

# **Diseño, construcción e implementación de una placa Shield para Arduino® en una tarea de Finger Tapping**

Yamil Alejandro González Ferreyra

Laboratorio 7

Depto. de Física – FCEyN - UBA

Diciembre de 2018

ALUMNO(S): Yamil Alejandro González Ferreyra – LU: 217/12  
yamferreyra.23@gmail.com

LUGAR DE TRABAJO: Universidad Nacional de Quilmes – [www.unq.edu.ar](http://www.unq.edu.ar)

DIRECTOR DEL TRABAJO: Rodrigo Laje – [rodrigo.laje@gmail.com](mailto:rodrigo.laje@gmail.com)

CODIRECTOR o COLABORADOR:

*Leandro Versaci* – [focodefoco@gmail.com](mailto:focodefoco@gmail.com)

## Resumen

Se diseñó y ensambló una placa para Arduino (MEGA 2560) para realizar experimentos de finger tapping en el marco de estudio de los mecanismos cerebrales de estimación y producción de tiempos en el rango de las centenas de milisegundos. Los requisitos para el diseño contemplaron la capacidad de reproducir estímulos auditivos, registrar y reproducir tonos de respuesta en base a tapeos por medio de un sensor resistivo de fuerza, el control diferenciado del volumen para ambas señales sonoras, LEDS para generar respuestas visuales de igual manera que las respuestas auditivas. Se diseñó un experimento con distintas condiciones basadas en estímulos y respuestas dióticos y dicóticos para investigar a qué nivel en la secuencia de procesamiento se resuelve la sincronización sensomotora.

## 1. Introducción

Distintas tareas como bailar, practicar algún deporte, caminar o cualquier tarea motriz requieren de la estimación y producción de tiempos y en general de la capacidad de procesar la información temporal relevante a cada tarea. Los intervalos temporales involucrados en estas tareas son del rango de las centenas de milisegundos, con lo cual estas tareas se enmarcan dentro del *millisecond timing*, crucial para el control motor, la actualización de la memoria de trabajo, generación y reconocimiento del habla, tocar música y bailar [Buhusi & Meck 2005, Meck 2005]. En particular, la sincronización sensomotora es la capacidad de realizar un movimiento sincronizado con un estímulo externo periódico. En una tarea de finger tapping (el ejemplo más utilizado de sincronización sensomotora), quien practica el experimento (sujetx) recibe un estímulo periódico auditivo o visual externo (bip) que debe sincronizar con el golpeteo de un dedo (tap), como siguiendo el pulso de la música, y es usual utilizar un bip adicional (con distinta altura), llamado feedback que corresponde a cada tap del sujeto. Es simple demostrar que en el cerebro existe un mecanismo encargado de mantener un nivel de sincronía promedio: basta con observar que, si silenciamos el estímulo mientras se le pide al sujeto que continúe golpeteando, las asincronías “virtuales” divergen muy pronto.

Existe un creciente número de reportes de regiones cerebrales involucradas en millisecond timing, pero estamos lejos de entender cómo se relacionan entre ellas o cuáles son los mecanismos involucrados (corteza sensorial y motora (Rao et al. 1997); corteza premotora medial (Merchant & Averbach 2017, macacos); corteza premotora dorsolateral (Pollock et al 2017, reproducción temporal); cerebelo (Paquette et al 2017, discriminación de intervalos); corteza sensomotora, área motora suplementaria, cerebelo, putamen, tálamo (Moritz et al 2000); red fronto-parietal: dPMC, M1, S1, LPi, SMA (Jancke et al. 2000); redes corticales y subcorticales, área motora suplementaria, cíngulo anterior (Jantzen, Ratcliff & Jantzen 2017))). Es por esto que los estudios de comportamiento siguen siendo una herramienta fundamental para entender el funcionamiento del mecanismo de corrección del error.

Los primeros investigadores que trabajaron con sincronización sensomotora observaron que los taps tienden a preceder al bip de estímulo (ver figura 1). Dicha anticipación temporal, conocido como Asincronía Negativa Media (NMA), o anticipation tendency, es del orden de las pocas decenas de milisegundos, y es la asincronía promedio de un sujeto para una secuencia de bips. Dichas asincronías son negativas e indican que el sujeto se adelanta a cada estímulo en la tarea (Repp 2005).

