

Oscar Javier Castelblanco Riveros
Damián Camilo Martínez Martínez
Meyer Humberto Acero Brochero

Diseño No. 1

DESCRIPCIÓN DEL SISTEMA

La Federación Colombiana de Fútbol (FCF) requiere identificar si un árbitro está lo suficientemente calificado o no, para dirigir los partidos del próximo campeonato. Para esto se realizará una prueba que consiste en efectuar 10 disparos al arco, en cada lanzamiento el árbitro indicara si fue gol o no.

Se recibirá una señal por parte del sistema de detección de gol la cual será comparada con la información ingresada por el árbitro, al finalizar la prueba se compara el número de aciertos contra el nivel que registró el árbitro y se informará si este subió, igualó o bajó su nivel.

El diseño contará con las siguientes señales.

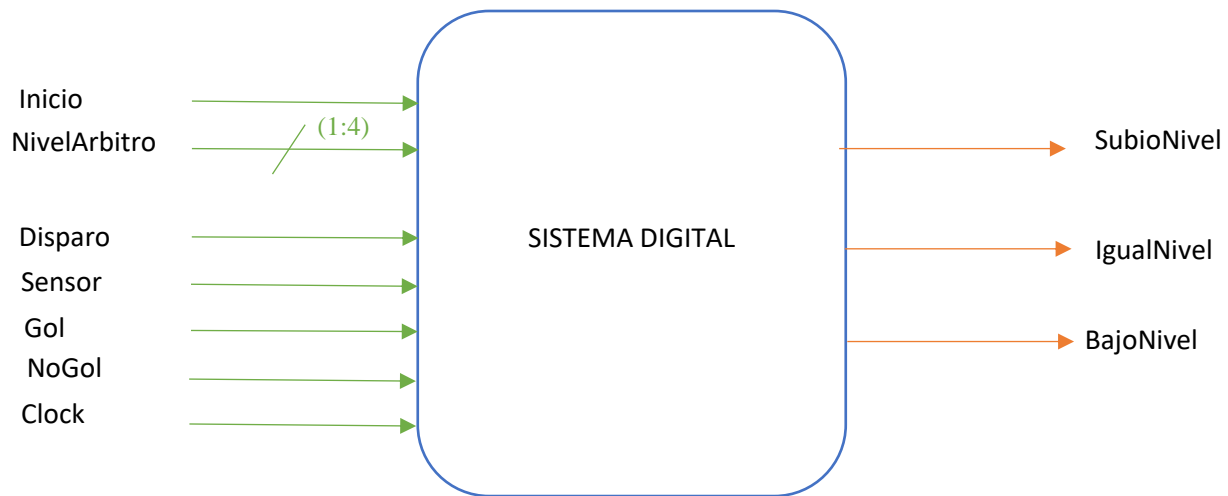
Señales de Entrada

- Inicio: Da comienzo al sistema de evaluación del arbitro
- NivelArbitro: Indica el nivel actual del árbitro.
- Disparo: Indica que se va a realizar un disparo.
- Sensor: Identifica si el disparo fue gol.
- Gol: Señal que el árbitro utiliza para indicar que fue gol.
- NoGol: Señal que el árbitro utiliza para indicar que no fue gol.

Señales de Salida

- SubioNivel: Indica que el árbitro subió su nivel.
- IgualNivel: Indica que el árbitro mantiene su nivel.
- BajoNivel: Indica que el árbitro bajó su nivel.

DIAGRAMA DE ENTRADAS Y SALIDAS



Entrada:

Inicio: Señal asincrónica de un bit que da inicio al sistema.

NivelArbitro: Señal asincrónica de 4 bits que recibe de forma paralela el nivel del árbitro.

Disparo: Señal asincrónica de un bit que habilita la lectura de la señal del sensor y de la respuesta del árbitro.

Sensor: Señal asincrónica de un bit que informa si fue gol o no.

Gol: Señal asincrónica de un bit dada por el árbitro para indicar al sistema que fue gol.

No gol: Señal asincrónica de un bit dada por el árbitro para indicar al sistema que no fue gol.

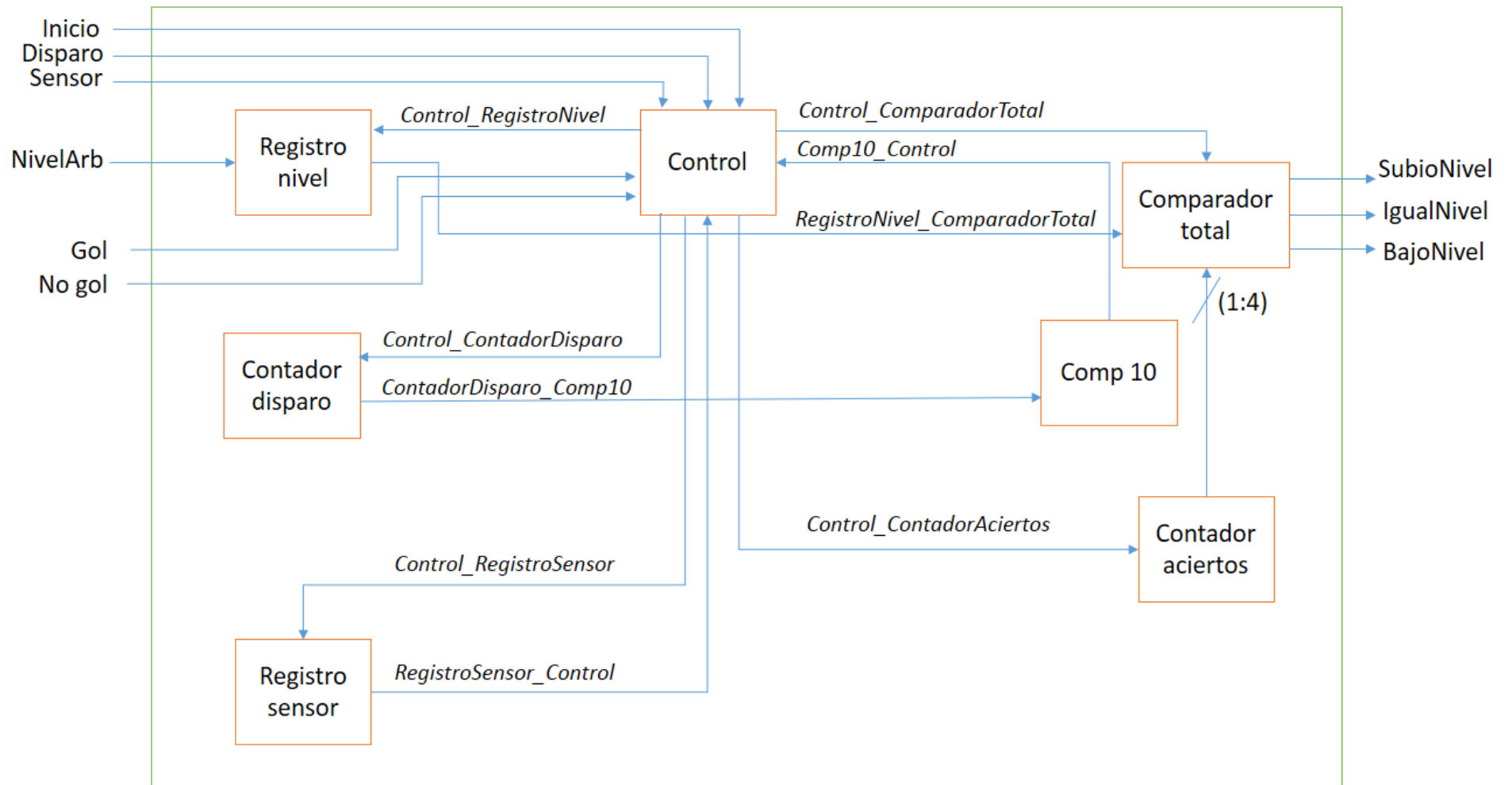
Salida:

SubioNivel: Señal sincrónica de un bit que representa que el árbitro subió su nivel, activa en alto.

IgualNivel: Señal sincrónica de un bit que representa que el árbitro igualó su nivel, activa en alto.

BajoNivel: Señal sincrónica de un bit que representa que el árbitro bajó su nivel, activa en alto.

Diagrama de Bloque



Descripción Diagrama de Bloques

Registro Nivel

Recibe la de forma serial nivel del arbitro por medio de la señal NivelArb, que lo almacenará cuando control por medio de una señal, le informe que lo guarde y luego cuando sea requerido se envíe al bloque ComparadorTotal.

Contador Disparo

Cuenta la cantidad de disparos cuando el bloque Control le informa que hubo uno. Manda una señal al bloque Cont10 (ContadorDisparo_Cont10) y otra al bloque ComparadorTotal (ContadorDisparo_ComparadorTotal).

Registro Sensor

Almacena la respuesta del sensor mandada por el bloque de control (Control_RegistroSensor) y lo envía por medio de una señal al bloque ComparadorAciertos (RegistroSensor_ComparadorAciertos).

Comp 10

Recibe la señal ContadorDisparo_Cont10 desde el bloque ContadorDisparo para comparar la cantidad de disparos, cuando llegue a 10 el bloque manda a una señal al bloque de Control (Cont10_Control) para informar que se terminaron los tiros.

Contador Aciertos

Recibe la cantidad de aciertos desde el bloque ComparadosAciertos y cuando control le informe que se terminaron los tiros, manda una señal con la cantidad de aciertos (ContadorAcierto_ComparadorTotal) al bloque ComparadorTotal.

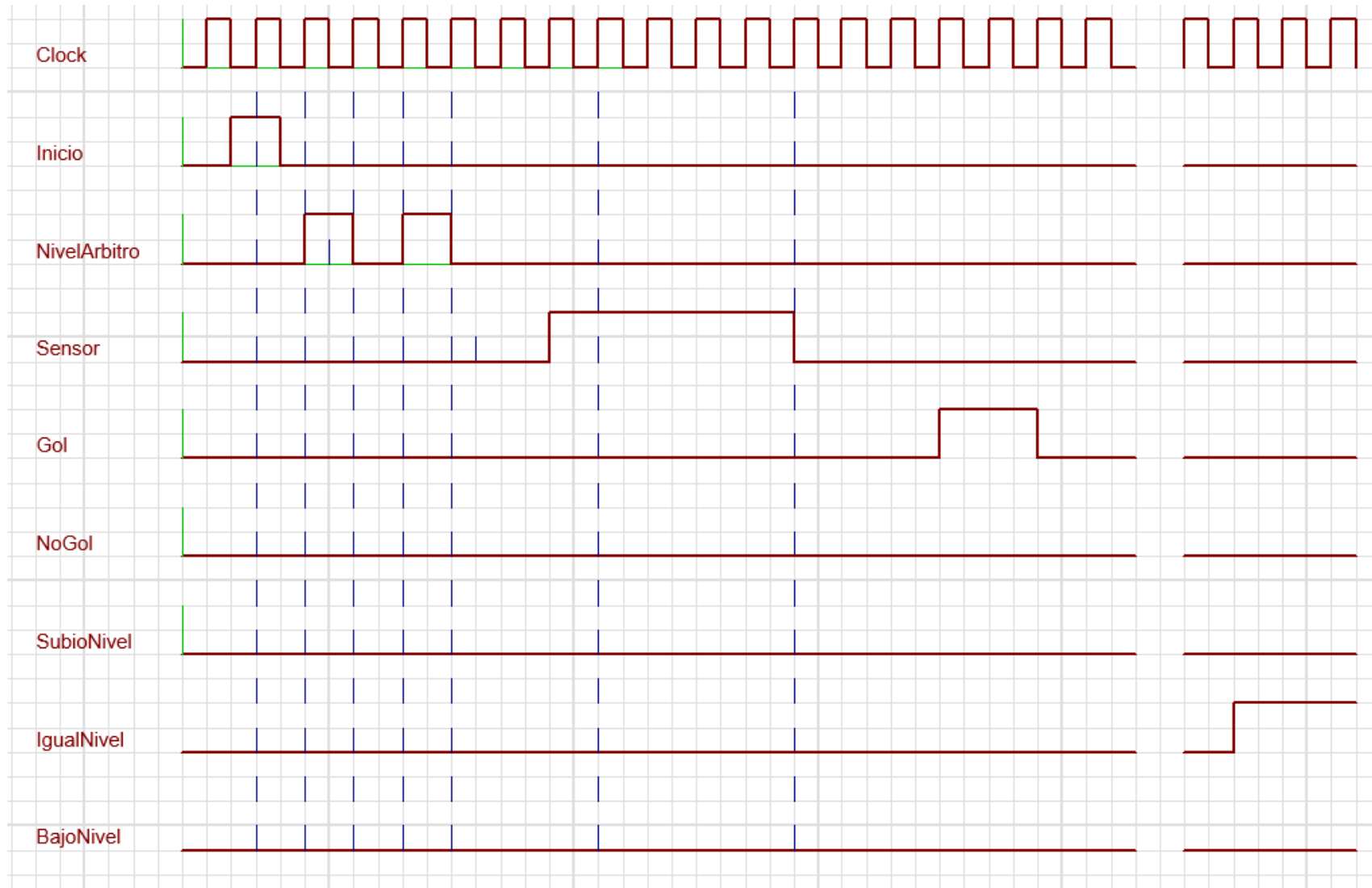
Comparador Total

Bloque combinatorio que compara el nivel del arbitro con la cantidad de aciertos por medio de las señales RegistroNivel_ComparadorTotal y ContadorAciertos_ComparadorTotal cuando recibe la señal Control_ComparadorTotal para saber que se terminaron los y dar la respuesta final por medio de alguna de las señales de salida (SubioNivel, IgualNivel y BajoNivel).

