

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

"LABORATORIO PARA EL ESTUDIO DE LA FISIOLOGÍA DE LA PERCEPCIÓN"

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

I N G E N I E R A

ELÉCTRICA ELECTRÓNICA

P R E S E N T A:

JANNET GUERRERO CHIMAL



DIRECTOR:

ING. ADRIÁN HERNÁNDEZ ALVA

OCTUBRE 2007

Dedicatorias …

A mis padres.

Por darme la vida,
llenarla de sueños
y enseñarme
a luchar por ellos.

Los amo.

Agradecimientos …

Gracias a mis padres por su ejemplo de fortaleza y voluntad, por creer en mi y apoyarme incondicionalmente, por que este importante paso en mi carrera lo hemos dado juntos, gracias por su ternura y comprensión pero principalmente por su amor.

Gracias a mis hermanos Martha y Erick por ser mis cómplices, guardar mis secretos y darme consejos, por sus cuidados y por la confianza que depositan en mí.

Gracias a mis pequeños sobrinos Christian, Oswaldo y Dereck por que con sus miradas inocentes y llenas de ilusión, sus sonrisas tiernas y sus dulces abrazos me inspiran para seguir adelante.

Gracias a Román Soriano Ramos por su amistad, su apoyo desmesurado, su infinita paciencia y por todos los consejos que, a lo largo de cuatro años, me ha brindado; pero le agradezco especialmente por todo su cariño.

Gracias a todos mis amigos(as) y compañeros(as) de la Facultad de Ingeniería, de la Preparatoria 2 y de gimnasia que compartieron momentos inolvidables de felicidad (y unos cuantos de angustia), gracias por luchar a mi lado, siempre los llevaré en mi memoria.

Gracias a mis amigos del Instituto de Fisiología que tanto me han ayudado, en especial a Leidy Ana Ortiz y a Manuel Álvarez, que en esos momentos complicados, me animaron con una sonrisa y un abrazo, a continuar.

Gracias a Alejandro Dávila por todos los consejos que sabia y oportunamente me ha dado, así mismo, por el tiempo y la amistad que me ha brindado.

Gracias a la Dra. Claudia Lerma por la oportunidad de trabajar juntas, por todas las cosas nuevas que me enseñó y por su amistad.

Gracias a la Universidad Nacional Autónoma de México y a la Facultad de Ingeniería por la formación profesional, por inculcarme la responsabilidad y la libertad, por forjar parte importante de mi carácter, gracias por permitirme ser parte de ti.

Gracias a todos mis profesores que me transmitieron, además de conocimientos, la pasión por la ingeniería, muy especialmente gracias a mis sinodales: el Ing. Francisco Rodríguez Ramírez, el Dr. Pablo Roberto Pérez Alcazar, el M.I. Juan Manuel Gómez González y el Ing. Miguel Eduardo González Cárdenas por revisar el texto original, por sus correcciones y comentarios.

Gracias mi director de tesis el Ing. Adrián Hernández Alva por la oportunidad de ser parte del laboratorio de neurociencias, por todas las enseñanzas, la paciencia y el tiempo que me dedicó.

Índice de temas.

1. Introducción..... 1

2. Análisis del sistema de control de experimentos de psicofísica..... 3

2.1 Descripción de los experimentos de psicofísica. 3

2.2 Descripción de los requerimientos de entrada y salida de las funciones principales del sistema. 7

2.3 Análisis detallado del ensayo. 11

2.4 Recursos utilizados actualmente. 17

2.5 Propuesta de nuevo hardware y software..... 18

3. Diseño del sistema de control de experimentos de psicofísica. 21

3.1 Módulos para la construcción de ensayos. 21

3.2 Señales de entrada y salida. 22

3.3 Configuraciones y controles manejados por el sistema. 24

3.4 Diagramas de flujo para el experimento y el ensayo. 30

4. Desarrollo de la aplicación en LabView. 33

4.1 Programación y prueba de los módulos..... 34

4.2 Integración de los módulos para formar el sistema. 43

5. Resultados 55

6. Conclusiones..... 57

7. Bibliografía. 59

Índice de tablas y figuras.

Tabla 1 Asociación de los botones con las respuestas según la tarea propuesta. 4

Figura 1 Tarea de discriminación. 5

Figura 2 Estimulador táctil (Vibratactile Stimulator BME Systems Inc.). 5

Figura 3 Diagrama de bloques de los experimentos de psicofísica. 9

Figura 4 Prototipo de la interfaz gráfica de usuario para configuración del sistema. 10

Figura 5 Prototipo de la interfaz gráfica de usuario para el control de la tarea y presentación de resultados. 11

Figura 6 División de un ensayo en estados de la tarea de detección. 13

Tabla 2 Secuencia de estados para la tarea de detección según el evento. 13

Figura 7 División de un ensayo de la tarea de discriminación en estados. 14

Tabla 3 Secuencia de estados para la tarea de discriminación y de patrones temporales. 15

Figura 8 Eventos esperados en un ensayo para la tarea de discriminación. 16

Figura 9 Diagrama de estados para la tarea de discriminación. 17

Figura 10 Diagrama lógico de notificación para el sistema. 22

Figura 11 Eventos esperados y señales que los originan. 23

Figura 12 Señales de entrada del puerto digital de la tarjeta de adquisición de datos. 23

Figura 13 Esquema del destino de las salidas analógicas 23

Figura 14 Componentes del contador (parte superior). Configuración en el modo 'buffered period measurement' (parte inferior). 24

Tabla 4 Estados que conforman los ensayos según el tipo de tarea. 25

Tabla 5 Contenido del archivo 'todas_las_tareas'. 25

Tabla 6 Protocolo para escribir los parámetros de los estímulos. 26

Figura 15 Diagrama de flujo para un experimento 30

Figura 16 Diagrama de flujo para un ensayo. 31

Figura 17 Diagrama de E/S del VI generador de funciones básico. 34

Figura 18 Módulo principal que genera estímulos continuos. 36

Figura 19 Módulo principal que genera rampas. 37

Figura 20 Diagramas de bloques para la generación del pulso (izquierda) y para la generación del tiempo muerto (derecha) de los patrones temporales. 38

Figura 21 Diagrama de bloques del VI para generación de patrones temporales. 38

Figura 22 Panel Frontal para generación de patrones temporales 39

Figura 23 Esquema de los bits del puerto paralelo de la PC usados para el control de las luces y la recompensa. 40

Figura 24 Panel Frontal y diagrama de bloques para controlar el encendido de las luces. 40

Figura 25 Diagrama de bloques del VI que apaga las luces. 41

Figura 26 Panel frontal y diagrama de bloques para la recompensa. 41

Figura 27 Panel frontal y diagrama de bloques para el módulo de duración del estado. 42

Tabla 7 Eventos correctos para cada estado. 42

Figura 28 Diagrama del VI para la evaluación del evento ocurrido. 43

Figura 29 Diagrama de bloques para la programación del contador..... 43

Figura 30 Panel Frontal de la interfaz para configurar el experimento. 45

Figura 31 Diagrama de bloques de la Interfaz para configurar del experimento. 46

Figura 32 Tabla generada por el VI interfaz_1 con los datos de la figura 36. 47

Figura 33 Diagrama de bloques y panel frontal del estado inicial. 49

Figura 34 Diagrama de bloques del estado de error. 50

Figura 35 Panel frontal y diagrama de bloques para el estado de rampa negativa. 51

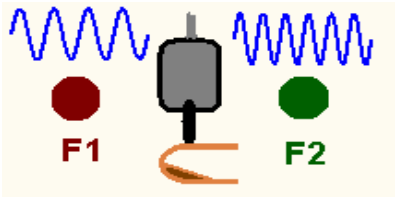
Figura 36 Panel frontal y diagrama de bloques para el estado de barra ocupada..... 51

Figura 37 Panel Frontal y diagrama de bloques para el estado de estimulación. 52

Figura 38 Panel frontal y diagrama de bloques para el estado presión del botón. 53

Figura 39 Panel frontal y diagrama de bloques para el estado final..... 53

Figura 40 Interfaz de usuario para el control del experimento y presentación de resultados. 54



1.Introducción

El Dr. Ranulfo Romo del Instituto de Fisiología Celular (IFC), perteneciente a la Universidad Nacional Autónoma de México, realiza estudios de la fisiología de la percepción. Para ello, experimenta aplicando a monos rhesus estímulos táctiles, cuyos parámetros, como la frecuencia, la duración, la amplitud y la forma de onda, están estrictamente controlados; al mismo tiempo, registra la actividad motora y/o neuronal de los sujetos de experimentación. Todo esto con el objetivo de asociar los atributos de los estímulos y las respuestas motoras del sujeto, con la respuesta neuronal y así descifrar la forma en la cual el cerebro transforma los parámetros de los estímulos sensoriales en experiencia útil.

Este laboratorio cuenta con un sistema que controla el tipo de tarea que realiza el sujeto de experimentación, los parámetros de los estímulos y la lectura, grabación y presentación de los resultados de la actividad motora y neuronal.

Actualmente, el funcionamiento del sistema lo llevan a cabo tres computadoras (una Pentium y dos 486), la primera de ellas tiene instalado el sistema operativo Windows 98 y las otras dos tienen instalado DOS. Los programas específicos para las tareas conductuales están elaborados en el lenguaje de programación C, mientras que las interfaces gráficas para la presentación de la respuesta motora y neuronal están elaboradas en el lenguaje de programación C++. Para el desarrollo de los experimentos este sistema usa tarjetas de adquisición y generación de señales del proveedor National Instrument que usan el bus ISA. La comunicación entre los elementos del sistema se realiza mediante el protocolo GPIB, finalmente, la adquisición de la respuesta neuronal se realiza mediante la tarjeta PCI 6071E, también de National Instrument. Además, para generar los estímulos táctiles se utiliza un generador de señales (HP8904A; Hewlett Packard) que debe ser controlado mediante interrupciones.

Duplicar este sistema es imposible porque la tecnología que se usó ya no existe, el bus ISA ha sido sustituido por PCI y PCI Express. Por otro lado, para realizar modificaciones en los parámetros de los experimentos se requiere gran cantidad de tiempo, así como de recursos materiales y humanos. Esto ha generado la necesidad de diseñar y construir un nuevo sistema que permita

adquirir y procesar la información con menores recursos, menores costos y mayor flexibilidad.

El objetivo del presente proyecto es: diseñar y construir un sistema electrónico-computacional que permita realizar experimentos del área de neurociencias, controlando todos los parámetros de los estímulos y registrando las respuestas motoras de los sujetos experimentales.

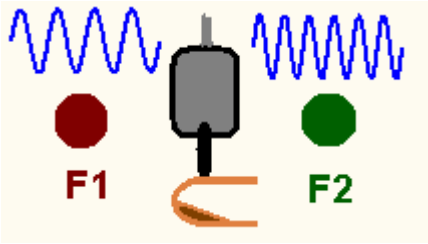
Como parte de las experiencias extracurriculares en las que he participado, he tenido contacto con el concepto de instrumentación virtual, el cual está relacionado con el hecho de que hoy en día, este tipo de instrumentación aprovecha las características de la computadora personal y permiten la programación de funciones que se almacenan, ejecutan y editan fácilmente, de modo que, la funcionalidad que en el instrumento tradicional estaba definida por el fabricante, en el instrumento virtual está definida por el usuario final. Es así como junto con los dispositivos de adquisición de datos, con su correspondiente software, permiten el desarrollo de sistemas de bajo costo y gran flexibilidad. Considerando el objetivo, propuse al Dr. Romo y a mi director de tesis el utilizar este tipo de tecnología para realizar el proyecto.

Para diseñar este sistema se utilizó un método llamado iterativo incremental, el cual propone analizar las características y especificaciones del proyecto, dividirlo en módulos generales determinando su jerarquía, definir las tareas específicas de dichos módulos para poder analizar, diseñar, implementar, probar y documentar cada módulo por separado. Esto permite reducir riesgos críticos y hacer cambios tácticos, basado en la opinión del usuario final al momento de incluir las nuevas características y mejoramientos.

Cada módulo se consideró como una iteración, donde en las primeras iteraciones se determinó el sistema, se dominaron los requisitos y eliminaron los riesgos críticos; en las iteraciones posteriores, se redujeron los riesgos menos graves y se implementaron los programas, así se añadieron incrementos hasta obtener un sistema final.

Las iteraciones permitieron limitar, atenuar y/o controlar los riesgos, de tal manera que se planificó y en algunas ocasiones se re-planificaron los proyectos, siempre en el margen de los requisitos. Se hicieron pruebas para eliminar los errores constantes y se redujeron los gastos al mínimo.

Al finalizar cada iteración, y antes de pasar a la iteración siguiente, se evaluó si todos los problemas habían sido resueltos, de no haber sido así se amplió la iteración hasta cumplir con lo requerido, así mismo se atendieron los aspectos del rendimiento y la disponibilidad del software y hardware observando si existían nuevos requisitos para la siguiente iteración y haciendo pruebas conjuntas, es decir si los módulos elaborados anteriormente trabajan bien con el modulo actual. Al pasar de una iteración a otra se decía que había un incremento.



2. Análisis del sistema de control de experimentos de psicofísica.

La percepción es el proceso mediante el cual la conciencia integra los estímulos sensoriales sobre objetos, hechos o situaciones y los transforma en experiencia útil. Uno de los métodos utilizados en neurociencias para el estudio de la percepción es la psicofísica, que se encarga de cuantificar las sensaciones a partir de la estimulación.

2.1 Descripción de los experimentos de psicofísica.

En un estudio psicofísico, un sujeto de experimentación realiza una ***tarea***, la cual consiste en presentarle estímulos al sujeto para que responda dependiendo del tipo de sensación que el estímulo le provoque. Esto permite investigar la relación entre los atributos de los estímulos y la percepción del sujeto manifestada a través de respuestas motoras.

En el laboratorio del Dr. Romo se estudia principalmente la modalidad sensorial táctil de la percepción para lo cual se aplican vibraciones mecánicas sobre los dedos usando cuatro tareas conductuales, que describiré a continuación:

a) *Tarea de detección.* Consiste en presentarle un estímulo que el sujeto de experimentación podrá o no percibir, de acuerdo a sus parámetros (amplitud, frecuencia o duración). Este experimento sirve para ubicar el umbral de detección (límite inferior percibido).

b) *Tarea de categorización*. Se presenta un estímulo cuya magnitud cambia en cada presentación entre un valor máximo y un valor mínimo. El sujeto debe establecer si el estímulo presentado se parece más al valor máximo o al valor mínimo. Por ejemplo, el sujeto clasificará el parámetro variado en rápido o lento (si el parámetro que varia es la frecuencia), largo o corto (si el parámetro es la duración) o en fuerte o suave (si el parámetro es la amplitud).

c) *Tarea de discriminación.* Se presentan dos estímulos y el sujeto de experimentación deberá establecer la relación entre la magnitud del segundo estímulo respecto del primero. Por ejemplo, una respuesta será: el segundo estímulo es más rápido, largo o fuerte que el primero, la otra respuesta será que el segundo estímulo es más lento, corto o suave que el primero.

d) *Tarea de reconocimiento de patrones temporales*. Se presentan dos estímulos con el mismo número de pulsos mecánicos (presión de corta duración), que pueden estar distribuidos de cualquier manera en un intervalo de tiempo, el sujeto de experimentación deberá identificar si los dos estímulos son iguales o no.

Por lo tanto, existen dos posibles respuestas en cada una de las tareas descritas. Esa respuesta es indicada por el sujeto al presionar de uno de dos botones que se encuentran ubicados frente a él. Uno de los botones es llamado botón central y el otro es llamado botón lateral. En la Tabla 1 se describen los estímulos según las tareas y la respuesta correcta.

Tabla 1 *Asociación de los botones con las respuestas según la tarea propuesta.*

	R e s p u e s t a s	
Tipo de tarea	Botón central	Botón lateral
Detección	Estímulo percibido	Estímulo no percibido
Categorización	Estímulo rápido, largo o fuerte (categoría alta)	Estímulo lento, corto o suave (categoría baja)
Discriminación	Segundo estímulo mas rápido, largo o fuerte que el primero (segundo estímulo de mayor magnitud)	Segundo estímulo mas lento, corto o suave que el primero (segundo estímulo de menor magnitud)
Patrones temporales	Estímulos iguales	Estímulos diferentes

Como ejemplo, en la Figura 1 se muestra la tarea de discriminación donde el parámetro variable en los dos estímulos presentados, es la frecuencia. En el caso A la frecuencia del segundo estímulo es menor que la frecuencia del primer estímulo y la respuesta correcta es la presión del botón lateral (BL). En el caso B

la frecuencia del segundo estímulo es mayor que la del primer estímulo y la respuesta correcta es la presión del botón central (BC).

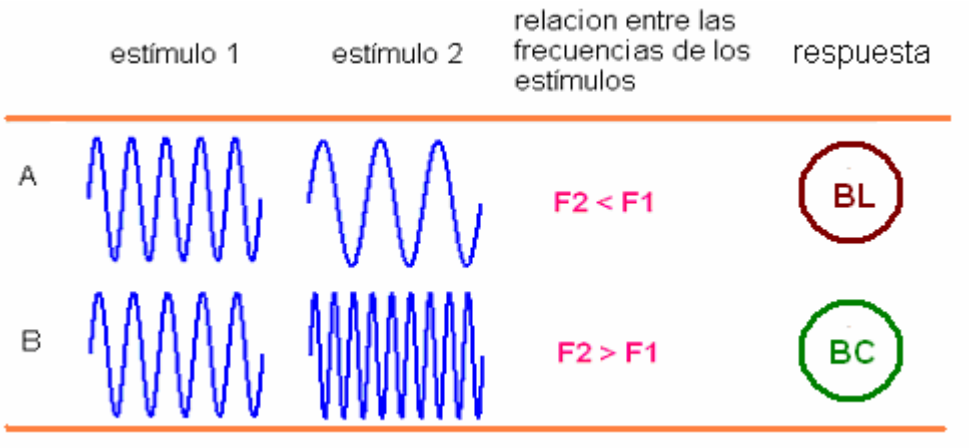


Figura 1 *Tarea de discriminación.*

Los estímulos tactiles son aplicados con la flecha de un motor lineal, el sujeto de experimentación debe mantener una mano en una misma posición sobre una mesa, con la palma hacia arriba, la flecha del motor hará contacto con la superficie de un dedo y proporcionará los estímulos al desplazarse verticalmente sobre la piel de la yema del dedo.

En la Figura 2 se muestra el motor lineal (B) estimulando a un sujeto de experimentación (C). El módulo A recibe las señales y las acondiciona para enviarla al motor lineal.

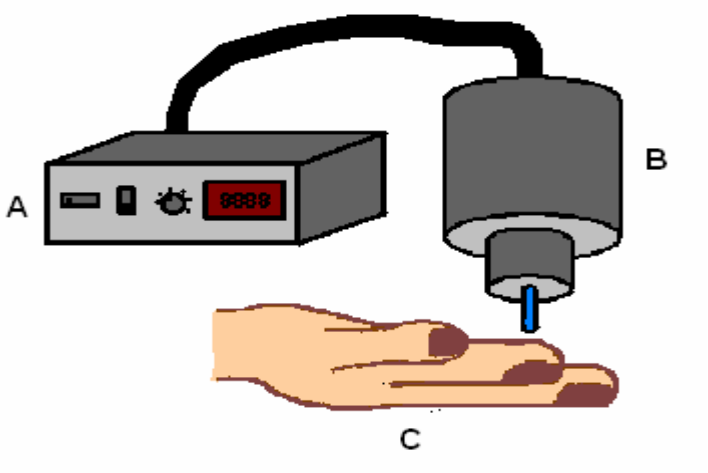


Figura 2 *Estimulador táctil (Vibratactile Stimulator BME Systems Inc.).*

Los experimentos de estímulos visuales consisten en señalar al sujeto el botón que debe oprimir mediante el encendido de alguno de dos indicadores. La

respuesta que deberá dar el sujeto será de acuerdo al botón que se encienda y coincidirá necesariamente con las condiciones descritas. Cada uno de los botones cuenta con un diodo emisor de luz en su interior y una tapa de acrílico de color.

Para indicarle al sujeto de experimentación que la presentación de los estímulos está próxima, el sistema desplaza la flecha del motor lineal desde una posición inicial, en donde no tiene contacto alguno con el dedo, hasta la posición donde hace contacto con el dedo.

