

## Proyecto #01 – Diseño y construcción de una FSM

Gabriel Alexander Fong Penagos

### Seguro y Alarma de un Automóvil

#### Caja Negra



#### Descripción

El seguro y Alarma de un automóvil será controlado mediante un control remoto o a distancia, el cual puede controlar el seguro de todas las puertas del automóvil, el encendido y apagado del motor del automóvil, el encendido y apagado de las funciones eléctricas del automóvil (aire acondicionado, radio, luces, entre otras), todas estas acciones se podrán ver en el tablero del automóvil y unos leds en el control remoto. Además, cuenta con 3 botones de emergencia, con los cuales mediante una serie de combinaciones que el usuario dueño del automóvil conocerá, podrá configurar una serie de alarmas, sirenas e incluso el polarizado del carro, cada combinación es un tipo de emergencia distinta. Por último, tiene un sensor de golpes el cual avisara al usuario por medio de una alarma.

La maquina del Seguro y alarma de un Automóvil tiene 6 entradas, 5 de un bit de tamaño y una de 3 bits (Botones de emergencia), y tiene 9 salidas, 5 de un bit (Puerta piloto, puertas secundarias, motor, mensaje de emergencia y funciones electrónicas) y 4 de 2 bits (tablero, alarma, sirena, polarizado).

## FSM 1

### Caja Negra



### Descripción

La FSM1 cuenta con 4 inputs de 1 bit y una entrada generada por la FSM2 (Señal P) y 5 señales de salida, 4 de un bit y una de 2 bits (tablero). La FSM1 es la encargada del control de seguro en las puertas del automóvil, encendido y apagado del motor y funciones electrónicas, además informa al usuario del estado del seguro de las puertas y del estado del motor y funciones electrónicas.

### Entradas

- Botón Locked (L) = es un botón tipo pulsador (normalmente abierto), el cual al presionarlo coloca el seguro en las puertas del automóvil. *Es de un bit. 1 = colocar seguro. 0 = ninguna acción.*
- Botón Unlocked (U) = es un botón tipo pulsador (normalmente abierto), el cual al presionarlo quita el seguro en las puertas del automóvil. *Es de un bit. 1 = quitar seguro. 0 = ninguna acción.*
- Start (ST) = es un botón normalmente cerrado, el cual al presionarlo enciende o apaga el motor del automóvil. *Es de un bit. 1 = encender motor. 0 = apagar motor.*
- ON (ON) = es un botón normalmente cerrado, el cual al presionarlo enciende o apaga las funciones electrónicas del automóvil. *Es de un bit. 1 = encender funciones. 0 = apagar funciones.*

\*Todas las entradas anteriores se encuentran en el control remoto.

- Señal P = la señal P es generada por la FSM2, mas adelante explicaremos como se genera, la cual al estar activa, automáticamente coloca seguro en todas las puertas del automóvil. *Es de un bit. 1 = colocar llave en todas las puertas. 0 = ninguna acción.*

### Salidas

- Puerta piloto (PP) = Es el seguro de la puerta del piloto. *Es de un bit. 1 = puerta con seguro. 0 = puerta sin seguro.*
- Puertas Secundarias (PS) = Es el seguro de las puertas secundarias ( de 1 a 4 puertas dependiendo el modelo del automóvil, incluye baúl si lo tiene). *Es de un bit. 1 = puertas con seguro. 0 = puertas sin seguro.*

- Motor (M) = Motor del automóvil. *Es de un bit. 1 = motor encendido. 0 = motor apagado.*
- Funciones Electrónicas (F) = Activa las funciones electrónicas del automóvil como el aire acondicionado, radio, luces externas e internas, entre otras dependiendo el automóvil. *Es de un bit. 1 = Funciones encendidas. 0 = funciones apagadas.*
- Tablero (TA) = el tablero nos indica el estado de seguro de las puertas y el encendido del motor. Esta señal se encuentra en el tablero del automóvil y en leds en el control remoto. *Es de 2 bits. 00 = sin seguro en las puertas. 01 = seguro solo en las puertas secundarias. 10 seguro en todas las puertas. 11 = motor encendido y seguro en todas las puertas.*

## FSM2

### Caja Negra



### Descripción

La FSM2 cuenta con 2 inputs, uno de un bit de tamaño (Sensor de golpe) y el de Botones de emergencia es de 3 bits. Tiene 5 salidas, 2 de un bit (Señal P y Mensaje de Emergencia) y 3 de 2 bits (Alarma, Sirena y polarizado). La FSM2 es la encargada del control de las alarmas, sirenas, polarizado, mensaje de emergencia y la señal P. Los botones de emergencia son los encargados de seleccionar cual es el tipo de emergencia, los cuales se deben de seleccionar con el código adecuado, el cual colocará una alarma, sirena y polarizado distinto, además en algunos casos manda un mensaje de emergencia a un teléfono seleccionado, el cual envía la ubicación GPS en tiempo real. El sensor de golpe nos indica si al automóvil lo golpearon o lo intentaron abrir, el cual también activa una sirena y alarma para alertar al dueño.

### Entradas

- Botones de Emergencia (BE) = Son 3 botones normalmente cerrados que se encuentran en el control remoto, los cuales mediante las combinaciones de los mismos al presionarlos seleccionan una emergencia distinta. *NADA = 000 - PANICO = 001 - EMERGERGENCIA = 010 - BUSCAR = 011 - MODO INCOGNITO = 100 - MODO ME ESTAN SIGUIENDO = 101*
- Sensor de Golpe (SE) = Es un señor el cual se encuentra en el automóvil, el cual manda una señal cuando golpearan o intentan abrir el automóvil forzosamente. *1 = Golpe. 0 = Normal.*

## Salidas

- Alarma (A) = La alarma se encuentra en el automóvil, la cual da una señal por medio auditivo. Cuenta con 4 alarmas distintas *APAGADA* = 00 - *ALARMA DE EMERGENCIA* = 01 - *ALARMA PANICO* = 10 *SONAR* = 11.
- Sirena (SR) = La sirena se encuentra en el automóvil, la cual da una señal visual por medio de colores de luces. Cuenta con 4 configuraciones distintas. *APAGADA* = 00 - *SIRENA DE PANICO* = 01 - *SIRENA DE EMERGENCIA* = 10 - *BLINK* = 11.
- Polarizado (PO) = El polarizado se encuentra en las ventanas del automóvil, cuenta con 3 intensidades de polarizado distintos. *SUAVE* = 00 - *MEDIO* = 01 - *FUERTE* = 10.
- Mensaje de Emergencia (ME) = El mensaje de emergencia, manda una señal de GPS en tiempo real a uno o varios números de teléfonos que indique el usuario, el cual se activa en el modo de Pánico, Emergencia y si siguen a la persona. *1* = *Mensaje de ubicación enviándose*. *0* = *no envía ningún mensaje*.
- Señal P (P) = Esta señal es la que se envía a la FSM1, la cual se activa en los casos de modo de Pánico, Emergencia y siguen a la persona y si el sensor de golpe se activa. *1* = *Señal activa*. *0* = *Señal desactivada*.

## FSM Anti-rebote



La FSM Anti-rebote su función es que, aunque se mantenga presionado un botón sin soltarlo, no provoque cambios de estados en cada flanco de reloj, si no que hasta que detecte otra señal de flanco alto se active. Este se colocará en los botones de *Locked* y *Unlocked*.

## Diagramas de Estados de Transición

### FSM 1

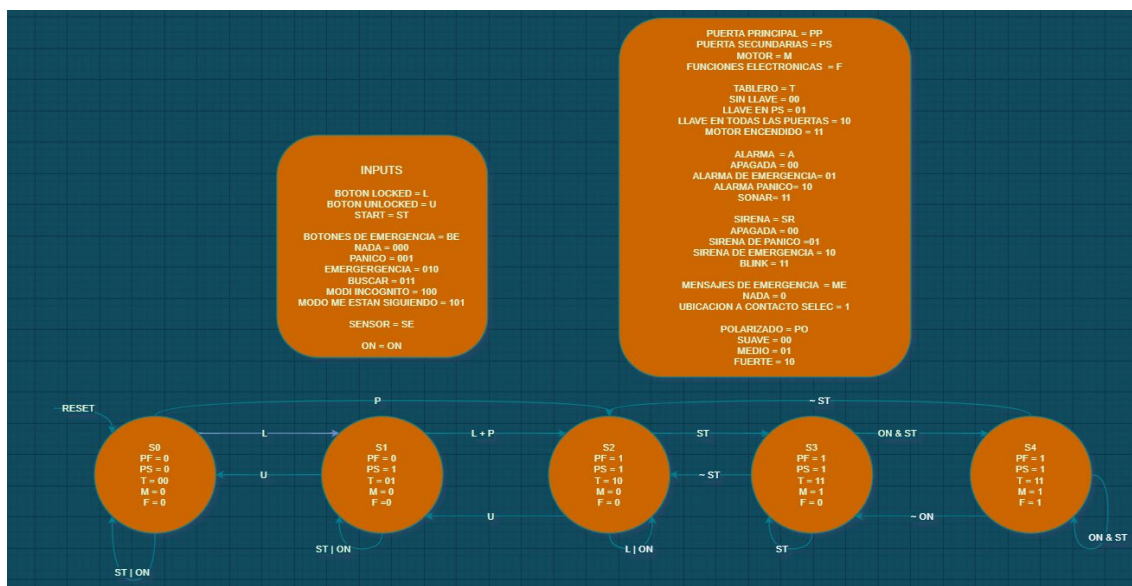


Tabla de estados FSM 1

L = Locked U = Unlocked

P = Señal P

ST = Start

ON = ON

TABLA DE ESTADOS SIN CODIFICAR								TABLA DE ESTADOS CODIFICADA										
S	L	U	P	ST	ON	S'		S2	S1	S0	L	U	P	ST	ON	S'2	S'1	S'0
S0	0	0	0	1	0	S'0		0	0	0	0	0	0	1	0	0	0	0
S0	0	0	0	0	0	S'0		0	0	0	0	0	0	0	0	0	0	0
S0	1	0	0	0	0	S'1		0	0	0	1	0	0	0	0	0	0	1
S0	X	X	1	X	X	S'2		0	0	0	X	X	1	X	X	0	1	0
S0	0	0	0	0	1	S'0		0	0	0	0	0	0	0	1	0	0	0
S0	1	1	0	X	X	S'0		0	0	0	1	1	0	0	0	0	0	0
S1	0	0	0	1	0	S'1		0	0	1	0	0	0	1	0	0	0	1
S1	0	1	0	X	X	S'0		0	0	1	0	1	0	X	X	0	0	0
S1	1	0	0	X	X	S'2		0	0	1	1	0	X	X	X	0	1	0
S1	X	X	1	X	X	S'2		0	0	1	X	X	1	X	X	0	1	0
S1	0	0	0	0	1	S'1		0	0	1	0	0	0	0	1	0	0	1
S1	0	0	0	0	0	S'1		0	0	1	0	0	0	0	0	0	0	1
S1	1	1	0	X	X	S'1		0	0	1	1	1	0	X	X	0	0	1
S2	1	0	0	0	0	S'2		0	1	0	1	0	0	0	X	0	1	0
S2	0	1	0	X	X	S'1		0	1	0	0	1	0	X	X	0	0	1
S2	X	X	X	1	X	S'3		0	1	0	X	X	X	1	X	0	1	1
S2	0	0	0	0	1	S'2		0	1	0	0	0	0	0	1	0	1	0
S2	0	0	0	0	0	S'2		0	1	0	0	0	0	0	0	0	1	0
S2	1	1	0	0	0	S'2		0	1	0	1	1	X	X	X	0	1	0
S2	X	X	1	X	X	S'2		0	1	0	X	X	1	0	X	0	1	0
S3	X	X	X	1	0	S'3		0	1	1	X	X	X	1	0	0	1	1
S3	X	X	X	1	1	S'4		0	1	1	X	X	X	1	1	1	0	0
S3	X	X	X	0	X	S'2		0	1	1	X	X	X	0	X	0	1	0
S4	X	X	X	1	1	S'4		1	0	0	X	X	X	1	1	1	0	0
S4	X	X	X	1	0	S'3		1	0	0	X	X	X	1	0	0	1	1
S4	X	X	X	0	X	S'2		1	0	0	X	X	X	0	X	0	1	0

