



FACULTAT  
**DE CIÈNCIES  
I TECNOLOGIA**

UVIC | UVIC·UCC

Treball de Fi de Grau Experimental

---

Estudi i desenvolupament d'un sistema BCI  
basat en SSVEP

---

Aaron Verdaguer Gonzalez

**Grau en Enginyeria Biomèdica**

Tutor/a: Jordi Solé Casals

Ubicació: Vic, Juny de 2021

# Agraïments

Primerament agrair al meu tutor Jordi Solé Casals l'ajuda proporcionada durant el desenvolupament del projecte i els contactes posats al meu abast. Agrair a Fan, estudiant de la Universitat de Nankai, els consells atorgats. També agraeixo a TaranDeep Singh Mandahar la seva predisposició a treballar junts per solucionar problemes comuns. Finalment reconèixer a tota la meva família i amics el suport que han prestat durant el transcurs del treball de fi de grau.

# Resum

**Títol:** Estudi i desenvolupament d'un sistema BCI basat en SSVEP

**Autor:** Aaron Verdaguer Gonzalez

**Tutor:** Jordi Solé Casals

**Data:** Juny de 2021

**Paraules clau:** Fast Fourier Transform (FFT), Steady State Visual Evoked Potentials (SSVEP), Brain Computer Interface (BCI), hardware, software, OpenBCI, MatLab, electroencefalografia (EEG).

Brain Computer Interface, BCI, és un conjunt de software i hardware que permet interaccionar amb el medi utilitzant els senyals provinents de l'electroencefalografia, EEG. La creació d'un sistema BCI consta de 5 fases: adquisició dels senyal, preprocessament, extracció de característiques, classificació, i la interfície de control. Aquest projecte adreça les primeres 4 etapes. Per facilitar la creació del sistema BCI el projecte s'ha dividit en 2 fases: fase offline i fase en viu. En els dos processos l'adquisició de la senyal és realitzat amb el hardware i software proporcionats per OpenBCI i MatLab. L'extracció de característiques és realitzada mitjançant la Fast Fourier Transform, FFT, a través de MatLab.

La FFT permet transformar el senyal d'EEG del domini temporal al de la freqüència, detectant els Steady State Visual Evoked Potential, SSVEP, a l'EEG. La generació dels SSVEP requereix l'exposició d'un usuari a un estímul visual en una freqüència determinada. La generació dels estímuls també s'adreça en aquest projecte. La classificació s'ha realitzat buscant el pic més alt del vector característic del senyal, o FFT. La creació de la interfície de control no s'ha realitzat per manca de temps.

# Abstract

**Title:** Study and development of a BCI system based on SSVEP

**Author:** Aaron Verdaguer Gonzalez

**Supervisor:** Jordi Solé Casals (UVic)

**Date:** June 2021

**Keywords:** Fast Fourier Transform (FFT), Steady State Visual Evoked Potentials (SSVEP), Brain Computer Interface (BCI), hardware, software, OpenBCI, MatLab, electroencephalography (EEG).

Brain Computer Interface, BCI, is a mixture of software and hardware that allows interacting with the environment using only electroencephalography, EEG, signals. The development of a BCI system consists of 5 phases: signal acquisition, processing, feature extraction, classification, and control interface. This project addresses the first 4 phases. TO facilitate the creation of a BCI the project has been diovided into 2 parts: offline and online. Signal acquisition is done with the OpenBCI software and hardware and MatLab in both parts. Feature extraction is done with the Fast Fourier Transform, FFT, using MatLab.

FFT allows to transform an EEG signal from the time domain to the frequency domain, detecting Steady State Visual Evoked Potentials, SSVEP, in the EEG. The generation of SSVEP requires a user to be exposed to a visual stimulus with a defined frequency. The stimuli generation is done in this project. Classification has been done by finding the highest peak value in the feature vector, or FFT. An action interface has not been developed due to the lack of time.