Después, el sistema espera a que el sujeto de experimentación indique que está listo para recibir los estímulos, lo cual es realizado por dicho sujeto mediante la colocación de su mano libre sobre una barra horizontal. La barra tiene emisores y receptores infrarrojos que detectan si el sujeto ha colocado o retirado la mano de dicha barra. La respuesta del sujeto debe ocurrir en un lapso de tiempo cuyo máximo valor ha sido determinado previamente por el investigador.

Una vez que se ha recibido la señal de que el sujeto está listo para recibir los estímulos, el sistema genera un retardo cuya duración fue establecida por el investigador. Al final del retardo se genera el o los estímulos tactiles con sus respectivos retardos entre estímulos. Una vez que se ha terminado la estimulación, el sistema le indica al sujeto de experimentación que debe dar la respuesta, esto lo hace desplazando la flecha del motor lineal, que aún hace contacto con el dedo, a una posición en donde ya no haga contacto; en ese instante el sujeto retira la mano de la barra y debe presionar el botón que considere adecuado.

A continuación el sistema evalúa si el ensayo concluyó exitosamente, de ser así, se recompensa al sujeto de experimentación con una cantidad específica de gotas de agua o jugo (en caso de que el sujeto de experimentación sea un mono rhesus), que son proporcionadas por medio de una válvula electromecánica. Si el sujeto ha errado su respuesta, se aplica un castigo que consiste en un retardo largo antes de iniciar la siguiente estimulación. Aquí también, las respuestas del sujeto para retirar la mano y presionar el botón deben ocurrir dentro de lapsos de tiempo cuyos valores máximos han sido determinados previamente por el investigador.

Al conjunto de acciones que el sistema y que el sujeto realizan desde que la punta del motor lineal se desplaza para hacer contacto con el dedo del sujeto y hasta que el sujeto recibe la recompensa(o castigo) se le llama ***ensayo***.

En caso de que el sujeto de experimentación no realice las acciones adecuadas en los tiempos definidos por el investigador, el sistema volverá a iniciar el ensayo independientemente del proceso en que se encuentre.

Considerando los lapsos de tiempo donde el sujeto debe responder se definen dos tipos de errores que provocan que el sistema reinicie el ensayo:

a) *Respuesta temprana*. Es cuando el sujeto de experimentación coloca la mano en la barra horizontal antes que la flecha del motor lineal haga contacto con el dedo a estimular, o bien, cuando el sujeto de experimentación retira la mano de la barra horizontal antes que la flecha del motor lineal haya dejado de hacer contacto con el dedo estimulado.

b) *Respuesta tardía*. Sucede cuando la flecha del motor hace contacto con el dedo del sujeto y este no pone la mano sobre la barra horizontal, también cuando la flecha del motor ha dejado de hacer contacto con el dedo del sujeto y éste no retira la mano de la barra horizontal, una última posibilidad ocurre cuando, habiéndose retirado la flecha del motor, el sujeto retira la mano de la barra horizontal, pero no presiona ningún botón.

El conjunto de parámetros definidos por el investigador para un estímulo (o par de estímulos) según la tarea se llama ***clase***, y existen tantas clases como estímulos diferentes, de modo que el número de veces que quieres que se repita cada clase multiplicado por el número de clases da como resultado el total de ensayos en un experimento.

Total de ensayos = (Número de clases) x (Numero de ensayos por clase)

Así, si el investigador define 8 clases diferentes y 10 repeticiones para cada clase, tendrá 80 ensayos totales. Al conjunto de ensayos así definido se le llama un ***experimento.***

Durante un experimento se desea observar en una gráfica el *desempeño* del sujeto, es decir el porcentaje de aciertos que tiene ensayo a ensayo, así como una tabla donde indique que clase se ha ejecutado y si el sujeto ha tenido un acierto o un error.

La secuencia en la cual se presentan las diferentes clases definidas por el investigador puede ser aleatoria o secuencial. Durante el desarrollo del experimento se utiliza un generador de ruido blanco con el fin de enmascarar la actividad ajena al experimento.

2.2 Descripción de los requerimientos de entrada y salida de las funciones principales del sistema.

Para llevar a cabo un experimento los investigadores deben elegir el tipo de tarea que realizará el sujeto, poner los parámetros de el (los) estímulo(s), es decir, definir las clases, indicar el número de veces que se desea repetir cada una, definir si la secuencia de las clases se realizará en orden o aleatoriamente, indicar el número de ensayos por clase, aplicar o no estímulo visual, e indicar si durante el experimento se escuchará o no un ruido blanco.

Existe la posibilidad de que el investigador desee observar la secuencia del experimento sin ningún sujeto de experimentación que lleve a cabo la tarea, en cuyo caso se le llama **experimento pasivo** y el sistema ejecuta las acciones como si el sujeto de experimentación tardara el tiempo máximo en realizar sus movimientos.

Una vez iniciado el experimento, los investigadores necesitan tomar decisiones durante la ejecución del mismo, como son: detener parcialmente el experimento (pausa), continuarlo, repetir el ensayo actual, repetir el ensayo en caso de error, detener el experimento, poner el tiempo de castigo y el nivel de recompensa y, al final, tomar la decisión de guardar o no los resultados.

Por lo tanto, el sistema requiere adquirir la información que le indica el tipo de tarea realizará, los datos de las clases, las demás configuraciones para el experimento y durante la ejecución del mismo se deben leer los controles. Toda esta información es aportada por el investigador.

La ejecución de un ensayo depende de las acciones del sujeto de experimentación y el control del experimento depende de las acciones tomadas por el investigador. El sistema debe adquirir toda esa información y actuar en consecuencia.

El sistema ejecuta las tareas de acuerdo a las configuraciones y comandos que el usuario ingresa a través de interfaces gráficas, también estimula al sujeto de experimentación y recibe las respuestas motoras, las cuales presenta al usuario como resultados[1]. Ver Figura 3.

[1] En el diagrama de bloques el usuario es el investigador.

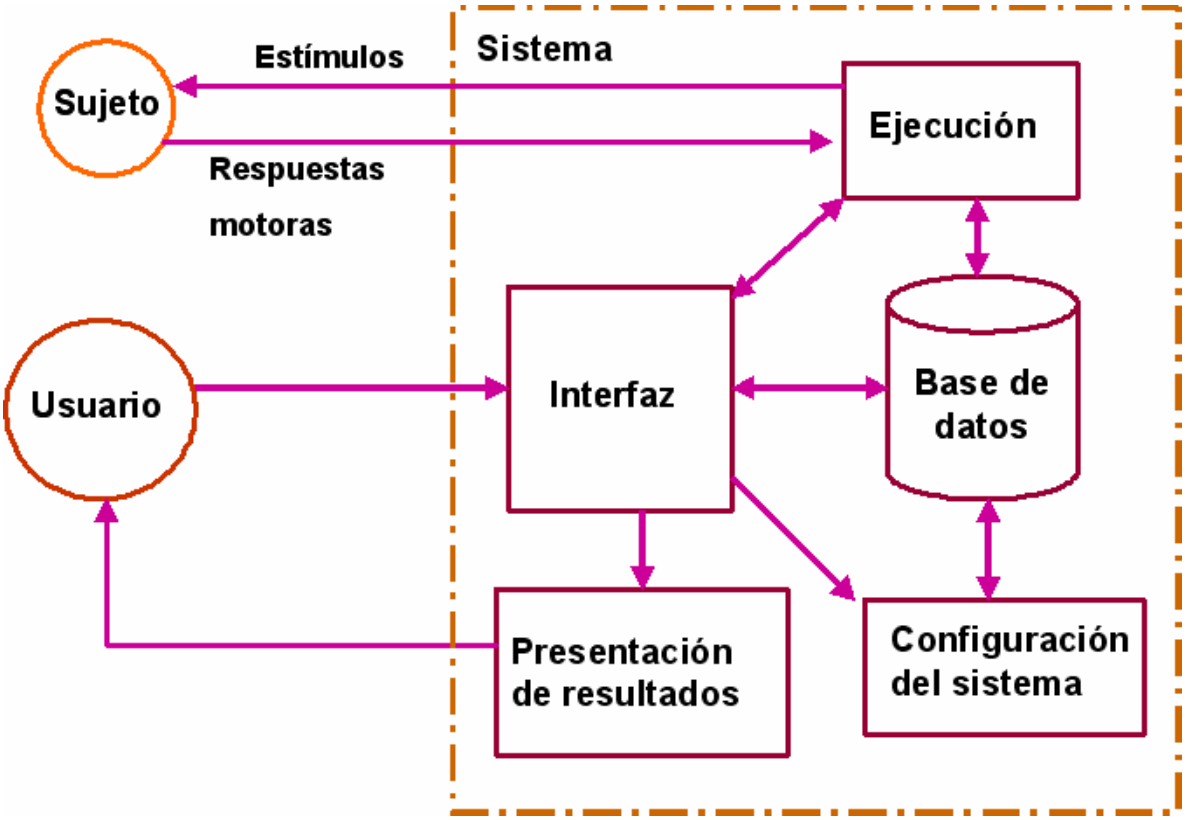


Figura 3 *Diagrama de bloques de los experimentos de psicofísica.*

Para llevar a cabo el experimento se seguirán los siguientes pasos:

Primero el investigador deberá establecer las clases para los estímulos, es decir, los parámetros de amplitud, duración, frecuencia, forma de onda, o el tipo de patrón temporal, así como la respuesta esperada y si habrá retardos para los estímulos. Esto se hace por medio de una *hoja de cálculo*.

Una vez elaborado el archivo de las clases, que se almacenará en formato de texto, dicho archivo formará parte de la *base de datos* y podrá usarse en cualquier momento.

A través de una primera interfaz gráfica (Figura 4) el usuario configura el sistema de la siguiente manera:

a) Se define el tipo de tarea: detección, categorización, discriminación o reconocimiento de patrones temporales.

b) Se definen las clases: de la base de datos se elige el archivo que contenga los parámetros de los estímulos que se desea presentar al sujeto.

c) Se teclea el número de repeticiones por clase en un control numérico.

d) Se elige la secuencia que habrá de seguirse, es decir, si las clases se presentarán en orden (clase 1, clase 2, etc) o aleatoriamente.

e) Se propone el tipo de experimento, es decir, si será pasivo (el sistema lleva a cabo los ensayos sin sujeto de experimentación) o si será activo (que el sistema lleve a cabo los ensayos de acuerdo a las decisiones que va tomando el sujeto de experimentación).

f) Se indicará si se presentará ruido blanco o no durante la ejecución del experimento.

g) La última configuración será si habrá estímulo visual, es decir si se usaran las luces.

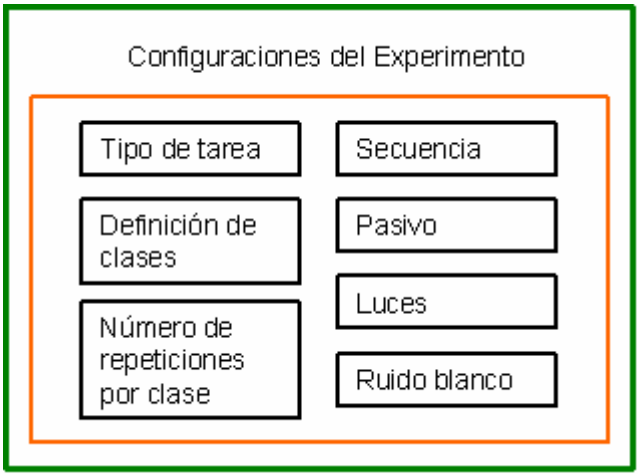


Figura 4 *Prototipo de la interfaz gráfica de usuario para configuración del sistema.*

Con una segunda *interfaz gráfica de usuario* se controla la *ejecución* del experimento, con botones que permiten: Iniciar/detener, Pausar/continuar el experimento, repetir error, repetir ensayo, y con controles numéricos para poner los niveles de recompensa y los niveles de castigo (ver Figura 5).

Una vez inicializado el experimento, se ejecutan las funciones que presentan los estímulos al sujeto de experimentación y también aquellas que leen su respuesta y la evalúan.

En esta misma interfaz gráfica se *presentarán los resultados* de cada ensayo, es decir, una gráfica de desempeño por clase; la gráfica de respuesta en el tiempo también por clase, donde es posible observar el tiempo que transcurre desde que el sujeto de experimentación liberó la barra hasta que presionó un botón; una tabla de resultados que indique, si el sujeto de experimentación acertó o erró; el número de ensayo que se está llevando a cabo y a que clase pertenece, así como, los parámetros de los estímulos.

Al finalizar el experimento, ya sea por que el investigador decide detenerlo o por que se llevaron a cabo todos los ensayos, aparecerá un último control que permitirá almacenar la información presentada si así lo desea el usuario.

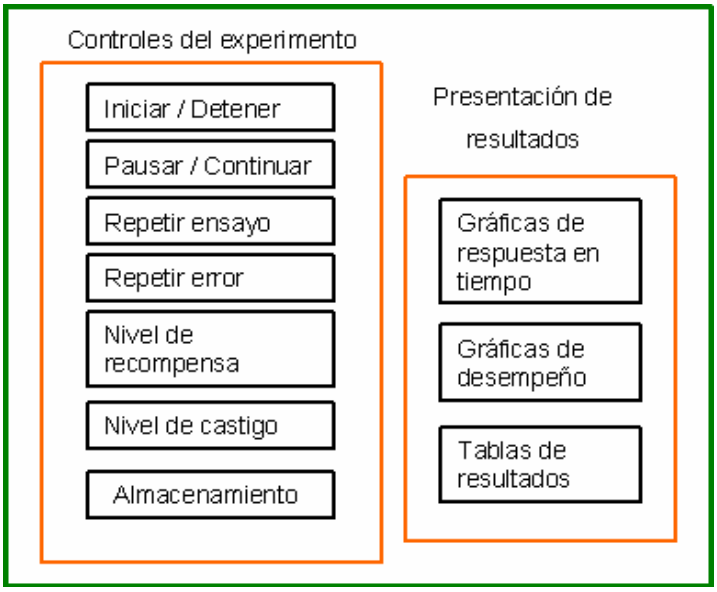


Figura 5 *Prototipo de la interfaz gráfica de usuario para el control de la tarea y presentación de resultados.*

2.3 Análisis detallado del ensayo.

El experimento consta de varios ensayos y cada ensayo consta de una serie de estados que se llevan a cabo antes y después de la estimulación. La estimulación a su vez es un estado y se rige por el tipo de tarea seleccionada y por los atributos de los estímulos, es decir, por las clases.

Una vez adquiridos los datos necesarios el sistema comienza a ejecutar los ensayos. En ese momento el sistema debe adquirir las respuestas motoras del mono, que en este caso son la barra horizontal libre u ocupada y la presión del botón central o del botón lateral; de acuerdo a estas señales y una señal extra que indica si el tiempo para cada etapa (estado) del ensayo ha transcurrido, el sistema debe elegir entre ejecutar el estado siguiente o interrumpir el ensayo. En la Figura 6 se muestra un ejemplo donde cada línea vertical indica el fin de un estado.

Los ensayos son divididos en **estados**, los cuales se definen como una condición estable del sistema donde se realiza una acción y se espera un evento que define el estado siguiente.

Los siguientes son los estados en los que se dividen los ensayos, independientemente de la tarea que se realiza:

- I: Estado inicial, se verifican los periféricos de entrada (palanca, botones, recompensa, ruido, etc.) se prende la luz del botón que indica la respuesta correcta, si así se desea.

- R-: Estado de rampa negativa, habrá un desplazamiento de la flecha estimuladora hasta tocar el dedo del sujeto.

- BO: Estado de barra ocupada, se espera que el sujeto pose su mano en la barra horizontal.

- r: Estado de retardo, habrá un retardo de duración variable.

- E: Estado de estimulación, se presentará el estímulo correspondiente.

- R+: Estado de rampa positiva, habrá un desplazamiento de la flecha estimuladora hasta la posición inicial.

- BL: Estado de barra libre, se espera que el sujeto retire su mano de la barra horizontal.

- PB: Estado de presión de botón, se espera que el sujeto presione el botón correcto.

- F: Estado final, se apaga la luz en caso de haberse encendido y se proporcionará recompensa en caso de que el sujeto acertara en su repuesta.

- Error: Estado de error, el cual apaga luces y sube la punta en caso de encontrarse en otra posición.

En la Tabla 2 se muestra para la tarea de detección la dependencia del estado siguiente en función del estado actual y el evento.

En la Figura 7 se muestra para la tarea de discriminación la división de un ensayo en estados, cada línea vertical indica el final de un estado.

En la Tabla 3 se muestra para la tarea de discriminación la dependencia del estado siguiente en función del estado actual y el evento.

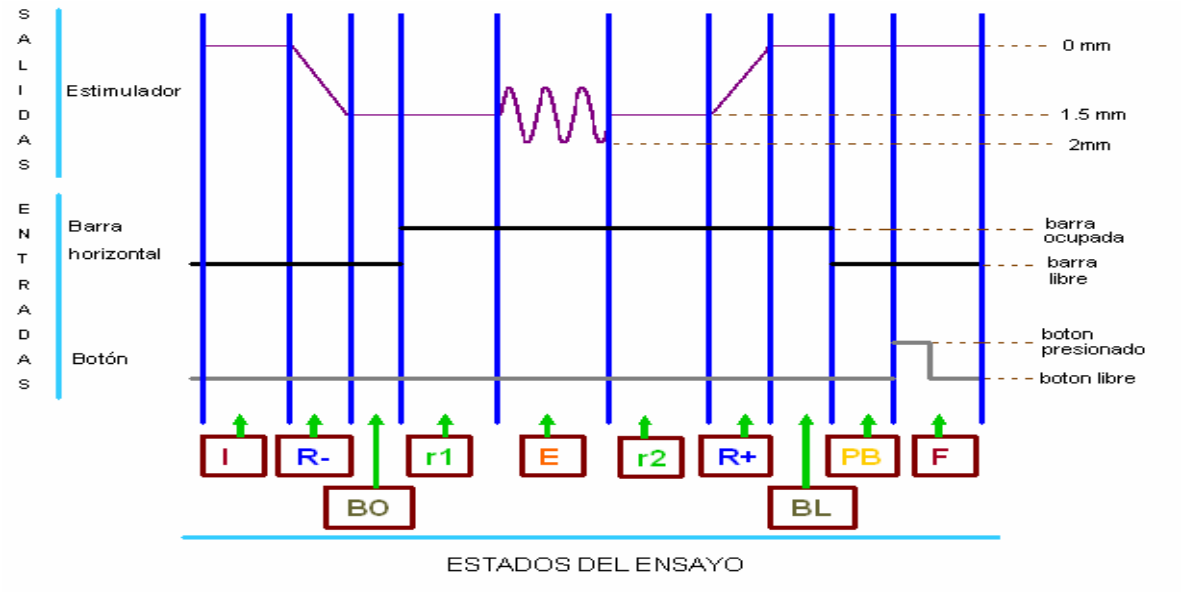


Figura 6 *División de un ensayo en estados de la tarea de detección.*

Tabla 2 *Secuencia de estados para la tarea de detección según el evento.*

Evento	Estado actual	Estado siguiente
Tiempo de estado transcurrido	I	R-
	R-	BO
	r1	E
	E	r2
	r2	R+
	R+	BL
	BO	error
	BL	
	PB	
Barra ocupada	BO	r1
Barra libre	BL	PB
	r1	
	E	
	r2	error
	R+	
Presión de algún botón	PB	F
Barra ocupada o Presión de algún botón	I	error
	R-	

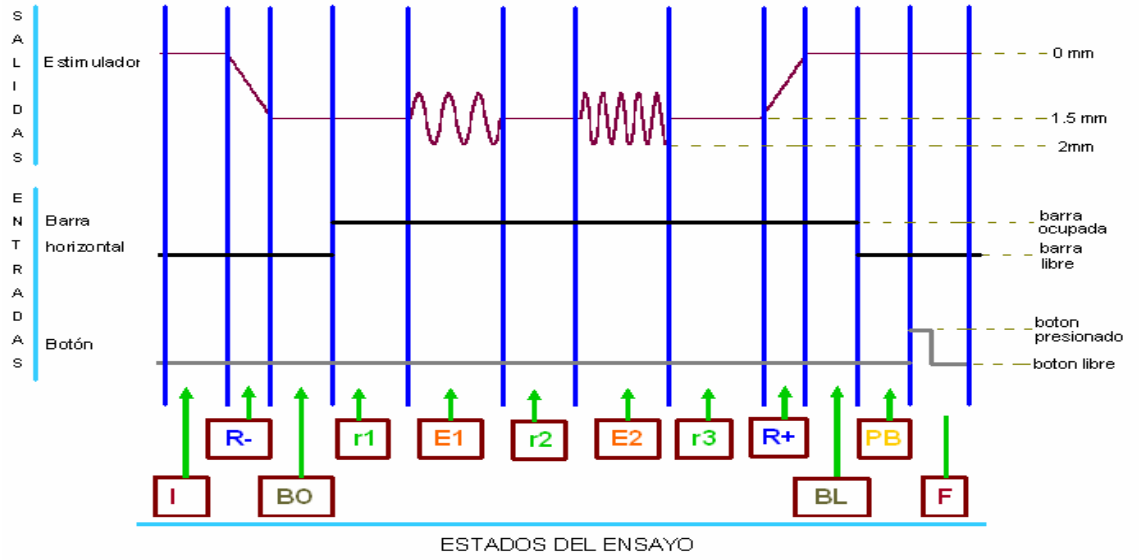


Figura 7 *División de un ensayo de la tarea de discriminación en estados.*

Tabla 3 *Secuencia de estados para la tarea de discriminación y de patrones temporales.*

Evento	Estado actual	Estado siguiente
Tiempo de estado transcurrido	I	R-
	R-	BO
	r1	E1
	E1	r2
	r2	E2
	E2	r3
	r3	R+
	R+	BL
	BO	error
	BL	
	PB	
Barra ocupada	BO	r1
Barra libre	BL	PB
	r1	error
	E1	
	r2	
	E2	
	r3	
	R+	
Presión de algún botón	PB	F
Barra ocupada o Presión de algún botón	I	error
	R-	

Los eventos esperados, serán entonces:

- Botón central presionado
- Botón lateral presionado
- Barra libre
- Barra ocupada
- Tiempo del estado transcurrido

El último evento esperado, ‘tiempo del estado transcurrido’ indica, en los estados en los que el sujeto de experimentación no realiza ninguna acción, que el estado ha terminado, y da la pauta para continuar con el estado siguiente. Para los estados en los cuales se espera que el sujeto de experimentación realice una acción, la duración del estado indica el tiempo máximo en el cual el sistema esperará la respuesta del sujeto, de modo que de no presentarse el evento esperado (acción del sujeto) se presentará el evento ‘tiempo del estado transcurrido’, y el estado siguiente será el estado error.