Estado	Codificación
S0	000
S1	001
S2	010
S3	011
S4	100

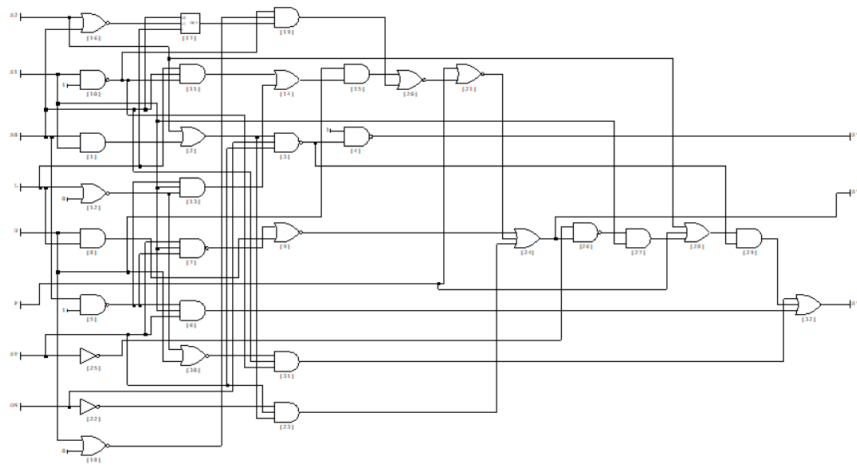
TABLA DE SALIDAS									
S2	S1	S0	PF	PS	T2	T1	M	F	
0	0	0	0	0	0	0	0	0	
0	0	1	0	1	0	1	0	0	
0	1	0	1	1	1	0	0	0	
0	1	1	1	1	1	1	1	0	
1	0	0	1	1	1	1	1	1	

Prioridades o combinaciones a tomar en cuenta en las tablas y los diagramas de estados

- Si los botones de Locked (L) y Unlocked (U) se presionan en el mismo momento, no ocurre ningún cambio de estado.
- La Señal P (P) tiene prioridad sobre el botón Unlocked (U).
- No se puede encender el motor (M) a menos que todas las puertas tengan seguro.
- No se puede encender las funciones electrónicas (F) si el motor no está encendido, esto para proteger la batería del automóvil y evitar descargas de la batería.

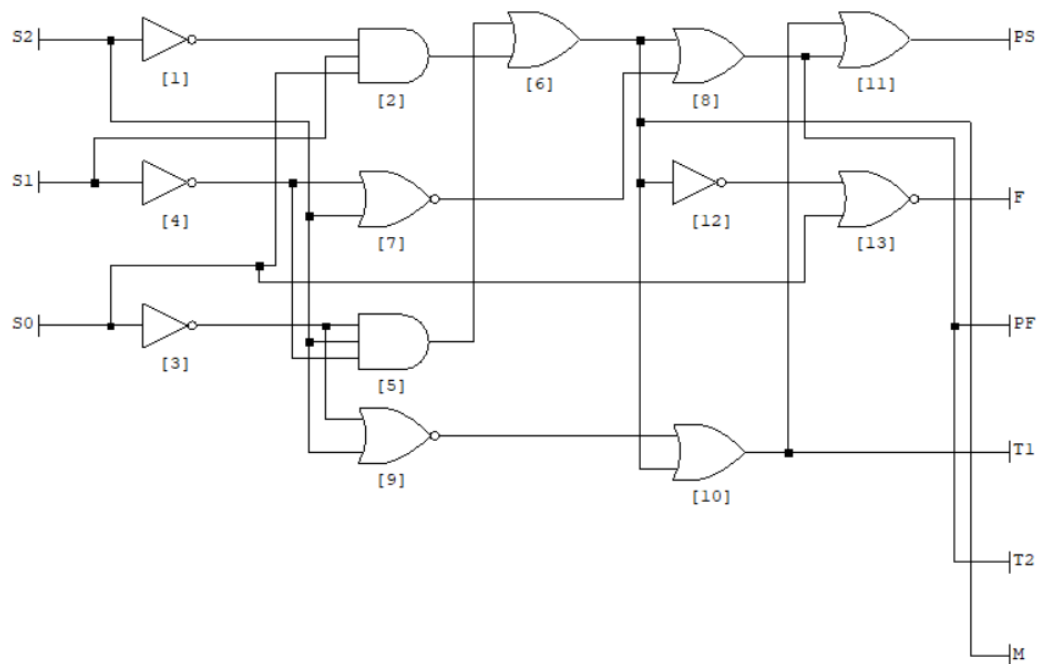
## Ecuaciones de Estados FSM 1, obtenidas de Logic friday

Minimized:  
 $S'2 = S2 \text{ ST ON} + S1 \text{ S0 ST ON};$   
 $S'1 = S2 \text{ ST}' + S2 \text{ ON}' + S1 \text{ S0}' \text{ ST} + S1 \text{ S0 ST}' + S1' \text{ S0 L U}' + S2' \text{ S1}' \text{ P} + S2' \text{ S0}' \text{ P} + S1 \text{ L ST}' + S1 \text{ U}' \text{ ST}' + S1 \text{ ST ON}';$   
 $S'0 = S2 \text{ ST ON}' + S1 \text{ S0}' \text{ L}' \text{ ST} + S1 \text{ S0}' \text{ U}' \text{ ST} + S1 \text{ S0 ST ON}' + S1' \text{ S0 L U P}' + S1 \text{ S0}' \text{ L}' \text{ U P}' + S1' \text{ S0 L}' \text{ U}' \text{ P}' + S2' \text{ S1}' \text{ S0}' \text{ L U}' \text{ P}' ;$

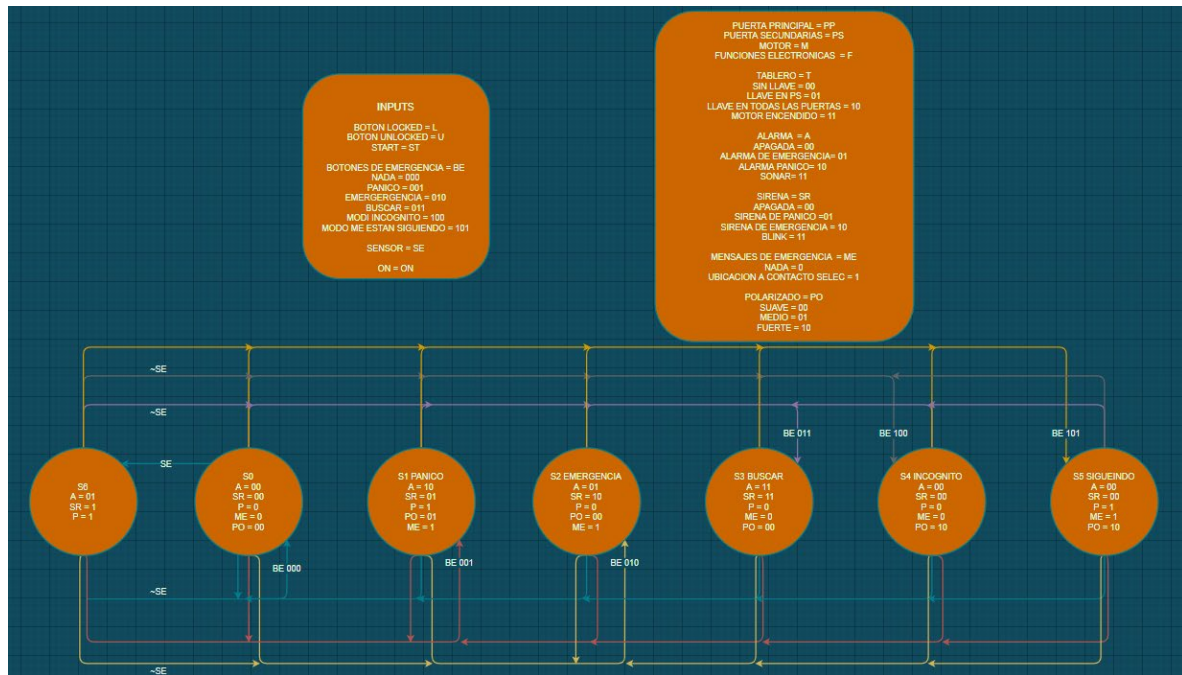


## Ecuaciones de Salidas de la FSM 1, obtenidas de Logic friday

Minimized:  
 $PF = S2' \text{ S1} + S2 \text{ S1}' \text{ S0}';$   
 $PS = S2' \text{ S1} + S2' \text{ S0} + S2 \text{ S1}' \text{ S0}';$   
 $T2 = S2' \text{ S1} + S2 \text{ S1}' \text{ S0}';$   
 $T1 = S2' \text{ S0} + S2 \text{ S1}' \text{ S0}';$   
 $M = S2' \text{ S1 S0} + S2 \text{ S1}' \text{ S0}';$   
 $F = S2 \text{ S1}' \text{ S0}';$



## FSM 2



### Tabla de estados FSM 2

BE2 = Botón de Emergencia 3

BE1 = Botón de Emergencia 2

BE1 = Botón de Emergencia 1

SE = Sensor de Golpe

TABLA DE ESTADOS SIN CODIFICAR						TABLA DE ESTADOS CODIFICADA											
S	BE2	BE1	BE0	SE	S'	S2	S1	S0	BE2	BE1	BE0	SE	S'2	S'1	S'0		
S0	0	0	0	0	S'0	0	0	0	0	0	0	0	0	0	0		
S0	0	0	1	X	S'1	0	0	0	0	0	1	X	0	0	1		
S0	0	1	0	X	S'2	0	0	0	0	1	0	X	0	1	0		
S0	0	1	1	X	S'3	0	0	0	0	1	1	X	0	1	1		
S0	1	0	0	X	S'4	0	0	0	1	0	0	X	1	0	0		
S0	1	0	1	X	S'5	0	0	0	1	0	1	X	1	0	1		
S0	0	0	0	1	S'6	0	0	0	0	0	0	1	1	1	0		
S0	1	1	X	0	S'0	0	0	0	1	1	X	0	0	0	0		
S0	1	1	X	1	S'6	0	0	0	1	1	X	1	1	1	0		
S1	0	0	0	X	S'0	0	0	1	0	0	0	X	0	0	0		
S1	0	0	1	X	S'1	0	0	1	0	0	1	X	0	0	1		
S1	0	1	0	X	S'2	0	0	1	0	1	0	X	0	1	0		
S1	0	1	1	X	S'3	0	0	1	0	1	1	X	0	1	1		
S1	1	0	0	X	S'4	0	0	1	1	0	0	X	1	0	0		
S1	1	0	1	X	S'5	0	0	1	1	0	1	X	1	0	1		
S1	1	1	X	X	S'1	0	0	1	1	1	X	X	0	0	1		
S2	0	0	0	X	S'0	0	1	0	0	0	0	X	0	0	0		
S2	0	0	1	X	S'1	0	1	0	0	0	1	X	0	0	1		
S2	0	1	0	X	S'2	0	1	0	0	1	0	X	0	1	0		
S2	0	1	1	X	S'3	0	1	0	0	1	1	X	0	1	1		
S2	1	0	0	X	S'4	0	1	0	1	0	0	X	1	0	0		
S2	1	0	1	X	S'5	0	1	0	1	0	1	X	1	0	1		
S2	1	1	X	X	S'2	0	1	0	1	1	X	X	0	1	0		