# Índex de Continguts

1. Introducció.....	10
1.1 Electroencefalografia – EEG.....	11
1.2 Brain Computer Interface – BCI.....	13
1.3 Steady State Visual Evoked Potential – SSVEP.....	15
1.4 Software & Hardware.....	17
2. Objectius.....	20
3. Metodologia.....	22
3.1. Generació dels estímuls.....	23
3.2. Obtenció de les dades pel processament offline.....	24
3.3. Processament de les dades offline.....	26
3.4. Extracció de característiques.....	27
3.5. Classificador.....	28
3.6. Obtenció de les dades en viu.....	29
3.7. Proceessament de les dades en viu.....	30
3.8. Classificació de les dades en viu.....	30
4. Resultats.....	31
4.1. Classificació offline.....	31
4.2. Classificació en viu.....	34
5. Discussió.....	36
5.1. Limitacions.....	36
5.2. Possibles aplicacions i millores.....	37
6. Conclusió.....	38
7. Bibliografia.....	39
Annex: I.....	i
Annex: II.....	ii

Annex: III..... iv

# Llista de taules

<b>Taula 1.</b> Bandes de freqüència de l'EEG de lent a ràpid i la seva relació amb l'estat mental. (van Atteveldt et al., n.d.).....	13
<b>Taula 2.</b> Format de les sessions. *L'observador a de mirar l'estímul corresponent.....	24
<b>Taula 3.</b> Format del fitxer de sortida.....	25
<b>Taula 4.</b> Resultats de la classificació offline per cada finestra de 4s i precisió del classificador a cada data set.....	33
<b>Taula 5.</b> Etiquetes corresponents a cada finestra de 4s per tots els data sets.....	33
<b>Taula 6.</b> Resultats de la classificació en viu per cada finestra de 4s i precisió del classificador a cada intent.....	36

# Llista de figures

<b>Figura 1.</b> Nombre de publicacions BCI per any, obtingut mitjançant la cerca de "interfície d'ordinador brainc" a PubMed. (Kiret Dhindsa, 2017).....	10
<b>Figura 2.</b> Electrode situat al cuir cabellut col·locat per recollir les senyals de les sinapsis. actives (Basic Science of EEG   Epilepsy Foundation, n.d.).....	12
<b>Figura 3.</b> Percentatge de recerca per cada tòpic per l'ús de dades d'EEG. (Soufineyestani et al., 2020).....	13
<b>Figura 4.</b> Esquema d'etapes per la generació d'un BCI. (Brain-Computer Interfaces / Technologies / The MoreGrasp Project, n.d.).....	15
<b>Figura 5.</b> Il·lustració conceptual del paradigma SSVEP en els dominis de temps i freqüència. (Norcia et al., 2015).....	16
<b>Figura 6.</b> Casc d'EEG Ultracortex Mark IV a l'esquerre i placa Cyton a la dreta. (OpenBCI Ultracortex Mark IV Headset (Left); Cyton Biosensing Board (Right)   Download Scientific Diagram, n.d.).....	17
<b>Figura 7.</b> Posició dels elèctrodes al casc d'EEG Ultracortex Mark IV a l'esquerre i la seva representació amb el software d'OpenBCI. (Ultracortex Mark IV · OpenBCI Documentation, n.d.).....	18
<b>Figura 8.</b> Interfície de l'OpenBCI_GUI. (An Arduino-Based Brain-Computer Interface).....	18
<b>Figura 9.</b> Esquema de les etapes a seguir durant el projecte.....	22
<b>Figura 10.</b> Disposició de la pantalla creada per la generació de SSVEP al senyal d'EEG.....	23
<b>Figura 11.</b> Comprovació de la generació dels SSVEP als sets de dades de 8 Hz i el base. Es pot observar un pic al primer gràfic al voltant dels 8 Hz, indicant que s'ha generat una resposta a l'estímul.....	25