La electrónica implementada genera un pulso TTL por cada evento, es decir, que cuando la palanca se libera se genera un pulso, lo mismo cuando se ocupa, cuando transcurre el tiempo de estado y cuando se presionan los botones. Los eventos esperados para cada estado se muestran en la Figura 8.

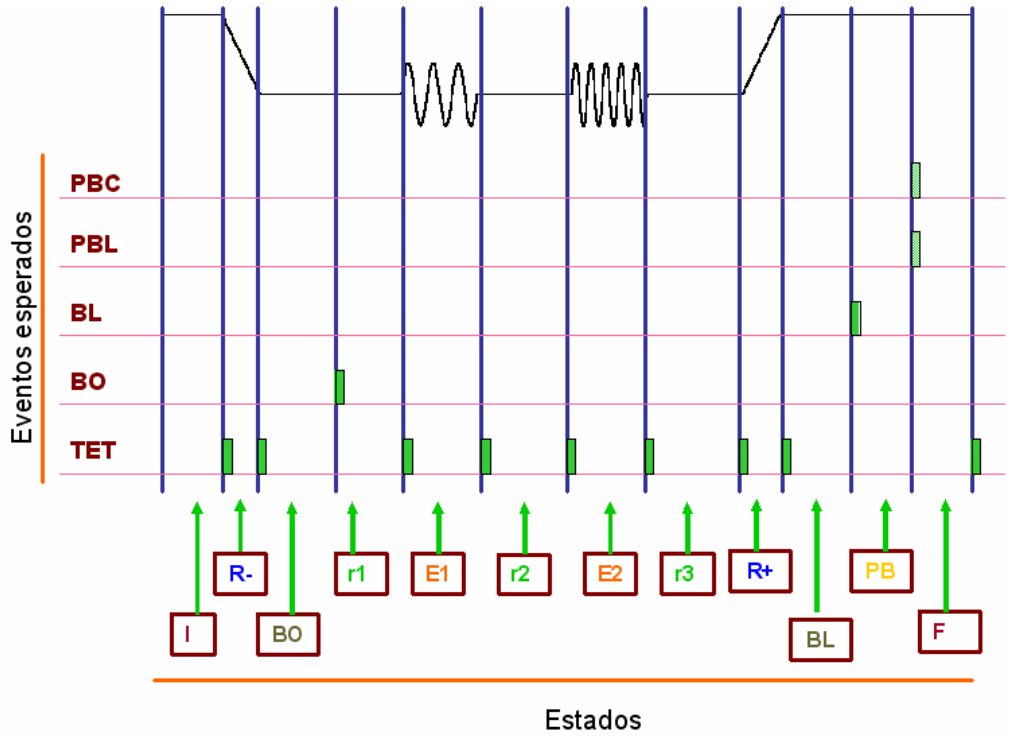


Figura 8 *Eventos esperados en un ensayo para la tarea de discriminación.*

En cada estado el sistema realiza una acción, registra el tiempo en que se realiza la acción o en todo caso el tiempo en que el sujeto da una respuesta, evalúa el evento ocurrido y llama al estado siguiente.

Para tener una idea clara en la Figura 9 se muestra el diagrama de estados que integra la información de las tablas y de las figuras para las tareas de discriminación y de patrones temporales. El estado inicial se representa con la letra I, el estado final con la letra F.

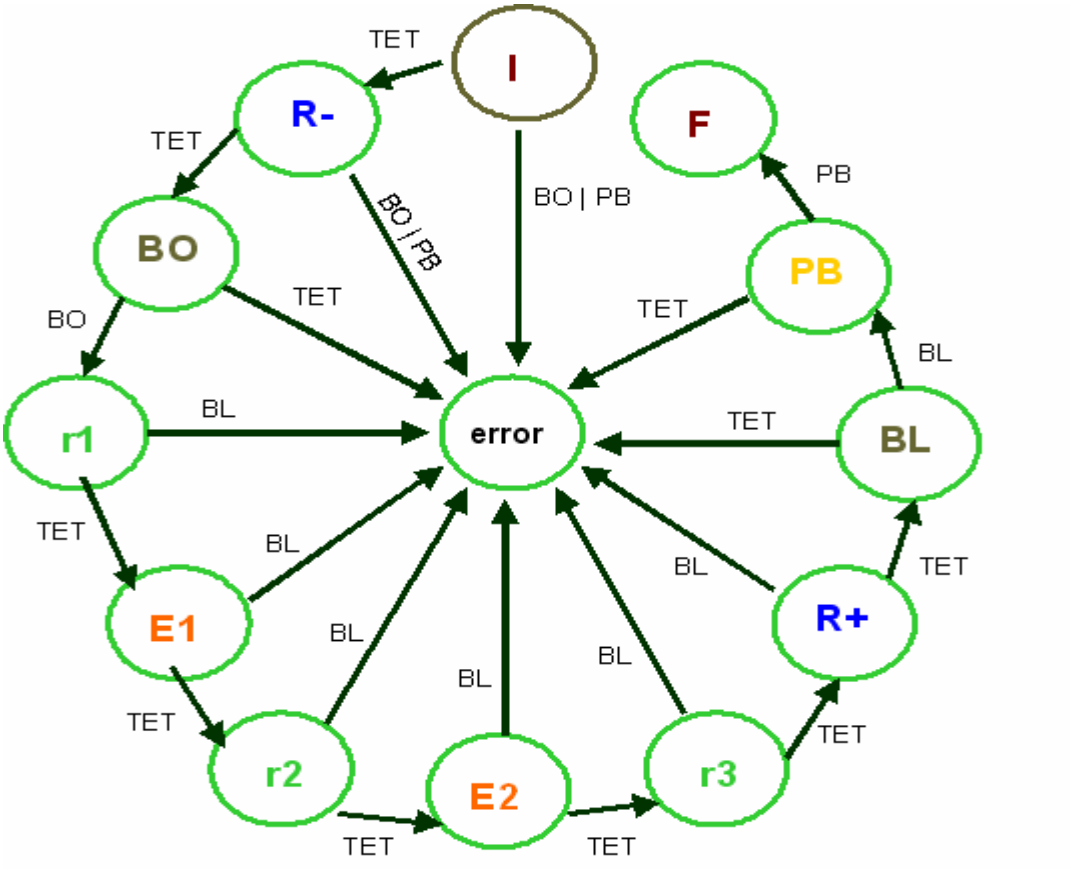


Figura 9 *Diagrama de estados para la tarea de discriminación.*

2.4 Recursos utilizados actualmente.

El sistema para el desarrollo de los experimentos de psicofísica, con el que actualmente cuenta el laboratorio, está compuesto por los siguientes elementos:

Una computadora llamada constructor, que trabaja bajo el sistema operativo Windows 98, la cual cuenta con la interfaz de usuario programada en C++ , su función consiste en permitir al investigador crear su base de datos, y con la interfaz de usuario el investigador configura y controla el experimento, toda la información que esta computadora recibe del investigador, la distribuye a otras dos computadoras, para el desarrollo del experimento.

Una computadora llamada controlador, esta computadora trabaja bajo el sistema operativo DOS, cuenta con funciones programadas en C, que adquieren la respuesta motora del sujeto de experimentación y con ello y la información enviada por “el constructor” dirigen la secuencia de los estados de cada ensayo, esta computadora cuenta, para poder adquirir la respuesta motora del sujeto de experimentación, con la tarjeta de adquisición y generación de señales digitales

PC-TIO-10, que además permite saber mediante temporizadores programables cuanto tiempo tardó el sujeto en dar su respuesta.

Una computadora llamada generador, esta computadora trabaja bajo el sistema operativo DOS, cuenta con funciones programadas en C, que permiten la generación de los estímulos y las rampas, por medio de una tarjeta de adquisición y generación de señales analógicas LAB-PC y un sintetizador de señales, “el controlador” le indica en que momento genera las rampas positiva y negativa. También en que momento la tarjeta LAB-PC se comunica con el sintetizador para generar los estímulos, previamente “el constructor” le envió los parámetros necesarios para las rampas y los estímulos. Todas las computadoras cuentan con tarjetas de comunicación que usan el protocolo GPIB (general purpose interface bus).

Como se observa, el sistema actual requiere de varios recursos, lo que implica, en el caso de requerir otro sistema, un costo elevado, y gran cantidad de tiempo para comunicar y sincronizar las computadoras y así llevar a cabo los experimentos. Además la tecnología usada en ese sistema ya no se consigue en el mercado por usar el bus de expansión ISA.

2.5 Propuesta de nuevo hardware y software.

La instrumentación tradicional se refiere a todo aquel instrumento tipo hardware rígido que se puede adquirir de distintos fabricantes y cuya funcionalidad viene definida por éstos.

Con el ingreso de la computadora personal, los instrumentos sacaron el máximo potencial de las mismas, abriéndose camino a dos nuevos conceptos: la instrumentación virtual y los sistemas de adquisición de datos basados en computadora.

Con la computadora personal, podemos procesar señales digitalizadas y almacenar los datos y resultados en disco duro junto con los programas que se ejecutan para procesar la señal. Los datos procesados pueden ser visualizados en pantalla. Por último los convertidores analógicos digitales necesarios para ingresar la señal a la computadora están contenidos en una tarjeta de expansión, las cuales también pueden incluir convertidores digitales-analógicos que nos permiten generar señales analógicas de salida. Estas tarjetas también cuentan con entradas y salidas digitales y en muchos casos con circuitos temporizadores programables.

Con todos esos recursos podemos definir un *Instrumento Virtual* como toda computadora en la cual se ha insertado o conectado interna o externamente un dispositivo de adquisición y/o generación de señales acompañado de un software lo suficientemente poderoso y flexible como para procesar, analizar, almacenar, distribuir y desplegar datos e información relacionados con la medición de una o

varias señales específicas, involucrando también la interfaz hombre-máquina y la comunicación con otros equipos, el usuario define la funcionalidad y apariencia del instrumento.

En la computadora se procesará la información, la cual puede ser visualizada en tiempo real, guardada en formatos específicos, transmitida a través de la red y analizada.

El software de control es la aplicación computacional a través de la cual se crea el instrumento virtual y se programa la funcionalidad deseada.

A través de un bus de conexión o un puerto de conexión se conectará la computadora con dispositivos externos como un acondicionador de señales.

El adquiridor de datos, es el elemento que toma los datos del acondicionador y hace la conversión analógica digital de la información.

La propuesta es que el sistema que para el control de experimentos de psicofísica sea un sistema de adquisición/generación de datos basado en la instrumentación virtual.

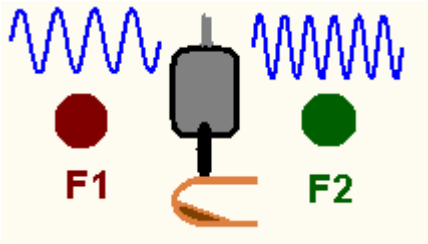
La computadora que procesará la información de la respuesta motora del sujeto de experimentación, donde estarán las interfaces de usuario, para configuración y control del experimento y donde se visualizará el desarrollo del experimento por medio de gráficas y tablas, es una computadora proporcionada por el laboratorio con las siguientes características: un procesador Intel Pentium 4 a 1.7 GHz, 1GB en memoria RAM, Disco duro de 20 GB, el sistema operativo Windows XP Home Edition.

El software seleccionado para desarrollar el instrumento virtual fue LabVIEW, este es un sistema de desarrollo cuyo fabricante es National Instrument quien también provee del hardware de adquisición/generación. Es un software gráfico que consta de un set de objetos que describen funciones como adquisición de datos, análisis, lectura y escritura de archivos y otros. Cada programa realizado en LabVIEW se denomina instrumento virtual o simplemente VI (por sus siglas en ingles 'virtual instrument') y consta de dos áreas bien definidas, la primera llamada Panel Frontal, que es la interfaz gráfica de usuario, la otra área es llamada Diagrama de Bloques, que es el área de programación donde se van colocando los objetos (los que describen las funciones). La secuencia se define cableando los objetos entre sí, lo que se asemeja a realizar un diagrama de flujo, siempre recordando que en la programación gráfica se define la secuencia en cuanto a que los objetos entregan salidas solo hasta tener todos los datos en sus entradas, lo que evita salidas erróneas o con información falsa. Una ventaja es que los VIs pueden anidarse uno dentro de otro, con lo cual el entorno es jerárquico y reusable.

La versión utilizada es LabVIEW 7.0, los requerimientos recomendados por National Instruments son: 256 MB en RAM, una resolución de pantalla de 1024x768 pixeles, un procesador Pentium 4 o equivalente y un espacio libre en disco duro de 550 MB, características que cumple la computadora que el laboratorio proporcionó.

El bus de conexión es PCI (peripheral component interconect). Este bus maneja 32 bits para la transmisión, además introduce la tecnología llamada “bus mastering” que permite a los dispositivos PCI tomar el control del bus y manejar directamente las transferencias a la memoria sin intervención de la CPU.

El hardware utilizado es una tarjeta PCI-6071-E de national Instrument, proporcionada por el laboratorio, esta tarjeta ofrece 64 entradas analógicas, 2 salidas analógicas, 8 enradas/salidas digitales y 2 contadores por hardware.



3. Diseño del sistema de control de experimentos de psicofísica.

3.1 Módulos para la construcción de ensayos.

El sistema debe permitir al investigador, por medio de interfaces gráficas, la selección de las características que definen el experimento y el control del desarrollo del mismo. Esto implica que el sistema debe ser capaz de: a) leer los parámetros para las señales del experimento de una base de datos generada por el investigador, b) integrar los estados necesarios y la secuencia de los mismos, c) leer controles para el inicio, fin, pausa y repetición de ensayos, y d) procesar cada estado hasta completar un ensayo, y repetir esto ensayo a ensayo hasta completar un experimento.

Para generar cada estado se requirió la implementación de los siguientes módulos.

I. Módulo “acción principal”. Define la acción de salida del estado, por ejemplo si es un estado rampa, la acción principal será mandar una rampa por el canal analógico de salida y directo al estimulador. Las salidas del sistema son: a) desplazamiento de la punta estimuladora (generación de rampas positiva y negativa y generación de estímulos táctiles); b) encendido de la señal visual (leds en los botones); c) activación de la recompensa. De modo que habrá una salida analógica y tres digitales. La respuesta del sistema para los estímulos no debe ser mayor a 1 ms.

II. Módulo “duración del estado”. Consiste en mandar una rampa por un canal analógico de salida que vaya de 0 V a 10 V en un tiempo definido por el usuario como duración del estado, de tal manera que esta señal será ingresada a un comparador y en el momento en que la señal llegue a 10 V el comparador mandará un pulso TTL a una entrada digital (este corresponde al evento ‘tiempo del estado transcurrido’), con la finalidad de que el sistema esté enterado que el tiempo del estado ha transcurrido.

III. Módulo “registro del tiempo”. Se realizo el circuito electrónico pertinente para que cuando se genere uno de los eventos esperados, el sistema sea notificado por medio de un pulso digital que se ha recibido un evento. La electrónica corresponde a una compuerta OR, cuyas entradas son los cinco pulsos de eventos esperados y cuya salida llamaremos EO (evento ocurrido). En la Figura 10 se observa el diagrama de la compuerta, sus entradas y la salida. La señal EO se conecto a la entrada de control de un temporizador (contador programable) de la tarjeta PCI-6071E

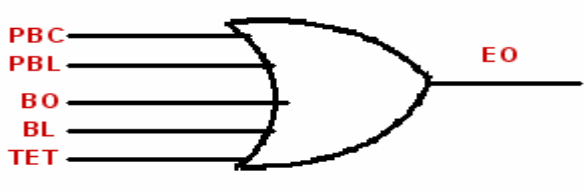


Figura 10 *Diagrama lógico de notificación para el sistema.*

Este módulo configura el temporizador para que a una frecuencia apropiada incremente su valor a partir de cero comenzando en el instante en que inicia cada estado. Cada vez que ocurra un pulso EO, el temporizador se detiene y guarda su valor en un registro, ese tiempo es leído y almacenado. Como resultado los estados en los cuales el sujeto de experimentación no realiza ninguna acción motora, el valor registrado debe ser igual al propuesto para el módulo de duración del estado y en los estados en los que se espera una acción motora del sujeto, el valor registrado será el tiempo de respuesta del sujeto.

IV. Módulo “validación de evento ocurrido”. Una vez recibido el pulso EO, se leen las entradas y se verifica que el evento ocurrido sea correcto para seguir con la secuencia de estados. Al ocurrir un evento incorrecto el sistema mandará como siguiente estado el estado error.

3.2 Señales de entrada y salida.

El sistema tendrá seis señales digitales de entrada: BL (barra libre), BO (barra ocupada), PBC (presión del botón central), PBL (presión del botón lateral), TET (tiempo de estado transcurrido) y EO (evento ocurrido), en la Figura 11 se muestran las señales que originan los eventos esperados. EO irá al contador, mientras que las otras cinco señales configurarán el puerto digital de la tarjeta de adquisición de datos.(ver Figura 12)

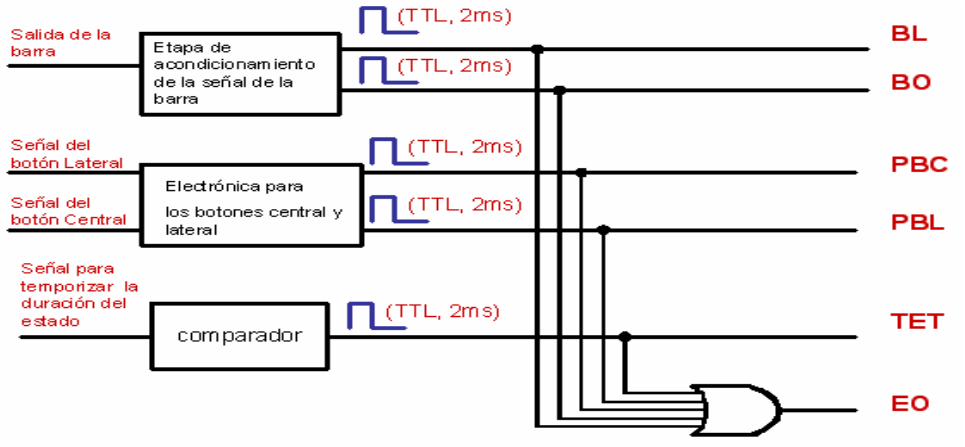


Figura 11 *Eventos esperados y señales que los originan.*

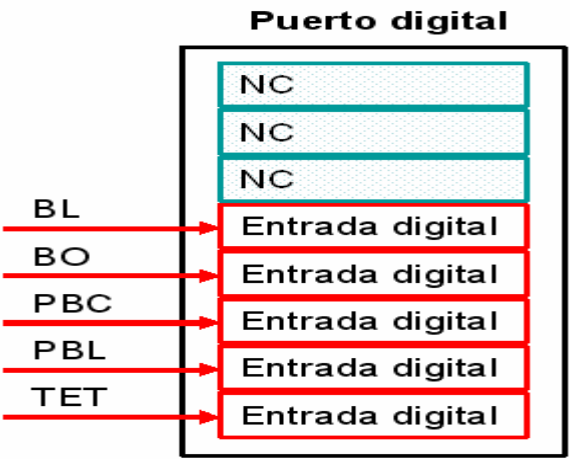


Figura 12 *Señales de entrada del puerto digital de la tarjeta de adquisición de datos.*

Las salidas analógicas del sistema son: la que controla la posición del estimulador (las rampas y los estímulos tactiles) y la rampa con duración igual a la duración del estado, es decir la rampa para el comparador que genera TET (ver Figura 13).