S3	0	0	0	X		S'0		0	1	1	0	0	0	X		0	0	0
S3	0	0	1	X		S'1		0	1	1	0	0	1	X		0	0	1
S3	0	1	0	X		S'2		0	1	1	0	1	0	X		0	1	0
S3	0	1	1	X		S'3		0	1	1	0	1	1	X		0	1	1
S3	1	0	0	X		S'4		0	1	1	1	0	0	X		1	0	0
S3	1	0	1	X		S'5		0	1	1	1	0	1	X		1	0	1
S3	1	1	X	X		S'3		0	1	1	1	1	X	X		0	1	1
S4	0	0	0	X		S'0		1	0	0	0	0	0	X		0	0	0
S4	0	0	1	X		S'1		1	0	0	0	0	1	X		0	0	1
S4	0	1	0	X		S'2		1	0	0	0	1	0	X		0	1	0
S4	0	1	1	X		S'3		1	0	0	0	1	1	X		0	1	1
S4	1	0	0	X		S'4		1	0	0	1	0	0	X		1	0	0
S4	1	0	1	X		S'5		1	0	0	1	0	1	X		1	0	1
S4	1	1	X	X		S'4		1	0	0	1	1	X	X		1	0	0
S5	0	0	0	X		S'0		1	0	1	0	0	0	X		0	0	0
S5	0	0	1	X		S'1		1	0	1	0	0	1	X		0	0	1
S5	0	1	0	X		S'2		1	0	1	0	1	0	X		0	1	0
S5	0	1	1	X		S'3		1	0	1	0	1	1	X		0	1	1
S5	1	0	0	X		S'4		1	0	1	1	0	0	X		1	0	0
S5	1	0	1	X		S'5		1	0	1	1	0	1	X		1	0	1
S5	1	1	X	X		S'5		1	0	1	1	1	X	X		1	0	1
S6	0	0	0	0		S'0		1	1	0	0	0	0	0		0	0	0
S6	0	0	1	X		S'1		1	1	0	0	0	1	X		0	0	1
S6	0	1	0	0		S'2		1	1	0	0	1	0	0		0	1	0
S6	0	1	1	0		S'3		1	1	0	0	1	1	0		0	1	1
S6	1	0	0	0		S'4		1	1	0	1	0	0	0		1	0	0
S6	1	0	1	X		S'5		1	1	0	1	0	1	X		1	0	1
S6	X	X	X	1		S'6		1	1	0	X	X	X	1		1	1	0

S

Estado	Codificación	TABLA DE SALIDAS										
		S2	S1	S0	A1	A0	SR1	SR0	P	PO1	PO0	ME
S0	000	0	0	0	0	0	0	0	0	0	0	0
S1	001	0	0	1	1	0	0	1	1	0	1	1
S2	010	0	1	0	0	1	1	0	0	0	0	1
S3	011	0	1	1	1	1	1	1	0	0	0	0
S4	100	1	0	0	0	0	0	0	0	1	0	0
S5	101	1	0	1	0	0	0	0	1	1	0	1
S6	110	1	1	0	0	1	0	1	1	0	1	0

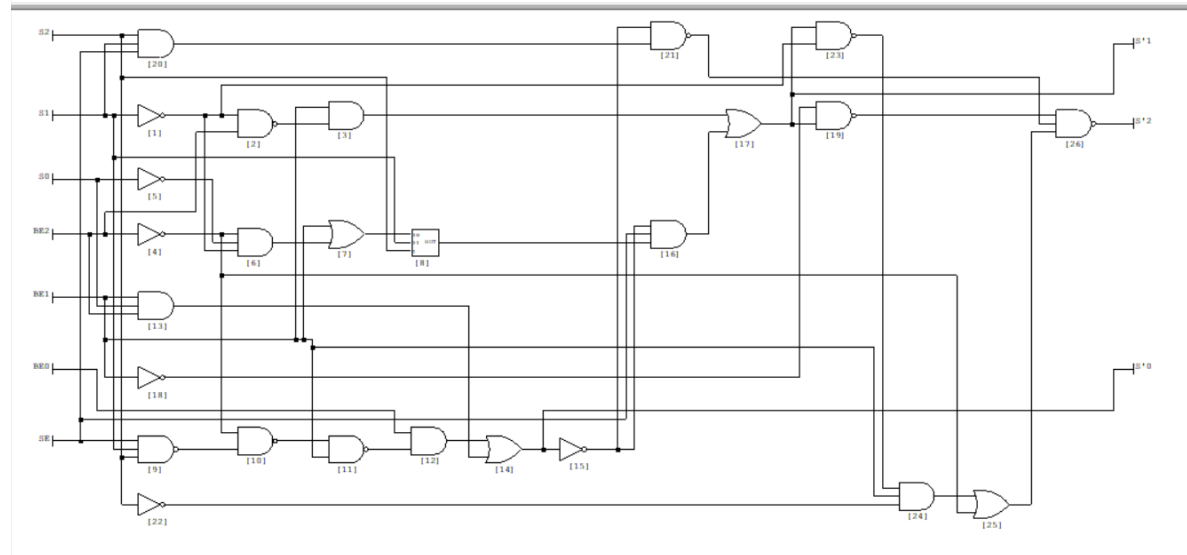
Prioridades o combinaciones a tomar en cuenta en las tablas y los diagramas de estados

- Entre los estados S0 al S5 se puede ir sin restricción mediante la codificación de los Botones de Emergencia
- Cuando se encuentra en S6 y SE = 1, se puede ir al S1 y al S1 y S5, en los otros casos SE tiene que ser 0.
- Las configuraciones de BE = 110 y 111 no existen, por lo tanto si se llegaran a presionar no cambian de estado y se quedan en donde se encontraban



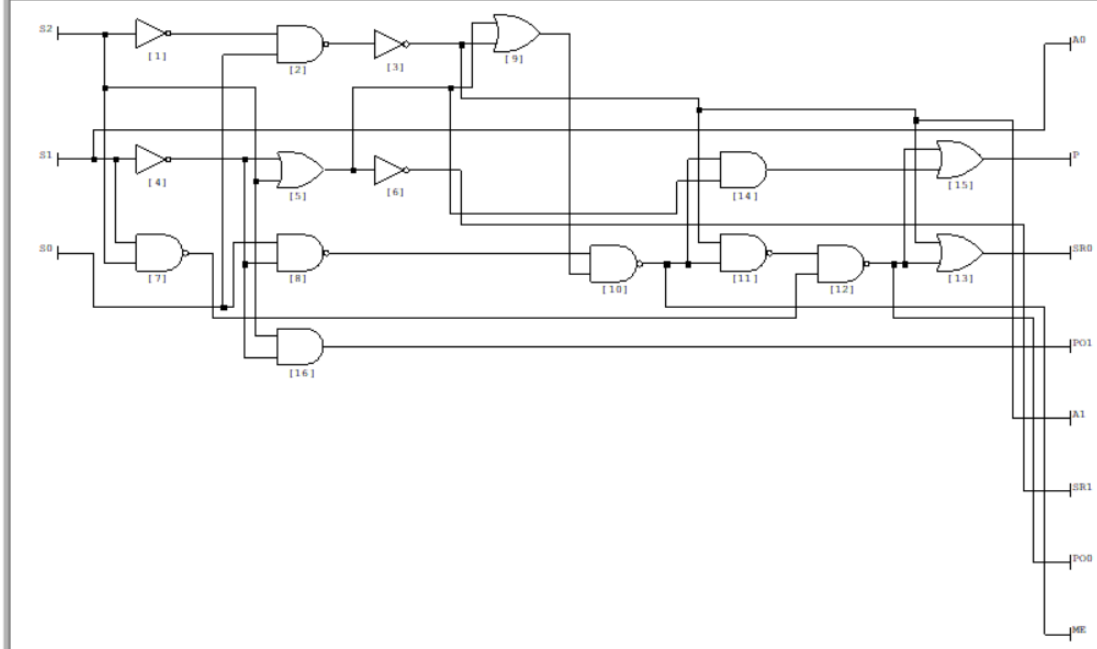
## Ecuaciones de Estados FSM 2, obtenidas de Logic Friday

Minimized:  
 $S'2 = S2 \text{ BE2} + \text{BE2 BE1}' + S1' S0' \text{ BE2 SE} + S2 S1 \text{ BE1 SE} + S2 S1 \text{ BE0}' \text{ SE} + S2' S1' S0' \text{ BE1}' \text{ BE0}' \text{ SE};$   
 $S'1 = S1 \text{ BE1} + \text{BE2}' \text{ BE1} + S2' S0' \text{ BE1 SE} + S2 S1 \text{ BE0}' \text{ SE} + S2' S1' S0' \text{ BE2}' \text{ BE0}' \text{ SE};$   
 $S'0 = \text{BE1}' \text{ BE0} + S0 \text{ BE2 BE1} + S2' \text{ BE2}' \text{ BE0} + S1' \text{ BE2}' \text{ BE0} + \text{BE2}' \text{ BE0 SE};$



## Ecuaciones de Salidas de la FSM 2, obtenidas de Logic Friday

Minimized:  
 $A1 = S2' S0;$   
 $A0 = S1;$   
 $SR1 = S2' S1;$   
 $SR0 = S2 S1 + S2' S0;$   
 $P = S2 S1 + S1' S0;$   
 $PO1 = S2 S1';$   
 $PO0 = S2 S1 + S2' S1' S0;$   
 $ME = S1' S0 + S2' S1 S0';$



## FSM Anti-rebote



La FSM Anti-Rebote sirve para cuando se mantiene presionado el Botón Locked o Unlocked no brinque o cambie varios estados, si no que hasta que detecte otro flanco alto, mande otra señal para que se cambie de estado.