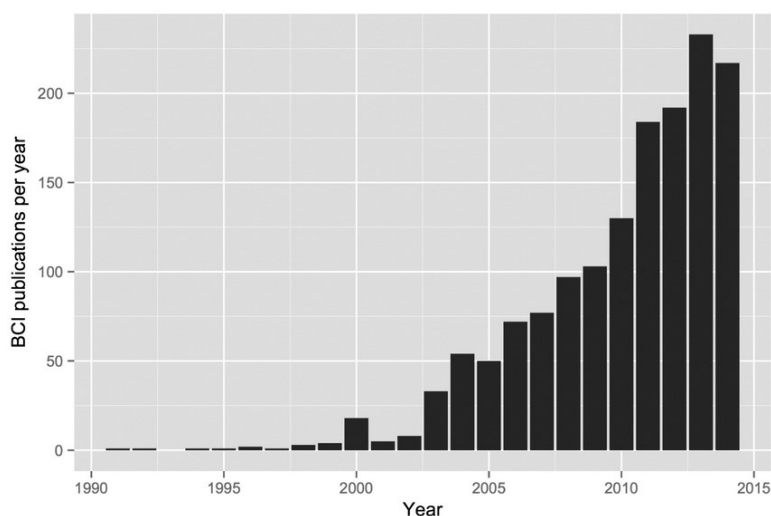


<b>Figura 12.</b> Filtre butterworth passa bandes entre 5 i 15 Hz d'ordre 4 (esquerra) i filtre butterworth elimina banda a 50 Hz (dreta).....	27
<b>Figura 13.</b> OpenBCI_GUI amb l'aplicació de xarxes al quadrant dret inferior. (Lab Streaming Layer (LSL) · OpenBCI Documentation, n.d.).....	29
<b>Figura 14.</b> Finestra de 4s del data set base previ al processament (dreta) i a posterior (esquerra).....	31
<b>Figura 15.</b> DFT de finestres estimulades i la corresponent localització dels SSVEP per a cada estímul.....	32
<b>Figura 16.</b> Exemple de localització d'un pic amb el classificador entre la freqüències pertanyent a la classe 8, estímul de 8.57 Hz.....	32
<b>Figura 17.</b> Finestra de 4s obtinguda en viu i processada.....	34
<b>Figura 18.</b> DFT d'una finestra de 4s obtinguda en viu.....	34

# 1. Introducció

En la societat en la que vivim no es pot imaginar un món on tothom tingui les mateixes possibilitats. Des de ben petits ens han ensenyat, malauradament, que no tothom té accés a educació, medicines, o fins i tot a l'aigua. Per aquest motiu d'una manera o altra algunes persones intentem que aquesta diferència d'oportunitats afecti el menys possible a les persones més desfavorides de la nostra societat. L'accés a matèries de primeres necessitats en la nostra comunitat no està tan limitat com en altres països, però per altra banda hi ha altres tipus de diferències entre els habitants. Un exemple podrien ser les persones amb mobilitat reduïda. Imagineu intentar baixar el volum del televisor, o canviar el canal sense poder moure els dits. Accions quotidianes que a la població sense cap mena de discapacitat no els hi requereixen cap mena d'esforç, per a les persones amb alguna discapacitat de mobilitat podria resultar en una depressió o altres malalties psicològiques a causa de la falta d'independència (Botero, 2013).

El motiu esmentat anteriorment, juntament amb la possibilitat de reduir al màxim les frustracions causades per falta de mobilitat o oportunitat, i les ganes de conèixer millor els senyals produïts pel cervell han estat la motivació per dur a terme aquest treball que consisteix en l'estudi i creació d'un sistema de Brain Computer Interface (BCI) emprant el paradigma SSVEP. En els últims anys es poden observar com les publicacions relacionades amb BCI van augmentant ja sigui per facilitar la vida d'alguns ciutadans, o per l'intent de tenir una millor comprensió del cervell humà.

