerraba para después subir y liberar el premio obtenido. La manera en que se hacían los movimientos en los tres ejes fue con los siguientes motores a pasos:

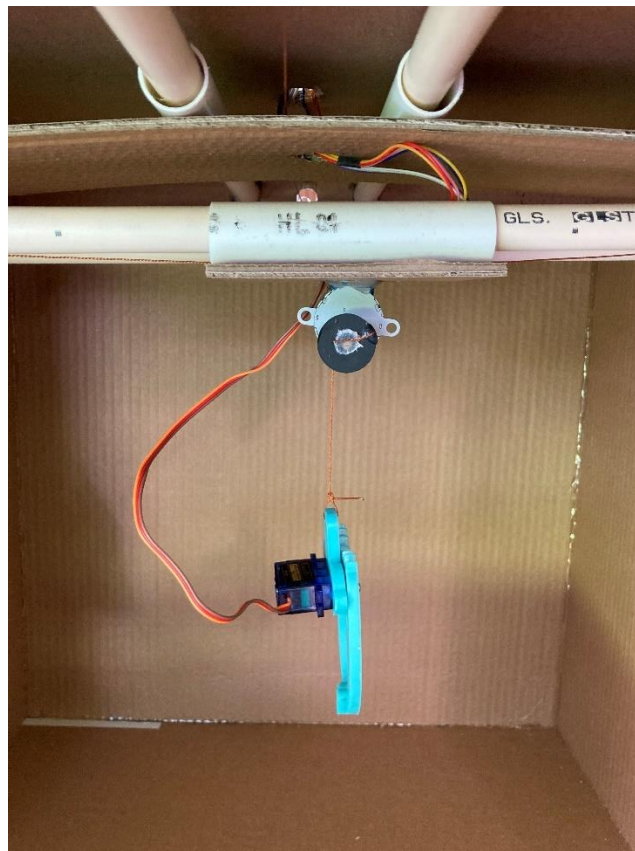
Eje y (adelante y atrás):



Eje x (izquierda y derecha):

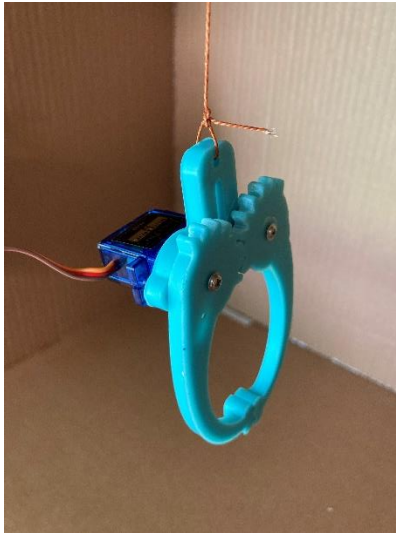


Eje z (abajo y arriba):

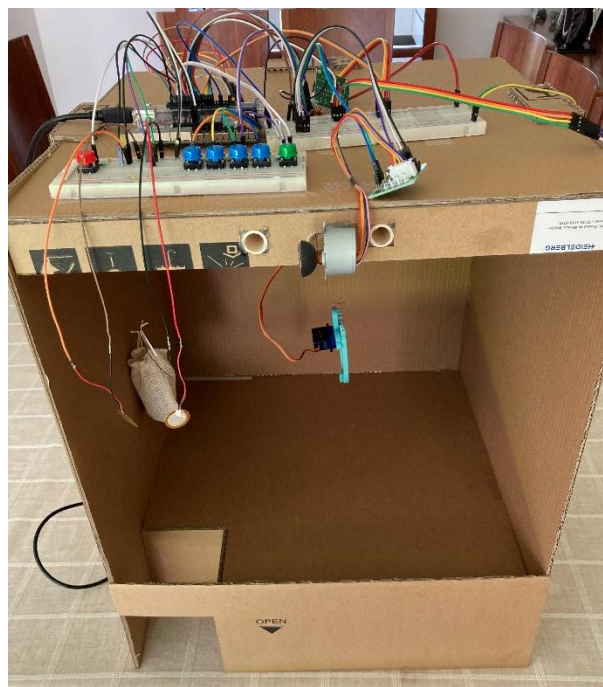


Para permitir el libre movimiento de los motores y por ende de la garra se desarrolló un sistema de ejes en el techo, los cuales nos permitían alcanzar todo el cuadrado de movimiento. También la forma más sencilla de mover hacia un lado y otro la garra con una sola cuerda fue usando contrapesos en el lado opuesto de los motores que jalaran cuando el motor desenrollara la cuerda.

Para la garra se utilizó un servomotor y una impresión en 3D que se obtuvo libremente en línea. Esta ya tenía las medidas necesarias por lo que sólo hubo que hacerla y armarla. Así el servo al moverse la abriría y cerraría según correspondiera.



El diseño final terminó siendo estéticamente muy atractivo y también funcional para los movimientos necesarios.



Conclusiones

El desarrollo de este proyecto final sí representó un reto, inclusive mayor al que me esperaba debido a la complejidad del sistema que se escogió desarrollar. Si bien es cierto que el resultado final estuvo más limitado que el planteamiento inicial, todavía creo que es un proyecto satisfactorio y que muestra un alto grado de aprendizaje a lo largo de la materia. El prototipo desarrollado era funcional para poderse mover en cualquier dirección (adelante, atrás, izquierda y derecha) y además se logró efectuar de buena manera que bajara y subiera la garra. Esto en un prototipo no es sencillo debido a que los cables normalmente nos estorban y el moverse en dos dimensiones se puede complicar.

Durante la elaboración, desde su conceptualización y planeación hasta la conexión eléctrica de los cables con la tarjeta de desarrollo, se estuvieron practicando los conceptos revisados en la asignatura. Realmente la parte más importante es la de diseño y no porque la codificación, conexión y construcción del sistema no fueran suficientemente complejas o difíciles, pero al final de cuentas todas dependen del diseño inicial del sistema. Nuestro diseño condicionó la estructura del prototipo (donde dejar los hoyos para las conexiones, calcular los pesos y movimientos, etc.) y fungió como la base para realizar la programación en lenguaje VHDL. En la vida real así debe de ser y a pesar de que después durante la implementación se cambie el diseño original por alguna comodidad o necesidad, éste seguirá siendo la base principal de todo lo demás.

El resultado del proyecto me deja satisfecho y con ganas de seguir aprendiendo más al respecto de los sistemas digitales que se pueden desarrollar. Se cumple con el objetivo de demostrar los conocimientos adquiridos en la asignatura y se logró dar solución al problema o planteamiento inicial.

