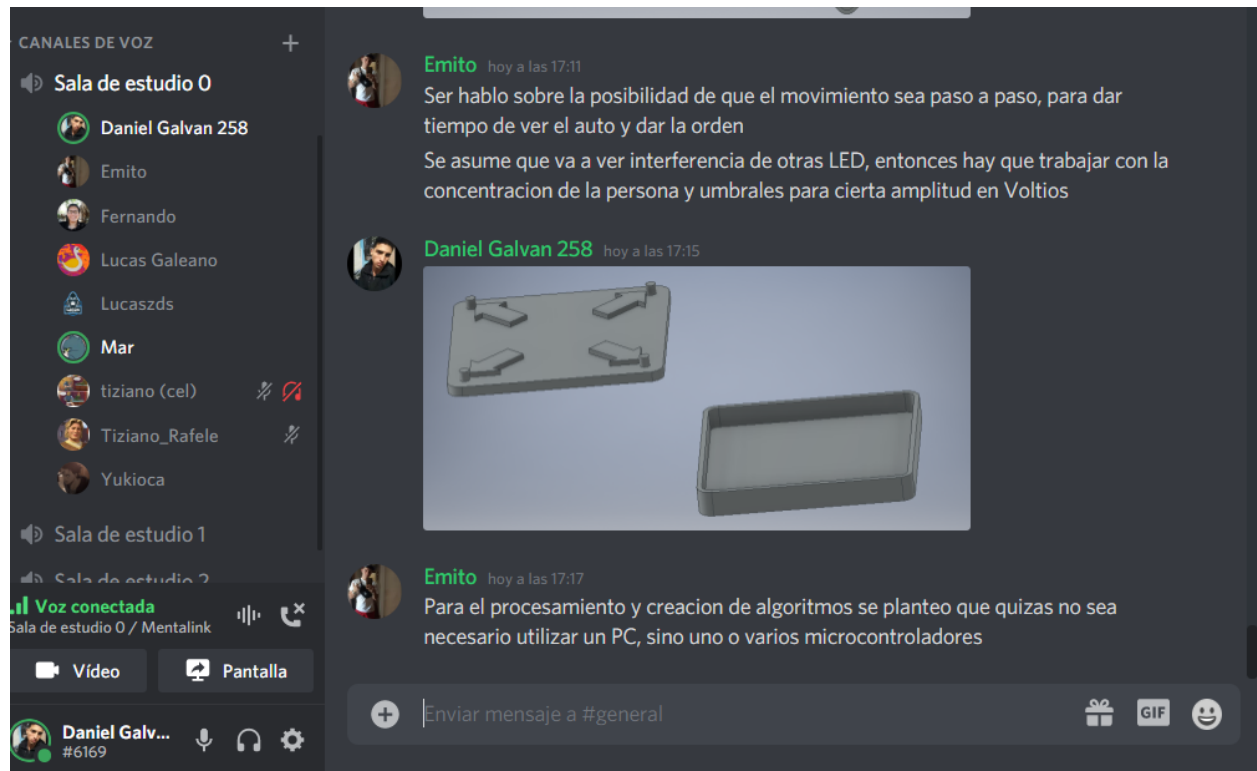


Segunda reunión de Mentalink



Objetivos

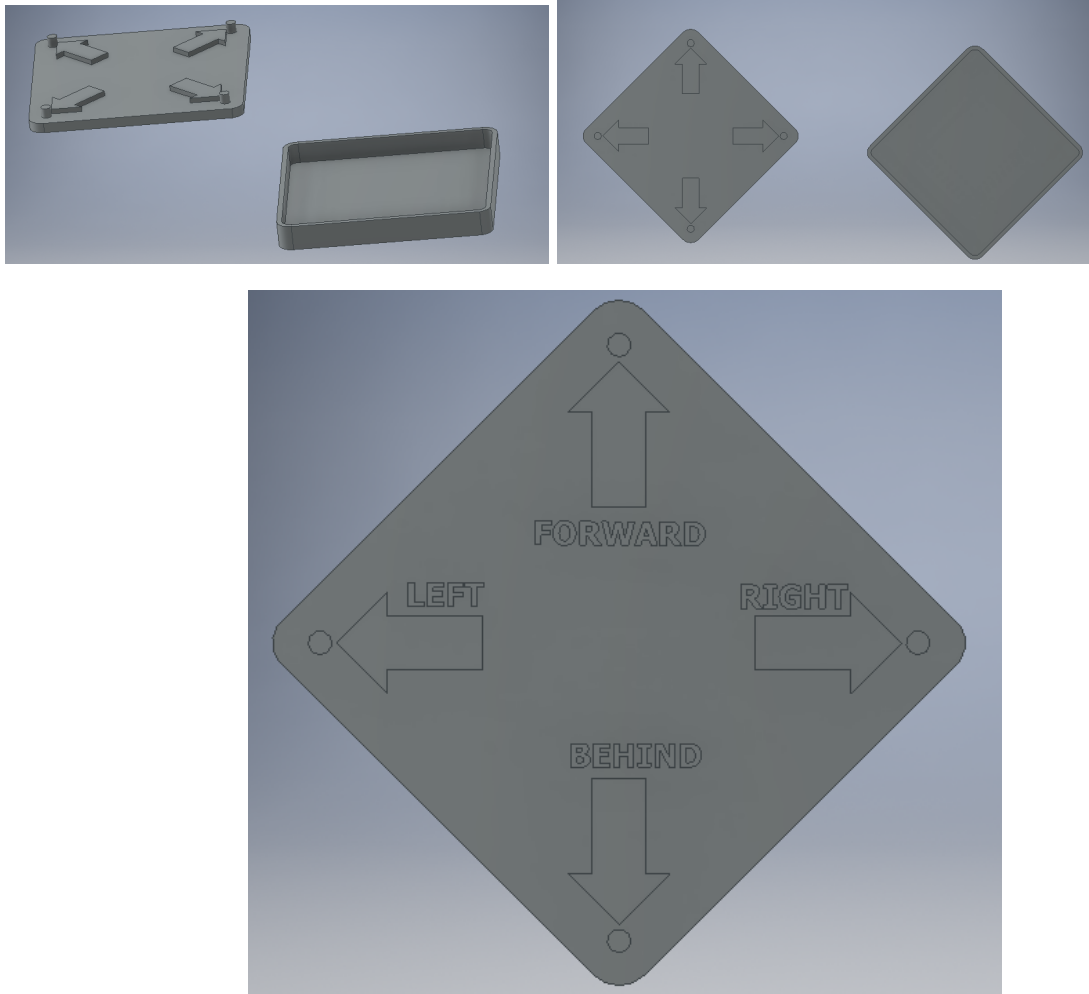
- Seguir plasmando la idea sobre el mando
- Charlar sobre el código
- Definir protocolo de comunicación Wireless
- Charlar sobre el cronograma
- Cómo va a ser el movimiento del vehículo
- Dividir en subgrupos para distribuir mejor el trabajo.
- Plantear dudas y una posible fecha para la reunión con el equipo coordinador.

COMANDO

Se asume que va a ver interferencia de otras LED, entonces hay que trabajar con la concentración de la persona y umbrales para cierta amplitud en Voltios.

En tanto a como varía la señal en relación a la concentración de la persona, el equipo de biomédica se va a encargar de la investigación (Emiliano Alvarez, Walter Mar y Daniel Galván).

Posible diseño de la consola de comando



- Se planteó la forma de un rombo para optimizar el área.
- Se planteó la posibilidad de una LED en el medio que al concentrarse en mirarla vuelva al vehículo a su estado inicial (sin movimiento) enseguida.

ETAPA DE PROCESAMIENTO Y ALGORITMO DE LAS ÓRDENES

Para el procesamiento y los algoritmos de la señal se planteó que quizás no sea necesario utilizar un PC, sino uno o varios microcontroladores. Esto traería varias ventajas, siendo la principal que el equipo en general sea más transportable y fácil de poner en marcha pensando en otras aplicaciones, como en una silla de ruedas.