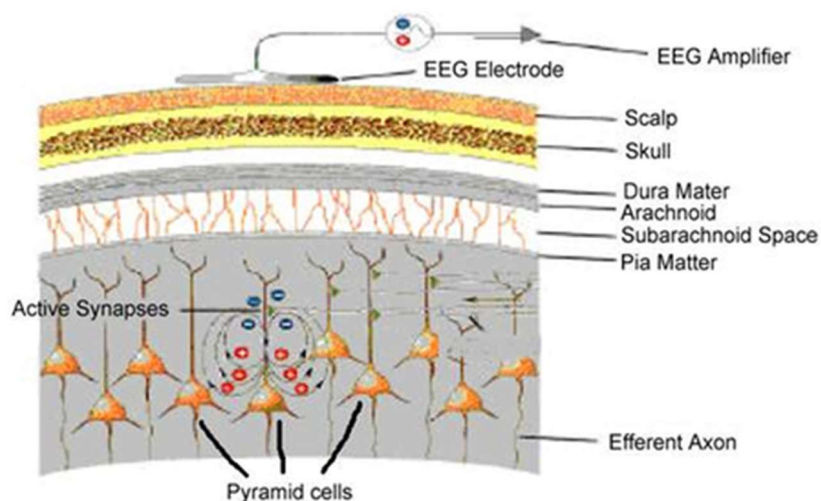


**Figura 1.** Nombre de publicacions BCI per any, obtingut mitjançant la cerca de "brain computer interface" a PubMed. (Kiret Dhindsa, 2017)

## 1.1 Encefalografia - EEG

Per tal de poder crear una interfície cervell-ordinador cal entendre d'on provenen i com es recullen els senyals produïts pel cervell per tal de ser analitzats posteriorment i generar una resposta basada en aquests. L'encefalografia (EEG) és la mesura no invasiva dels camps elèctrics del cervell mitjançant la col·locació d'elèctrodes en el cuir cabellut. Els elèctrodes recullen els potencials generats pels moviments d'ions a les neurones i al seu voltant. Cal remarcar que l'electroencefalografia és una tècnica antiga, amb gairebé un segle d'antiguitat fet que li ha permès tenir una gran varietat d'aplicacions en molts àmbits, com en rehabilitació, diagnòsica, o domèstica (Biasiucci et al., 2019). Un altre avantatge de l'EEG és la seva accessibilitat. Avui en dia gairebé tothom pot tenir accés a cascs que permeten recollir senyals d'EEG i a codi gratuït per tal de poder processar aquests senyals i interpretar-los, o fins i tot l'ús d'aquests per interaccionar d'alguna manera amb ordinadors.



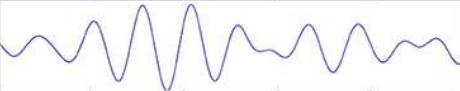

Per altra banda, conèixer quins són els senyals que generen aquesta encefalografia o com es capturen ajuda a obtenir un millor coneixement sobre com processar-los. Els senyals que estem analitzant no són els potencials d'acció de les neurones com a tal, sinó els potencials postsinàptics generats per l'activació de grups locals de neurones. Els potencials d'acció són un canvi ràpid d'ions que canvien el potencial intracel·lular de la neurona de  $-70$  mV a  $-55$  mV, incrementant el seu potencial fins a  $30$  mV, i que causen que aquesta generi una resposta (*Neuron Action Potentials: The Creation of a Brain Signal (Article) | Khan Academy, n.d.*). Aquests canvis són tan ràpids i abruptes que només es poden recollir amb alguns mètodes invasius. Per altra banda, aquesta resposta que generen les neurones alliberen neurotransmissors als axons per provocar l'activació de la següent neurona o grup de neurones. Aquesta alliberació genera un canvi de potencial a l'espai extracel·lular que sí que pot ser recollit per elèctrodes situats al cuir cabellut (Biasiucci et al., 2019).