Control

Bloque de control que recibe la señal Inicio para iniciar el sistema y reiniciar los contadores mandando las señales Control_Cont10 y Control_ContadorAciertos. Recibe la señal disparo e informa de ella al bloque ContadorDisparo por medio de la señal Control_ContadorDisparo. Recibe la señal Sensor y manda al bloque RegistroSensor la respuesta del sensor (señal Control_RegistroSensor). Recibe la respuesta del arbitro, ya sea por medio de la señal Gol o de la señal NoGol e informa a sus respectivos bloques de la misma. Recibe la señal Cont10_Control para saber que se terminaron los tiros y de permiso al bloque ComparadorTotal por medio de la señal Control_ComparadorTotal de mostrar la salida determinada.

DIAGRAMA DE TIEMPO



AHPL

1.
 $\rightarrow (\text{Inicio} \times 2) + (\overline{\text{Inicio}} \times 1)$
2.
 $\text{RegistroNivel}(1) \leftarrow \text{NivelArbitro}$
 $\text{RegistroNivel}(2) \leftarrow \text{RegistroNivel}(1)$
 $\text{RegistroNivel}(3) \leftarrow \text{RegistroNivel}(2)$
 $\text{RegistroNivel}(4) \leftarrow \text{RegistroNivel}(3)$
3.
 $\text{RegistroNivel}(1) \leftarrow \text{NivelArbitro}$
 $\text{RegistroNivel}(2 \dots 4) \leftarrow \text{RegistroNivel}(1 \dots 3)$
4.
 $\text{RegistroNivel}(1) \leftarrow \text{NivelArbitro}$
 $\text{RegistroNivel}(2 \dots 4) \leftarrow \text{RegistroNivel}(1 \dots 3)$
5.
 $\text{RegistroNivel}(1) \leftarrow \text{NivelArbitro}$
 $\text{RegistroNivel}(2 \dots 4) \leftarrow \text{RegistroNivel}(1 \dots 3)$
6.
 $\rightarrow (\text{Disparo} \times 7) + (\overline{\text{Disparo}} \times 6)$
7.
 $\text{ContadorDisparo} \leftarrow \text{Inc}(\text{ContadorDisparo})$
8.
 $\rightarrow (\text{Sensor} \times 9) + (\overline{\text{Sensor}} \times 10)$
9.
 $\rightarrow (\text{Sensor} \times 9) + (\overline{\text{Sensor}} \times 11)$
10.
 $(\text{RegistroSensor} \leftarrow 0)$
 $x12$
11.
 $(\text{RegistroSensor} \leftarrow 1)$
 $x12$
12.
 $(\text{Gol} \times 13) + (\overline{\text{Gol}} \times 12) + (\text{NoGol} \times 14) + (\overline{\text{NoGol}} \times 12) + \text{And}(\text{Gol}, \text{NoGol})x12$
13.
 $\rightarrow (\text{RegistroSensor}x15) + (\overline{\text{RegistroSensor}}x16)$

14.

$\rightarrow (\text{RegistroSensor} \times 16) + (\overline{\text{RegistroSensor}} \times 15)$

15.

$\text{ContadorAcierto} \leftarrow \text{Inc}(\text{ContadorAcierto})$

16.

$(\text{Comparador10}(\text{ContadorDisparo}, 10) \times 17) + (\overline{\text{Comparador10}(\text{ContadorDisparo}, 10)} \times 6)$

17.

$(\text{ComparadorMayor}(\text{ContadorAcierto}, \text{RegistroNivel}) \times 17$
 $+ (\text{ComparadorMenor}(\text{ContadorAcierto}, \text{RegistroNivel}) \times 18$
 $+ (\text{ComparadoIgual}(\text{ContadorAcierto}, \text{RegistroNivel}) \times 19)$

18.

$(\text{SubioNivel} \leftarrow 1) \times 1$

19.

$(\text{IgualNivel} \leftarrow 1) \times 1$

20.

$(\text{BajoNivel} \leftarrow 1) \times 1$

//Vuelve al estado 1 para esperar otro arbitro

ECUACIONES

$\text{Mayor} = \text{ContAciertos}(3) \wedge \overline{\text{RegistroNivel}(3)} +$

$(\text{ContAciertos}(2) \wedge \overline{\text{RegistroNivel}(2)}) \wedge (\text{ContAciertos}(3) \odot \text{RegistroNivel}(3)) +$

$(\text{ContAciertos}(1) \wedge \overline{\text{RegistroNivel}(1)}) \wedge ((\text{ContAciertos}(3) \odot \text{RegistroNivel}(3)) \wedge ((\text{ContAciertos}(2) \odot$
 $\text{RegistroNivel}(2)) +$