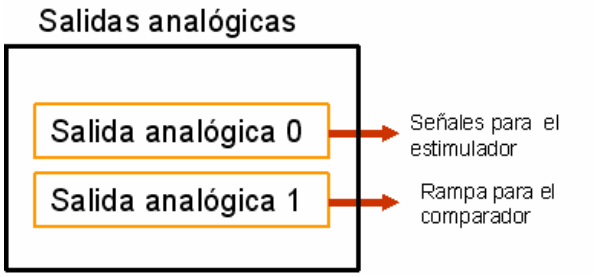


Figura 13 *Esquema del destino de las salidas analógicas*

El sistema tendrá tres salidas digitales: dos corresponden al pulso para las luces (estímulo visual: botón lateral y botón central) y la otra corresponde al control de la recompensa, dichas salidas son enviadas al puerto paralelo de la computadora.

Los componentes del contador son: a) el source que es una señal de entrada que cambia el conteo del registro, b) el gate, otra señal de entrada que determina si la señal en el sourse cambiará o no el conteo, c) el output que es una señal de salida que genera pulsos o una serie de ellos y d) el counter registrer almacena el conteo actual (parte superior de la Figura 14).

El pulso EO se ingresó al contador configurado en el modo “buffered period measurement” para detenerlo y guardar la cuenta del tiempo en un registro (parte inferior de la Figura 14)y posteriormente evaluar las entradas del puerto digital.

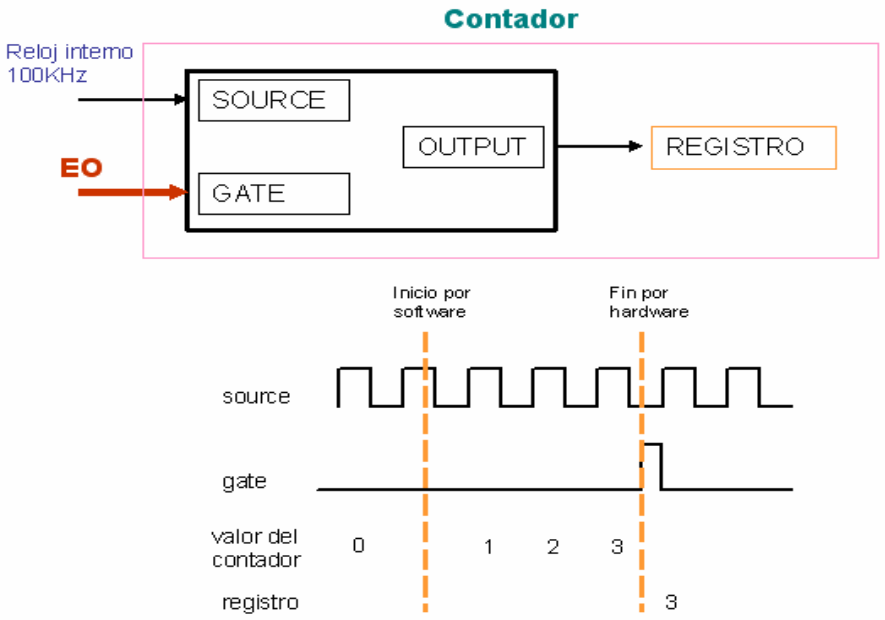


Figura 14 *Componentes del contador (parte superior). Configuración en el modo ‘buffered period measurement’ (parte inferior).*

3.3 Configuraciones y controles manejados por el sistema.

Para realizar un experimento de psicofísica se debe tomar en cuenta lo siguiente:

a) Realizar la configuración del experimento. En una primera interfaz gráfica podrán realizar las siguientes configuraciones:

- **Tipo de tarea**. Se determinará por medio de un menú desplegable el tipo de tarea que se realizará ya sea detección, categorización,

discriminación o reconocimiento de patrones temporales. El número de estados que componen el ensayo dependerá la tarea elegida (ver Tabla 4).

Tabla 4 *Estados que conforman los ensayos según el tipo de tarea.*

Tarea elegida	Estados que se llevan a cavo.
Todas las tareas	Inicio (I),rampa positiva (R+), barra ocupada(BO), rampa negativa(R-), barra libre (BL), presión de botón (PB), final (F).
Detección y categorización	Primer retardo (r1), primer estímulo (E1), segundo retardo (r2).
Discriminación y patrones temporales	Primer retardo (r1), primer estímulo (E1), segundo retardo (r2), segundo estímulo (E2), tercer retardo (r3).

Así, para las tareas de detección y de categorización, existen 10 estados, y para las tareas de discriminación y patrones temporales existen 12 estados.

Propuse crear un único archivo llamado 'todas_las_tareas', en donde se definen los parámetros para los estados comunes a todas las tareas (ver Tabla 5).

Tabla 5 *Contenido del archivo 'todas_las_tareas'.*

Estado	Duración [ms]	Amplitud [micrometros]
Inicio	1000	
Rampa negativa	100	1500
Barra ocupada	1000	
Rampa positiva	100	1500
Barra libre	1000	
Presión del botón	1000	
Final	1000	

- Definición de **clases**. Este es un control de ruta de archivo donde se seleccionará el nombre del *archivo de clases*.

Para los estados que no son comunes a todas las tareas se diseñó un formato para especificar sus parámetros, es decir los parámetros para los estímulos y los retardos pre-, post- e inter-estímulo.

Se debe seguir un protocolo al momento de indicar los parámetros de los estímulos, dicho protocolo se lleva a cabo en una hoja de cálculo ilustrada en la Tabla 3, los 16 parámetros descritos corresponden a las columnas del archivo, se escribirán tantas filas como clases se quieran.

Tabla 6 *Protocolo para escribir los parámetros de los estímulos.*

Parámetro	Significado
CLASE	Número que se le asigna al conjunto de parámetros de los estímulos.
R1	Tiempo de retardo para el primer estímulo en milisegundos.
R2	Tiempo de retardo entre el primer estímulo y el segundo, en milisegundos.
R3	Tiempo de retardo entre el segundo estímulo y el fin del ensayo, en milisegundos
F1	Frecuencia de la señal del primer estímulo, en Hertz.
F2	Frecuencia de la señal del segundo estímulo, en Hertz.
A1	Amplitud de la señal del primer estímulo, en micras.
A2	Amplitud de la señal del segundo estímulo, en micras.
D1	Duración del primer estímulo, en milisegundos.
D2	Duración del segundo estímulo, en milisegundos.
TO	Tipo de onda para ambos estímulos. 1 para una onda seniodal, 2 para una cuadrada, 3 para una triangular y 4 para una onda de diente de sierra.
B	Será uno en caso de que la respuesta sea el botón central y cero en caso de que la respuesta sea el botón lateral.
CPT1	Número de casillas del patrón temporal del primer estímulo.
CPT2	Número de casillas del patrón temporal del segundo estímulo.
PT1	Es el patrón temporal del primer estímulo.
PT2	Es el patrón temporal del primer estímulo.

Este archivo será leído por la aplicación y según la tarea elegida se *leerán únicamente* los parámetros que se necesitan para desarrollar dicha tarea.

Por ejemplo, si se ha seleccionado la tarea de detección únicamente se leerán las columnas de: CLASE, R1, R2, F1, A1, D1, TO y B, pero si se ha seleccionado la tarea de discriminación, se leerán las columnas de: CLASE, R1, R2, R3, F1, F2, A1, A2, D1, D2, TO y B.

- Número de **repeticiones por clase**. Es un control numérico que indica cuantas veces se repetirá cada clase, lo que al mismo tiempo definirá cuantos ensayos totales se tendrán, pues como recordará el lector, el número de repeticiones por clase multiplicado por el numero de clases da el total de ensayos.

- **Secuencia**. En este caso se tendrá un menú desplegable en donde se elegirá secuencia aleatoria o secuencia ordenada. En el primer caso las clases se presentan en orden aleatorio, es decir, podría surgir la siguiente secuencia: clase 7, clase 2, clase 9, etc. En el caso de secuencia ordenada, se irá desde la clase 1 ascendentemente de 1 en 1 hasta la clase n (1,2,3,4,..,n).

- **Experimento**. Con un menú desplegable se elegirá ya sea un experimento pasivo o un experimento activo.

- **Luces**. Con una caja de verificación se elige si el experimento será visual (por lo que se prenderán las luces de los botones de respuesta) o si el estímulo será puramente táctil.

- **Ruido Blanco**. Con una caja de verificación se elige si el sujeto de experimentación escuchará o no el ruido blanco durante la presentación de los ensayos

Con los datos del tipo de tarea, el archivo de clases, la secuencia y el número de repeticiones por clase se prepara una *tabla* llamada 'todos los ensayos' con tantas filas como total de ensayos y tantas columnas como parámetros de los estados necesarios para desarrollar la tarea (sin contar los estados comunes a todas las tareas). En esta tabla existen bloques de clases, es decir que si el experimento tiene 6 clases se presentaran estas seis clases, ya sea en orden o de manera aleatoria, y a continuación se presentarán nuevamente las mismas 6 clases, otra vez de manera ordenada o aleatoria, dependiendo de lo elegido, y así hasta juntar tantos bloques de clases como número de repetición por clases.

Con la tabla 'todos los ensayos' creada y con la tabla leída del archivo 'todas_las_tareas' se tendrán todos los parámetros necesarios para ejecutar el experimento.

b) Controlar la ejecución del experimento (desde una segunda interfaz).

Una vez hechas las configuraciones, el sistema esta listo para comenzar la presentación de los ensayos, que iniciarán con una indicación del investigador, que se hace a través de un botón de tipo booleano llamado botón de INICIO. Una vez activado este botón, la leyenda cambiará a DETENER, que significa que si se desactiva el sistema se detendrá. Cuando esto sucede aparecerá automáticamente un mensaje que preguntará si deseas GUARDAR los resultados del experimento, si es así, entonces, se presentará un control de texto para que se teclee el nombre del archivo en que se guardará la información. Este archivo tiene formato de texto y su contenido se almacena en el siguiente orden: ensayo, clase, acierto, frecuencia(s), amplitud (es), duración (es).

Otro requerimiento dentro de los controles del experimento, será el de PAUSAR, que también es un botón booleano y al activarse cambia su leyenda a CONTINUAR, el sistema se detendrá, pero no aparecerán los controles para guardar la información, el sistema esperará, hasta la siguiente indicación del investigador, una vez desactivado, simplemente se continúa el experimento desde donde se había pausado.

Un control importante es sin duda el de REPETIR ERROR. Este control booleano al activarse cambia su leyenda a NO REPETIR ERROR y verifica cada vez que se comente un error para leer los mismos parámetros del ensayo recién efectuado y volverlos a mandar. Al desactivarse este botón, simplemente se sigue la secuencia del ensayo según la tabla.

El último control booleano es el de REPETIR ENSAYO. Una vez activado, cambia su leyenda a NO REPETIR ENSAYO, este control permite repetir el mismo ensayo, independientemente si fue acierto o error, hasta que el investigador decida ya no repetir el ensayo y continuar con la presentación del resto de los ensayos.

Una vez que se presenta un ensayo y el sujeto de experimentación ha acertado en su respuesta se le recompensa con unas gotas de agua. La cantidad de agua con la cual se le recompensa depende de cuanto tiempo se abre la válvula que suministra el agua, de modo que se puede mandar un pulso alto y lo único que se requiere saber es cuanto tiempo debe durar, dicho tiempo es ingresado por el investigador al sistema por medio de un control numérico llamado NIVEL DE RECOMPENSA presentado junto con los demás controles del experimento.

En este caso el nivel 1 corresponderá a un pulso de 0.5s y el nivel 8 corresponderá a 4s, es decir que por cada nivel de recompensa se abrirá la válvula un tiempo determinado por 0.5[s] multiplicado por el nivel.

Si el sujeto de experimentación es un ser humano, no se le da a beber el agua, sin embargo puede escuchar el sonido de la válvula al abrirse y saber que ha tenido un acierto.

En caso de que se cometa un error, el sistema presenta un retardo antes de iniciar el siguiente ensayo, el tiempo que se tardará el sistema en presentarle el ensayo correspondiente también lo determinará el investigador por medio de otro control numérico llamado NIVEL DE CASTIGO (duración).

Todos estos controles se leerán y se llevarán a cabo las acciones pertinentes en cada fin de ensayo y en ese momento se presentarán en esta interfaz gráfica los resultados de dicho ensayo.

Una vez iniciado el experimento, se llevarán a cabo todas las funciones pertinentes, cuyas entradas corresponden a la información proporcionada por el investigador en las configuraciones (tal como la presentación del ruido blanco y la creación de la tabla ‘todos los ensayos’)

Entonces se manda el primer ensayo, donde se ejecuta cada estado, pasando al estado correspondiente según el evento ocurrido. Al terminarse el primer ensayo, si se realizó por completo, se leen los controles y se presentan los

resultados. Si el ensayo se interrumpió por haberse presentado un evento incorrecto, se repite el ensayo en cuestión, en este momento también se leen los controles.

Se repite esta acción hasta que se termine la presentación de todos los ensayos. Al terminar el último ensayo no interrumpido se apaga el ruido blanco si es que se presentó, se muestra el mensaje para guardar los resultados.

Los resultados que se observan en esta interfaz de usuario son:

a) Una tabla llamada 'tabla de resultados', se podrán observar el número de ensayo que se realiza, la clase a la que pertenece dicho ensayo, si el sujeto ha cometido un acierto o un error y por último se observarán los parámetros de él o los estímulos presentados, justamente esta es la información que se guarda al finalizar el experimento (si así el investigador lo desea).

b) Una gráfica llamada 'tiempo de respuesta', se observará el rango de tiempo que el sujeto de experimentación tarda desde el momento en que retira la mano de la barra horizontal hasta el momento en que presiona uno de los dos botones (independientemente de que sea acierto o error) para cada una de las clases.

c) Una grafica de barras el desempeño por clase del sujeto de experimentación.

En este caso para cada clase se divide el número de aciertos que ha tenido el sujeto entre el número de veces que se le ha presentado la clase a evaluar:

$$\text{Desempeño del sujeto en la clase } x = \frac{\text{No. de aciertos de la clase } x}{\text{No. de presentaciones de la clase x}}$$

Por ejemplo el desempeño del sujeto en la clase 4 será de 0.333 si se le hubiera presentado 6 veces esta clase y solo hubiera tenido 2 aciertos.

3.4 Diagramas de flujo para el experimento y el ensayo.

En la Figura 15 se muestra el diagrama de flujo para el experimento según lo descrito anteriormente:

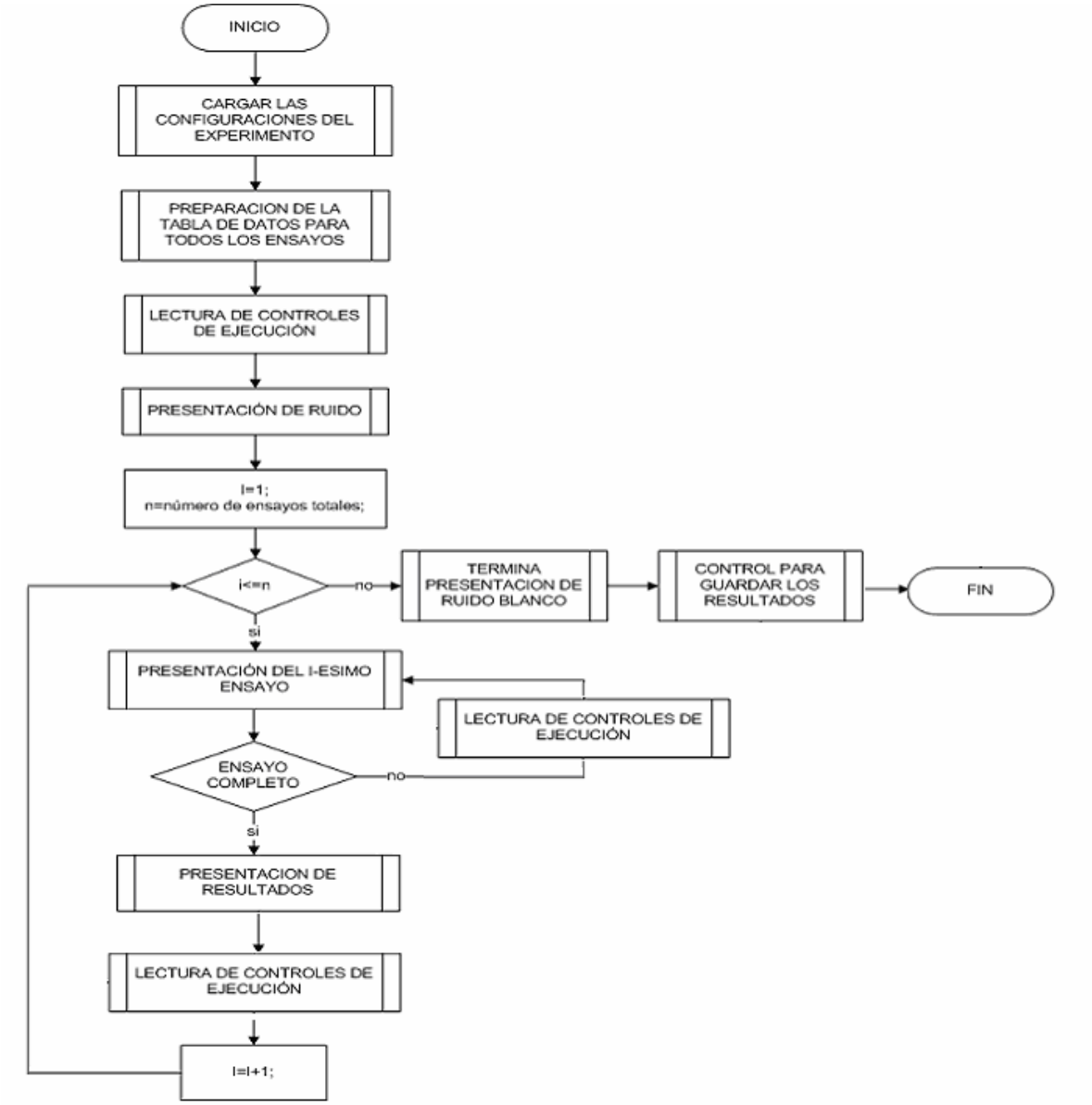


Figura 15 *Diagrama de flujo para un experimento.*

La presentación del i-esimo ensayo se muestra en Figura 16 en un diagrama de flujo, basado en el análisis previo y en el diagrama de estados mostrados en el capítulo 1:

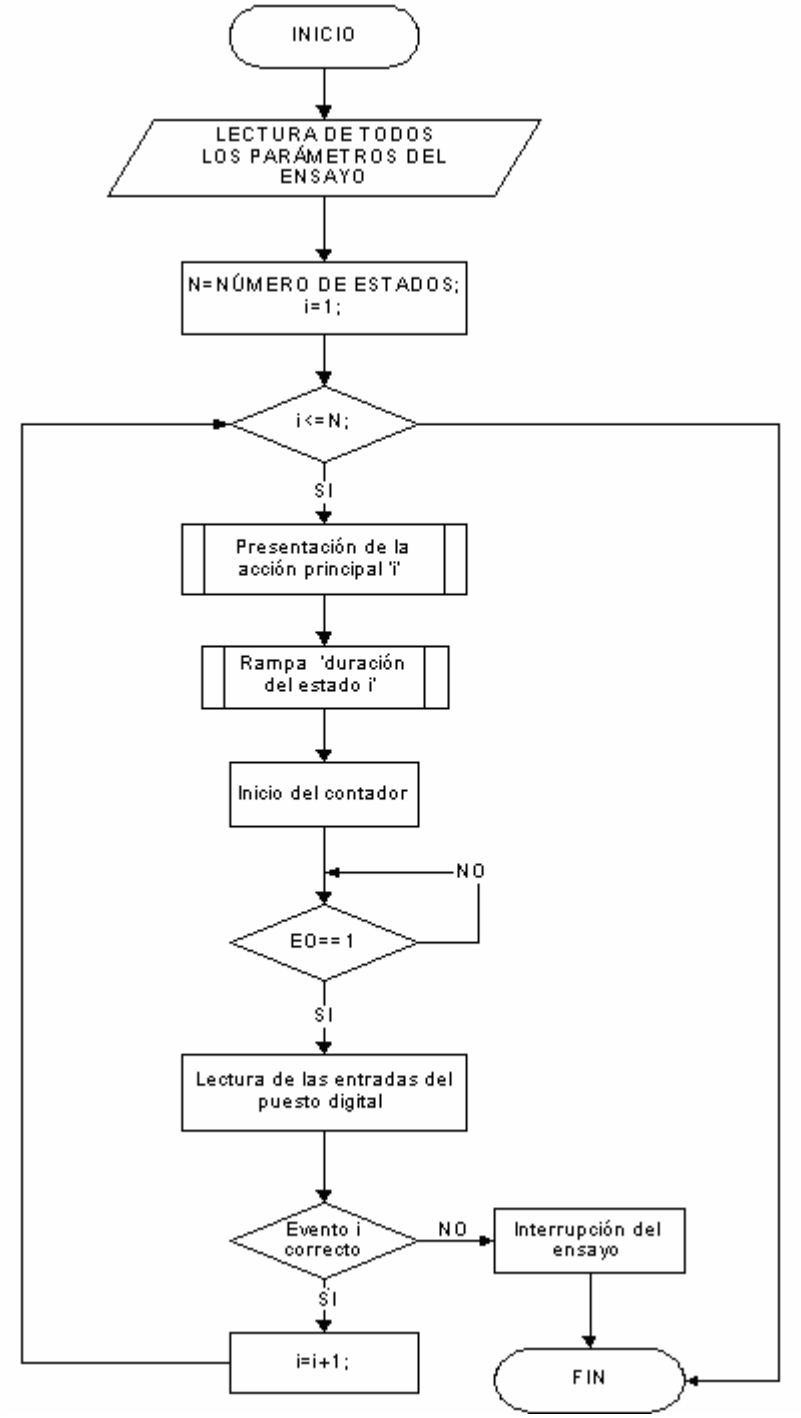
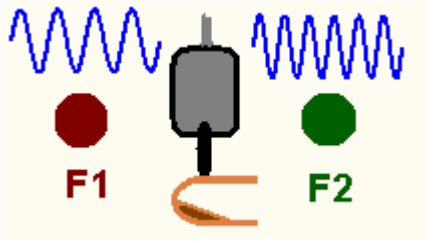


Figura 16 *Diagrama de flujo para un ensayo.*



4.Desarrollo de la aplicación en LabView.

La instrumentación virtual es una combinación de elementos de hardware y software usados sobre una computadora personal por sus capacidades de productividad, despliegue y conectividad, que tienen en conjunto la funcionalidad de un instrumento tradicional. La instrumentación virtual es usada en aplicaciones tales como: prueba, automatización, diseño y control pero principalmente para **adquirir una variedad de señales de diversas fuentes y generar señales de control para diversos dispositivos**.

El software usado para desarrollar este sistema fue LabView 7.0, con el API Traditional NI-DAQ y la tarjeta de adquisición de datos PCI 6071 E.

El sistema permite desarrollar un experimento de psicofísica. Cada experimento está conformado por cierto número de estados y cada estado por módulos. Los módulos que conforman un estado son los siguientes:

a) Módulo Acción Principal. Este módulo tiene como función generar las siguientes señales:

- Para los estados rampa: generar una rampa, en un caso con pendiente positiva y en otro caso con pendiente negativa. Los parámetros necesarios son la duración y la amplitud.

- Para los estados de estimulación existen dos tipos de señales de estimulación, una corresponde a estimulación continua y otra a estimulación llamada patrones temporales, en donde se presentan una serie de pulsos en determinado orden. La estimulación de patrones temporales tiene como parámetro el orden de presentación de ciclos senoidales individuales llamados pulsos. La estimulación continua tiene como parámetros la forma de onda, amplitud, duración y frecuencia.

- Para el estado inicio la función principal será prender la luz del botón correspondiente (en caso de que se quieran las luces). El parámetro requerido es cual luz deberá prenderse (la lateral o la central).

- Para el estado final, las funciones serán: proporcionar la recompensa si el sujeto de experimentación acertó, es decir, generar el pulso qe abre la válvula cuya duración es igual a el nivel de recompensa multiplicada por 0.5[s]. Si previamente se encendieron las luces, la otra función será apagarlas. Existen dos posibilidades: si el sujeto acertó, con la misma función de recompensa las luces se apagarán automáticamente, si no acertó existe una función que las apagará.

b) Módulo duración del estado. La función de este módulo es mandar una rampa de amplitud fija (10 V) con pendiente positiva. Su único parámetro será la duración. la cual corresponde a la duración máxima del estado, de modo que esta rampa es una entrada para un comparador. Cuando la señal alcance el máximo nivel (10 volts) el comparador generará un pulso TTL que será enviada a una entrada digital de la tarjeta de adquisición y su significado es Tiempo de Estado Terminado (TET).

c) Módulo registro del tiempo. La función de este módulo es iniciar un contador de la tarjeta de adquisición de datos, de tal manera que el valor del contador será almacenado en un registro cada vez que ocurra una señal de evento ocurrido (conectado a la terminal “gate”).

d) Módulo de validación de evento ocurrido. Su función es leer el puerto digital de la tarjeta de adquisición que corresponde al evento ocurrido y compararlo con el evento correcto. Si son iguales, el sistema continuará con el siguiente estado, si son diferentes, el sistema manda al estado error y se dice que el ensayo se ha interrumpido.

4.1 Programación y prueba de los módulos.

Para generar las señales necesarias para la estimulación se usa el VI ‘Basic Function Generator’ (Figura 17) incluido en las librerías de LabView, que admite entradas para la amplitud, la frecuencia, la forma de onda, la fase y la componente de directa.

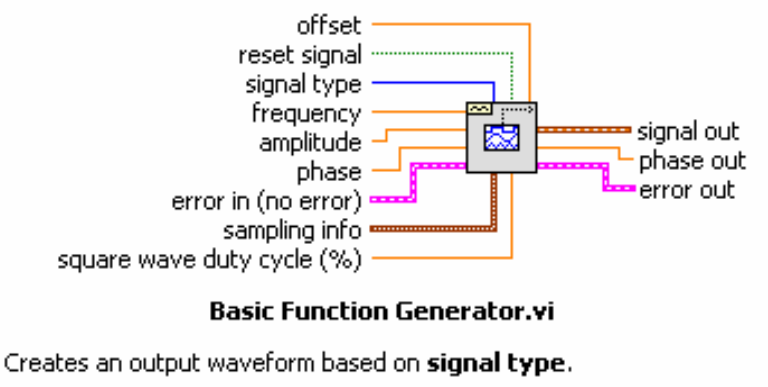


Figura 17 *Diagrama de E/S del VI generador de funciones básico.*

Los investigadores entienden por amplitud el desplazamiento de la punta estimuladora y lo manejan en micrómetros. El rango que puede desplazarse la punta estimuladora esta entre 0.1 y hasta 2000 micras. Sin embargo el valor que debe ingresarse a la función debe darse en volts y además debe ser voltaje pico (Vp), de modo que, sabiendo que la punta estimuladora se desplaza 1milímetro por cada 5 volts, la conversión será que por cada micrómetro que deba desplazarse la punta se deberán enviar 5 milivolts. Es así que se llama a la función 'Expression Node', que permite crear una expresión matemática de una sola variable, la expresión será la siguiente: (*x*)*(0.005), donde la '*x* ' es la amplitud ingresada por los investigadores en micrómetros, el factor 0.005, es la escala para que por cada micra que deba desplazarse se envíen 5 milivolts.

La función 'Basic Function Generator' tiene además dos entradas importantes: la frecuencia de generación de muestras y el número de muestras generadas, estas entradas están contenidas en el cluster 'sampling info'. Por definición, la frecuencia de generación es de 1000 muestras por segundo y el número de muestras es también 1000, lo que implica una señal de duración de 1 segundo. Para cambiar la duración, solo basta multiplicar el valor de la duración deseada por la frecuencia de generación y el resultado ingresarlo a el número de muestras.

Número de muestras**=** duración [s] **x** frecuencia [S/s]

Debido a que los investigadores especifican el valor de la duración en milisegundos, y la función admite estos valores en segundos se recurre a otra expresión matemática: x/1000, donde x es el valor de la duración pero en milisegundos.

La frecuencia base máxima de una señal para los estímulos es de 300 Hz, por lo que para construir la señal requerida debemos tomar en cuenta el teorema de Nyquist, lo que implica que la frecuencia de muestreo de salida de la señal deberá ser de por lo menos, en este caso, de 600 Hz. En el laboratorio se utiliza una frecuencia de muestreo de salida de 3000 Hz, de modo que cumplimos con el teorema de muestreo de Nyquist y obtenemos una señal de calidad.

Una estructura de casos permite elegir el tipo de onda a través de un control numérico, donde el número "1" corresponde a una señal senoidal, el "2" a una señal cuadrada, el "3" a una triangular y el "4" a una señal de diente de sierra.

Es posible enviar la señal con un componente de directa (CD); los investigadores ingresan este valor en micrómetros.

Para generar un estímulo continuo solo deberán ubicarse los controles para cada uno de los parámetros que se desea modificar, haciendo las consideraciones para la amplitud y la duración, e ingresarlos en el VI 'Basic Function Generation', los controles se encuentran en la paleta de controles del panel frontal y el VI se encuentra en la paleta de funciones del diagrama de bloques. En la Figura 18 se

muestra el panel frontal (izquierda) donde se ubican todos los controles para la generación de los estímulos y el diagrama de bloques de la programación (derecha).

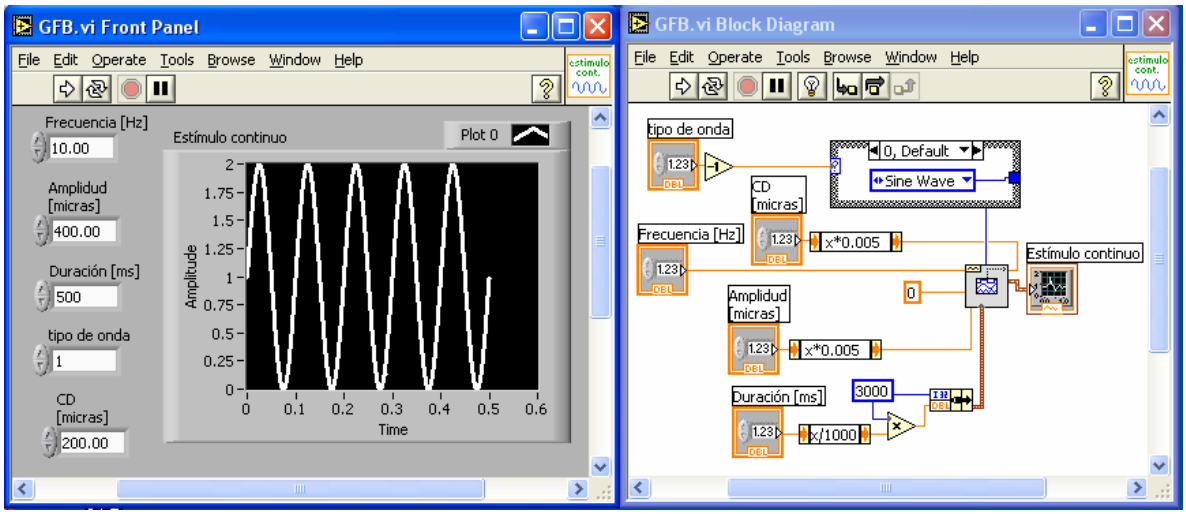


Figura 18 *Módulo principal que genera estímulos continuos.*

Para generar las señales rampa, se requirió utilizar 'Basic Function Generator', en este caso se usó la forma de onda triangular. Para lograr una rampa con pendiente positiva iniciamos la señal en 270 °, mientras que para una pendiente negativa la señal se comienza en 90°; tomando en cuenta que se usó una señal triangular, el parámetro de duración correspondía únicamente a la mitad del periodo, de modo que la frecuencia sería el inverso de dos veces la duración. Como en el caso de los estímulos, también se escribe la duración en milisegundos y la amplitud en micrómetros. En la Figura 19 se muestra el panel frontal (izquierda) donde se ubican todos los controles para la generación de la onda y el diagrama de bloques (derecha) de la programación de la rampa.

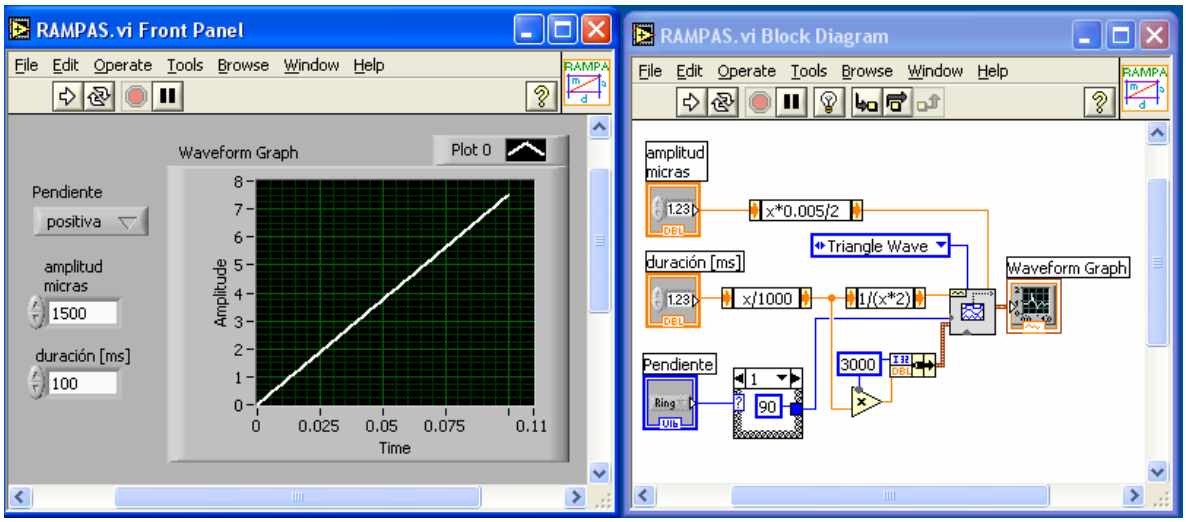


Figura 19 *Módulo principal que genera rampas.*

Los patrones temporales son un conjunto de ciclos senoidales, cada ciclo es llamado pulso; los cuales tendrán un orden de presentación según lo requiera el investigador y juntos forman una serie, esta serie tiene como característica que a su inicio y fin tienen un pulso. Para lograrlo, el investigador debe elaborar un arreglo, es decir un vector cuyos elementos sean ceros y unos, de tal manera que cada uno representa un pulso y cada cero representa un espacio de tiempo llamado tiempo muerto.

Para generar el pulso, conociendo que la frecuencia es de 50 Hz, y por lo tanto el periodo es de 20 ms, además que la frecuencia de muestreo es de 3000 Hz, el número de muestras a generar debe ser: 3000 x 0.02, es decir se deben generar 60 muestras. Debido a que la amplitud del pulso es de 100 micras, se debe generar una señal de 0.5 volts de pico a pico, o lo que sería lo mismo de 0.25 volts pico. Estos valores se ingresan como constantes al VI ‘sine waveform’, que es equivalente a usar el VI ‘basic function generation’ con una señal senoidal (Figura 20 lado izquierdo)

Para generar el tiempo muerto se genera una señal de 0 Hz, con 60 muestras, lo que dará un desplazamiento de DC de 0, si se deseara subir o bajar ese desplazamiento bastaría con agregar un control de amplitud ya que la señal generada es senoidal con fase de 90° (Figura 20 lado derecho).

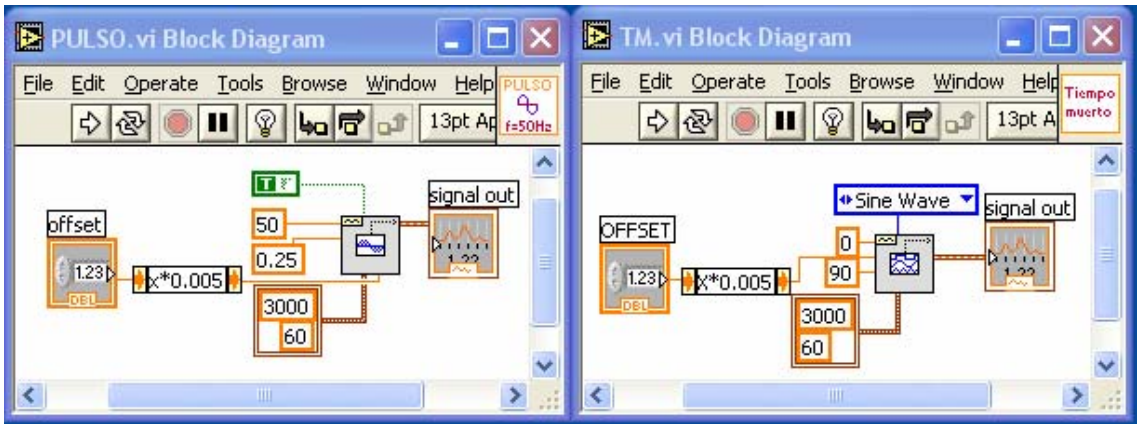


Figura 20 *Diagramas de bloques para la generación del pulso (izquierda) y para la generación del tiempo muerto (derecha) de los patrones temporales.*

Para facilitar la programación se generan SubVI's, es decir la equivalencia de funciones en otros lenguajes de programación, de modo que se editan los iconos y las conexiones para los valores de entrada y salida, posteriormente estos SubVI's son llamados en otros VI's.

Una vez generados los SubVI's podemos generar el patrón de pulsos, leyendo los valores de un arreglo, comparando si es uno, llamar al SubVI para el pulso, si es cero, llamando al del tiempo muerto, y concatenando dentro de un ciclo "for", hasta obtener todo el Patrón.

En la Figura 21se muestran los diagramas de bloques para el patrón de pulsos. A la izquierda se llama al sub VI encargado de generar los tiempos muertos, a la derecha se llama al sub VI encargado de generar los pulsos.