**Figura 2.** Electrode situat al cuir cabellut col·locat per recollir les senyals de les sinapsis actives.  
*(Basic Science of EEG | Epilepsy Foundation, n.d.)*

La versatilitat i accessibilitat de l'EEG poden tenir conseqüències negatives respecte al seu ús. L'EEG permet mesurar activitat neuronal del cervell en temps real i caracteritzar una gran quantitat de vies neurofisiològiques com estats de consciència o son, disfuncions dinàmiques del cervell, o càrrega cognitiva (Biasiucci et al., 2019). Per altra banda aquest no deixa de ser una mesura de potencials elèctrics. Això genera una limitació de la informació rebuda i cal comprendre la diferència entre les àrees que generen una acció i les de referència. També cal esmentar que per culpa de la velocitat dels potencials en el cervell no podem assumir a la totalitat que els senyals recollits pels electrodes reflecteix el potencial exacte de la seva àrea en tot moment. Les millores en els aparells de mesura actuals permeten reduir cada vegada més els desavantatges que té l'EEG i això fa que actualment aquest tingui una gran varietat d'usos (Soufneyestani et al., 2020).

Moltes de les aplicacions actuals requereixen un coneixement previ dels senyals que formen l'EEG en el domini de la freqüència. És possible descompondre l'EEG en ones delta ( $\delta$ :  $\sim 0.2\text{--}3.5$  Hz), theta ( $\theta$ :  $\sim 4\text{--}7.5$  Hz), alpha ( $\alpha$ :  $\sim 8\text{--}13$  Hz), beta ( $\beta$ :  $\sim 14\text{--}30$  Hz), gamma ( $\gamma$ :  $\sim 30\text{--}90$  Hz), i altes freqüències ( $>90$  Hz). Exemples on aquests senyals es veuen alterats d'alguna manera podrien ser les mesures dels estats conscients com es veu a la taula 1, càrrega cognitiva, o l'ús d'activitat evocada. L'activitat evocada és observada en resposta a un esdeveniment o estímul, per exemple visual, i genera un increment d'amplitud dels senyals en una determinada banda de freqüència.

Frequency band	Speed (Hz)	Mental state	Electroencephalography (EEG) recording
Delta	1-4	Deep sleep	
Theta	4-8	Drowsy	
Alpha	8-12	Relaxed	
Beta	12-30	Focused	

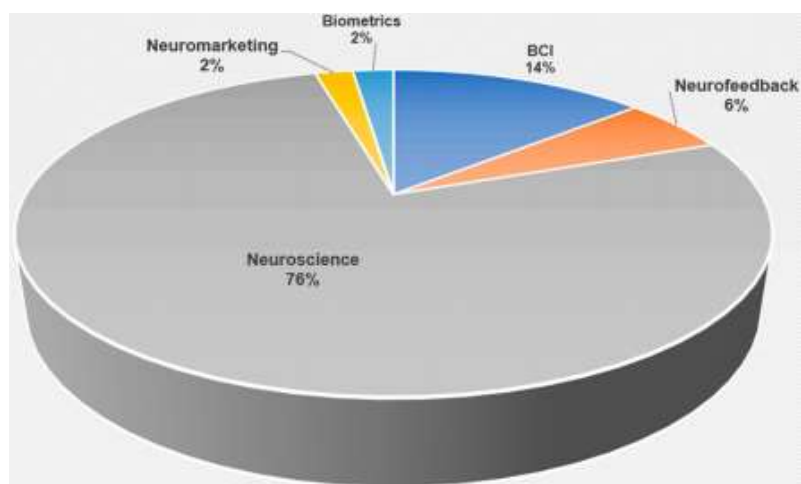
1 second

Slow  
↓  
Fast

**Taula 1.** Bandes de freqüència de l'EEG de lent a ràpid i la seva relació amb l'estat mental. (van Atteveldt et al., n.d.)

## 1.2 Brain Computer Interface - BCI

Actualment el tòpic més comú en l'ús d'EEG continua sent la neurociència, però podem veure un increment en l'ús d'aquesta eina per Brain Computer Interface (BCI) o per neuromàrqueting com mostra la figura 3. Per tal d'entendre el perquè està incrementant aquesta tendència cal definir que és un BCI o una interfície cervell-ordinador, en què consisteix i les seves possibles aplicacions en el món real.