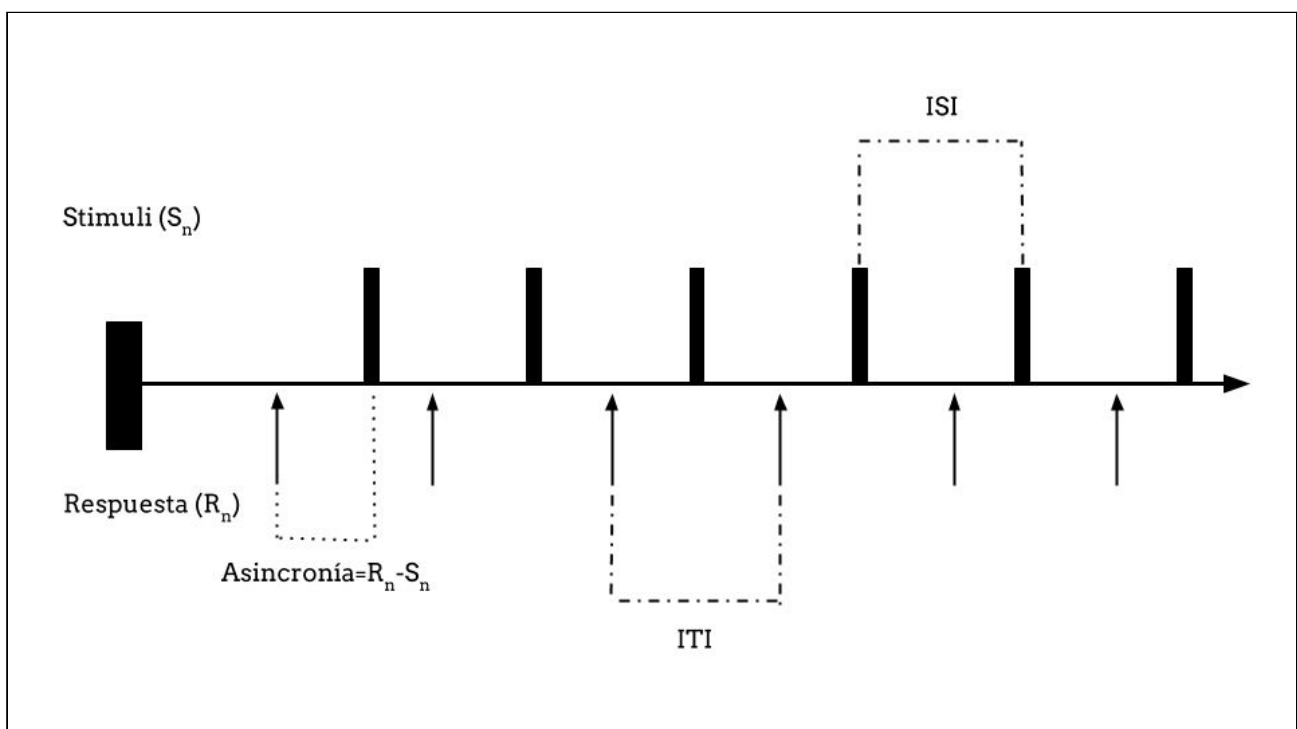


Figura 1. Esquema de serie temporal de bips y taps de respuesta.

Si bien estas asincronías se presentan en la mayoría de los experimentos y estudios de Sincronización Sensomotora, las causas de la NMA siguen sin ser totalmente comprendidas. Una de las explicaciones más simples se basa en el hecho de una menor velocidad de transmisión y procesamiento para la modalidad somatosensorial (la información sobre el tap que es primordialmente tacto y propiocepción) en comparación con la auditiva (que es por donde habitualmente llega el estímulo). Para que las percepciones de respuesta y

estímulo coincidan a nivel de procesamiento central, el dedo debe anticiparse al bip (hipótesis de Fraise-Paillard).

Como se dijo, la sincronización está asociada a la información de los órganos sensoriales. Dicha información, proveniente de distintos órganos o distintos canales (como pueden ser oídos izquierdo y derecho, u ojos izquierdo o derecho), no solo se procesa en distintas regiones cerebrales, sino que para cada canal se procesa contralateralizada y tiene distintos estadios de proceso (por ejemplo ingresa en corteza sensorial, pasa por corteza frontal si se resuelve conscientemente, termina en corteza motora si conlleva una respuesta de parte del sujeto, etc). Es en las etapas superiores de este procesamiento donde la información diferenciada se junta y complementa para dar paso a una acción en el plano motor (ver figura 2).

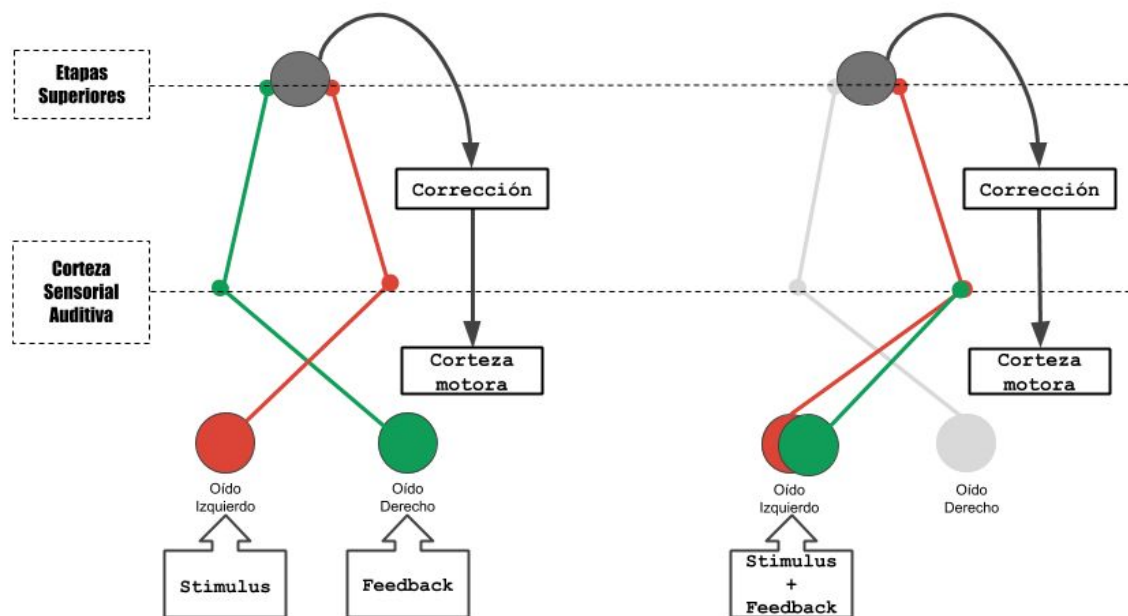


Figura 2. Esquema de procesamiento de la información sensorial auditiva.

Esto nos lleva a plantear los siguientes interrogantes e hipótesis de partida: en una tarea de finger tapping, a qué nivel en la secuencia de procesamiento se resuelve la sincronización sensomotora? A nivel de corteza sensorial, o a niveles más altos como corteza frontal o corteza motora? La condición normal en la que un sujeto recibe la información auditiva es, en general, a través de ambos oídos simultáneamente (diótica). Para probar la hipótesis de este trabajo se presentará la condición diótica usual y además condiciones dicóticas donde los oídos no reciben la misma información, como por ejemplo estímulo y feedback en el mismo oído (el otro oído no recibe nada) y estímulo y feedback en oídos diferentes. Se espera encontrar diferencias si el procesamiento se lleva a cabo a nivel de corteza auditiva, pero no si se lleva a cabo en niveles superiores de procesamiento.

## 2. Diseño y construcción de la placa

### Requerimientos

Como proyecto para **Laboratorio 6**, se buscó diseñar esta placa para realizar experimentos de finger tapping más avanzados que los tradicionales, como por ejemplo medir la fuerza que ejerce el dedo durante el golpe, controlar servos para perturbaciones espaciales (por ejemplo cambio de altura del punto de contacto), emitir estímulos dióticos (iguales para ambos oídos) o dicóticos (diferentes para cada oído), elegir diversos timbres para los sonidos, etc. El diseño experimental contempla los siguientes requerimientos y permite las siguientes manipulaciones experimentales: 1. presentación de estímulos periódicos auditivos y visuales; 2. feedback auditivo o visual de las respuestas del sujeto; 3. adición de ruido blanco para enmascarar posibles distractores; 4. independencia entre canales de audio izquierdo y derecho; 5. capacidad de realizar perturbaciones espaciales como por ejemplo mover el punto de contacto del dedo con un servo; 6. doble sensor de fuerza para experimentos específicos; 7. sincronización con otros equipos (por ejemplo un acelerómetro o electroencefalógrafo).