MODIFICACIONES AL DIAGRAMA DE BLOQUES

Como se planteó la posibilidad de no se necesite un ordenador para la etapa de procesamiento, se modificó de cierta forma el diagrama de bloques:

CONTROL-EGG ---wireless-- ETAPA DE PROCESAMIENTO -(posibilidad de wireless o no, si el microcontrolador de procesamiento y el de algoritmo está en el propio vehículo) - MOVIMIENTO VEHÍCULO.

Variante

CONTROL-EGG ---wireless-- ETAPA DE PROCESAMIENTO (se encuentra dentro del panel de control con las LEDS (otro microcontrolador)) --wireless -- VEHÍCULO -FEEDBACK.



Se iban a utilizar a lo sumo 5 microcontroladores:

- Comando (LEDs), probablemente una placa Arduino o un ATMEGA
- El que se encarga de la adquisición de la señal en el casco, ADC y su pre procesamiento (este lo proporciona OPEN BCI)
- Un microcontrolador potente que se encargue de realizar el algoritmo de obtención de las órdenes (lectura de los datos provenientes del casco, FFT y umbrales). En esta etapa puede dividirse en uno o varios microcontroladores, se analizará posteriormente. Un SP32 o una placa Raspberry Pi eran una de las posibilidades que surgieron.
- Uno que al recibir la señal del microcontrolador encargado del procesamiento lo interprete y controle los motores del vehículo. Placa Arduino o SP32 con un puente H.
- Autonomía del vehículo (frenado, feedback, ect.) Un Arduino probablemente.