Ecuaciones de Estados FSM Anti-rebote, obtenidas de Logic Friday

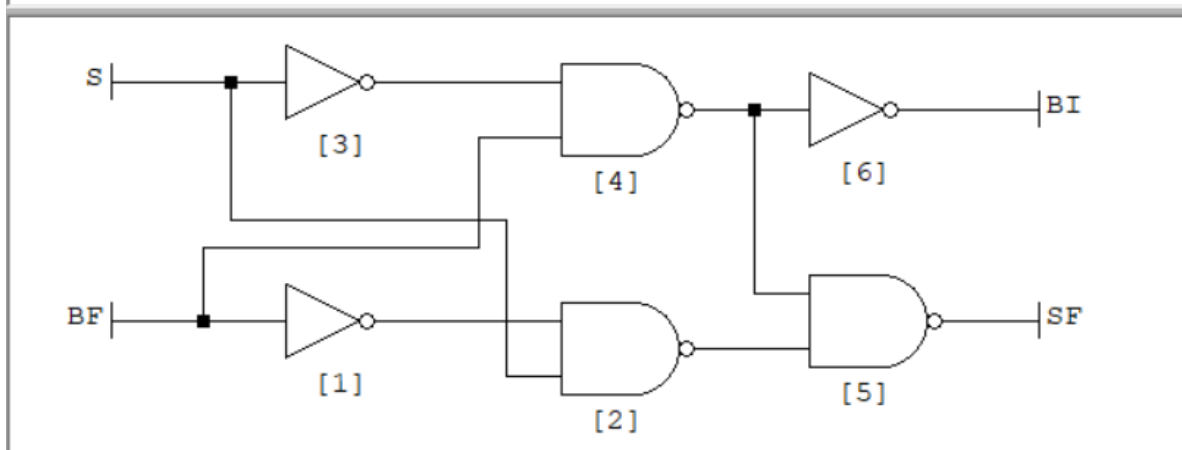
TABLA DE ESTADOS SIN CODIFICAR

S	BF	S'	BI
S0	0	S'0	0
S0	1	S'1	1
S1	1	S'1	0
S1	0	S'0	0

TABLA DE ESTADOS CODIFICADA

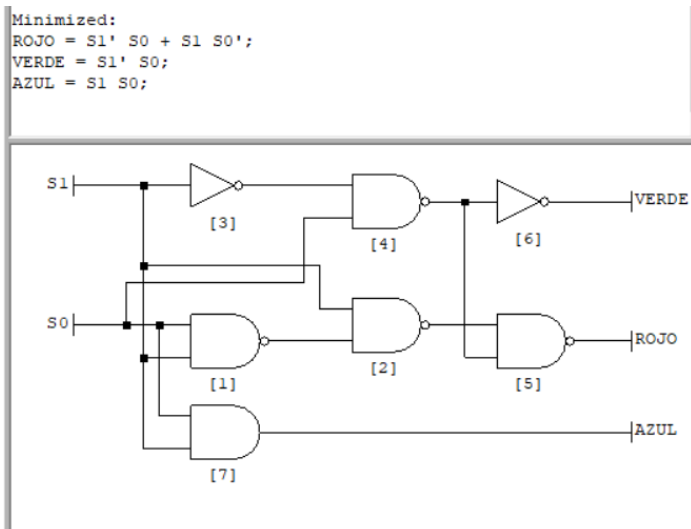
S	BF	S'	BI
0	0	0	0
0	1	1	1
1	1	1	0
1	0	0	0

Minimized:  
 $SF = S' BF + S BF'$   
 $BI = S' BF$



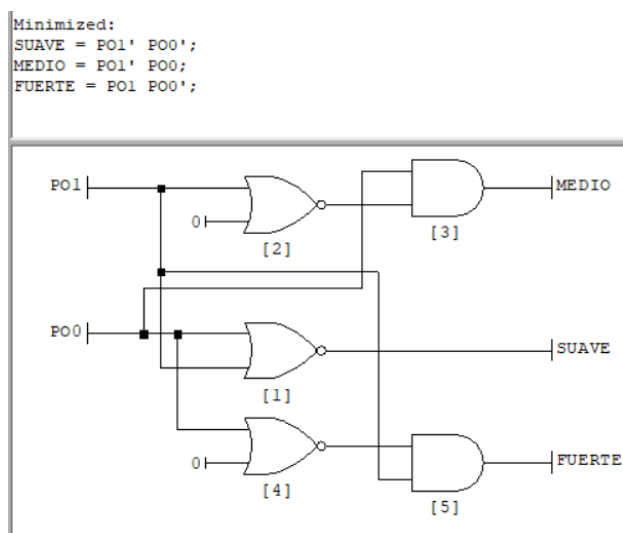
### Control de la Alarma simulado con la LED RGB de circuit verse

BIT1	BIT0		ROJO	VERDE	AZUL
0	0		0	0	0
0	1		1	1	0
1	0		1	0	0
1	1		0	0	1



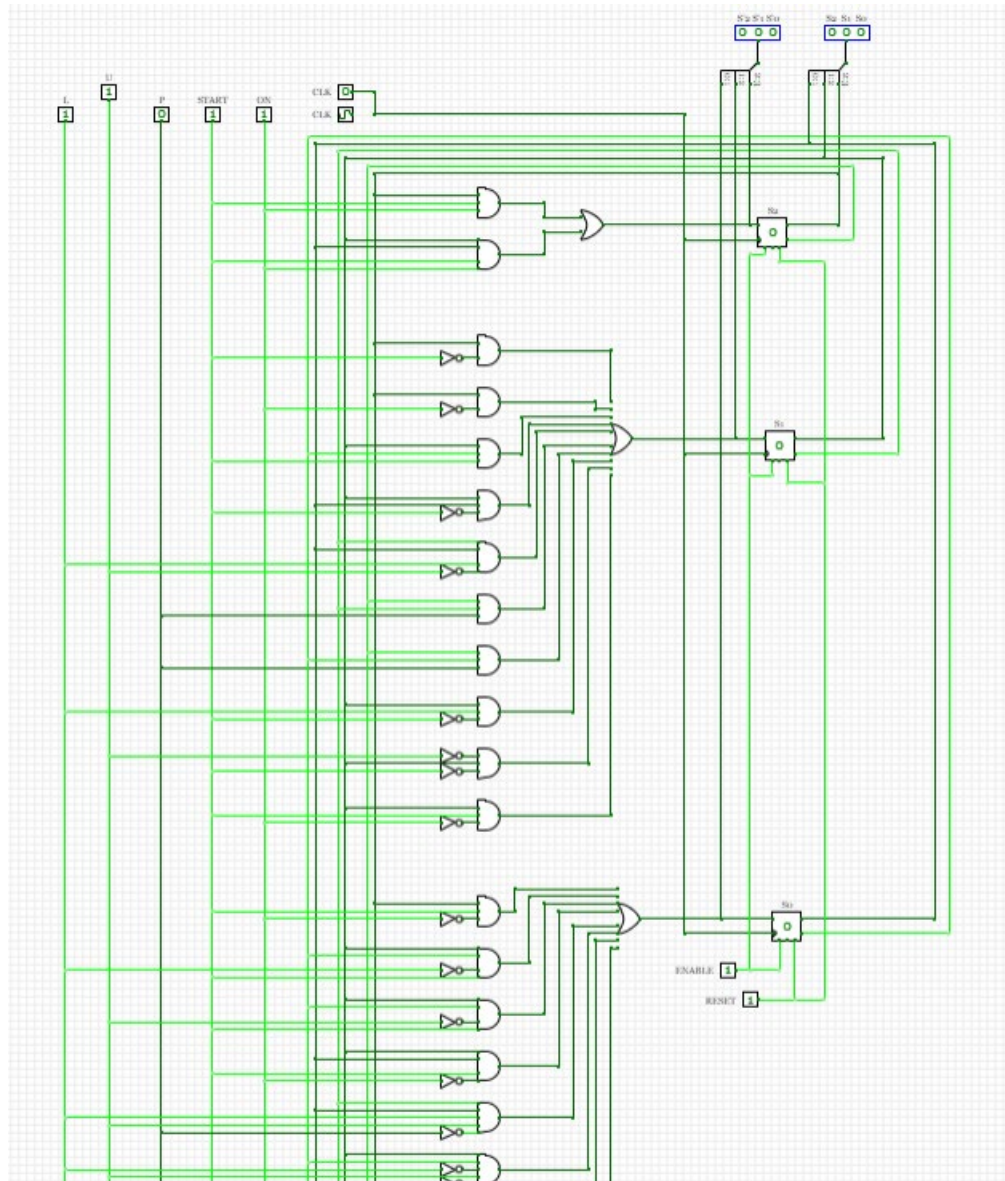
### Control del Polarizado simulado con la LED RGB de circuit verse

BIT1	BIT0		SUAVE	MEDIO	FUERTE
0	0		1	0	0
0	1		0	1	0
1	0		0	0	1
1	1		0	0	0

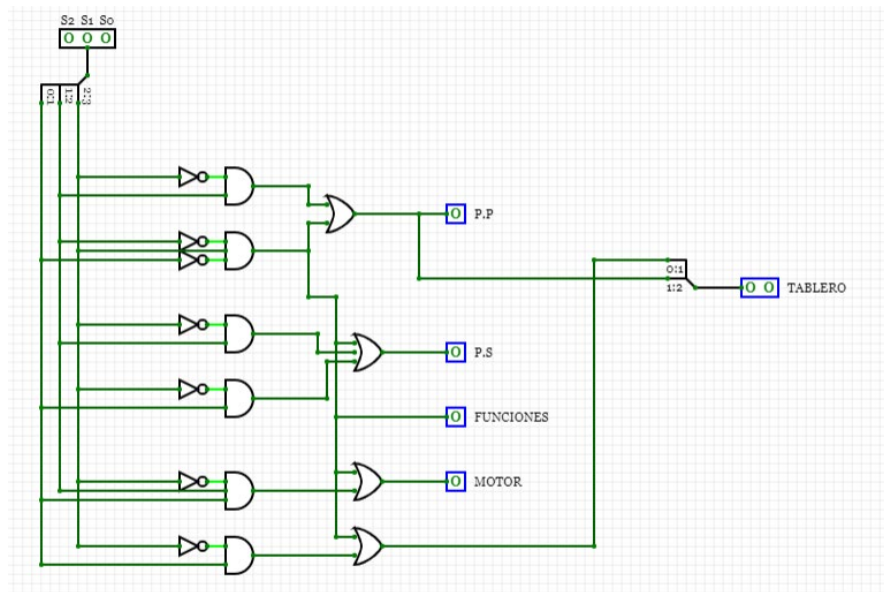


## Circuit Verse

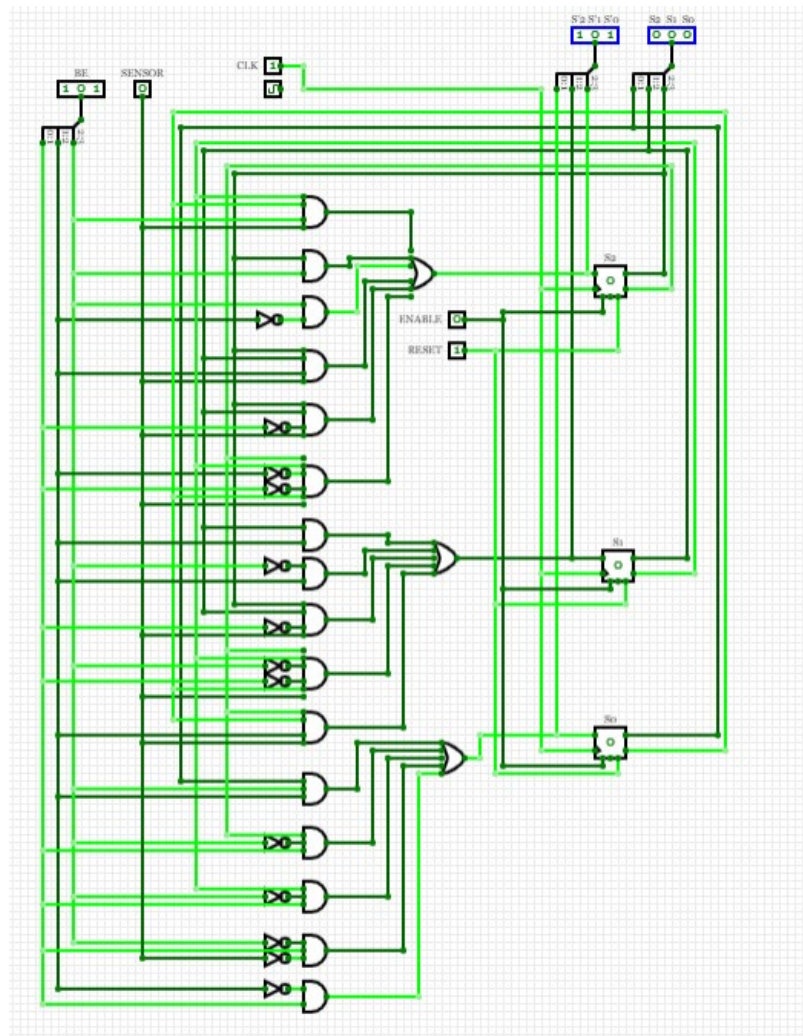
FSM 1. Subcircuito de estados.