### Descripción por bloques lógicos y detalles de construcción

En la figura 3 se observan las conexiones lógicas de el circuito diseñado y la separación en distintos bloques de acuerdo a sus funcionalidades.

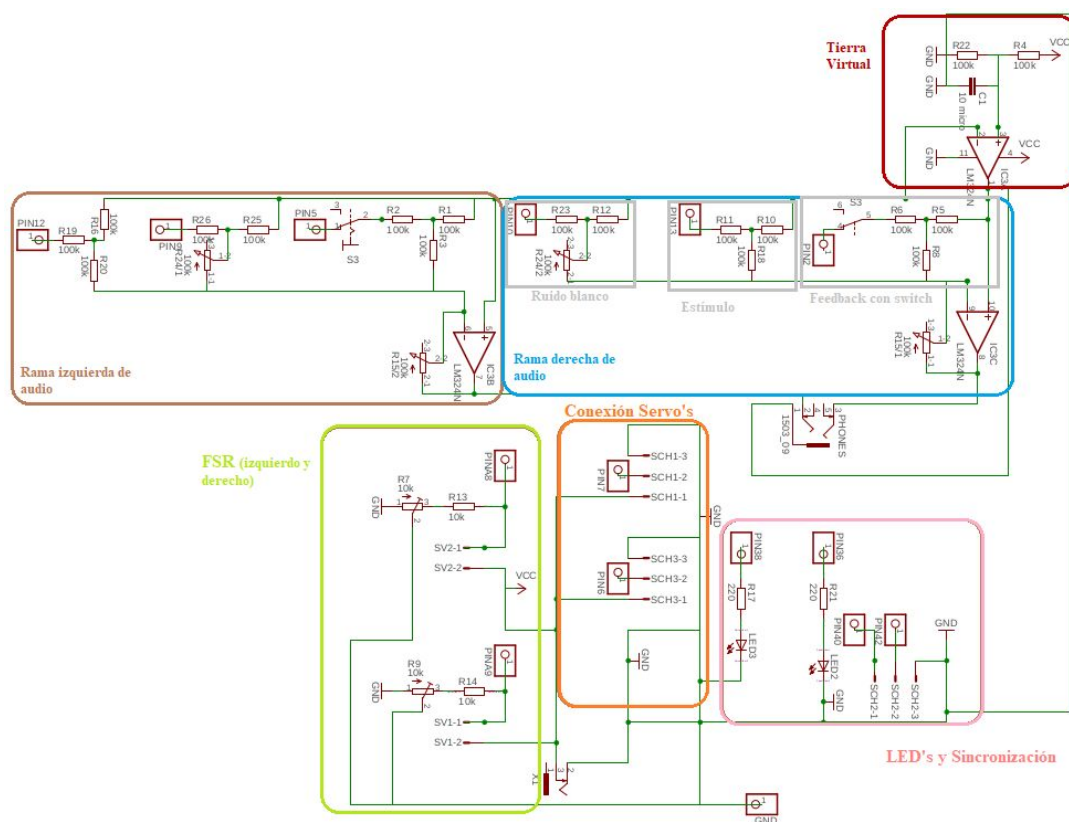


Figura 3. Conexiones y bloques lógicos del diseño del Shield

## 2.1 Audio

Para lograr estímulos dióticos o dicóticos, los canales están discriminados en izquierdo y derecho. Así mismo, feedback y ruido blanco también necesitan ir por canales independientes. Cada canal de audio está constituido por tres ramas: estímulo, feedback y ruido blanco. Cada una de éstas comienza en un pin del Arduino (que brinda como máximo 5V), pasa por un divisor de tensión para reducir su amplitud y entra a un sumador-amplificador. Finalmente llega al plug del auricular (ver figura 3).

**Pines y timers** La placa Arduino MEGA 2560 posee distintos tipos de pines; los elegidos para trabajar son del tipo PWM (Pulse Width Modulation) ya que permiten generar señales analógicas a partir de una salida digital y, además, permiten reproducir formas de onda diferentes a la cuadrada (estándar). De esta manera se logra que estímulo y feedback tengan timbres arbitrarios como por ejemplo sinusoidal o triangular, y el sujeto pueda diferenciarlos. Los PWM están asociados a timers, que son relojes implementados en hardware dentro del Arduino con resolución temporal de algunos microsegundos. Los timers pueden tener una resolución de 8 y 16 bits. Particularmente, para las ramas de estímulo y feedback se eligieron sendos timers de 16 bits (asociados a los pines 2, 9, 11 y 13), mientras que para las ramas de ruido blanco, con 8 bits se obtiene suficiente resolución. Estos pines PWM, además, permiten configurar un valor medio de la señal enviada, lo cual resultó muy útil para el diseño del ruido blanco que enmascara el sonido alrededor del sujeto. Si bien las ramas provienen de pines diferentes, no son totalmente independientes, ya que comparten un timer en común. Lo cual tiene como fin evitar pequeñas diferencias de fase entre lo que recibe un oído con respecto al otro en los estímulos dióticos.

**Amplificación** Para combinar las tres señales estímulo-feedback-ruido dentro de cada rama y amplificarlas con la misma ganancia, se utilizaron Amplificadores Operativos (de aquí en adelante Op-Amp) que, a su vez, fueron configurados como sumadores con una resistencia variable (potenciómetro doble) que permite controlar la ganancia del voltaje de salida del Op-Amp.