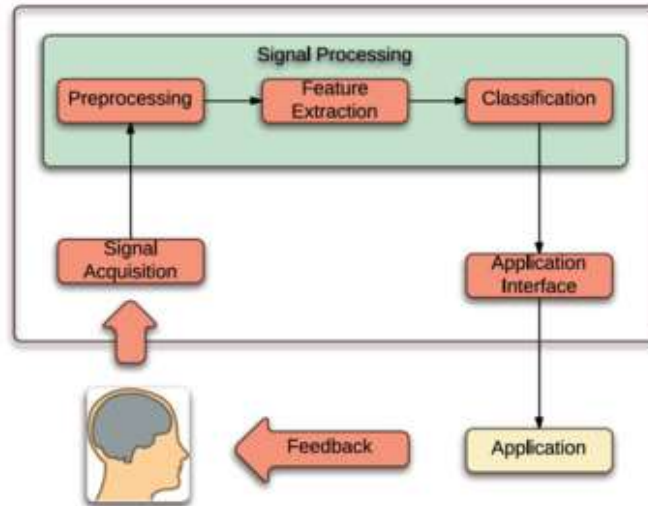


**Figura 3.** Percentatge de recerca per cada tòpic per l'ús de dades d'EEG. (Soufneyestani et al., 2020)

Una interfície cervell-ordinador o BCI és un conjunt de software i hardware comunicant-se que permet a l'usuari interaccionar amb el medi que l'envolta sense emprar nervis perifèrics o músculs utilitzant els senyals provinents de l'EEG. Les BCI generen un nou canal de comunicació entre les intencions d'una persona i un dispositiu extern com podrien ser ordinadors, generadors de veu, o pròtesis sense emprar músculs (Fernando Nicolas-Alonso & Gomez-Gil, 2012). Com s'ha esmentat anteriorment això és particularment atractiu per persones amb mobilitat reduïda perquè els hi proporciona una nova eina per tal de millorar la seva qualitat de vida, i al mateix temps reduir el cost de la vigilància intensa gràcies a permetre una major independència. Aquestes BCI poden reconèixer certs patrons a l'EEG en 5 etapes; adquisició dels senyal, preprocessament, extracció de característiques, classificació, i la interfície de control (Khalid et al., n.d.).

La primera etapa captura els senyals del cervell. També s'encarrega de fer una amplificació i redueix el soroll del senyal. L'etapa de preprocessament passa el senyal per un seguit de filtres i la prepara per a la següent etapa, agafant només la part desitjada del senyal. L'extracció de característiques genera un vector d'informació rellevant del senyal d'EEG que està sent tractat. Aquesta tasca és una de les més complexes perquè moltes vegades el senyal d'EEG està distorsionat per soroll. Aquest pot provenir de diverses fonts, com per exemple la xarxa elèctrica o altres senyals provinents del mateix usuari.