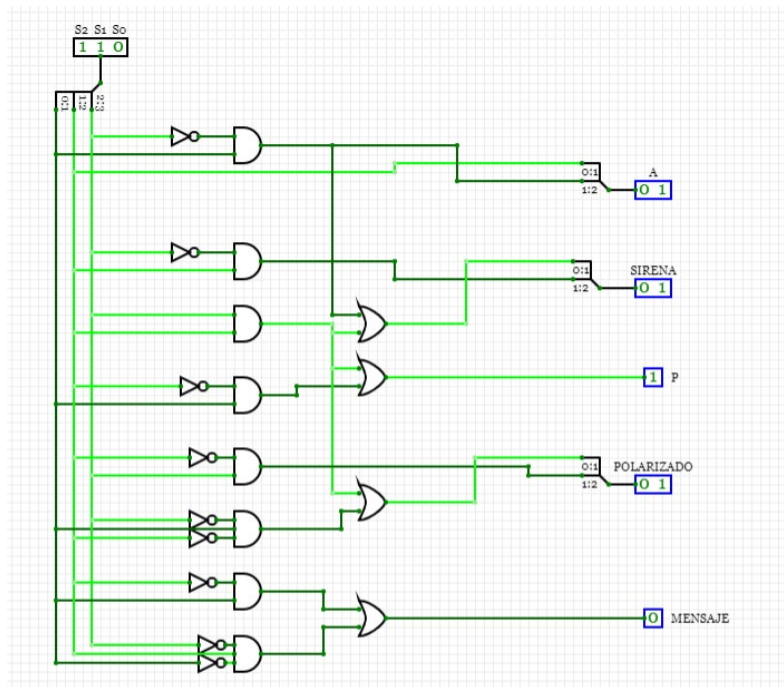
FSM 1. Subcircuito de salidas.



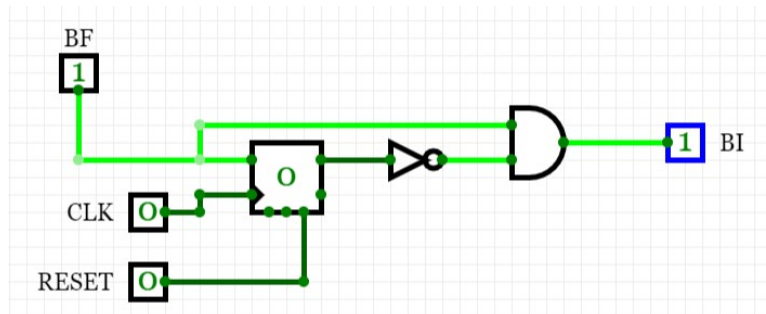
FSM 2. Subcircuito de estados.



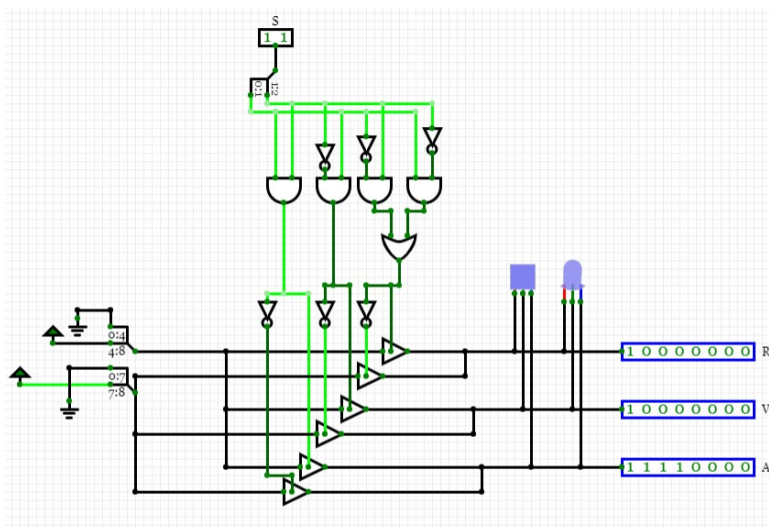
## FSM 1. Subcircuito de salidas.



## FSM Anti-rebote



## Subcircuito Alarma



EL Sub-Circuito de Alarma sirve para simular los distintos tipos de alarma con el LED RGB de circuit verse.

Rojo = Pánico.

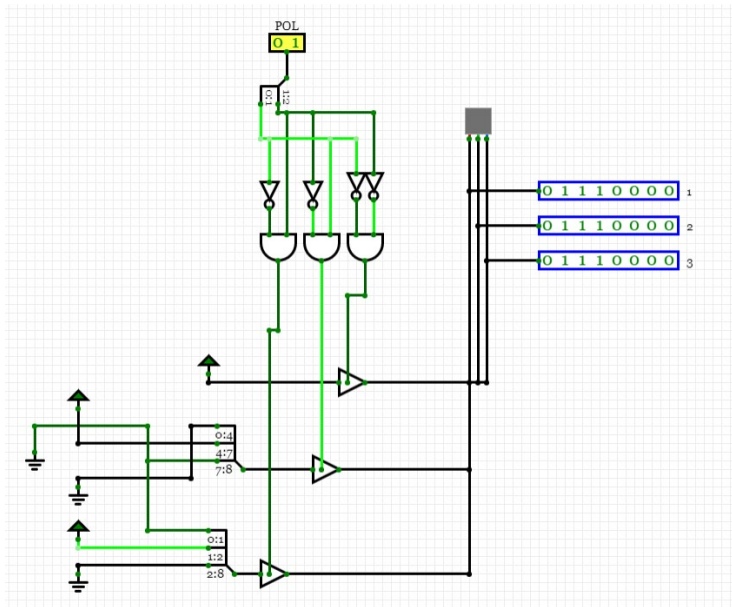
Amarillo = Emergencia

Azul = Sonar

Gris = apagada



## FSM Polarizado



El Sub-Circuito de Alarma sirve para simular los distintos tipos de alarma con el LED RGB de circuit verse.

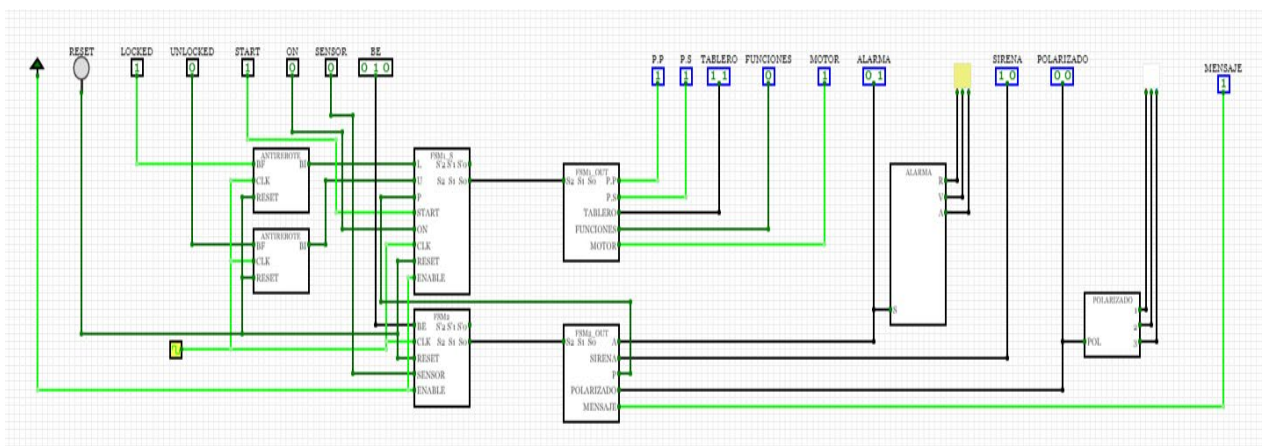
Rojo = Pánico.