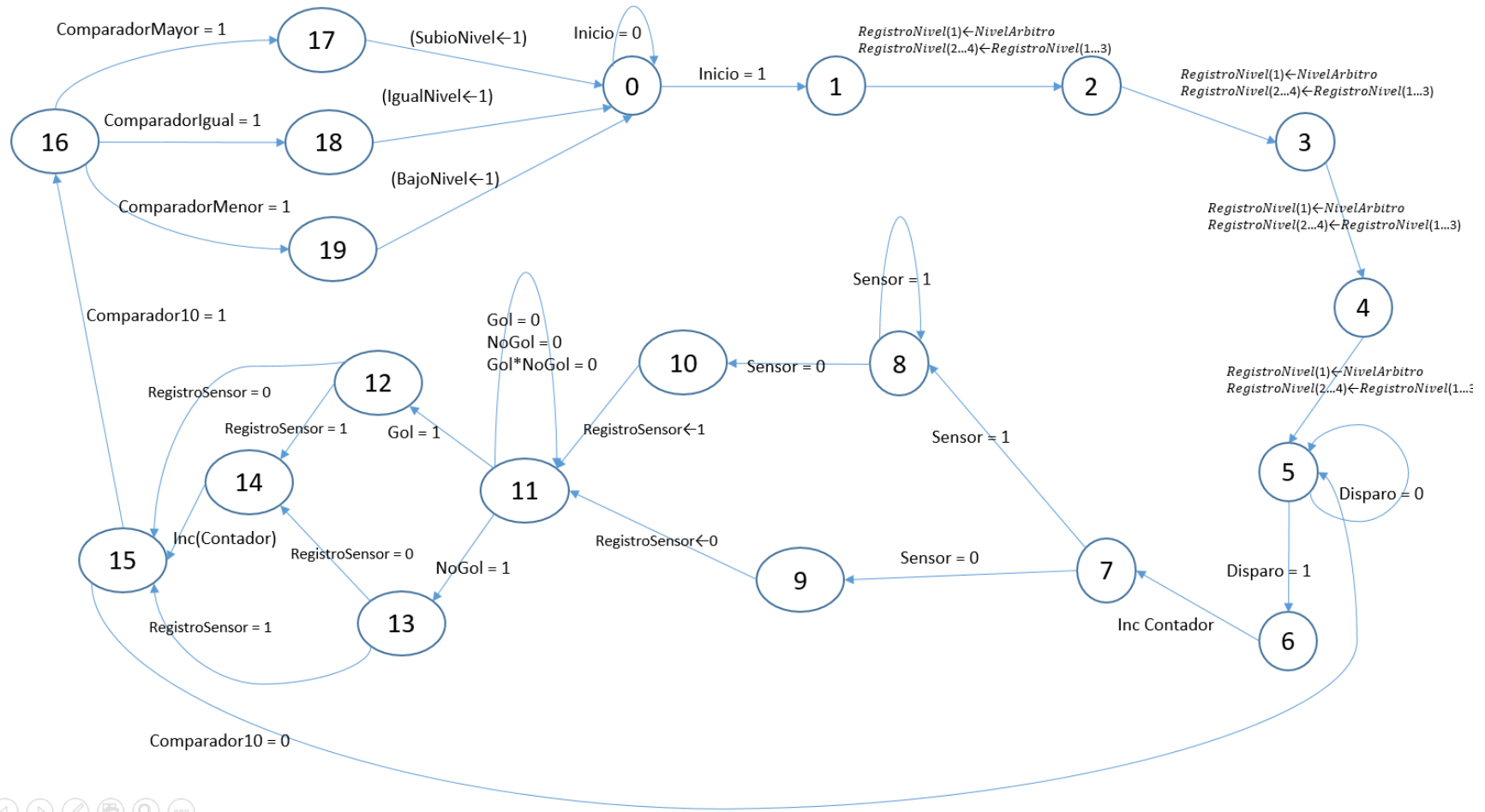
$(\text{ContAciertos}(0) \wedge \overline{\text{RegistroNivel}(0)}) \wedge ((\text{ContAciertos}(3) \odot \text{RegistroNivel}(3)) \wedge ((\text{ContAciertos}(2) \odot$
 $\text{RegistroNivel}(2)) \wedge ((\text{ContAciertos}(1) \odot \text{RegistroNivel}(1)))$

$\text{Igual} = ((\text{ContAciertos}(3) \odot \text{RegistroNivel}(3)) \wedge ((\text{ContAciertos}(2) \odot \text{RegistroNivel}(2)) \wedge$

$$((\text{ContAciertos}(1) \odot \text{RegistroNivel}(1)) \wedge ((\text{ContAciertos}(0) \odot \text{RegistroNivel}(0)))$$

Menor = Mayor ^ Igual

DIAGRAMA DE ESTADOS



Protocolo de Pruebas:

Para la realización de las pruebas del sistema se tendrán en cuenta los estados mas importantes para la comprobación correcta del nivel del arbitro.

Se utilizará un nivel de árbitro menor o igual a dos, esto con la finalidad de comprobar al finalizar el sistema que la opción de subió nivel es correcta. Adicionalmente, para comprobar si los aciertos del árbitro están siendo calculados de manera adecuada se realizarán varios tiros siguiendo la siguiente secuencia

Resultado del Sensor	Opción del Árbitro
Gol	Gol
Gol	NoGol
NoGol	Gol
NoGol	NoGol
Gol	Gol
Gol	Gol
Gol	Gol
Gol	Gol
Gol	Gol
Gol	Gol

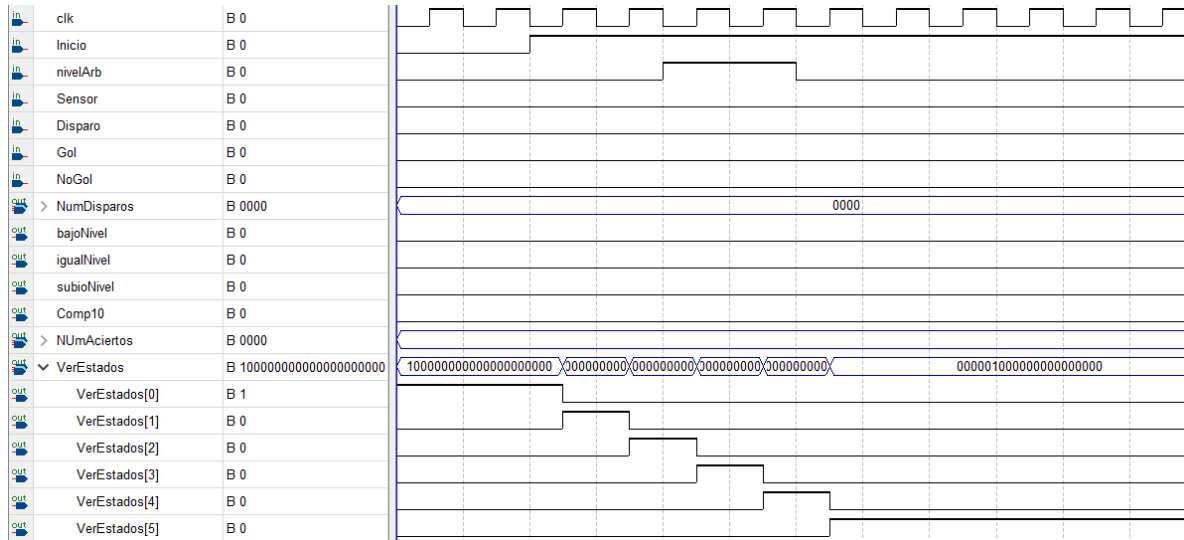
Asi que el resultado obtenido finalmente será de 8 aciertos y 2 equivocaciones del árbitro con lo cuál se deberá obtener un resultado de superó Nivel.

Protocolo de Pruebas

Protocolo de pruebas para simulación:

Para conocer si nuestro sistema funciona correctamente debemos revisar varios casos importantes en la simulación los cuales son:

- Nuestro sistema inicie en el estado 0 , pase al estado 1 cuando la señal de Inicio este en alto y luego por cada flanco de subida del reloj pase al estado 2,3 ,4 y 5.
En los estados 1, 2 ,3 y 4 guarda la información de NivelArbitro y se queda en el estado 5 esperando que la señal de Disparo cambie a alto.