PROTOCOLOS DE COMUNICACIÓN WIRELESS

Se nombró la posibilidad de 3 protocolos Wireless: WiFi, Bluetooth y LiFi.

Este último (LiFi) era desconocido para la mayoría de los integrantes del equipo, donde surgió una charla muy interesante para informarnos entre todos de que se trataba. Aunque al final lo optamos por descartar ya que era una tecnología muy nueva y era innecesario para este proyecto.

WI-FI vs LI-FI	
By - @python3learning	
	
• 54 - 250 Mbps speed	• 1- 3.5 Gbps
• 20 - 100 Meter Range	• 10 Meter Range
• Point to Multipoint	• Point to Point Range
• 2.4 GHz	• 100 Times of TeraHZ
• 801.11b IEEE Standard	• 802.15.7 IEEE Standard

Se habló sobre la posibilidad de que los protocolos WiFi y Bluetooth interfieran entre sí ya que utilizan el mismo rango de frecuencia.

O que la interferencia se produzca entre dos señales que utilicen el mismo protocolo.

Bluetooth 4.0 LE (también: Bluetooth smart)	Diciembre de 2009	24 Mb/s	Pila de protocolos Low Energy (LE) para distintos métodos de ahorro de energía (p. ej., perfil GATT) para dispositivos pequeños; corrección de errores mejorada; cifrado de 128 bits
Bluetooth 4.1	Diciembre de 2013	25 Mb/s	Los dispositivos más pequeños ya no necesitan intermediarios; IPv6
Bluetooth 4.2	Diciembre de 2014	25 Mb/s	Mejoras generales
Bluetooth 5.0	Diciembre de 2016	50 Mb/s	Aumento considerable del alcance y de las tasas de transmisión de datos

Cuadro comparativo de Bluetooth vs Wifi

Parte	Wifi	Bluetooth:
Ancho de banda	Alto	Laag
Requisito de hardware	Adaptador inalámbrico en todos los dispositivos dentro de la red y también un enrutador inalámbrico.	Adaptador Bluetooth en todos los dispositivos que conectamos entre nosotros.
Facilidad de uso	Es más complejo y también requiere la configuración de hardware y software.	Bastante fácil de usar y vincular dispositivos también es más fácil.
Alcance	metros 100	metros 10
Seguridad	Las funciones de seguridad son mejores. Aún existen algunos riesgos.	Menos seguro
El consumo de energía	Hoog	Laag
Rango de frecuencia	2.4 GHz y 5 GHz	2.400 GHz y 2.483 GHz
Flexibilidad	También es compatible con una amplia variedad de dispositivos.	El soporte de Interlink se limita solo a unos pocos dispositivos

Fuentes: <https://es.itpedia.nl/2018/07/12/wifi-en-bluetooth-wat-is-het-verschil/>

De este tema quedó encargado el equipo de LTI, conformado por Fernando Pérez y Lucas Galeano.

CIRCUITO DE SEGURIDAD

Se habló también de que el circuito autónomo de frenado del auto sea simplemente con dos sensores de ultrasonido y un algoritmo básico. Uno ultrasonido delante y otro detrás del vehículo.

También se analizó la posibilidad de que se diera FEEDBACK a la persona que maneja el vehículo, tanto si no puede avanzar más por un obstáculo o si se realiza marcha atrás se de una alarma.

MOVIMIENTO DEL AUTO

Se habló sobre la posibilidad de que el movimiento sea paso a paso, para dar tiempo de ver el auto y dar la orden.

La velocidad iba a ser si o si constante y maniobrable. Esto ya que implementar un algoritmo de control de la velocidad sin depender de un tercero y no utilizar otra parte del cuerpo iba a ser demasiado complicado.

El movimiento del auto puede ser:

1. Dos motores, uno para cada rueda trasera. Dos ruedas locas delante.
 - Cuando se quiera doblar hacia la derecha solo girara la rueda izquierda trasera.
 - Cuando se quiera doblar hacia la izquierda solo girara la rueda derecha trasera.
 - Si el movimiento es hacia adelante girarán ambas ruedas con el mismo torque.
 - Si se quiere ir hacia atrás ambas ruedas girarán en sentido contrario y mismo torque.

Los beneficios de realizarlo de esta forma es que es sencillo, fácil de implementar y más controlable para la situación que se nos demanda.

Se planteó la posibilidad de que las ruedas delanteras patinen en ciertos momentos, se investigará más.

2. Un motor para ambas ruedas traseras y un servomotor que cambie la dirección del auto.

Este método es más complicado de implementar, ya que se necesitan varias órdenes dependiendo el ángulo que se mueva el servomotor para doblar.