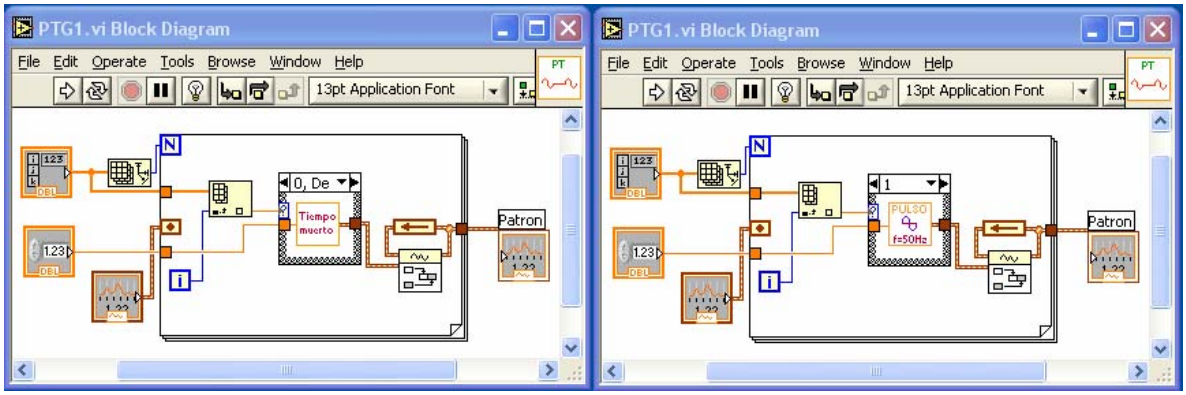


Figura 21 *Diagrama de bloques del VI para generación de patrones temporales.*

En la Figura 22 se muestra el Panel Frontal de la generación de patrones temporales. El primer elemento del arreglo es el de la parte superior, se observa que contiene diez elementos, considerando que cada dura 0.02 s, la duración de este patrón será de 0.2 s, es decir 200ms

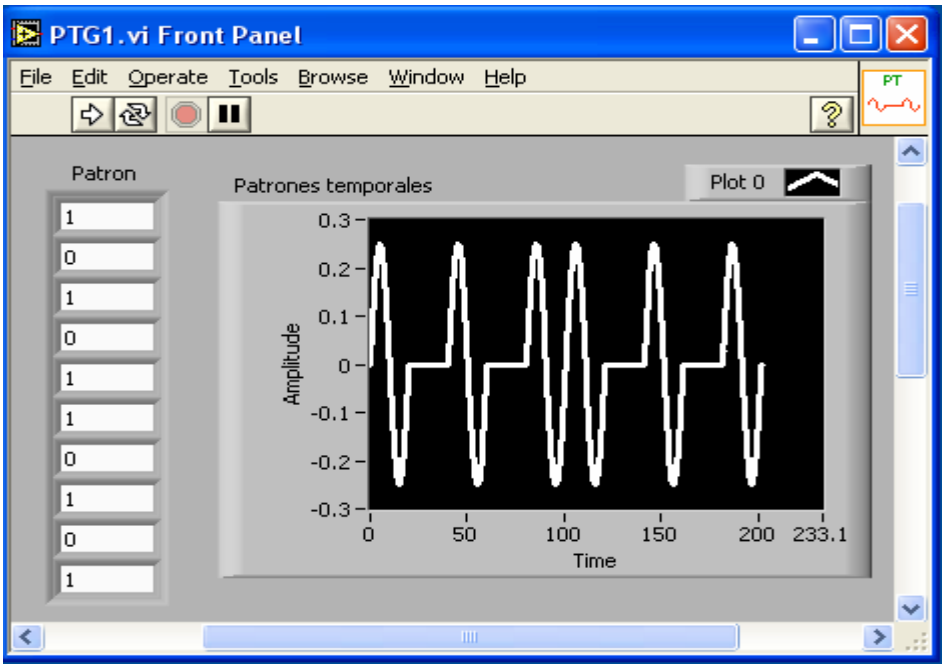


Figura 22 *Panel Frontal para generación de patrones temporales*

Tanto las luces como la válvula para la recompensa requieren estados lógicos. Las luces permanecerán encendidas mientras el pulso que las controle esté en alto y apagadas cuando esté en bajo. El mismo caso se tiene para la válvula de la recompensa, mientras el pulso que lo controla esté en alto la válvula permanecerá abierta y cuando esté en bajo, la válvula permanecerá cerrada.

Las luces (la central y la lateral) y la recompensa son controladas a través del puerto paralelo de la PC. se utilizan los bits menos significativos para su control (Figura 23).

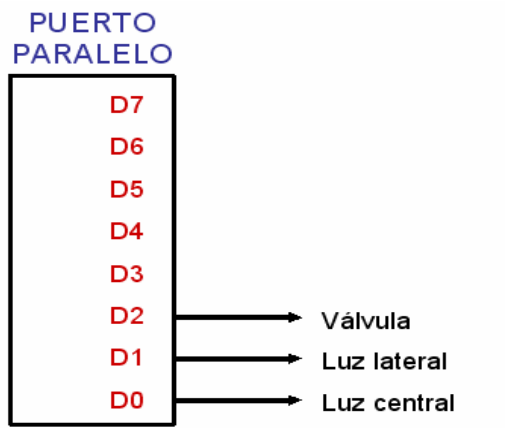


Figura 23 *Esquema de los bits del puerto paralelo de la PC usados para el control de las luces y la recompensa.*

Para el estado en que se debe encender alguna luz, solo deberá especificarse con un '0' si la luz a encender será la central, y con un '1' si se desea encender la lateral, el programa mandará un "1" binario al puerto paralelo para encender la luz central y un "2" binario para encender la luz lateral (ver Figura 24).

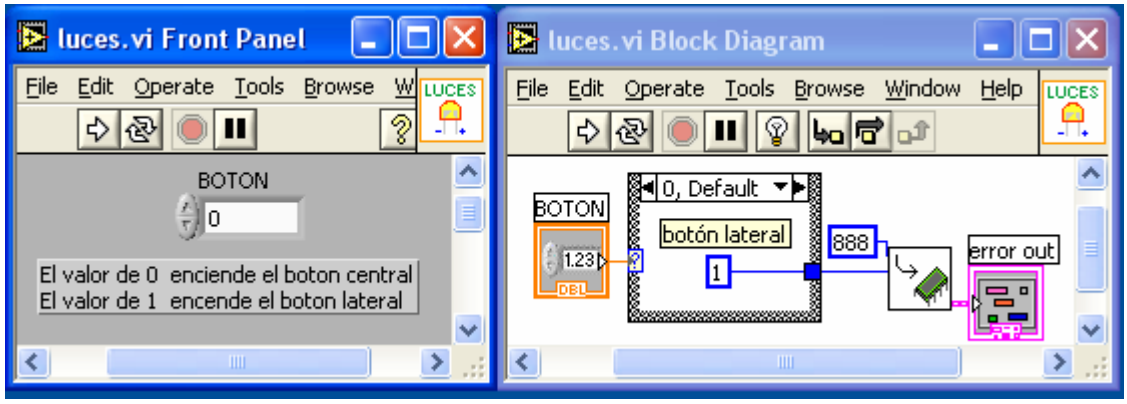


Figura 24 *Panel Frontal y diagrama de bloques para controlar el encendido de las luces.*

El apagado de las luces y la recompensa son funciones del estado final, en caso de que el sujeto de experimentación haya acertado en la respuesta, se mandará un valor de "4" binario al puerto paralelo, lo que automáticamente apaga las luces. En caso de que el sujeto de experimentación no acierte, se manda llamar una función que apague las luces mandando un "0" binario al puerto paralelo (Figura 25).

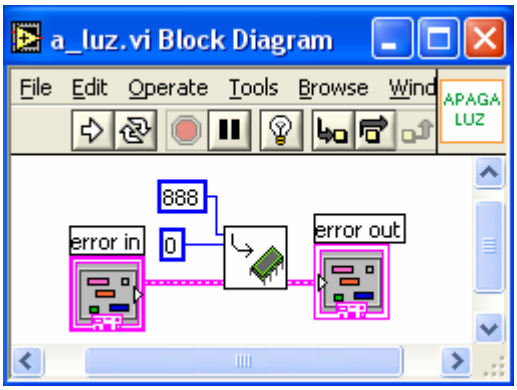


Figura 25 *Diagrama de bloques del VI que apaga las luces.*

Es necesario que la válvula se abra únicamente un tiempo determinado por el investigador, de modo que la función que controla la recompensa manda el valor de “4” y llama una función de retardo variable, una vez trascurrido el tiempo manda el valor de “0” (Figura 26), de modo que todas las salidas están en bajo.



Figura 26 *Panel frontal y diagrama de bloques para la recompensa.*

Para el módulo de duración del estado se manda una rampa, siempre con pendiente positiva, y amplitud fija a 10 [V] por un canal analógico de salida. El único parámetro de entrada es la duración en milisegundos (Figura 27).

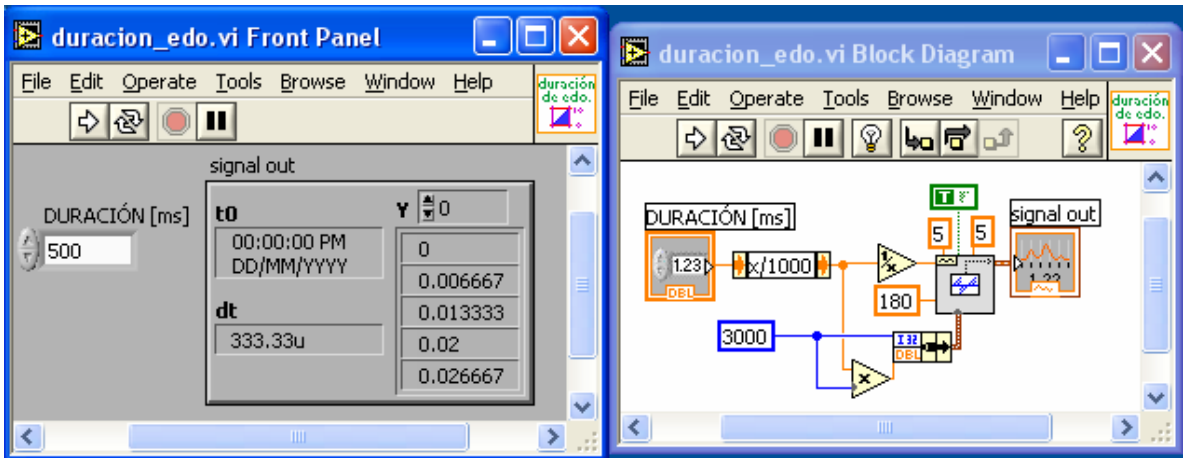


Figura 27 *Panel frontal y diagrama de bloques para el módulo de duración del estado.*

Para el módulo de validación del evento ocurrido, se lee completamente el puerto digital con el VI "Read from digital port" quedando el bit menos significativo a la derecha y el mas significativo a la izquierda. El bit menos significativo 'b0' del puerto indica si el tiempo del estado ya ha transcurrido, el siguiente bit "b1" indica si se ha presionado el botón lateral, el b2 indica si se ha presionado el botón central, el b3 y b4 indican si la palanca esta ocupada o si esta libre respectivamente. El evento ocurrido se compara con el evento correcto para determinar el estado siguiente (Figura 28).

El evento correcto para cada estado se muestra en la Tabla 7.

Tabla 7 *Eventos correctos para cada estado.*

ESTADO	EVENTO CORRECTO
Barra libre	11110000
Barra ocupada	11101000
Presión del botón	11100100 (botón central) 11100010 (botón lateral)
Los estados restantes	11100001

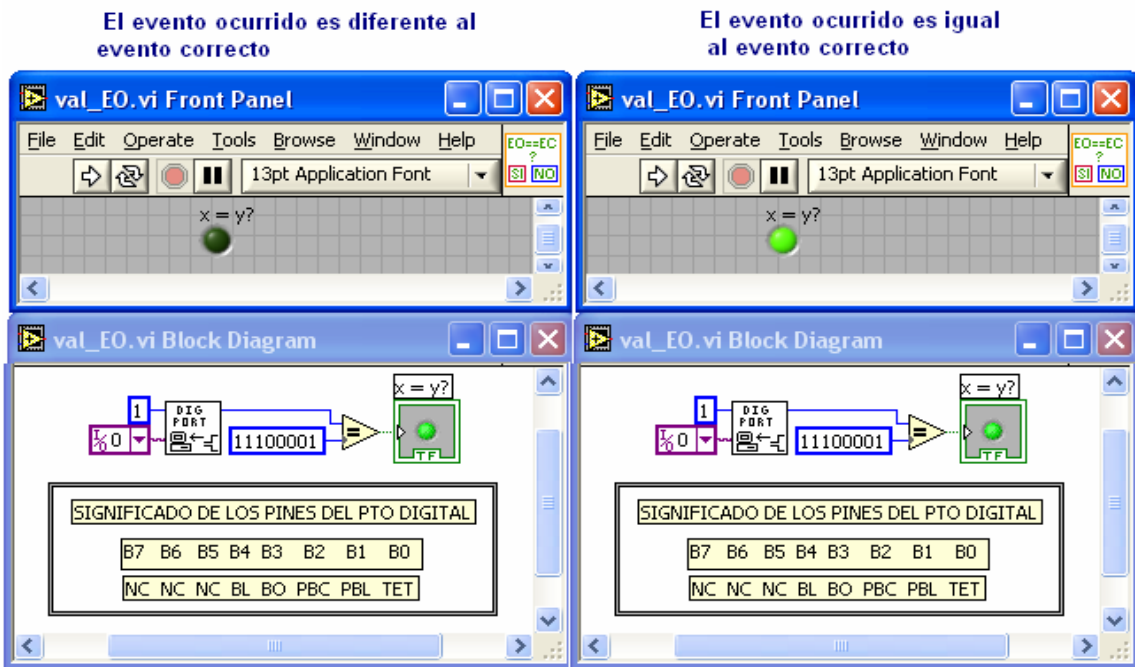


Figura 28 *Diagrama del VI para la evaluación del evento ocurrido.*

Para el módulo “registro del tiempo” el contador se configuró en modo de “Bufer period Measurement” y el buffer en modo “single”. Una vez activado el VI, el contador se inicia incrementando su cuenta en cada cambio de estado en el source (en este caso corresponde al reloj interno de la tarjeta configurado a 100 kHz) y se detendrá en el momento en que cambie de estado el gate (correspondiente a la ocurrencia de un evento). Al detenerse el contador también el VI se detiene y arroja el valor del registro, el cual se convierte a milisegundos (Figura 29).

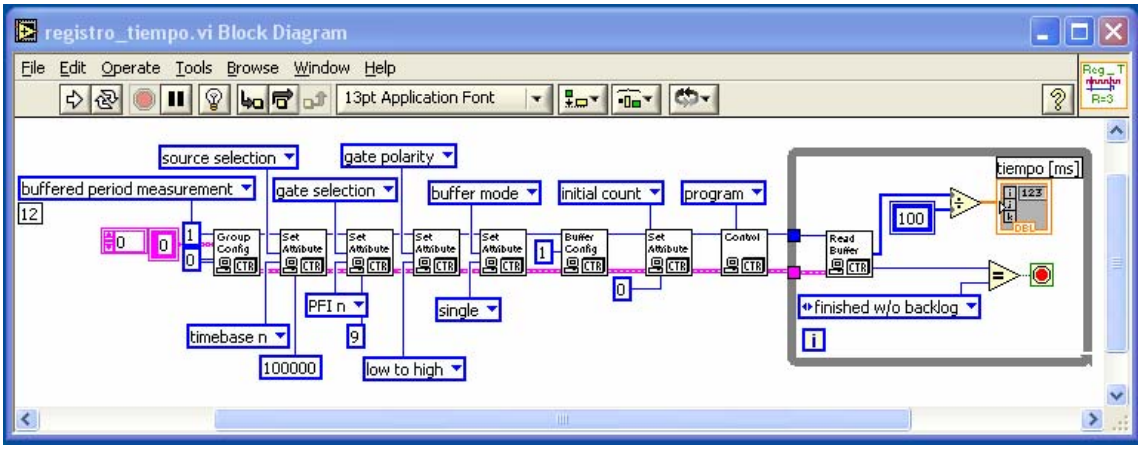


Figura 29 *Diagrama de bloques para la programación del contador.*

4.2 Integración de los módulos para formar el sistema.

Una vez programados los VI necesarios para los módulos, éstos se conjuntan para formar cada estado. Será necesario especificar el número de ensayos, y la clase que se presentará en cada ensayo, así como el número de estados, el tipo de estados y la secuencia de los mismos dependiendo de la tarea.

Así se podrá construir una tabla con todos los parámetros de los estímulos y de los tiempos inter-estímulos, de acuerdo a la tarea especificada. También se configurará el número de estados y la secuencia de los mismos en un vector, para lo que será necesario llamar tanto al archivo que contenga el valor del evento esperado en cada estado como al archivo que define los parámetros para el desplazamiento inicial y final de la punta estimuladora, así como los parámetros del estado final y el estado inicial.

El objetivo es conjuntar toda la información necesaria y organizarla para el desarrollo del experimento.

La primer interfaz está diseñada para generar las tablas con los parámetros requeridos para la realización del experimento (Figura 30).

El control del tipo de tarea sirve para saber qué columnas se leerán del archivo de clases, el archivo de clases es seleccionado por medio del control llamado definición de clases, esta nueva tabla corresponderá a un bloque de ensayos, pero este bloque deberá enviarse tantas veces como el controlador de numero de repeticiones lo indique, por último la presentación de los ensayos de cada bloque se pueden mandar o no aleatoriamente, según lo indique el controlador de secuencia. Con la información del control del tipo de tarea también se generará el vector que indica los estados y la secuencia.

Así por ejemplo si se eligiera la tarea de detección, en archivo de clases seleccionado se leerán las columnas correspondientes a la clase, el primer y segundo retardo, la frecuencia, la amplitud y la duración del estimulo, el tipo de onda y la respuesta esperada.

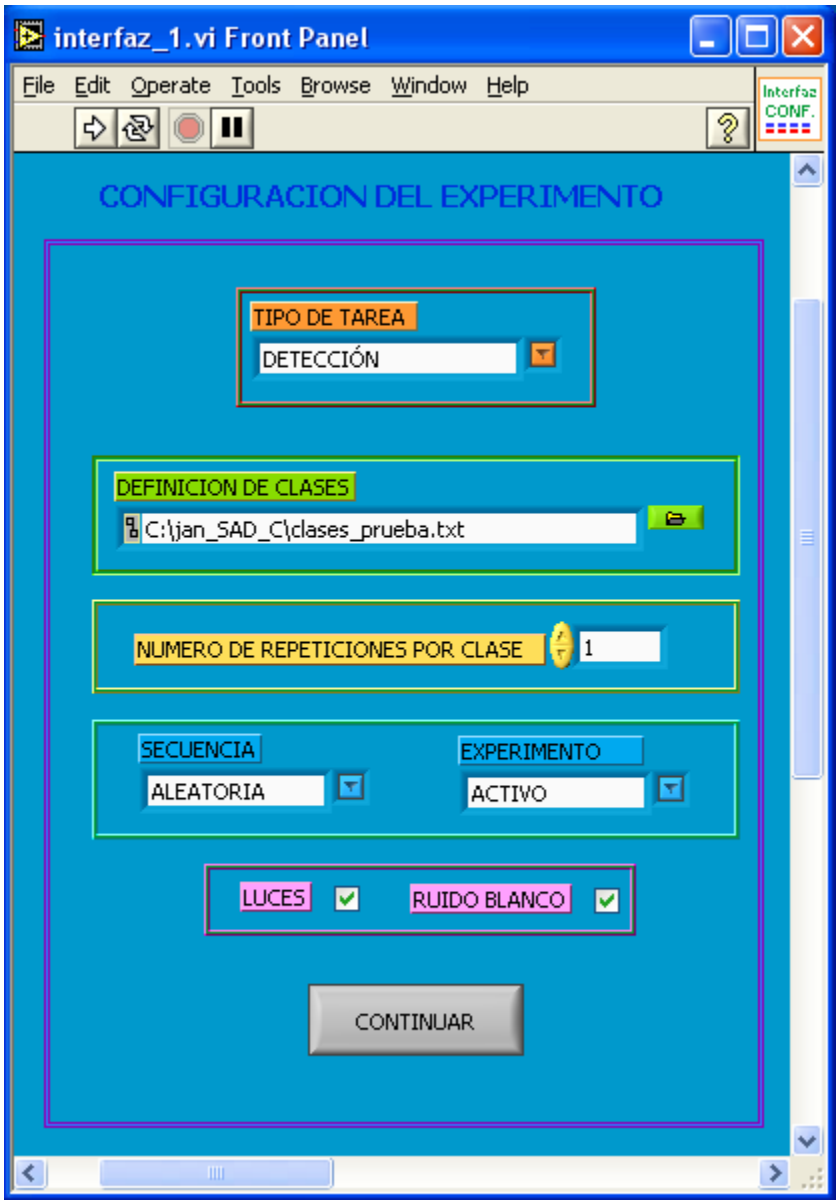


Figura 30 *Panel Frontal de la interfaz para configurar el experimento.*

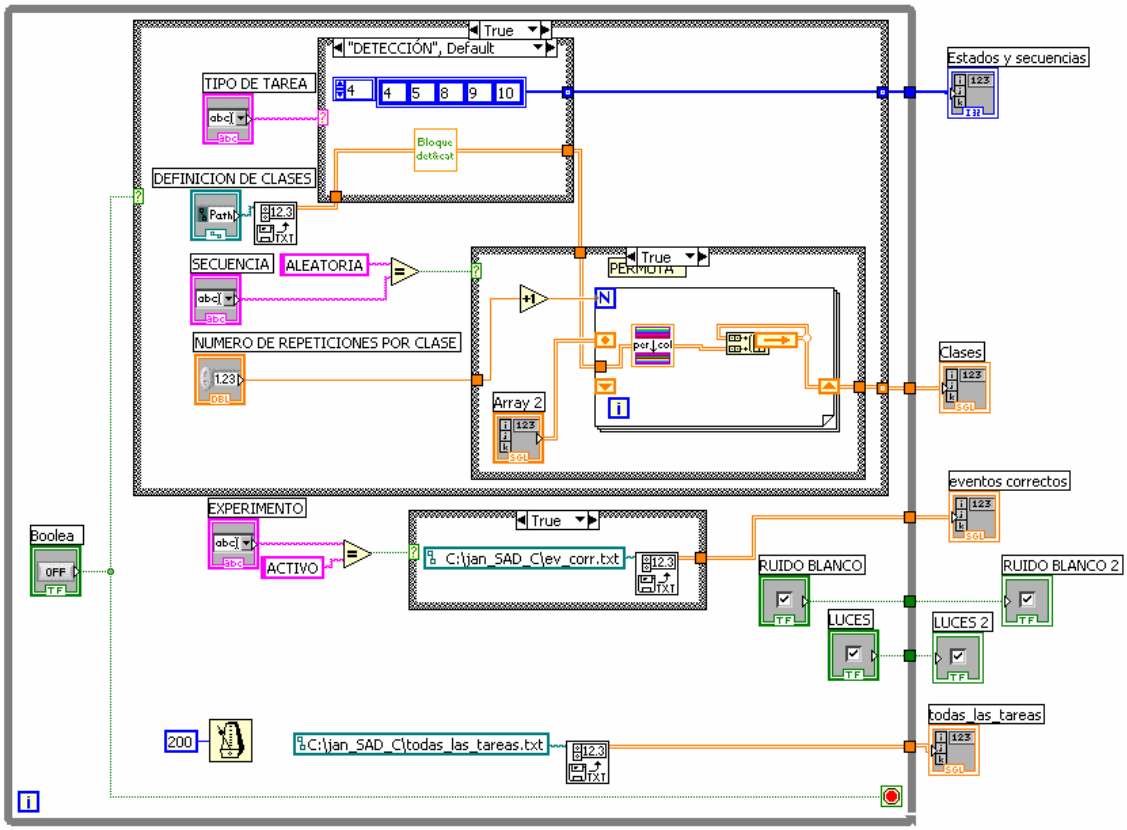


Figura 31 *Diagrama de bloques de la Interfaz para configurar del experimento.*

A partir de estos datos se generarán las tablas completas de los parámetros de cada ensayo de todo el experimento y los indicadores de las luces y el ruido blanco (ver Figura 32).