- Cuando la señal Disparo se pone en alto y el sistema pasa al estado 6 donde se aumenta el contador de NumDisparos y evaluamos 4 casos posibles

Caso 1: Señal sensor permanece en LOW y la señal de Gol cambia a High

En este caso no se toma como acierto debido a que el sistema toma como No gol el disparo y el arbitro dice que fue Gol.

Caso 2: Señal *Sensor* permanece en LOW y la señal de NoGol cambia a High

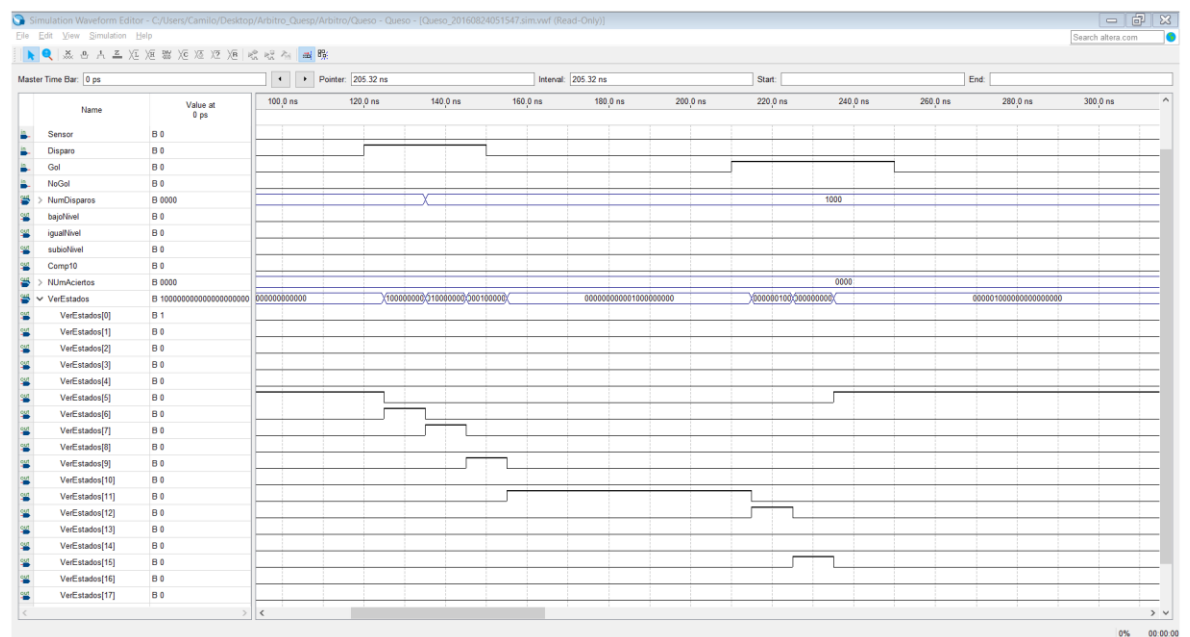
Se cuenta como acierto pues tanto el sistema como el arbitro coinciden que el disparo fue No gol

Caso 3: Señal *Sensor* cambia a High y luego cambia a LOW y la señal de Gol cambia a High

Se toma como acierto pues el sistema toma como Gol cuando la señal Sensor cambia de High a Low en los estados 7 y 8

Caso 4: Señal *Sensor* cambia a High y luego cambia a LOW y la señal de NoGol cambia a High

En este caso el arbitro no acierta pues no coinciden las respuesta del arbitro con la del sistema.



Analisis de las simulaciones

En la simulación se puede observar la realización de varios tiros para probar el sistema. Se observan los cuatro casos previamente descritos en las primeras cuatro filas de la tabla de protocolo de pruebas. Se puede visualizar que la salida de la cantidad de aciertos es igual de a dos. Lo que concuerda con el resultado esperado debido a que hay dos combinaciones donde el arbitro tiene una respuesta correcta.

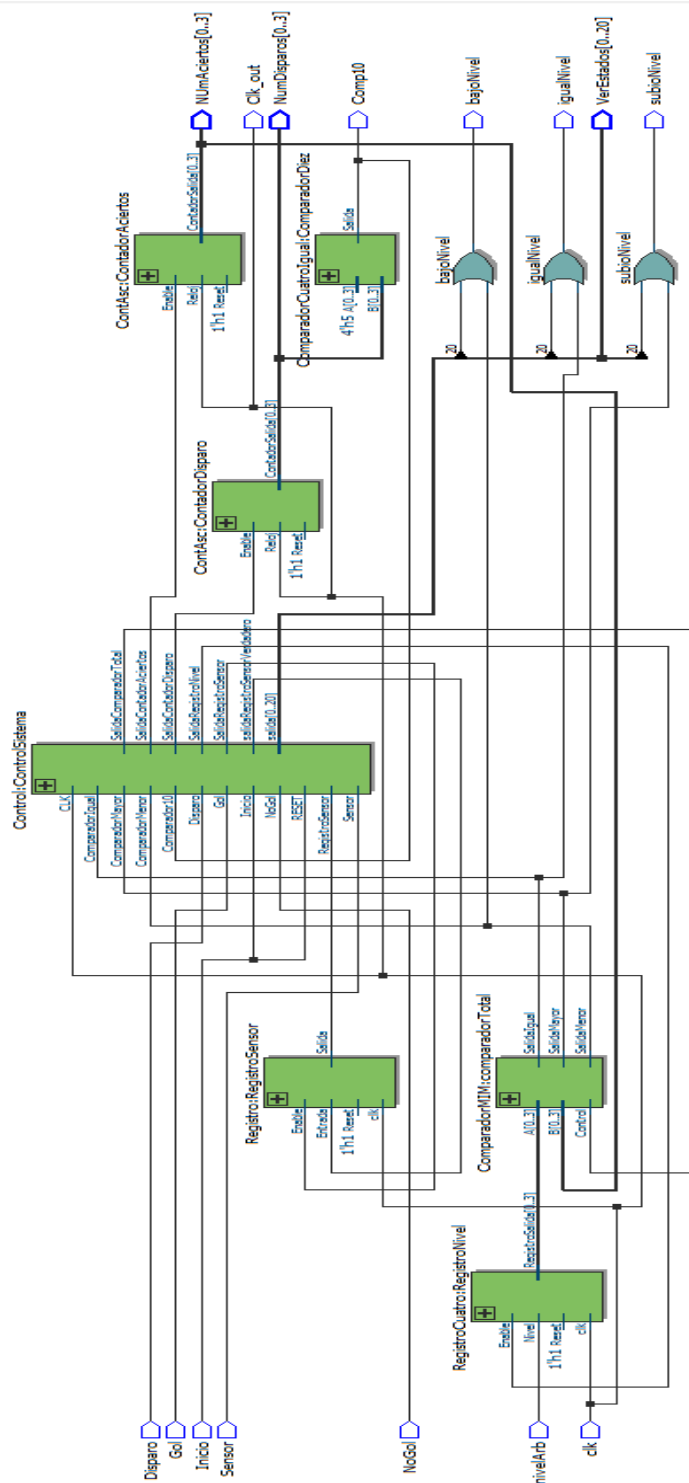
Por otro lado, podemos ver que el movimiento en el diagrama de estados es adecuado, es decir que las transiciones se realizan de manera adecuada. Por ejemplo podemos ver que el sistema retorna al estado 5 siempre después de comprobar si la respuesta del arbitro fue un acierto o no.