Amarillo = Emergencia

Azul = Sonar

Gris = apagada

## FSM COMPLETO





## Código de Programación PROYECTO\_FMS.v

```
1 //PROYECTO MAQUINA DE ESTADOS FINITOS
2 //ELECTRONICA DIGITAL
3 //OCTUBRE DE 2020
4 //GABRIEL ALEXANDER FONG PENAGOS 19722
5 //FSM
6
7 //CREAMOS MODULO FLIP FLOP TIPO D, DE 2 BITS
8 module flipflopD(input wire clk, reset,
9                 input wire D,
10                 output reg Q); //COLOCAMOS NUESTRA SALIDA COMO REG
11     always @(posedge clk, posedge reset) begin //COLOCAMOS ALWAYS PARA QUE SIEMPRE SE PUEDA REALIZAR LA FUNCION
12         if(reset) begin
13             Q <= 1'b0; //COLOCA LA SALIDA CON VALOR 0
14         end
15         else
16             Q <= D; //COLOCA EN LA SALIDA EL VALOR QUE TENGA D
17         end
18     endmodule
19
20 module antirebote(input wire BF,clk, reset,
21                 output wire BI);
22     wire S;
23     flipflopD U1(clk, reset, BF, S);
24     assign BI = (~S & BF);
25 endmodule
26
27 module FSM1(input wire clk, reset, L, U, P, ST, ON,
28            output wire PP, PS, M, F,
29            output wire [1:0]TA,
30            output wire [2:0]SA,
31            output wire [2:0]SF); //SA = ESTADO ACTUAL, SF = ESTADO FUTURO
32
33     wire S2, S1, S0;
34     //REALIZAMOS LAS ECUACIONES PARA LOS ESTADOS FUTUROS
35     assign SF[2] = (S2 & ST & ON) | (S1 & S0 & ST & ON);
36     assign SF[1] = (S2 & ~ST) | (S2 & ~ON) | (S1 & ~S0 & ST) | (S1 & S0 & ~ST) | (~S1 & S0 & L & ~U) | (~S2 & ~S1 & P) | (~S2 &
37     assign SF[0] = (S2 & ST & ~ON) | (S1 & ~S0 & ~L & ST) | (S1 & ~S0 & ~U & ST) | (S1 & S0 & ST & ~ON) | (~S1 & S0 & L & U & ~
38     //REALIZAMOS LAS CONEXIONES PARA NUESTROS FLIP FLOP PARA ESTADOS ACTUALES
39     flipflopD U1(clk, reset,SF[2], S2);
40     flipflopD U2(clk, reset,SF[1], S1);
41     flipflopD U3(clk, reset,SF[0], S0);
```

```
42 module FSM2(input wire clk, reset, SE,
43            input wire [2:0]BE,
44            output wire P, ME,
45            output wire [1:0]A,
46            output wire [1:0]SR,
47            output wire [1:0]PO,
48            output wire [2:0]SA,
49            output wire [2:0]SF);
50
51     assign SA[2] = S2;
52     assign SA[1] = S1;
53     assign SA[0] = S0;
54     //REALIZAMOS LAS ECUACIONES PARA LAS SALIDAS
55     assign PP = (~S2 & S1) | (S2 & ~S1 & ~S0);
56     assign PS = (~S2 & S1) | (~S2 & S0) | (S2 & ~S1 & ~S0);
57     assign TA[1] = (~S2 & S1) | (S2 & ~S1 & ~S0);
58     assign TA[0] = (~S2 & S0) | (S2 & ~S1 & ~S0);
59     assign M = (~S2 & S1 & S0) | (S2 & ~S1 & ~S0);
60     assign F = (S2 & ~S1 & ~S0);
61 endmodule
```

```

64     wire S2, S1, S0, BE2, BE1, BE0;
65     assign BE2 = BE[2];
66     assign BE1 = BE[1];
67     assign BE0 = BE[0];
68     //REALIZAMOS LAS ECUACIONES PARA LOS ESTADOS FUTUROS
69     assign SF[2] = (S2 & BE2) | (BE2 & ~BE1) | (~S1 & ~S0 & BE2 & SE) | (S2 & S1 & BE1 & SE) | (S2 & S1 & ~BE0 & SE) | (~S2 & ~
70     assign SF[1] = (S1 & BE1) | (~BE2 & BE1) | (~S2 & ~S0 & BE1 & SE) | (S2 & S1 & ~BE0 & SE) | (~S2 & ~S1 & ~S0 & ~BE2 & ~BE0
71     assign SF[0] = (~BE1 & BE0) | (S0 & BE2 & BE1) | (~S2 & ~BE2 & BE0) | (~S1 & ~BE2 & BE0) | (~BE2 & BE0 & ~SE);
72     //REALIZAMOS LAS CONEXIONES DE LOS FLIP FLOPS PARA LOS ESTADOS ACTUALES
73     flipflopD U1(clk, reset, SF[2], S2);
74     flipflopD U2(clk, reset, SF[1], S1);
75     flipflopD U3(clk, reset, SF[0], S0);
76
77     assign SA[2] = S2;
78     assign SA[1] = S1;
79     assign SA[0] = S0;
80
81     //REALIZAMOS LAS ECUACIONES PARA LAS SALIDAS
82     assign A[1] = (~S2 & S0);
83     assign A[0] = S1;
84     assign SR[1] = (~S2 & S1);
85     assign SR[0] = (S2 & S1) | (~S2 & S0);
86     assign P = (S2 & S1) | (~S1 & S0);
87     assign PO[1] = (S2 & ~S1);
88     assign PO[0] = (S2 & S1) | (~S2 & ~S1 & S0);
89     assign ME = (~S1 & S0) | (~S2 & S1 & ~S0);
90 endmodule
91

```

## Código de programación PROYECTO\_FSM\_tb.v

```

PROYECTO_FSM_tb.v
1 //PROYECTO MAQUINA DE ESTADOS FINITOS
2 //ELECTRONICA DIGITAL
3 //OCTUBRE DE 2020
4 //GABRIEL ALEXANDER FONG PENAGOS 19722
5 //TESTBENCH
6
7 module testbench ();
8     //VARIABLES DE LA FSM1
9     reg clk, reset1;
10    real t;
11    reg L, U, P, ST, ON; //ENTRADAS FSM1
12    wire LI, UI;
13    wire PP, PS, M, F; //SALIDAS DE 1 BIT FSM1
14    wire [1:0]TA; //SALIDAS DE 2 BIT FSM1
15    wire [2:0]SA1; //ESTADOS ACTUALES
16    wire [2:0]SF1; //ESTADOS FUTUROS
17
18    //VARIABLES DE LA FSM2
19    reg SE;
20    reg [2:0]BE; //ENTRADA DE 3 BITS FSM2
21    wire P2, ME; //SALIDA DE 1 BIT FSM2
22    wire [1:0]A;
23    wire [1:0]SR;
24    wire [1:0]PO; //SALIDAS DE 2 BITS FSM2
25    wire [2:0]SA2;
26    wire [2:0]SF2; //ESTADOS DE FSM2
27
28
29 always //REALIZAMOS LA SECUENCIA DEL CLK CAMBIO DE FLANCO DE 1 UNIDAD DE TIEMPO
30 begin
31     clk <= 0; #1 clk <= ~clk; #1;
32 end
33 always //ESTE CONTEO t SE UTILIZARA PARA MOSTRAR EN QUE UNIDAD DE TIEMPO SE ENCUENTRA Y SE PUEDA VISUALIZAR EN EL MONITOR
34 begin
35     t <= t + 1; #1;
36 end
37
38 antirebote U0(L, clk, reset1, LI); //FSM ANTIREBOTE PARA LOCKED
39 antirebote U1(U, clk, reset1, UI); //FSM ANTIREBOTE PARA UNLOCKED
40 FSM1 U2(clk, reset1, LI, UI, P, ST, ON, PP, PS, M, F, TA, SA1, SF1); //FSM 1
41 FSM2 U3(clk, reset1, SE, BE, P2, ME, A, SR, PO, SA2, SF2); //FSM 2
42

```

```

42 initial begin
43     reset1 = 1;
44     L = 0; U = 0; P = 0; ST = 0; ON = 0; t = 0; //INICIALIZAMOS NUESTRAS ENTRADAS RESETEAMOS NUESTRAS FSM
45     #1
46     $display("\n");
47     $display("ESTADO ACTUAL = SA \n ESTADO FUTURO = SF \n PP = PUERTA PILOTO \n PS = PUERTA SECUNDARIA \n M = MOTOR \n F = FUNCIONES ELECTRICAS \n TA = TABLERO \n");
48     $display("t \t CLOCK \t L \t U \t P \t ST \t ON | \t SA \t SF \t PP \t PS \t M \t F \t TA ");
49     $monitor("%d \t %b \t %b \t %b \t %b \t %b \t %b | \t %b \t %b \t %b \t %b \t %b \t %b",t, clk, L, U, P, ST, ON, SA[2:0], SF[2:0], PP, PS, M, F, TA);
50     reset1 = 0;
51     #1 ST = 1; //INTENTAMOS ENCENDER EL MOTOR
52     #2 ST = 0; ON = 1; //INTENTAMOS ENCENDER LAS FUNCIONES
53     #4 L = 1; ON = 0; //COLOCAMOS SEGURO A PS
54     #4 L = 0; U = 1; //QUITAMOS SEGURO, DEJAMOS PASAR 4 UNIDADES DE TIEMPO PARA OBSERVAR SI LA FUNCION DE ANTIREBOTE DE L FUNCIONA
55     #2 P = 1; U = 0; //LA SEÑAL P NOS LLEVA S2, APAGAMOS LA ENTRAADA U PARA PODER ANTIVARLA DESPUES POR EL ANTIREBOTE
56     #2 U = 1; P = 0; //ACTIVAMOS U PARA IR A S1, LA SEÑAL P SE APAGA
57     #2 L = 1; //ACTIVAMOS L PARA REGRESAR S2
58     #2 ST = 1; P = 0; U = 0; L = 0; //ACTIVAMOS ST PARA ENCENDER EL MOTOR
59     #2 ON = 1; //ACTIVAMOS LAS FUNCIONES
60     #2 ST = 0; //APAGAMOS EL MOTOR, TAMBIEN SE APAGA LAS FUNCIONES
61     #2 ST = 1; ON = 0; //ENCENDEMOS EL MOTOR
62     #2 ON = 1; //ENCENDEMOS LAS FUNCIONES
63     #2 ON = 0; //APAGAMOS LAS FUNCIONES
64     #2 ST = 0; //APAGAMOS EL MOTOR
65     #2 U = 1; //QUITAMOS SEGURO EN PP S1
66     #2 U = 0; //DEJAMOS DE PRESIONAR U
67     #2 U = 1; //QUITAMOS SEGURO EN PS S0
68     #2 U = 0; //DEJAMOS DE PRESIONAR U
69     #2 U = 1; L = 1; //SI PRESIONAMOS U Y L NO CAMBIAMOS DE ESTADO

```

```

70     #2 U = 0; L = 0;
71     #2 U = 1; P = 1; //LA SEÑAL P TIENE PRIORIDAD ANTE LA SEÑAL U. VAMOS A S2
72     #2 reset1 = 1; U = 0; P = 0; //RESETEAMOS
73     #1 reset1 = 0;
74     #1 L = 1; P = 1; //LA SEÑAL P TIENE PRIORIDAD ANTE L
75     #2 L = 0; P = 0;
76     #2 ST = 1;
77     #2 ON = 1;
78     end
79
80 initial begin
81     #60 //INICIAMOS LA FSM2
82     SE = 0; BE[2:0] = 0; reset1 = 1;
83     $display("\n");
84     $display("SA = ESTADO ACTUAL \n SF = ESTADO FUTURO \n A = ALARMA \n BE = BOTONES DE EMERGENCIA \n SE = SENSOR DE GOLPE");
85     $display("t \t CLOCK \t BE \t SE | \t SA \t SF \t A \t SR \t PO \t ME \t P ");
86     $monitor("%d \t %b \t %b \t %b | \t %b \t %b \t %b \t %b \t %b \t %b",t,clk,BE, SE, SA[2:0], SF[2:0], A[1:0], SR[1:0], PO[1:0], ME, P2);
87     #2 BE = 3'b001; reset1 = 0; //CAMBIAMOS A ESTADO DE PANICO
88     #2 BE = 3'b010; //DE ESTADO DE PANICO CAMBIAMOS A EMERGENCIA
89     #2 BE = 3'b011; //DE ESTADO DE EMERGENCIA CAMBIAMOS BUSCAR
90     #2 BE = 3'b100; //DE ESTADO BUSCAR A MODO INCOGNITO
91     #2 BE = 3'b101; //DE ESTADO INCOGNITO A MODO ME ESTAN SIGUIENDO
92     #2 BE = 3'b110; //COMPROBAMOS QUE LA CODIFICACION 110 NO CAMBIA DE A NINGUN ESTADO POR QUE YA NO EXISTEN MAS EMERGENCIAS
93     #2 BE = 3'b111; //COMPROBAMOS QUE LA CODIFICACION 111 NO CAMBIA DE A NINGUN ESTADO
94     #2 BE = 3'b000; //REGRESAMOS AL ESTADO INICIAL
95     #2 SE = 1; //SE ACTIVA EL SENSOR DE GOLPE VAMOS A S6
96     #2 SE = 0; //APAGAMOS EL SENSOR PARA DEMOSTRAR UNA CONDICION DE SALIDA. SEGUIMOS EN S6
97     #2 BE = 3'b000; //SALIMOS DE S6 Y VAMOS A S0
98     #2 SE = 1; //SE ACTIVA EL SENSOR DE GOLPE VAMOS A S6
99     #2 BE = 3'b001; //CON SE = 1 PODEMOS SALIR DE S6 Y VAMOS A MODO PANICO
100     #2 SE = 0; BE = 3'b000; //REGRESAMOS A CONDICION DE ESTADO INICIAL
101     #2 SE = 1; //SE ACTIVA EL SENSOR DE GOLPE VAMOS A S6
102     #2 BE = 3'b101; //CON SE = 1 PODEMOS SALIR DE S6 Y VAMOS A MODO ME ESTAN SIGUIENDO
103     #2 BE = 3'b000; SE = 0; //REGRESAMOS A CONDICION DE ESTADO INICIAL
104     #2 SE = 1; //SE ACTIVA EL SENSOR DE GOLPE VAMOS A S6
105     #2 BE = 3'b010; //COMPROBAMOS QUE CON LAS OTRAS CONFIGURACIONES NO LOGRAMOS SALIR DE S6
106     #2 BE = 3'b011;
107     #2 BE = 3'b100;
108     #2 BE = 3'b101; // SALIMOS DE S6 VAMOS A PROBAR LA MAYORIA DE CAMBIOS DE ESTADO
109     #2 BE = 3'b001; //DE ME ESTAN SIGUIENDO VAMOS A MODO PANICO
110     #2 BE = 3'b011; //DE PANICO VAMOS A BUSCAR
111     #2 BE = 3'b101; //DE BUSCAR A ME ESTAN SIGUIENDO
112     #2 BE = 3'b001; //DE ME ESTAN SIGUIENDO A PANICO
113     #2 BE = 3'b100; //DE PANICO A MODO INCOGNITO
114     #2 BE = 3'b001; //DE INCOGNITO A PANICO
115     #2 BE = 3'b101; //DE PANICO A ME ESTAN SIGUIENDO
116     #2 BE = 3'b010; //DE ME ESTAN SIGUIENDO A EMERGENCIA
117     #2 BE = 3'b100; //DE EMERGENCIA A MODO INCOGNITO
118     #2 BE = 3'b010; //DE INCOGNITO A EMERGENCIA
119     #2 BE = 3'b101; //DE EMERGENCIA A ME ESTAN SIGUIENDO
120     #2 BE = 3'b011; //DE ME ESTAN SIGUIENDO A BUSCAR
121     #2 BE = 3'b101; //DE A BUSCAR A ME ESTAN SIGUIENDO
122     #2 BE = 3'b100; //DE ME ESTAN SIGUIENDO A MODO INCOGNITO
123     #2 $finish;
124     end