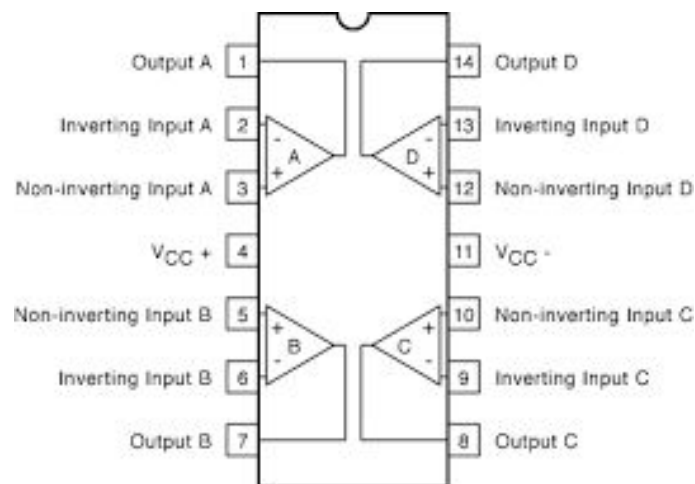


Figura 4. Chip LM324N

Se empleó un chip LM324N que tiene cuatro opamps y alimentación "single supply" 0-5 V, ya que todos los voltajes que entran o salen del Arduino están en ese rango. Dos de los opamps son utilizados para amplificar los canales derecho e izquierdo de audio. Los mismos fueron configurados en modo sumador / amplificador inversor, es decir, que cada amplificador operacional unifica la suma y amplificación de las señales de audio de cada canal.

**Tierra virtual** Un tercer opamp fue utilizado para poder suplir de una tierra virtual a los canales de audio. La misma, alimentada con un voltaje de 5vpp, ofrece un valor medio de 2.5vpp alrededor del cual oscilarán las señales de audio. Tanto el sumador/amplificador como la tierra virtual tienen diseños específicos para opamps de single supply, que difieren ligeramente de los diseños más comunes con dual supply (-5,0,5 V). Mientras que el estímulo está constantemente encendido, sendas ramas de feedback cuentan con un interruptor que permite encender o apagar dicha respuesta; los canales de ruido blanco, por su parte, cuentan también con un potenciómetro doble que permite regular el volumen del mismo, diferenciado de las otras señales.

## 2.2 Registro de respuestas

Para registrar el golpe del dedo se utilizaron dos sensores resistivos FSR Interlink Electronics; el funcionamiento de los mismos se basa en que cuando no son presionados ofrecen una "resistencia infinita", la cual desciende rápidamente al ser presionado el sensor. Se colocan como parte de un divisor de tensión y, de acuerdo a la resistencia del sensor (es decir, a si está presionado o no) el punto medio del divisor variará su voltaje lo que es registrado por un pin analógico del Arduino. Para regular la sensibilidad del sensor se agregó un resistor trimpot; así mismo, se logra evitar saturaciones y respuestas múltiples, y que entregue hasta 5v.

## 2.3 Otras características

**Comunicación con otros equipos** Para poder agregar más información desde otros dispositivos como acelerómetros, encefalogramas, entre otros, se añadieron pines de entrada y salida digital para adquirir datos con un eje de tiempo común a todos los instrumentos de adquisición.

**Servos y perturbaciones espaciales** En pos de poder generar perturbaciones espaciales, dos conectores fueron destinados para sincronización con motores servos. Estos harán subir y bajar, de manera arbitraria al sensor, agregando así un nuevo grado de libertad al experimento.

**Software de diseño** Para trabajar todo el diseño, se utilizó el software de diseño EAGLE de Auto-desk. El mismo admite el armado de un esquema de conexiones lógicas para los elementos de un circuito que luego se puede pasar al diseño de un board, en el cuál se realizan las conexiones físicas. Este último cuenta con una función de autoruteo que se encarga de optimizar las conexiones entre elementos (en base a las conexiones lógicas del esquemático) evitando el cruce pistas de cobre.



## 2.4 Testeo y caracterización

Para corroborar que el diseño y las conexiones lógicas funcionan, se armó cada rama del diseño sobre un protoboard y se testeó con un osciloscopio. Para evitar fugas de corriente en las ramas de audio, las resistencias de los divisores de tensión y la de los sumadores son de distinto orden de magnitud, siendo de  $10\text{k}\Omega$  y  $100\text{k}\Omega$  respectivamente. Estas últimas, además, son de dos órdenes de magnitud menores que la impedancia de entrada de los opamps. Una vez establecido un diseño preliminar del armado, donde se hicieron todas las conexiones lógicas mediante el uso del software EAGLE, se procedió a ubicar los elementos en una plantilla para imprimir sobre una placa de cobre. El programa admite una función de autoruteo que se encarga de encontrar recorridos posibles de pistas de conexión entre los elementos del PCB (Printed Circuit Board). Se decidió la ubicación inicial de los elementos de manera que el autoruteo logre una resolución de las conexiones muy cercana al 100 %.

Para ello, primero se ubicaron los pines correspondiéndose con la distribución de una placa de Arduino Mega 2560. Luego, el elemento principal sobre el cuál se ordena el resto, es el Chip LM324N. Es importante que a la hora de trazar las conexiones, ninguna pista pase entre patas del integrado, ya que son muy cercanas y en el lavado es difícil que se separen, lo cual puede armar una conexión indeseada. Se tuvo en cuenta, cuando se armó el diseño esquemático, asignar los opamp's del chip de manera que las conexiones fueran fáciles de rutear. Se optó por realizar un ruteo de doble capa de manera que se solucionaran la mayor cantidad de conflictos de pistas de cobre. En la figura 4 se observa el ruteo final de ambas capas. Luego se procedió a imprimir y transferir el diseño a la placa de cobre, y posteriormente hacer el *lavado* de la placa (se sumerge la placa en una solución de cloruro férrico para que este disuelva el cobre expuesto y solamente queden aquellas trazas cubiertas por la tinta del estampado).