Uno de los errores que se arreglaron durante la realización de las mismas fue la detección de una mala conexión entre bloques lo cuál hacia que una de las entradas de uno de los bloques estuviera dirigida hacia tierra y por ende la señal de salida no era la esperada. Es por esta razón que decidieron utilizarse nombres más representativos para la señales que conectan bloques. El formato que se escogió fue el siguiente

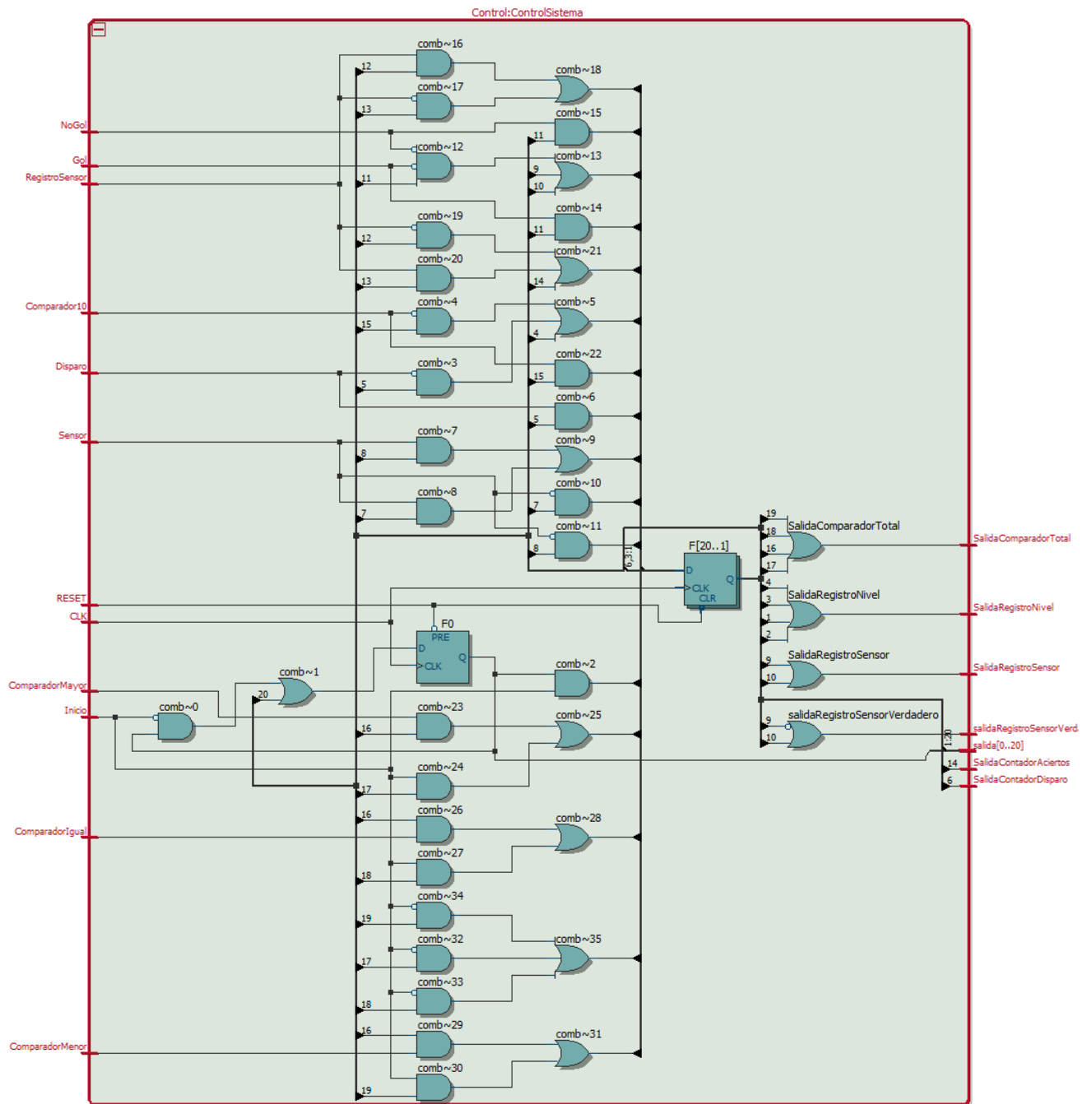
BloqueDeEntrada_BloqueDeSalida

La segunda observación que se realizó cuando se hizo la simulación fue la necesidad de tener pre establecidas las señales de sensor antes de realizar el disparo cuando se realice la ejecución práctica. Esto debido a que la velocidad del reloj es muy alta y la del ser humano no es comparable con respecto a esta. Además como la lectura del sensor se realizó un ciclo de reloj después de leer disparo, es imperativo tener pre configurado el valor de sensor.