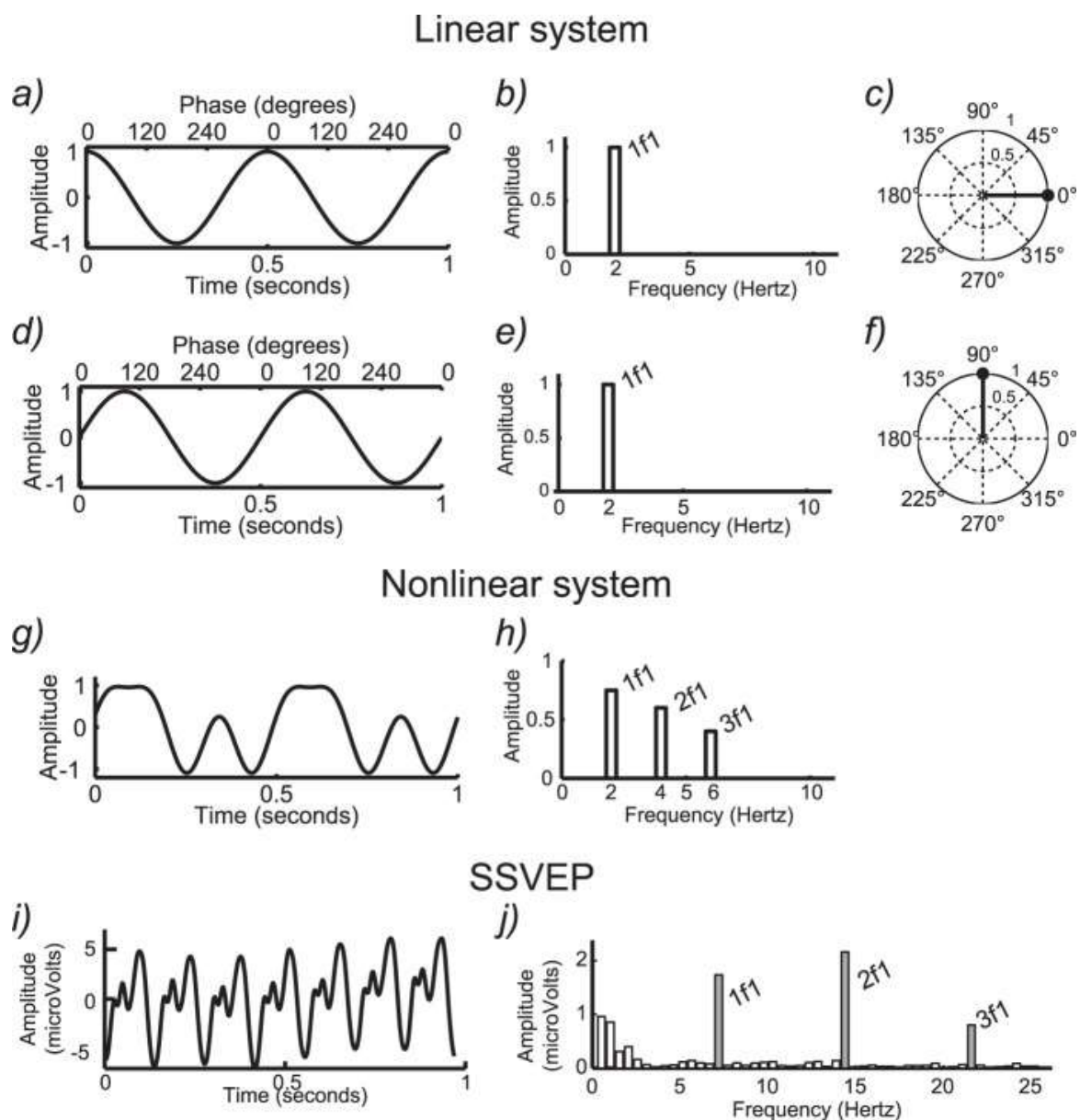
El vector característic extret no pot ser gaire extens, ja que aquest ha de processar-se en viu. A la mateixa hora ha de mantenir tota la informació rellevant de l'EEG (Fernando Nicolas-Alonso & Gomez-Gil, 2012). La classificació divideix els senyals que estan sent processades basant-se en el vector de característiques i genera una classe per cada moment en el temps. Finalment la interfície de control genera una comanda depenent de la classe rebuda. Aquestes comandes poden anar des de moure una cadira de rodes fins a moure el ratolí de l'ordinador.



**Figura 4.** Esquema d'etapes per la generació d'un BCI. (*Brain-Computer Interfaces / Technologies / The MoreGrasp Project*, n.d.)