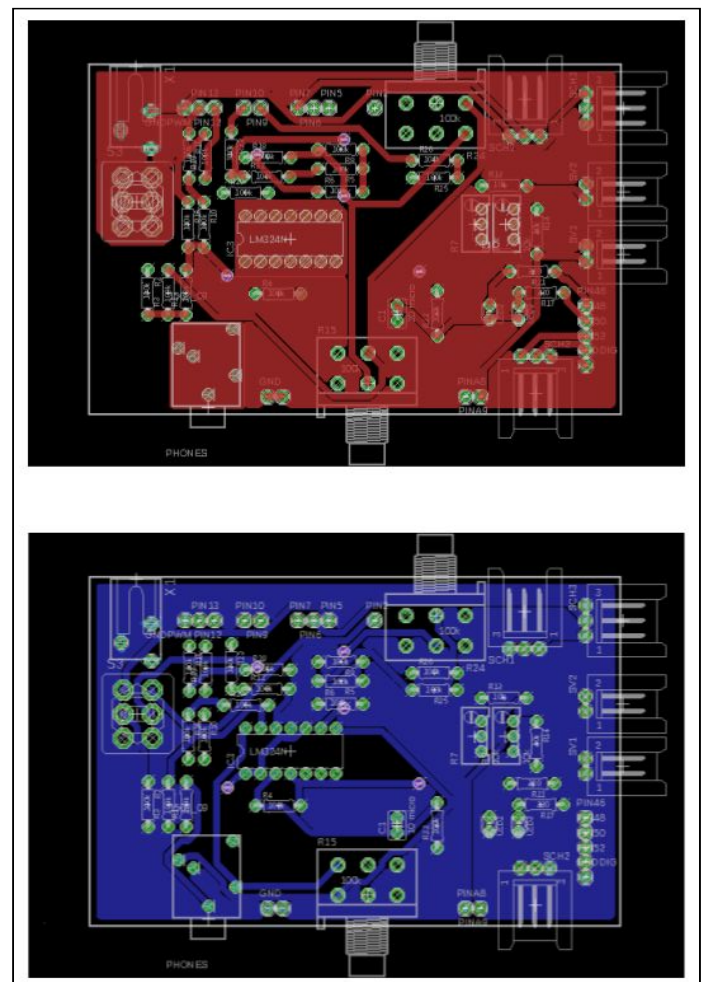


Figura 5: ruteo superior (arriba), ruteo inferior (abajo)

En la figura 6 se observa el montaje final de la placa y los diseños de las dos capas. Como propuesta a futuro, se propone resolver de una manera más completa el circuito, verificar que en los sucesivos guardados del archivo de diseño del pcb, se constate que no se corten conexiones y finalmente proceder con el montaje de los elementos a la placa. Asimismo, observamos y recomendamos el uso de transfer siliconado para hacer la transferencia del diseño a la placa, ya que con papel transfer común, el mismo es muy difícil de retirar y para diseños con pistas tan próximas entre sí es muy fácil que la pelusa del papel genere conexiones innecesarias entre pistas.

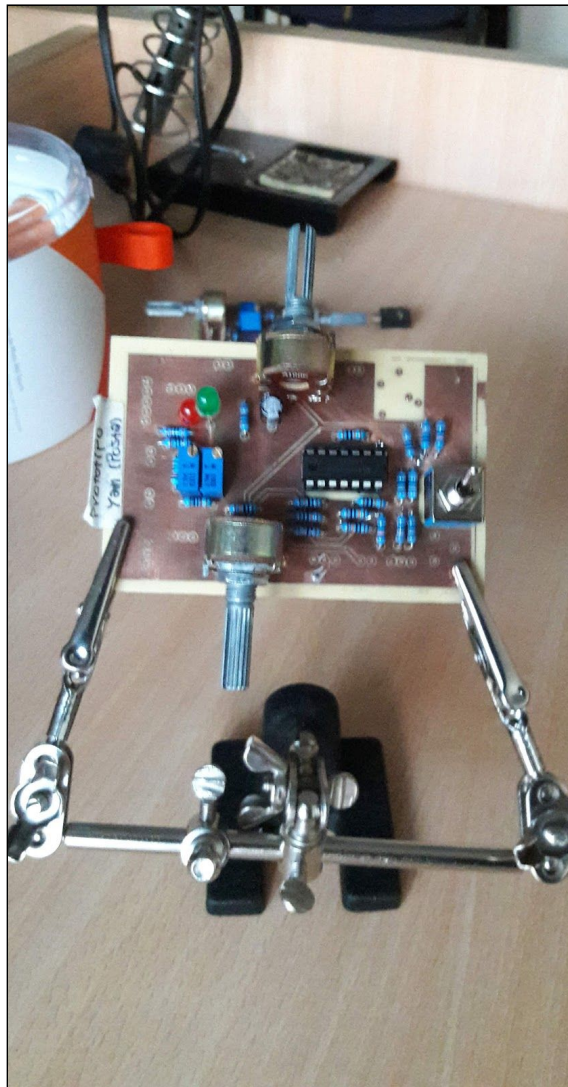


Figura 6. Montaje final de la placa diseñada.

### 3. Diseño Experimental

Como parte de la materia **Laboratorio 7**, y con mismo lugar de trabajo que **Laboratorio 6**, para testear la hipótesis sugerida en este trabajo se diseñó un experimento psicofísico en el cual a 16 sujetos se le presentan estímulos periódicos y también un tono de feedback en base a la respuesta registrada. Desarrollamos dos códigos para realizar el experimento. Primero, para controlar el Arduino se escribió un código en C que genera estímulos de 495 Hz y tonos de feedback de 1800 hz, y registra las respuestas. Este código recibe instrucciones al comienzo de cada trial por parte de un código Matlab que es el manager general del experimento y es el que administra en qué momento se presentan las 5 condiciones distintas (ver figura 7), una diótica y 4 dicóticas (dos con estímulo y feedback cruzado y dos con estímulo y feedback en el mismo oído y el otro oído sin información). Al final de cada trial el código del Arduino envía los datos registrados y devuelve al control al Matlab.

		STIMULI		
		Right	Left	Both
FEEDBACK	Right	SL-FR	SL-FL	
	Left	SR-FL	SL-FL	
	Both			BTH-BTH

Figura 7. Bloques de condición

Cada condición está constituida por 12 trials de 30 bips cada uno, y un período o intervalo inter-stimuli (ISI) de 444 ms. El período fue elegido de manera que el sujeto no reciba un estímulo con una periodicidad que le resulte familiar (como puede ser un segundo o medio segundo). En el caso de que el sujeto se adelante o atrase demasiado al estímulo ( $>|250\text{ ms}|$ ), el programa considerará que éste se perdió un bip o tapeó de más, con lo cual agrega un trial más al bloque correspondiente, de manera que se cumplan los 12 bien hechos. En la figura 8 se observa el desarrollo del experimento: las condiciones son presentadas aleatoriamente al sujeto, a través del Arduino; toda la información recopilada por éste es luego analizada con Matlab.