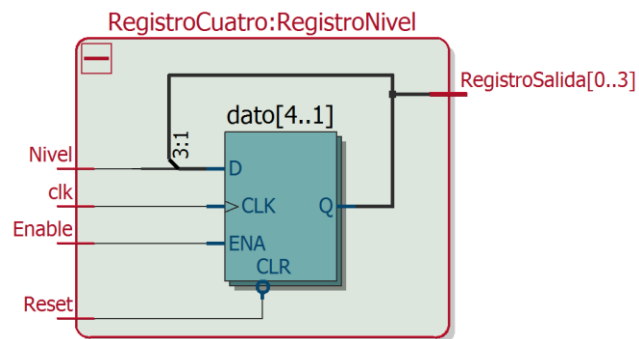
Bloque General



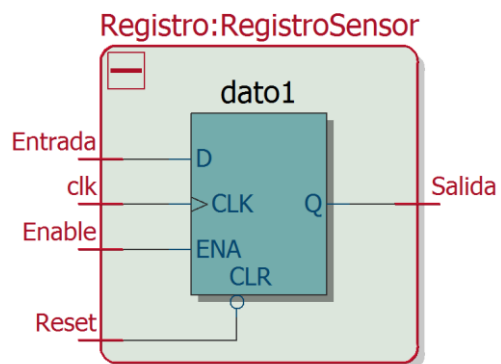
Bloque control



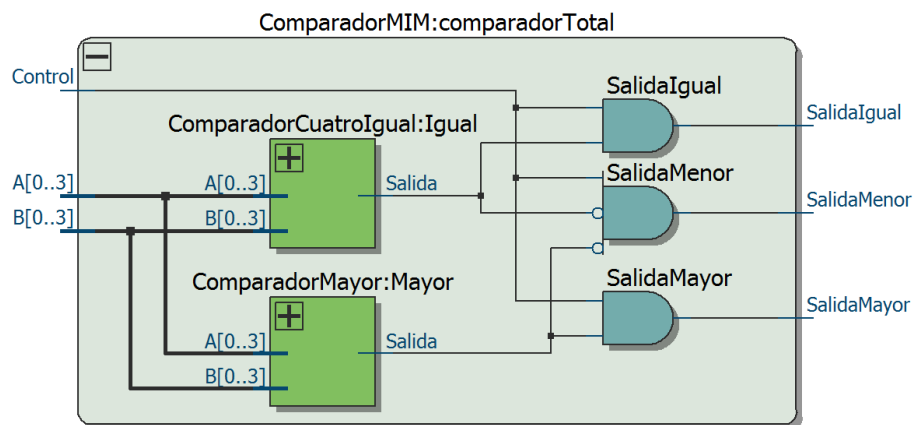
Bloque registro nivel



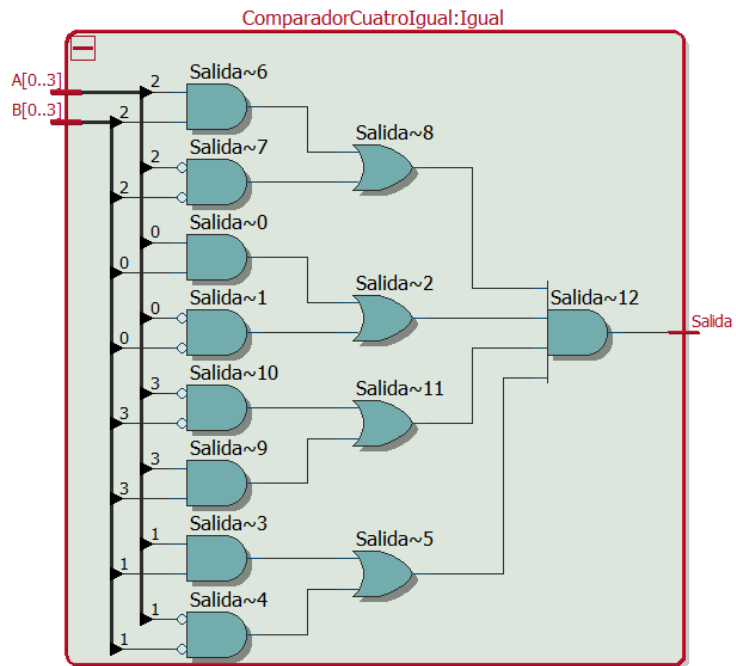
Bloque registro sensor



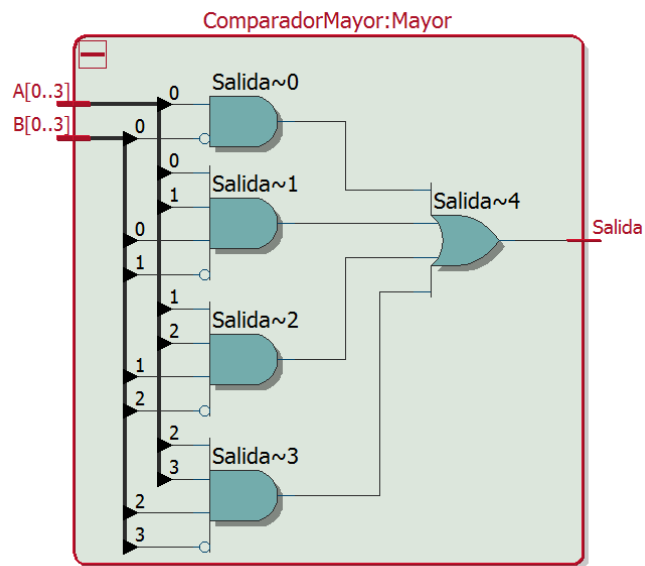
Bloque Comparador Total



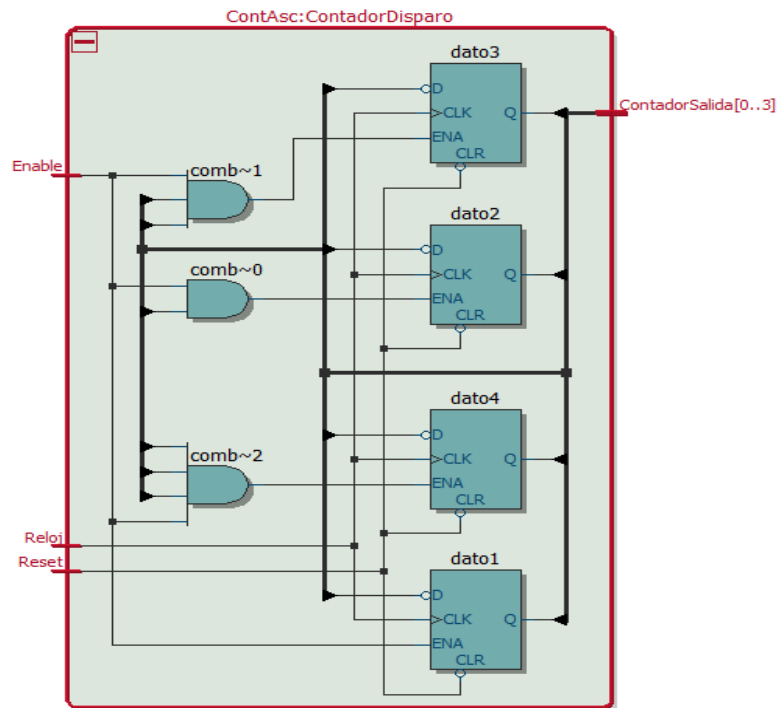
Bloque interno comparador igual



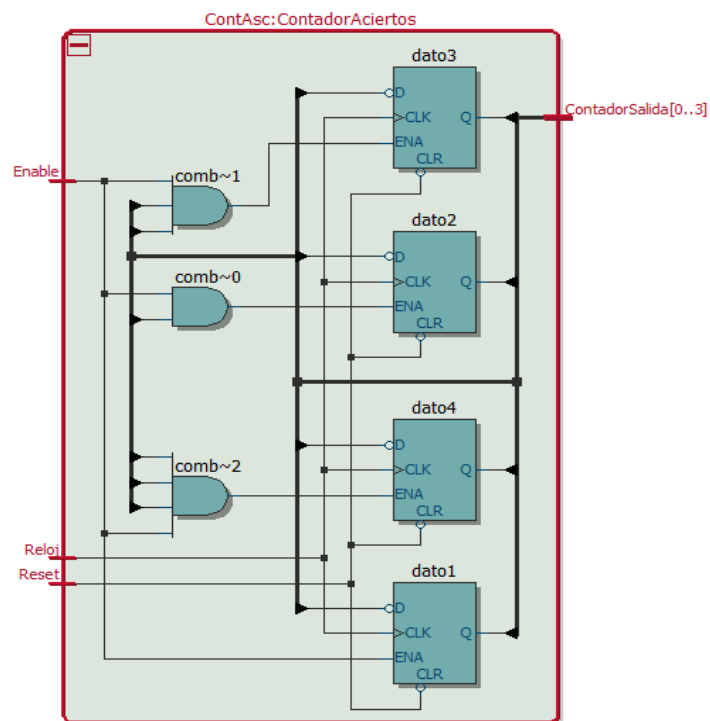
Bloque interno comparador mayor



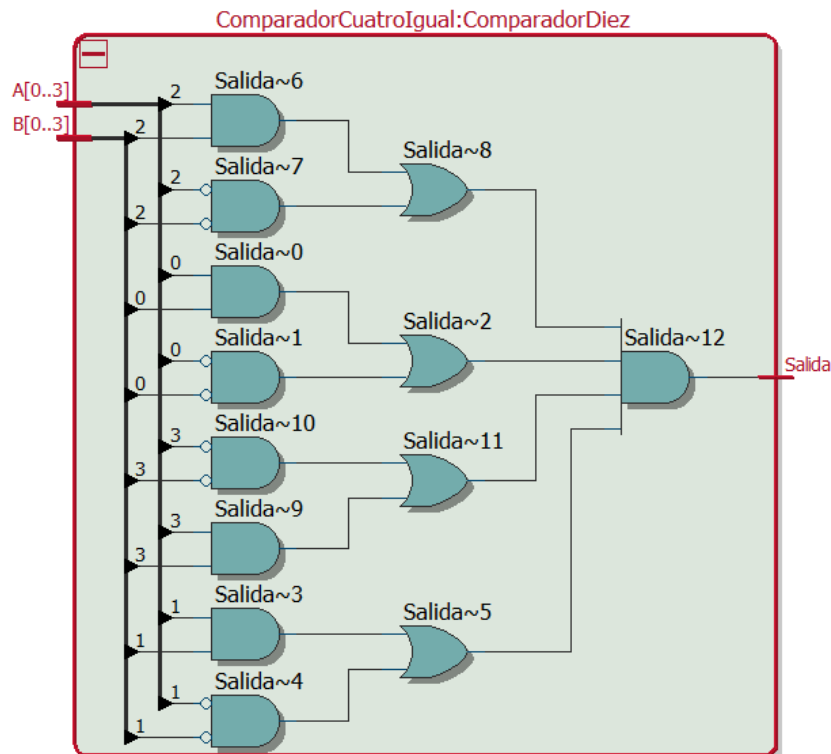
Bloque contador disparo



Bloque contador aciertos



Bloque comparador10



Análisis de la entrega en hardware – incluir video de la entrega

Las entradas fueron implementadas por medio de los switches de la tarjeta y la salida por medio de los leds de la misma. Para definir el nivel actual del árbitro se deben tener en cuenta las distintas respuestas del árbitro y del sensor (entradas) para dar un juicio justo y compararlo con el nivel inicial y ver su actual desempeño. Se pueden presentar cuatro casos de entradas, cada una se muestra en los videos; la primera es cuando es gol y el árbitro está en lo correcto y dice gol también, entonces se debe activar primero la entrada sensor y luego disparo, así el sistema sabe que vendrá un disparo con respuesta gol, toma el valor de 1 lógico la entrada del árbitro diciendo gol y obteniendo así un acierto; otra manera de acertar es cuando no es gol y el árbitro indica que no lo fue, se activa la entrada disparo, vuelve a bajar y se espera la respuesta del árbitro. En los casos en los que el árbitro no acierta, es decir la lectura de la entrada del sensor es la inversa a la respuesta del árbitro, cuando se sube y baja disparo y el árbitro dice gol o cuando se sube sensor, se sube disparo y luego bajan y el árbitro dice que no fue gol. Al finalizar la cantidad de tiros y se hagan las comparaciones para determinar el nivel actual este se muestra apagando un led de la tarjeta. Todos los leds están encendidos y se apagará el del nivel actual.