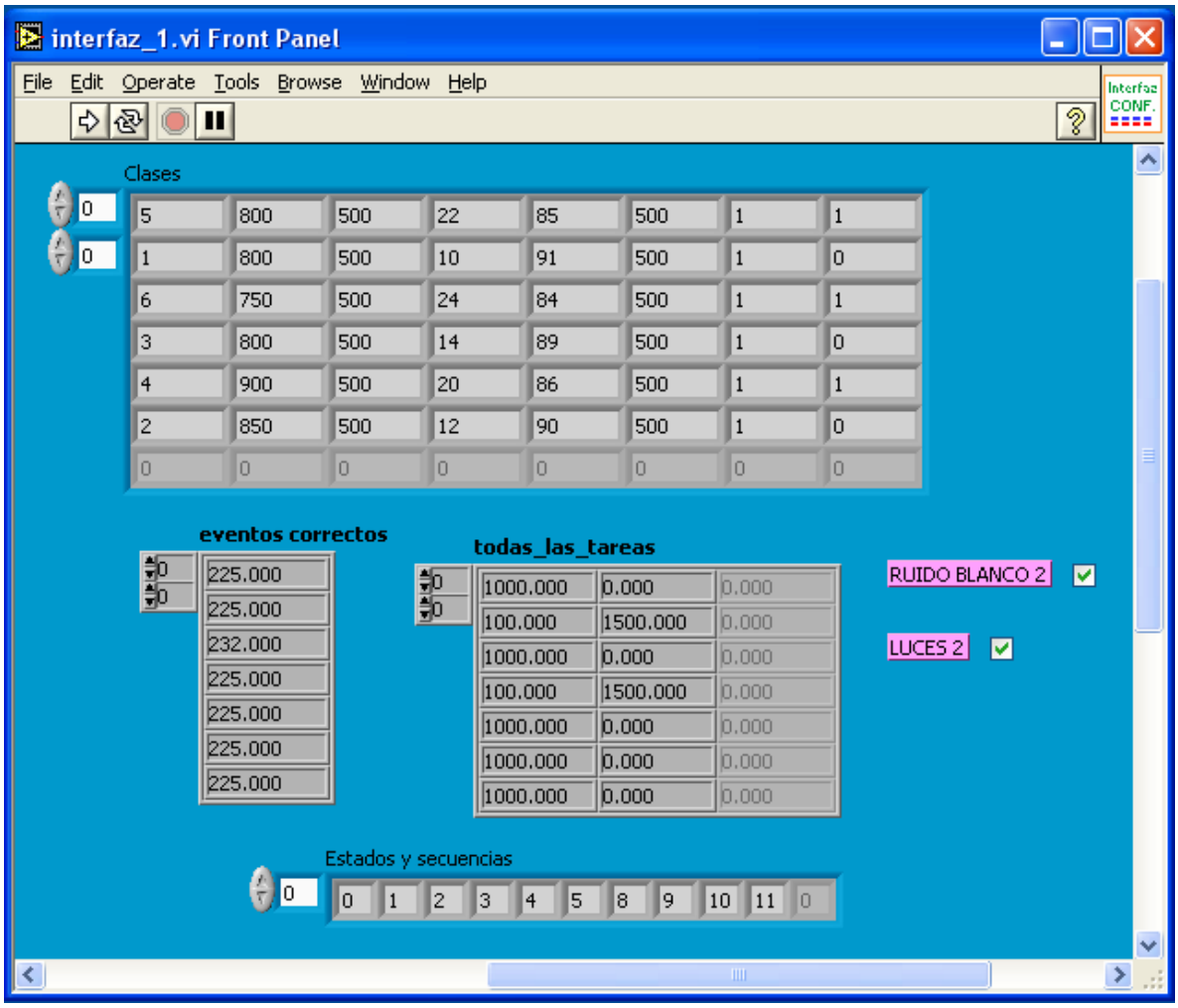


Figura 32 *Tabla generada por el VI interfaz_1 con los datos de la figura 36.*

Con la información completa y organizada es momento de iniciar el ensayo, para lo cual esta información se manda a un sub VI que será el encargado de seguir la secuencia de los estados. Este sub VI inicia con el valor del estado actual y define la acción principal al mismo tiempo que define la duración del estado, es decir manda llamar las funciones del módulo principal y las funciones del módulo de duración de estado, este sub VI también inicia el contador. Una vez que se produce cualquier evento este sub VI está programado para evaluar la respuesta y con ella indicar el estado siguiente.

Las tareas de detección y categorización constan de 10 estados (inicial (0), rampa negativa (1), barra ocupada (2), primer retardo (3), primer estímulo (4), segundo retardo (5) , rampa positiva (8), barra libre (9), presion del botón (10) y estado final (11).

Las tareas de discriminación y patrones temporales tienen a parte de los estados anteriores, el estado de segundo estímulo (6) y tercer retardo (7).

Para el estado Inicio la acción principal es prender las luces en caso de que los investigadores así lo deseen, captando la información del botón de respuesta correcta, el sub VI “Luces” prenderá el botón correcto; al mismo tiempo que se ejecuta la acción principal se ejecutan los módulos de duración de estado y de registro de tiempo, de modo que se envía una rampa por una salida analógica de la tarjeta[1], a un comparador y cuando la salida de este comparador tenga un pulso en alto indica que el tiempo máximo del estado ha transcurrido, éste o cualquiera de los otros posibles eventos detendrán el contador iniciado en el subVI “Registro de tiempo”, una vez que se ha registrado el valor, inmediatamente el sub VI “evento correcto” evalúa el evento ocurrido, si es correcto arroja un valor lógico verdadero y permite la secuencia de los estados, si el valor es correcto, la bandera llamada “completo?” se activará y el sistema entenderá que el estado se concluyó satisfactoriamente. El indicador de estado permite ver que estado se ejecuta actualmente, durante un ensayo. (Figura 33)

[1] Para mandar cualquier información por los canales analógicos de salida de la tarjeta es necesario configurar el canal y la frecuencia. Una vez generados los datos que se desean mandar, éstos se escriben en el buffer de salida, para posteriormente iniciar la salida de los datos, una vez enviados los datos, se limpia y cierra el canal analógico de salida.

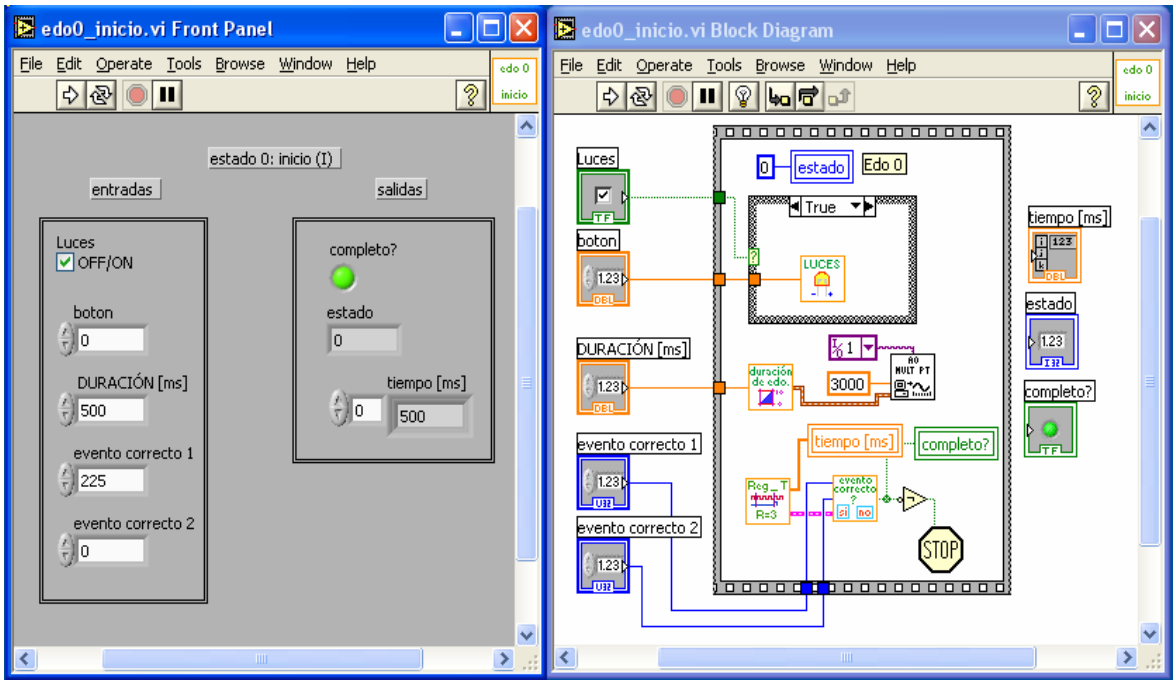


Figura 33 *Diagrama de bloques y panel frontal del estado inicial.*

Si el estado es incompleto (es decir la bandera “completo?” Tiene un estado lógico bajo) se ejecuta el estado error, donde se apagará la luz en caso de estar encendida e inicialmente pregunta cual estado fue el último que se ejecutó, para saber si tendrá que subir o no la punta estimuladora enviando una rampa de la amplitud y duración correspondiente a la del estado de rampa positiva (Figura 34).

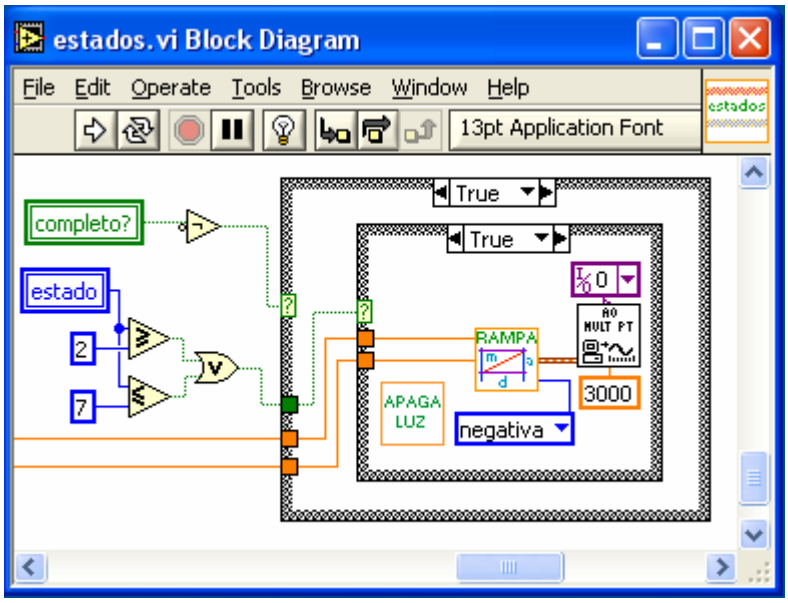


Figura 34 *Diagrama de bloques del estado de error.*

El estado 1 es el llamado estado rampa negativa cuya finalidad es desplazar la punta estimuladora para que haga contacto con el dedo del sujeto de experimentación (bajar la punta), como en todos los estados, también se manda una señal para la duración del estado, se registra el tiempo y se evalúa la respuesta ocurrida (Figura 35). Este estado el similar al estado 8 llamado rampa positiva cuya acción principal es desplazar la punta a la posición inicial donde no hace contacto con el dedo del sujeto (subir la punta). En estos estados los investigadores asocian la pendiente de la rampa con la dirección del desplazamiento, por ejemplo si la punta baja se asocia con una rampa de pendiente negativa y si la punta sube, lo asocian con una pendiente positiva, sin embargo, en realidad se manda para el caso en que la punta deba bajar una rampa que va desde cero volts a diez, lo que significa que es una señal de rampa positiva y viceversa.

En este estado es podrá observar, a diferencia del estado inicial, que se ocupan los dos canales analógicos de salida, usando una función de las librerías de LabView llamada AO Generate Waveforms.VI para que la generación sea simultanea.

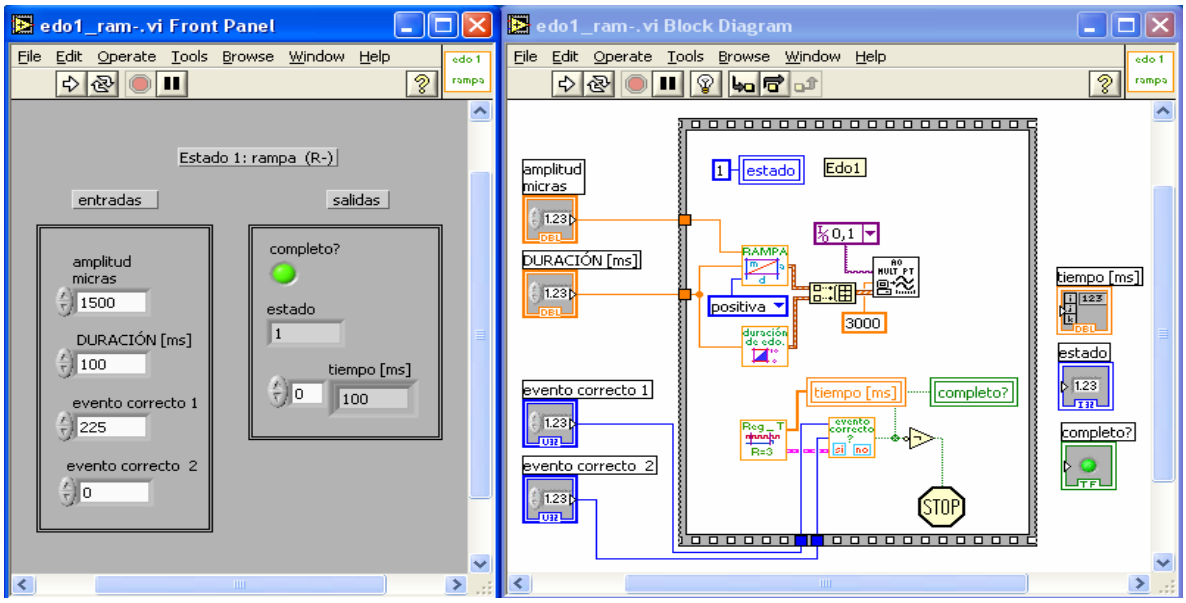


Figura 35 *Panel frontal y diagrama de bloques para el estado de rampa negativa.*

El estado 2 es el llamado estado de barra ocupada, donde se espera que el sujeto de experimentación ponga su mano sobre la barra, no existe una acción principal, es decir no hay señal analógica de salida, al igual que el resto de los estados se tiene una duración de estado, se registra el tiempo y se verifica el evento ocurrido. Este estado es similar a los estados de retardos y al estado de barra libre (3,5,7 y 9 respectivamente), lo único que cambia es el evento esperado (Figura 36).

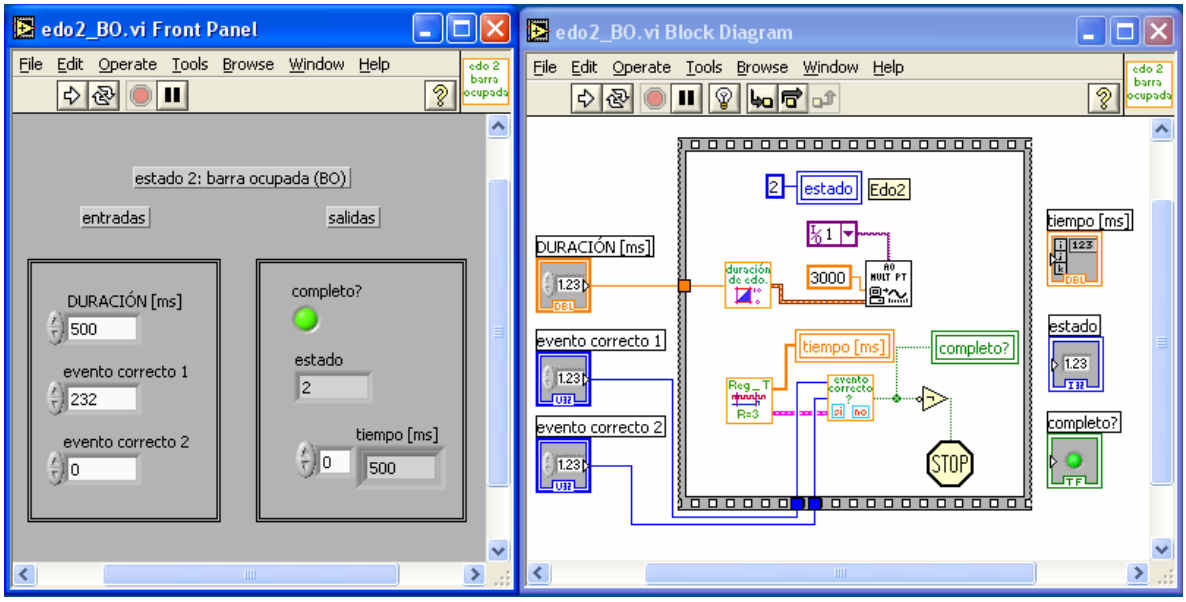


Figura 36 *Panel frontal y diagrama de bloques para el estado de barra ocupada.*

El estado 4 (Figura 37) es el llamado estado de estimulación, en el cual la acción principal es la de mandar un estímulo, ya sea uno continuo o un patrón temporal, según sea el caso, este estado es similar al estado 6 en el caso de las tareas de discriminación y de patrones temporales, lo único que varia son los parámetros para el estímulo.