Si se llegara a implementar sería lo mejor, ya que da más libertad de movimiento y se acercará más a lo que realiza un vehículo en la realidad.

De la implementación e indagación de estas opciones se encargará el equipo de Mecatrónica, conformado por Lucas López, Tiziano Torterolo y Elias Hernandez.

CHARLA SOBRE EL CÓDIGO DADO EN EL TALLER 2

Para los compañeros que no tenían tanto conocimiento en programación, procesamiento de señales y espectro de la frecuencia; se repasó el código y vimos varias cosas que nos serían de utilidad para la etapa de procesamiento.

Códigos que nos iban a resultar de utilidad:

`plotEEG()`

`filterEEG()` (este utilizaba un pasa banda siempre, ya que así fue creado en Utils), investigamos el código con el que se obtenían los parámetro del filtro y se utilizaban aca.

`butter()`, muy útil si se quieren obtener los parámetros de un filtro Butterworth de cualquier tipo y orden.

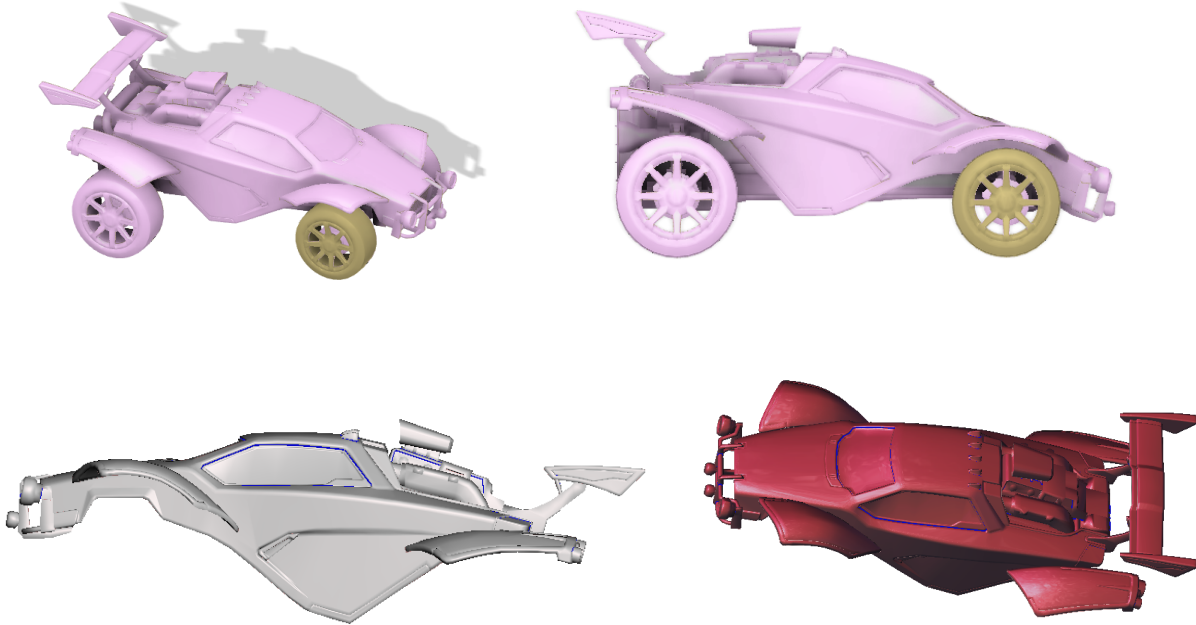
`getSpectrum` y `plotSpectrum`, hace la FFT en la señal de EEG. Muy útil ya que es de donde se va a extraer la información sobre los potenciales evocados para realizar los algoritmos de movimiento del vehículo.

También se creó un código por medio de un for que graficaba todos los EEG de los diferentes pacientes para cierto blanco y los filtraba. Se puede utilizar de base para la creación de otros algoritmos, como por ejemplo el que realiza la FFT o utilizarlo para solo una persona y varios blancos. Va a ser adjunto al repositorio de GitHub.

DISEÑO DE UN POSIBLE CHASIS PARA EL VEHÍCULO

Entre la charla se sugirió utilizar un chasis de un vehículo del juego Rocket League para el vehículo. Este iba a quedar mucho más estético.

Estábamos tan motivados que con el Software Meshmixer desmantelamos el sólido y aislamos solo el chasis en 3D



Conclusión

Logramos avanzar mucho en la conceptualización del trabajo y la idea a presentar.

Aunque vamos a trabajar siempre juntos (de manera transdisciplinaria) dividimos las tareas para avanzar más rápido dependiendo la carrera y los temas que pueden resultar más sencillos de implementar a ellas.

Plantemos las dudas para preguntarlas posteriormente a los coordinadores en una reunión.