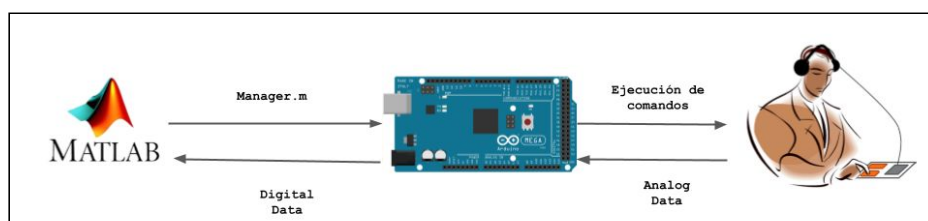


Figura 8. Desarrollo experimental

El armado experimental que finalmente se utilizó no incluyó la placa diseñada, debido a que la misma tenía algunas conexiones sin resolver, como se mencionó al final de la última sección. Los experimentos se realizaron con un protoboard. En este se respetó el armado lógico de la placa shield, exceptuando la utilización de los leds, los conectores de sincronización con otros dispositivos y los trimpots, por lo tanto no se pudo regular la sensibilidad del FSR. El armado final utilizado se observa en la siguiente figura X. Se utilizaron auriculares Sony In-Ear con el objetivo de insonorizar el ambiente, ya que el programa se modificó para no utilizar el ruido blanco para enmascarar los ruidos del ambiente. En la figura 9 se observa el armado final del set-up experimental.

## 4. Resultados y discusión

### Resultados

En la siguiente figura se observa un trial individual de la condición normal (5) para uno de los sujetos. En la misma se observa que, en promedio para este trial, las asincronías tienen mayormente valores negativos como se describe en la introducción; sin embargo, se observan algunas asincronías positivas que muestran que el sujeto, en ese tap, estuvo atrasado con el estímulo.

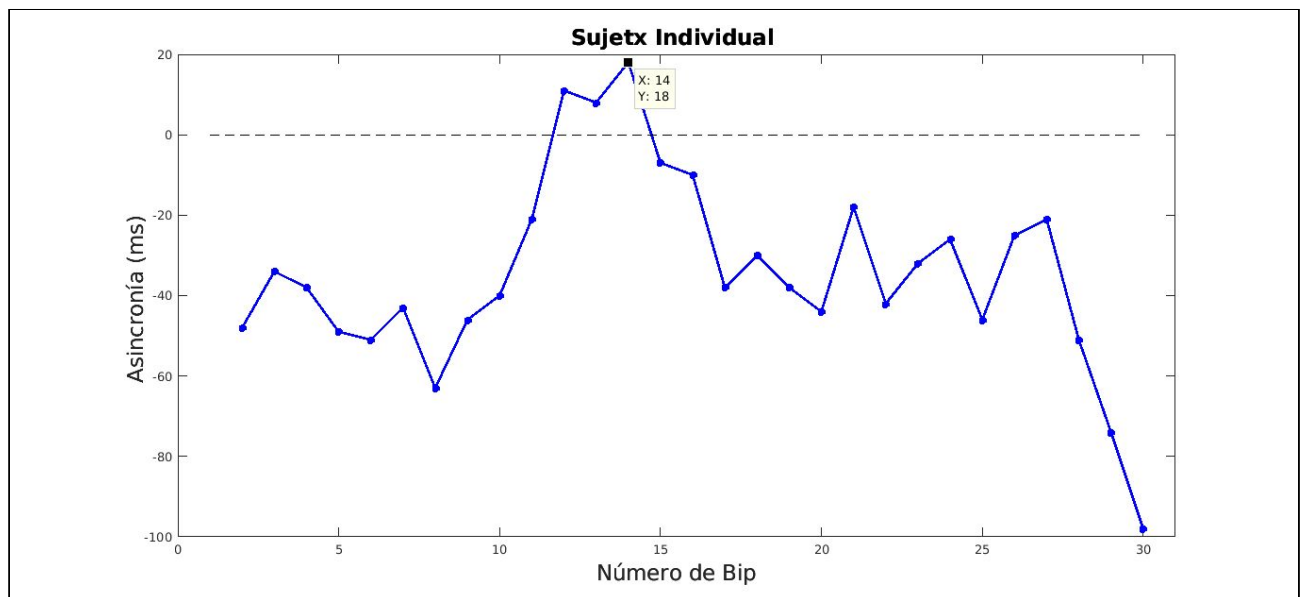


Figura 9. Trial individual

Cada sujetx realizó 5 bloques aleatorios y distintos entre sí, de 12 trials con 30 bips de estímulo cada uno. En la siguiente figura se puede observar que la señal es muy ruidosa para cada condición, con lo cual se ratifica que las asincronías varían a lo largo de cada trial:

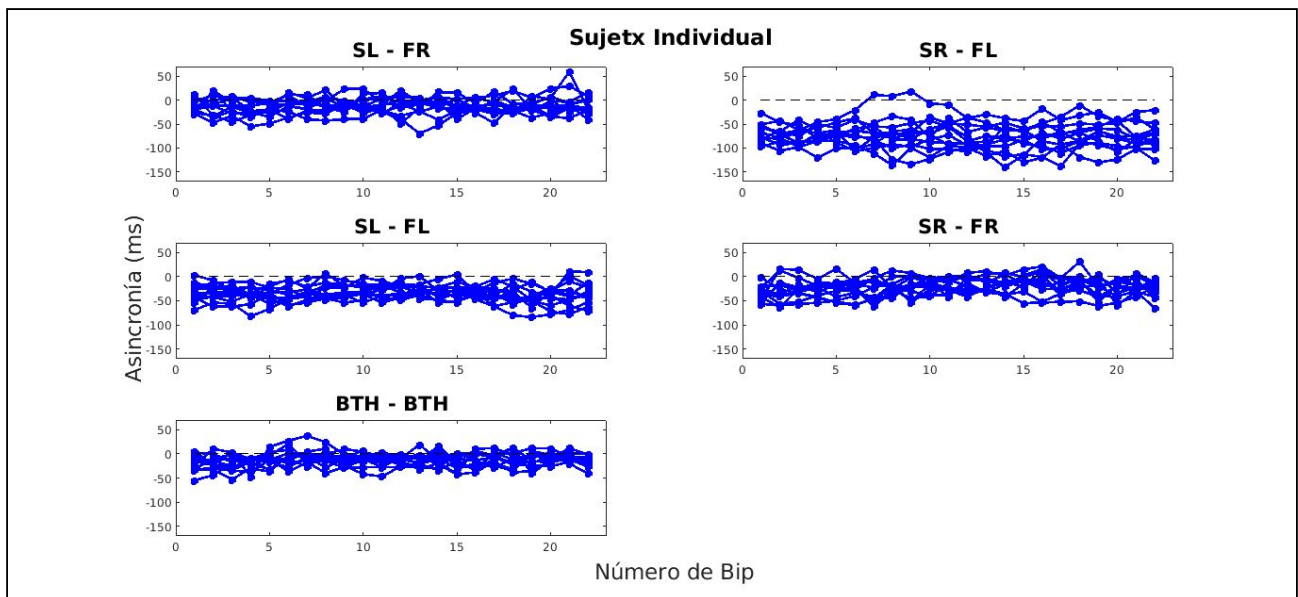


Figura 10. Experimento completo realizado por unx sujetx

## Análisis de trials: drift y outliers

Para analizar los trials, lo primero fue poder determinar los taps outliers a retirar.

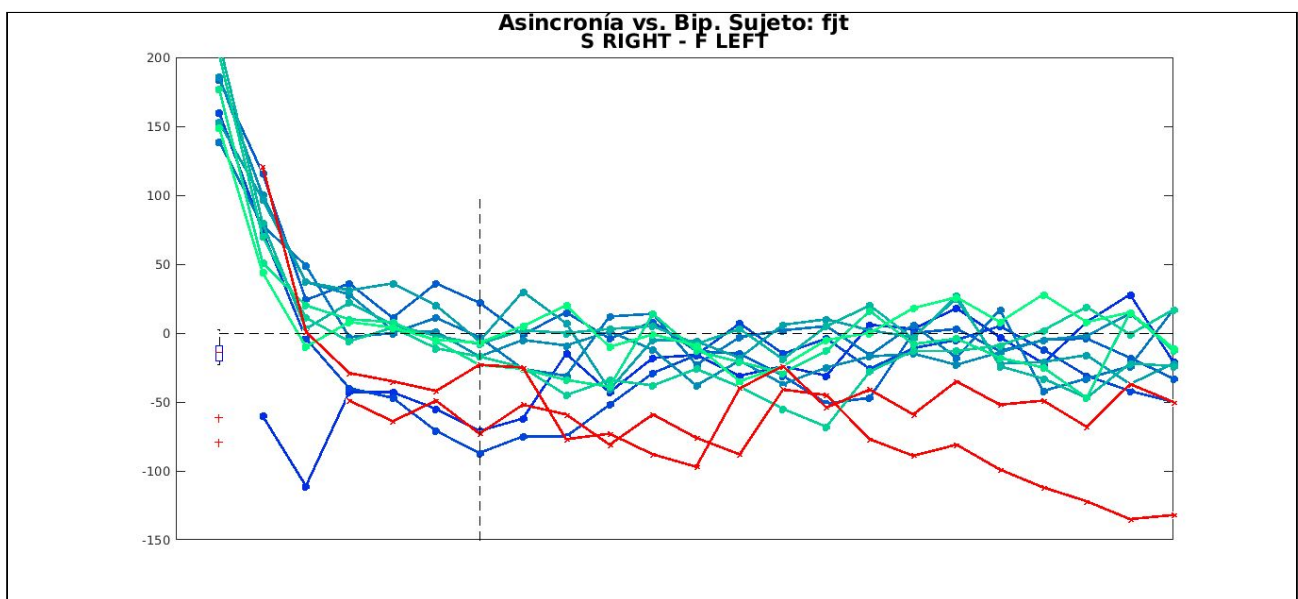


Figura 11. Un bloque de trials para la condición 3 (SR-FL) con trials outliers en rojo y drift en los primeros taps.

En la figura 11 podemos observar la variabilidad entre trials con los outliers marcados en rojo por el criterio de BoxPlot. Éste es un método estandarizado para mostrar la distribución de datos basado en 5 valores estadísticos, denominados cuartiles; los mismos corresponden a el mínimo de la distribución, el primer cuartil (Q1), la mediana, el segundo cuartil (Q2), y el máximo. Este criterio nos permite determinar los trials más distantes en nuestro experimento, así como también determinar si la distribución



de puntos se tuerce (en este caso se observa claramente que el outlier rojo está alejado precisamente por cambiar su dirección respecto del resto de los trials). El código con el cual se analizaron los datos fue escrito para limpiar las dispersiones de estos puntos, y así obtener una dispersión representativa. Se observó que en los primeros taps de cada trial lxs sujetxs presentan una pendiente significativa, que llamaremos drift, el cual hace que la NMA cambie un poco. La dispersión de dicho drift no es representativa, ya que se puede observar que la dispersión tiende a quedar confinada en una manga de puntos (delimitada en verde claro) y asociamos dicho drift a que el sujeto es tomado por sorpresa por el comienzo del trial. En la mayoría de las dispersiones de los datos de cada sujeto se observa una corrección notable a lo largo de los taps, que se puede apreciar como una curva que se acerca al eje de abscisas.

También se observa (en la figura 11) que si bien algunos sujetos no llegan a una sincronización en la cual la curva converge a cero, la misma sí oscila alrededor de un valor de NMA esto indica que la sincronización existe y está defasada.