```

```

125 initial begin //INICIAMOS PARA COLOCAR LOS DATOS EN GTK WAVE
126     $dumpfile("PROYECTO_FSM_tb.vcd");
127     $dumpvars(0, testbench);
128     end
129 endmodule

```

Diagrama de Timing

FSM1

ESTADO ACTUAL = SA  
ESTADO FUTURO = SF  
PP = PUERTA PILOTO  
PS = PUERTA SECUNDARIA  
M = MOTOR  
F = FUNCIONES ELECTRICAS  
TA = TABLERO  
L = LOCKED  
U = UNLOCKED  
P = SEAL/AL P  
ST = STAR

t	CLOCK	L	U	P	ST	ON	SA	SF	PP	PS	M	F	TA
1	1	0	0	0	0	0	000	000	0	0	0	0	00
2	0	0	0	0	1	0	000	000	0	0	0	0	00
3	1	0	0	0	1	0	000	000	0	0	0	0	00
4	0	0	0	0	1	0	000	000	0	0	0	0	00
5	0	0	0	0	0	1	000	000	0	0	0	0	00
6	1	0	0	0	0	1	000	000	0	0	0	0	00
7	0	0	0	0	0	1	000	000	0	0	0	0	00
8	1	0	0	0	0	1	000	000	0	0	0	0	00
9	0	1	0	0	0	0	000	001	0	0	0	0	00
10	1	1	0	0	0	0	001	001	0	1	0	0	01
11	0	1	0	0	0	0	001	001	0	1	0	0	01
12	1	1	0	0	0	0	001	001	0	1	0	0	01
13	0	0	0	1	0	0	001	000	0	1	0	0	01
14	1	0	0	1	0	0	000	000	0	0	0	0	00
15	0	0	0	1	0	0	000	010	0	0	0	0	00
16	1	0	0	1	0	0	010	010	1	1	0	0	10
17	0	0	1	0	0	0	010	001	1	1	0	0	10
18	1	0	1	0	0	0	001	001	0	1	0	0	01
19	0	1	1	0	0	0	001	010	0	1	0	0	01
20	1	1	1	0	0	0	010	010	1	1	0	0	10
21	0	0	0	0	1	0	010	011	1	1	0	0	10
22	1	0	0	0	1	0	011	011	1	1	1	0	11
23	0	0	0	0	1	1	011	100	1	1	1	0	11
24	1	0	0	0	1	1	100	100	1	1	1	1	11
25	0	0	0	0	0	1	100	010	1	1	1	1	11
26	1	0	0	0	0	1	010	010	1	1	0	0	10
27	0	0	0	0	1	0	010	011	1	1	0	0	10
28	1	0	0	0	1	0	011	011	1	1	1	0	11
29	0	0	0	0	1	1	011	100	1	1	1	0	11
30	1	0	0	0	1	1	100	100	1	1	1	1	11
31	0	0	0	0	1	0	100	011	1	1	1	1	11
32	1	0	0	0	1	0	011	011	1	1	1	0	11
33	0	0	0	0	0	0	011	010	1	1	1	0	11
34	1	0	0	0	0	0	010	010	1	1	0	0	10
35	0	0	1	0	0	0	010	001	1	1	0	0	10
36	1	0	1	0	0	0	001	001	0	1	0	0	01
37	0	0	0	0	0	0	001	001	0	1	0	0	01

38	1	0	0	0	0	0	0	001	001	0	1	0	0	01
39	0	0	1	0	0	0	0	001	000	0	1	0	0	01
40	1	0	1	0	0	0	0	000	000	0	0	0	0	00
41	0	0	0	0	0	0	0	000	000	0	0	0	0	00
42	1	0	0	0	0	0	0	000	000	0	0	0	0	00
43	0	1	1	0	0	0	0	000	000	0	0	0	0	00
44	1	1	1	0	0	0	0	000	000	0	0	0	0	00
45	0	0	0	0	0	0	0	000	000	0	0	0	0	00
46	1	0	0	0	0	0	0	000	000	0	0	0	0	00
47	0	0	1	1	0	0	0	000	010	0	0	0	0	00
48	1	0	1	1	0	0	0	010	010	1	1	0	0	10
49	0	0	0	0	0	0	0	000	000	0	0	0	0	00
50	1	0	0	0	0	0	0	000	000	0	0	0	0	00
51	0	1	0	1	0	0	0	000	010	0	0	0	0	00
52	1	1	0	1	0	0	0	010	010	1	1	0	0	10
53	0	0	0	0	0	0	0	010	010	1	1	0	0	10
54	1	0	0	0	0	0	0	010	010	1	1	0	0	10
55	0	0	0	0	1	0	0	010	011	1	1	0	0	10
56	1	0	0	0	1	0	0	011	011	1	1	1	0	11
57	0	0	0	0	1	1	0	011	100	1	1	1	0	11
58	1	0	0	0	0	1	1	100	100	1	1	1	1	11
59	0	0	0	0	1	1	1	100	100	1	1	1	1	11
60	1	0	0	0	0	1	1	100	100	1	1	1	1	11

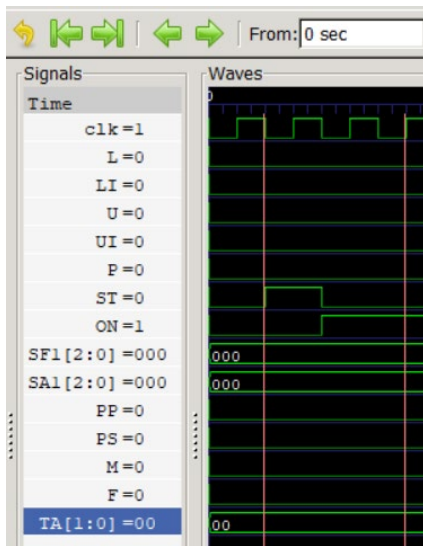
FSM2

SA = ESTADO ACTUAL  
SF = ESTADO FUTURO  
A = ALARMA  
BE = BOTONES DE EMERGENCIA  
SE = SENSOR DE GOLPE

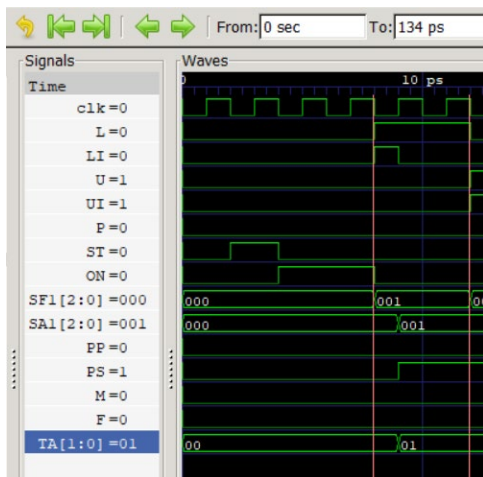
t	CLOCK	BE	SE	SA	SF	A	SR	PO	ME	P
61	0	000	0	000	000	00	00	00	0	0
62	1	000	0	000	000	00	00	00	0	0
63	0	001	0	000	001	00	00	00	0	0
64	1	001	0	001	001	10	01	01	1	1
65	0	010	0	001	010	10	01	01	1	1
66	1	010	0	010	010	01	10	00	1	0
67	0	011	0	010	011	01	10	00	1	0
68	1	011	0	011	011	11	11	00	0	0
69	0	100	0	011	100	11	11	00	0	0
70	1	100	0	100	100	00	00	10	0	0
71	0	101	0	100	101	00	00	10	0	0
72	1	101	0	101	101	00	00	10	1	1
73	0	110	0	101	101	00	00	10	1	1
74	1	110	0	101	101	00	00	10	1	1
75	0	111	0	101	101	00	00	10	1	1
76	1	111	0	101	101	00	00	10	1	1
77	0	000	0	101	000	00	00	10	1	1
78	1	000	0	000	000	00	00	00	0	0
79	0	000	1	000	110	00	00	00	0	0
80	1	000	1	110	110	01	01	01	0	1
81	0	000	0	110	000	01	01	01	0	1
82	1	000	0	000	000	00	00	00	0	0
83	0	000	0	000	000	00	00	00	0	0
84	1	000	0	000	000	00	00	00	0	0
85	0	000	1	000	110	00	00	00	0	0
86	1	000	1	110	110	01	01	01	0	1
87	0	001	1	110	001	01	01	01	0	1
88	1	001	1	001	001	10	01	01	1	1
89	0	000	0	001	000	10	01	01	1	1
90	1	000	0	000	000	00	00	00	0	0
91	0	000	1	000	110	00	00	00	0	0
92	1	000	1	110	110	01	01	01	0	1
93	0	101	1	110	101	01	01	01	0	1
94	1	101	1	101	101	00	00	10	1	1
95	0	000	0	101	000	00	00	10	1	1
96	1	000	0	000	000	00	00	00	0	0
97	0	000	1	000	110	00	00	00	0	0
98	1	000	1	110	110	01	01	01	0	1
99	0	010	1	110	110	01	01	01	0	1
100	1	010	1	110	110	01	01	01	0	1
101	0	011	1	110	110	01	01	01	0	1