Videos anexos donde se muestra el funcionamiento del sistema.

Parte 1 muestra los cuatro casos de entradas

<https://www.youtube.com/watch?v=7H2vTGU1I2o>

Parte 2 muestra la salida al finalizar los 10 tiros

<https://www.youtube.com/watch?v=Y4f4Au3iYuQ>

Autoreflexión

Oscar: Considero que se realizó un buen trabajo en grupo desde el planteamiento del diseño hasta su ejecución. Sin embargo, considero que se debe reservar un espacio a la semana entre los miembros del grupo para la realización del proyecto. Esto debido a que nos dejamos coger un poco el tiempo con la realización de la implementación práctica del sistema.

Meyer: Como todo diseño se presentaron problemas a la hora de la implementación pero se solucionaron la mayoría, no todos debido a algunas limitaciones físicas de la tarjeta y otras por falta de tiempo. El diseño se realizó paso a paso como es debido pero por administrar mal el tiempo no se alcanzaron a solucionar todos a pesar de las limitaciones que tiene el diseño para simplificarlo y hacerlo cada vez más realizable. Se trabajó arduamente pero no se le dedicó el tiempo esperado y la comunicación entre los integrantes del grupo no fue la mejor, por lo que se pudo haber implementado de mejor manera y con menos errores.

Damian: Poder trabajar en grupo es de vital importancia en los ingenieros ,con este diseño nos hemos podido dar cuenta de las dificultades que a veces se presentan y como una buena comunicación entre los miembros del grupo sirve para solucionar los problemas que se presenten.