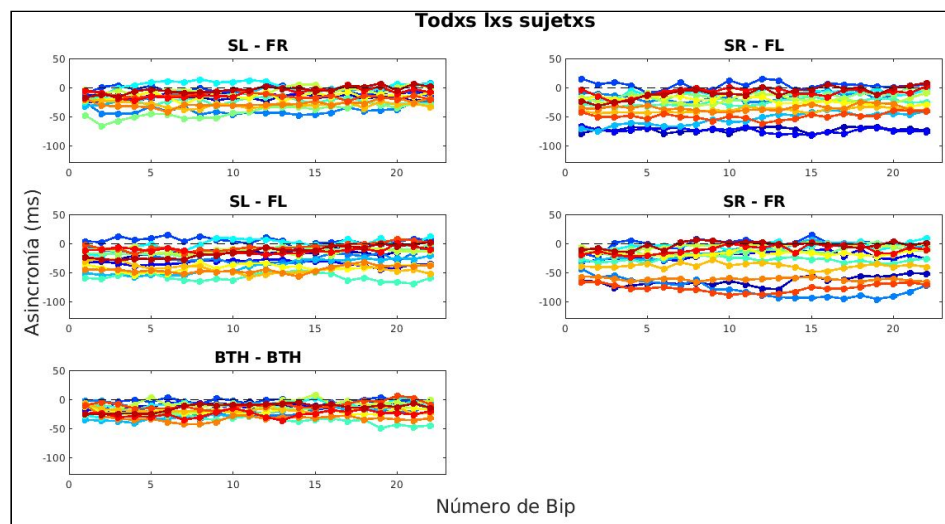


Figura 12. Conjunto de todas las NMA de todxs lxs sujetxs (un color por sujetx), en cada condición.

Todxs lxs sujetxs demostraron tener NMA's muy variadas al comparar sus condiciones entre sí (figura 12). Sin embargo, se observa que la mayoría de las asincronías, en promedio tienden a ser menores en las condiciones donde la escucha del estímulo involucra al oído izquierdo y podría estar relacionado al fenómeno de "oído dominante". condiciones 1 (SL-FR), 3 (SL-FL) y 5 (BTH-BTH) tienen anchos similares de dispersión.

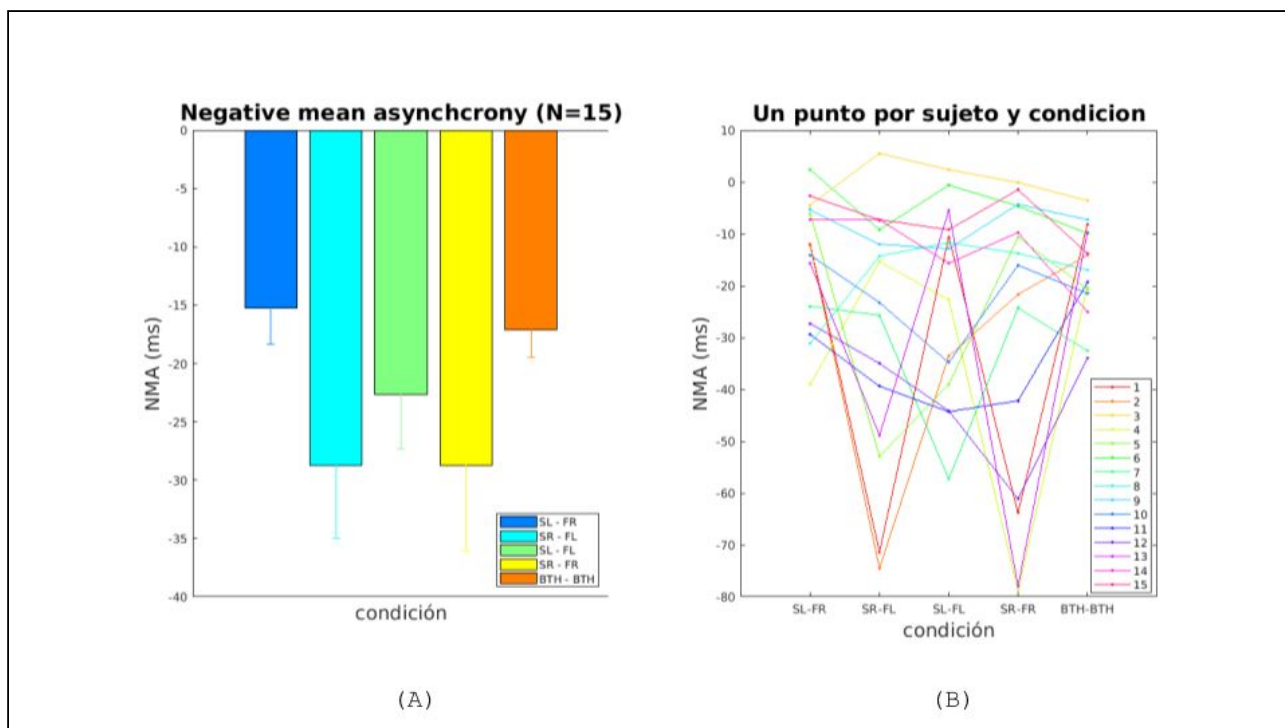


Figura 13. Diagrama de barras con los valores de NMA globales por condición (A); Valor de NMA por sujeto en cada condición (B).

Se analizaron los valores medios de todos en cada condición con el fin de comparar cada condición con la condición 5, que es la condición normal (barra naranja en la figura 13.a) y determinar si la manipulación experimental produce diferencias. Al analizar la NMA promedio entre sujetos para cada condición (Figura 13.a), se observan diferencias aparentes que no son respaldadas por la estadística (ANOVA de dos factores, Estímulo x Feedback, dos niveles cada factor, medidas repetidas: no significativo  $p > 0.05$ ; comparaciones entre condiciones individuales, test de Student para medidas repetidas, corregido por comparaciones múltiples: no significativo  $p > 0.05$ ). Este resultado puede analizarse en mayor detalle al graficar los mismos datos pero discriminados por sujeto (Figura 13.b): se observa que el comportamiento no es sistemático para todos los sujetos, algunos suben entre condición 1 y 2 mientras otros bajan y el hecho de que el diseño experimental sea de medidas repetidas hace que este comportamiento no sistemático juegue en contra. Considerando que un diseño experimental de medidas repetidas es lo correcto en este caso, podemos concluir que el resultado estadístico refleja lo que se ve en los datos de manera fiel.

## 5. Conclusiones

- conclusión de labo 6: se logró diseñar y probar un circuito que permite realizar experimentos psicofísicos de procesamiento temporal "state-of-the-art" de primer nivel.
- conclusión de labo 7: diseñamos y llevamos a cabo un experimento psicofísico, recolectando datos de 15 sujetos. encontramos que no hay diferencias significativas entre condiciones, lo que de acuerdo a nuestras consideraciones iniciales significa que el procesamiento temporal se lleva a cabo en niveles superiores más allá de la corteza sensorial. a pesar de obtener un resultado negativo (no hay diferencias), este experimento es de gran interés para el LDSM y se consideró como una importante prueba piloto