101	0	011	1	110	110	01	01	01	0	1
102	1	011	1	110	110	01	01	01	0	1
103	0	100	1	110	110	01	01	01	0	1
104	1	100	1	110	110	01	01	01	0	1
105	0	101	1	110	101	01	01	01	0	1
106	1	101	1	101	101	00	00	10	1	1
107	0	001	1	101	001	00	00	10	1	1
108	1	001	1	001	001	10	01	01	1	1
109	0	011	1	001	011	10	01	01	1	1
110	1	011	1	011	011	11	11	00	0	0
111	0	101	1	011	101	11	11	00	0	0
112	1	101	1	101	101	00	00	10	1	1
113	0	001	1	101	001	00	00	10	1	1
114	1	001	1	001	001	10	01	01	1	1
115	0	100	1	001	100	10	01	01	1	1
116	1	100	1	100	100	00	00	10	0	0
117	0	001	1	100	001	00	00	10	0	0
118	1	001	1	001	001	10	01	01	1	1
119	0	101	1	001	101	10	01	01	1	1
120	1	101	1	101	101	00	00	10	1	1
121	0	010	1	101	010	00	00	10	1	1
122	1	010	1	010	010	01	10	00	1	0
123	0	100	1	010	100	01	10	00	1	0
124	1	100	1	100	100	00	00	10	0	0
125	0	010	1	100	010	00	00	10	0	0
126	1	010	1	010	010	01	10	00	1	0
127	0	101	1	010	101	01	10	00	1	0
128	1	101	1	101	101	00	00	10	1	1
129	0	011	1	101	011	00	00	10	1	1
130	1	011	1	011	011	11	11	00	0	0
131	0	101	1	011	101	11	11	00	0	0
132	1	101	1	101	101	00	00	10	1	1
133	0	100	1	101	100	00	00	10	1	1
134	1	100	1	100	100	00	00	10	0	0
135	0	100	1	100	100	00	00	10	0	0

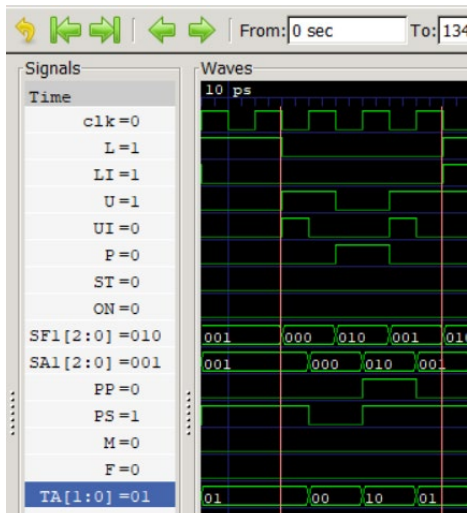
## FSM1 PRUEBA DE ENCENDIDO DE MOTOR Y FUNCIONES EN S0



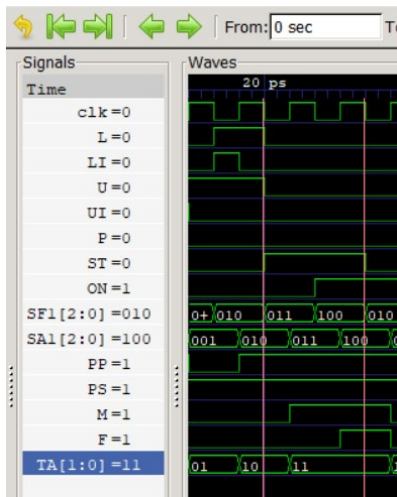
## FSM1 PRUEBA DEL FUNCIONAMIENTO DEL ANTIREBOTE EN L



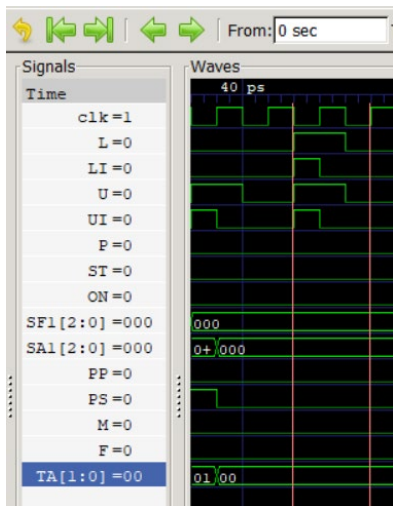
## FSM1 PRUEBA DE ANTIREBOTE EN U



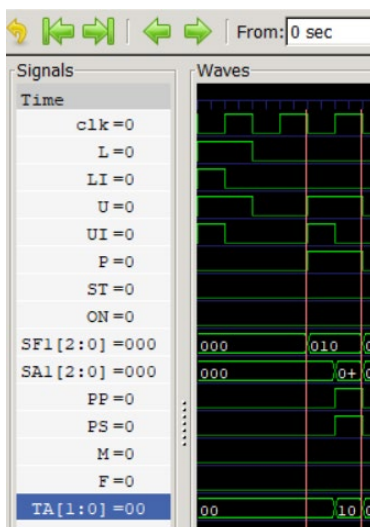
## FSM1 PRUEBA DE ENCENDIDO DE MOTOR Y FUNCIONES



## FSM1 PRUEBA DE PRESIONAR L Y U AL MISMO TIEMPO

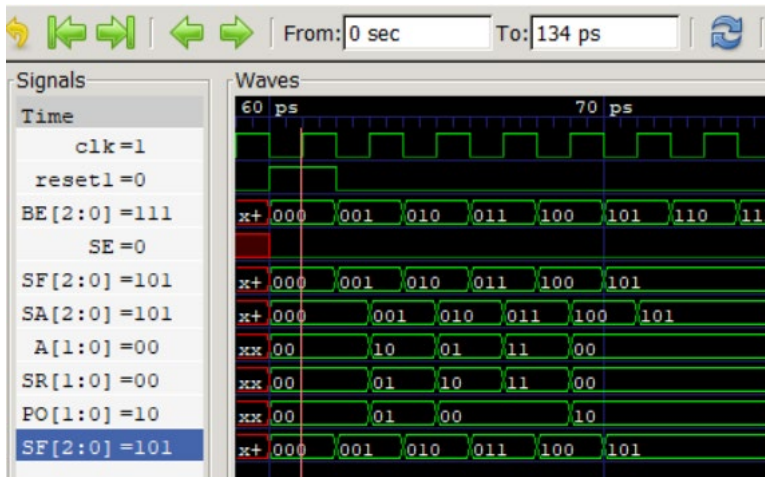


## FSM1 PRUEBA DE PRIORIDAD DE P SOBRE U

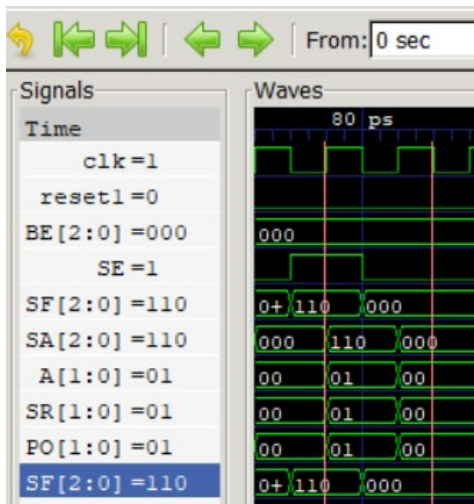




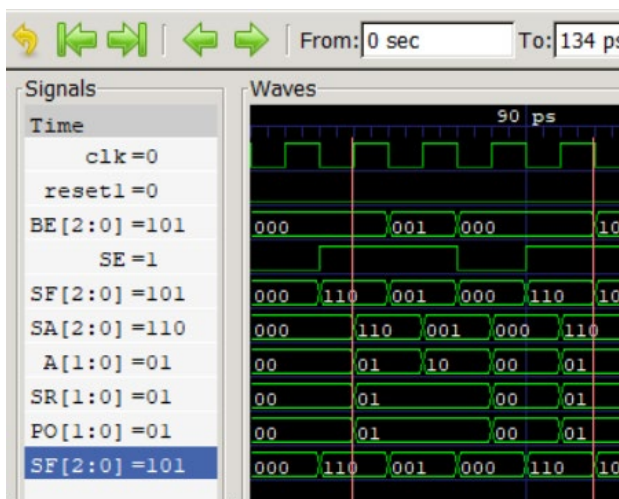
## FSM2 PRUEBAS DE CAMBIO DE EMERGENCIA



## FSM2 PRUEBA DE SALIDA DE S6 CON SENSOR DE GOLPE APAGADA

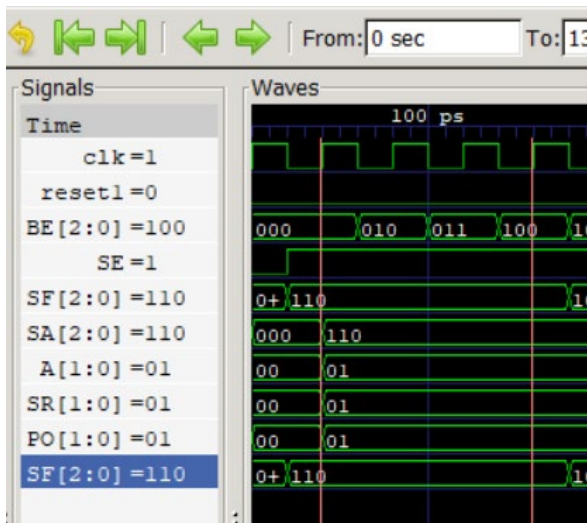


## FSM2 PRUEBA DE SALIDA DE S6 CON SENSOR DE GOLPE ENCENDIDO

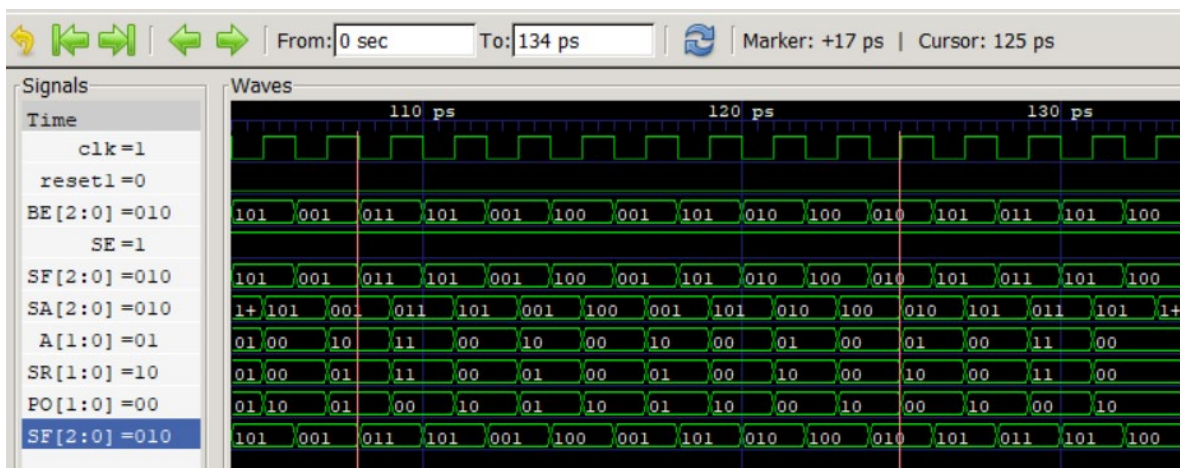




## FSM 2 PRUEBA DE SALIDA DE S6 CON SENSOR ENCENDIDO



FSM2 PRUEBAS DE CAMBIO DE EMERGENCIA DE CUALQUIER EMERGENCIA EN LA QUE SE ENCUENTRE



### LINK DEL REPOSITORIO

- <https://github.com/GAFong/LABORATORIOS/tree/master/PROYECTO%20FSM>

### LINK DEL VIDEO

<https://www.youtube.com/watch?v=lSu-7uoF9Cg>