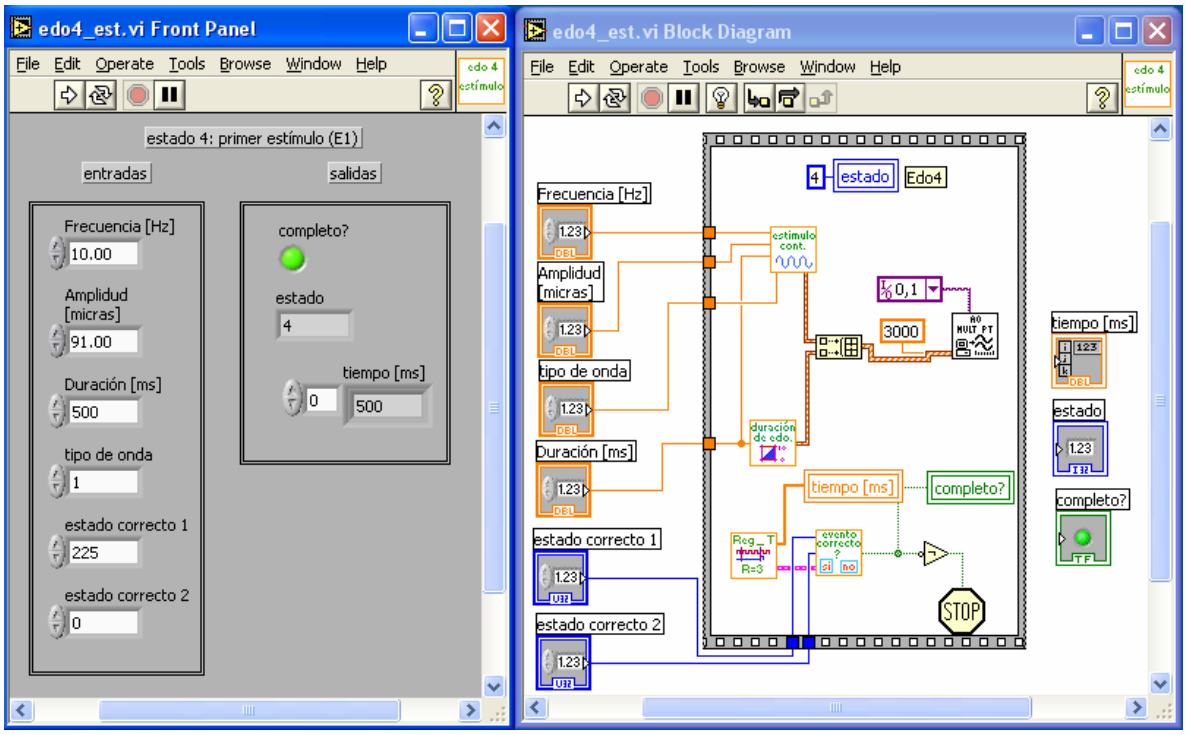


Figura 37 *Panel Frontal y diagrama de bloques para el estado de estimulación.*

El estado 10 (Figura 38) es el estado llamado de presión del botón, donde se espera que el sujeto de experimentación presione cualquier botón, cuando lo hace, el sistema manda el valor del registro del contador a un indicador, dicho valor se utilizará para las gráficas de desempeño en tiempo, también se obtiene el valor del botón presionado, de tal forma que se transfiere al estado final donde se evalúa para proporcionar o no la recompensa.

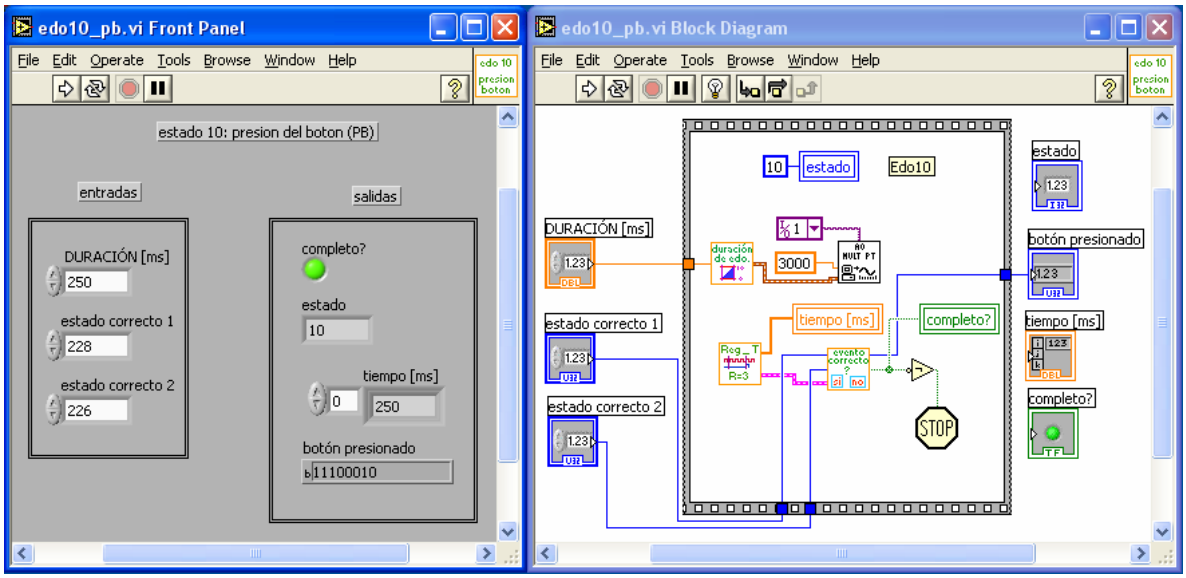


Figura 38 *Panel frontal y diagrama de bloques para el estado presión del botón.*

El estado 11 (Figura 39) llamado estado final, el valor que se transfiere es el de acierto, si es falso, implica que el sujeto de experimentación erró la respuesta, si es verdadero indica que el sujeto acertó. En caso de acertar se apaga la luz al mismo tiempo que se proporciona la recompensa, en caso de errar únicamente se apaga la luz.

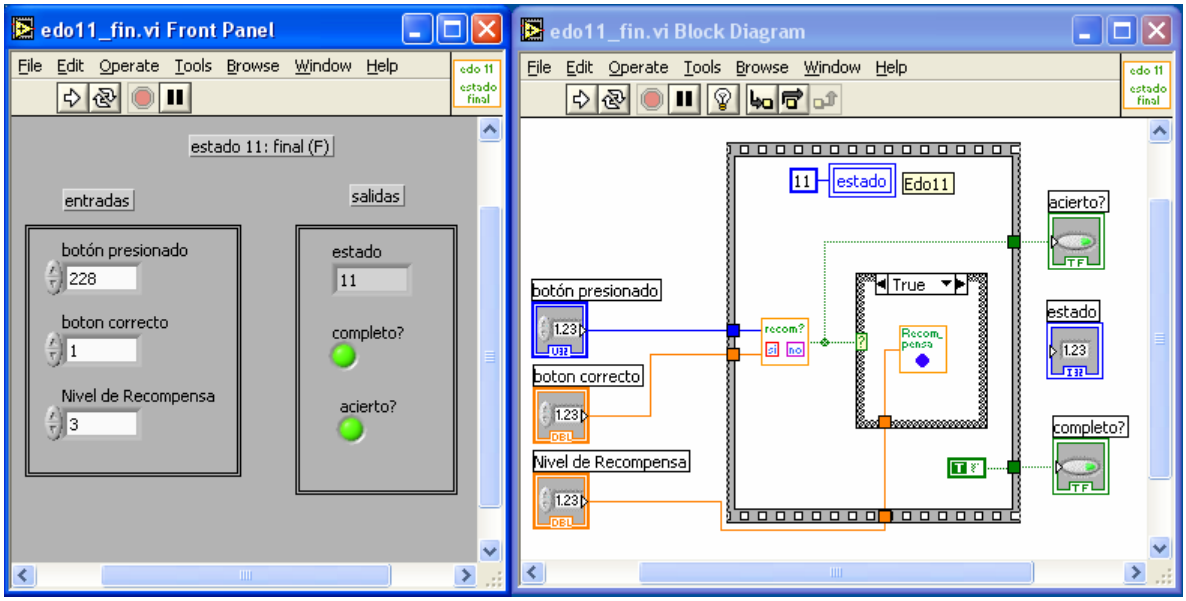


Figura 39 *Panel frontal y diagrama de bloques para el estado final.*

Al unir cada uno de los estados formamos un ensayo entero, al finalizar cada ensayo se puede obtener en un vector el valor de duración de cada estado, que sirve para asociar la respuesta neuronal (en caso de que se registre), con

cada uno de los estados. Además podemos saber si el sujeto de experimentación acertó y el tiempo de respuesta, es decir el tiempo que tarda desde que levanta la mano de la barra hasta que presiona el botón de respuesta.

Estos valores sirven para actualizar los gráficos y la tabla de resultados. En una segunda interfaz (ver Figura 40) el usuario es capaz de iniciar, pausar y detener el experimento, así como repetir ensayos. A la vez es posible observar graficas de respuesta en tiempo y de desempeño en porcentaje, así como observar tablas de resultados.

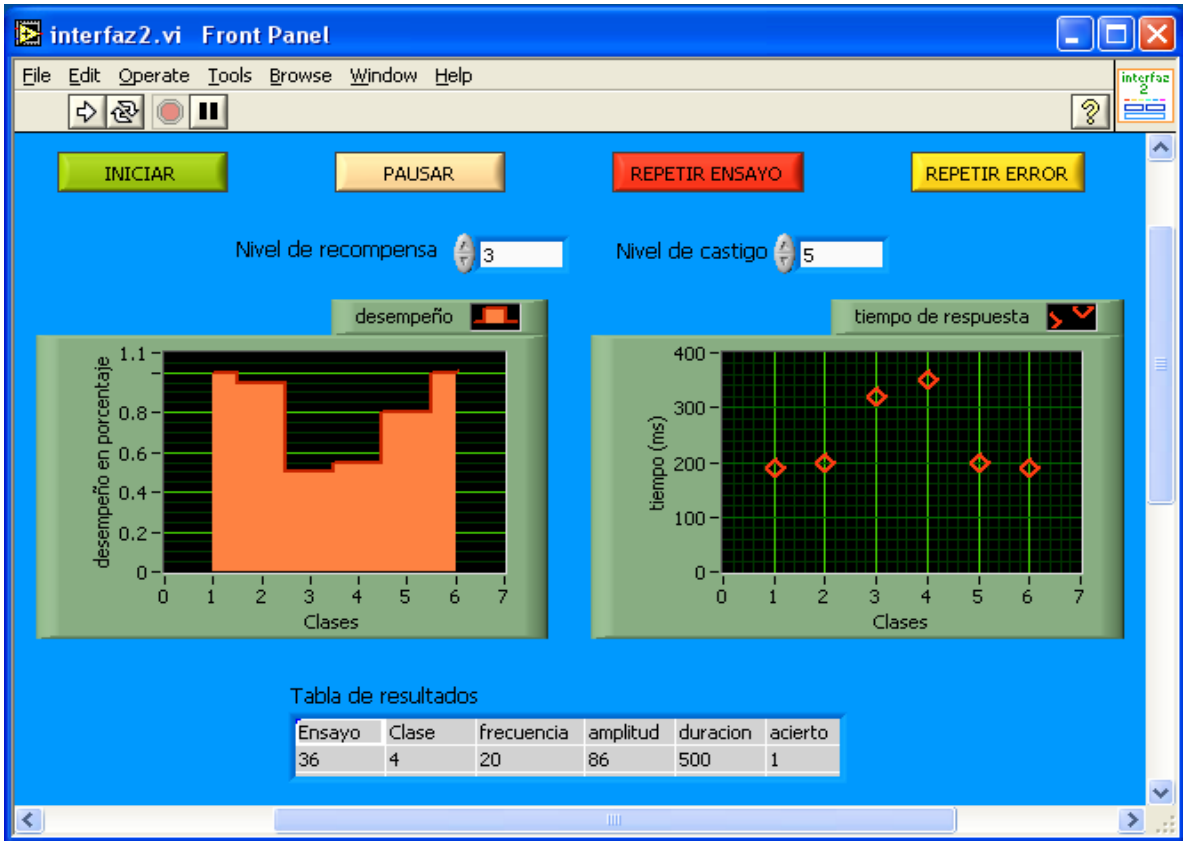
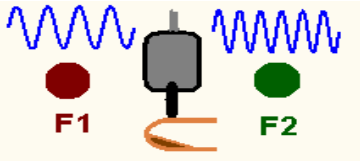


Figura 40 *Interfaz de usuario para el control del experimento y presentación de resultados.*

Con estas dos interfaces es posible configurar y controlar fácilmente experimentos de psicofísica, para el correcto entrenamiento de los sujetos de experimentación, la programación de módulos individuales que se integran en una etapa final permite integrar otros estados y de otro tipo al sistema sin necesidad de grandes modificaciones de fondo.



5. Resultados

Se programaron por completo las funciones de los módulos: las acciones principales (apagar luces, prender luces, generar rampas, estímulos continuos y patrones temporales), la duración del estado, el registro del tiempo y la evaluación de los eventos ocurridos.

Se integraron las funciones para desarrollar cada estado y a partir de ello desarrollar un ensayo. Se programó la interfaz que configura los experimentos y genera las tablas con toda la información de los parámetros requeridos para todas las funciones usadas en cada caso; por último se programó la interfaz que controla el experimento permitiendo iniciarlo, detenerlo, pausarlo, continuarlo, repetir error y repetir ensayo, así como observar las respuestas del sujeto de experimentación, es decir el tiempo de respuesta y el desempeño del sujeto.

Cada función implementada era probada por separado y conforme se integraba con otras funciones eran probadas en conjunto. Cuando se integraron las funciones y módulos para formar las dos interfaces con las que trabajarán los investigadores, se hicieron varias pruebas con diferentes datos de entrada y pruebas para todos los controles y así observar que el sistema respondiera correctamente, antes de ser evaluados por los investigadores.

Las interfaces electrónicas, tales como la barra, los botones y la válvula para la recompensa, necesarias para desarrollar las pruebas, fueron heredadas del sistema anterior.

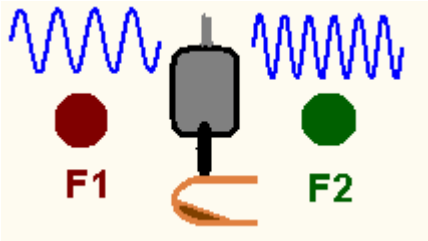
Ejemplos de la captura de datos a través de las interfaces correspondientes se observan en la Figura 30 y 40.

Las formas de onda generadas por el sistema se observan en la figura 18, 19 y 22.

En las pruebas elaboradas para las señales analógicas de salida se observa un retardo de 2 milisegundos, en el tiempo de reacción del sistema, sin embargo éste no interfiere con el desempeño en tiempo real del sistema.

No se observaron deformaciones en las pruebas realizadas sobre la generación de las señales ocupadas durante los experimentos.

El sistema diseñado facilita a los investigadores el desarrollo de los experimentos por medio de dos interfaces: la primera guía al investigador con menús amigables a la completa configuración del experimento; y la segunda interfaz le permite, de manera flexible, controlar el experimento, visualizar los resultados, para tomar decisiones acerca del entrenamiento de los sujetos de experimentación y almacenar datos para su posterior análisis.



6. Conclusiones

Logré migrar el sistema para el desarrollo de experimentos de psicofísica de las plataformas DOS y win98 a la plataforma de Windows XP, facilitando la configuración y control de los experimentos de psicofísica; permitiendo la visualización del desarrollo del experimento y de los resultados de desempeño del sujeto de experimentación, y permitiendo el almacenamiento de manera sencilla de los datos para su posterior análisis.

Este sistema combina la funcionalidad del anterior y agrega flexibilidad mediante la instrumentación virtual, todo a un costo relativamente bajo, en comparación con el costo del sistema anterior. El sistema funciona correctamente frente a las respuestas del sujeto de experimentación cualesquiera que estas sean; además, la flexibilidad del sistema permite realizar nuevos experimentos con estímulos diferentes y/o agregar estados al ensayo.

La adición de una nueva tarea, como podría ser la experimentación con estímulos auditivos, requerirá únicamente la programación de dichos estímulos, la integración de la información de los parámetros en el archivo de las clases y se podrá utilizar el resto de los programas e interfaces para desarrollar las tareas. Esto significa un ahorro de tiempo en la programación de un sistema nuevo al cambiar de modalidad de percepción.

Para el desarrollo del sistema utilicé una tarjeta de adquisición de datos PCI 6071 de la serie E, la cual permite enviar señales de control, y el software dedicado LabView versión 7.0, que permite programar las acciones de acuerdo a dichas señales de control e implementar con gran facilidad las interfaces gráficas para los usuarios, que en este caso son los investigadores. La principal ventaja que ofrece la tarjeta al sistema es la adquisición del tiempo de ocurrencia de las variables de control por medio de sus contadores y sus registros.

Los conocimientos adquiridos en la facultad me permitieron el crecimiento hacia la programación en el lenguaje gráfico y la implementación de sistemas de

adquisición de datos, así como el manejo de las tarjetas de adquisición de datos, herramientas necesarias para el desarrollo profesional en la industria y en la investigación en el área de la ingeniería en general.

El sistema es fácilmente portable, lo que ahorrará tiempo y dinero en la investigación en el laboratorio de neurociencias en el instituto de fisiología.

Uno de los problemas técnicos más relevantes de este trabajo fue el asociado a la generación de los estímulos para las tareas de patrones temporales. En el sistema anterior era necesario controlar el generador de señales (8904A) mediante interrupciones, con la finalidad de que los intervalos entre pulsos fueran estrictamente de la duración apropiada. Lo anterior requería: 1) una tarjeta PCTIO10 (con el temporizador programable), para generar las señales de interrupción en los tiempos adecuados (dos interrupciones por cada pulso mecánico); 2) una rutina de interrupción, programada en lenguaje C, la cual se ejecuta sobre el sistema operativo DOS, y enviaba los comandos de control a el generador de señales (8904A); y, finalmente, 3) el generador de señales, el cual es un aparto extremadamente caro.

El sistema diseñado elimina la necesidad de todos esos recursos y a pesar de ejecutarse sobre Windows XP (un sistema operativo no recomendado para sistemas en tiempo real), el diseño permite que el sistema no produzca retardos que comprometan el experimento realizado. En este caso, la totalidad de la forma de onda del estímulo es creada antes de iniciar el ensayo y la tarjeta PCI6071-E simplemente debe almacenar la forma de onda en un espacio de memoria y a través de una salida analógica enviar la señal al estimulador mecánico. Con esto se logra que la tarjeta se comporte como un dispositivo periférico y hace todo el proceso en forma independiente de lo que Windows XP este ejecutando sobre el procesador de la computadora.

7. Bibliografía.

1.- Kenneth E. Kendall, Julie E. Kendall. *Análisis y diseño de sistemas*. Prentice-Hall Hispanoamericana, S.A. México D.F. 3ª Ed. 1997.

2.-Fuentes Pasten Brenda, Rey Meneses Zelene. *Desarrollo de interfaces computacionales y electrónicas para un laboratorio de neurofisiología y psicofísica*. Tesis Facultad de Ingeniería, UNAM. 2000.

3.-Lemus Sandoval Luis Alonso. *Codificación neural de la memoria de trabajo en la corteza prefrontal del mono rhesus*.Tesis, Facultad de Ciencias, UNAM.2002.

4.- Alvarez López Jesús Manuel. *Diseño y construcción de un sistema para captura y procesamiento de actividad neuronal*. Tesis Facultad de Ingeniería, UNAM.2004.

5.- *Labview Basico I: Introducción,* Manual del curso. *National Instruments*. Agosto 2003.

6.-*Labview Basico II: Desarrollo,* Manual del curso*. National Instruments*. Octubre 2003.

7.-*PCI E Series Registrer-Level Programer Manual. Multifunction I/O Board for PCI Bus Computers*. National Instruments. Noviembre 1998.

8.- *PCI E Series User Manual. Multifunction I/O Board for PCI Bus Computers*. National Instruments. Enero 1999.

9.-*NI 6071E/6071E Family Specifications*. National Instruments. 2003-2005.

10.- *DAQ-STC Technical Reference Manual, System Timing Controller for Data Acquisition*. National Instruments. Enero 1999.

11.- *NI-DAQ for PC Compatibles. Version 6.9.1 Data Acquisition Software for the PC*. Febrero 2001.

12.-D.F.Kutz, N. Marzocchi. *Real-time Supervisor system based on trinary logia to control experiments hit behaving animals and humans*. JNeurophysiol 93:3674-3686,2005.