### 1.3 Steady State Visual Evoked Potentials - SSVEP

Per comprendre el paradigma SSVEP primer hem d'entendre que és un Visual Evoked Potential (VEP). Els VEP són modulacions de l'activitat cerebral en rebre un estímul visual (Galloway, 1990). Aquestes alteracions són fàcils de detectar perquè incrementen l'amplitud del senyal sinusoidal que conforma l'EEG en la freqüència estímul com es pot veure a la figura 5.j. Els VEP són classificats basant-se en 3 criteris: per la morfologia de l'estímul visual, per la freqüència de l'estímul, i pel camp d'estimulació (Fernando Nicolas-Alonso & Gomez-Gil, 2012). El primer criteri defineix com és la morfologia de l'estímul. Aquest pot ser una llum pampalluguejant a una determinada freqüència o un seguit de patrons gràfics. Depenent de la freqüència d'estímul els VEP poden ser VEP transitoris (TVEP) on la freqüència està situada per sota dels 6 Hz o SSVEP si està per sobre (Galloway, 1990). Referint-nos a l'últim criteri divideix els estímuls en camp complet, mig camp, o camp parcial depenent d'on centri l'atenció l'usuari respecte a l'estímul. Com que les respostes SSVEP són periòdiques estan confinades a un conjunt de freqüències finit, múltiple de la freqüència a la qual correspon l'estímul. Per aquest motiu les respostes generades pel cervell a un determinat estímul són tractades en el domini de la freqüència. També cal remarcar que la zona del cervell que genera la resposta és la zona occipital (Galloway, 1990). A la figura 5 es poden observar diferents estímuls representats en el domini del temps i la corresponent representació en el de la freqüència. També es pot observar un exemple de SSEVP.



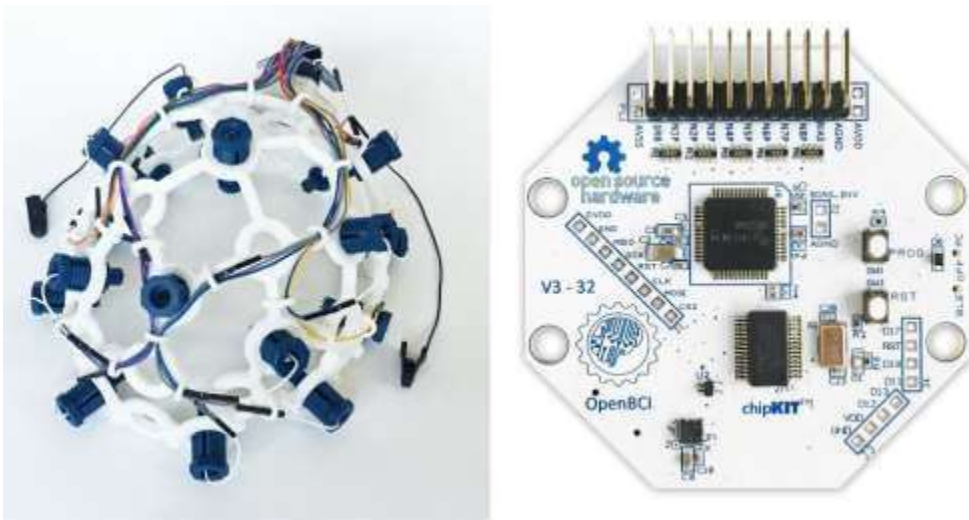
**Figura 5.** Il·lustració conceptual del paradigma SSVEP en els dominis de temps i freqüència. (a) Senyal sinusoidal com a resposta d'un sistema lineal en el temps. (b) Espectre corresponent del senyal (a). (c) Representació vectorial del senyal (a). (d) Resposta sinusoidal retardada  $90^\circ$  respecte al senyal (a). (e) L'espectre corresponent al senyal (d) és igual al del senyal (a) ja que la amplitud és igual. (f) Representació vectorial de (d). (g) Resposta formada per varis components d'un sistema no linier en el temps. (h) Ja que el senyal (g) està compost per varis senyals sinusoidals a diferents freqüències ( $1f$ ,  $2f$ ,  $3f$ ) la resposta en el domini de la freqüència mostra varis pics ( $1f$ ,  $2f$ ,  $3f$ ). (i) SSVEP representat en domini temporal en una freqüència de  $7.2\text{ Hz}$  ( $f_1$ ). (j) La resposta corresponent en el domini de la freqüència genera 3 pics múltiples de  $f_1$  ( $1f_1$ ,  $2f_1$ ,  $3f_1$ ).

(Norcia et al., 2015)



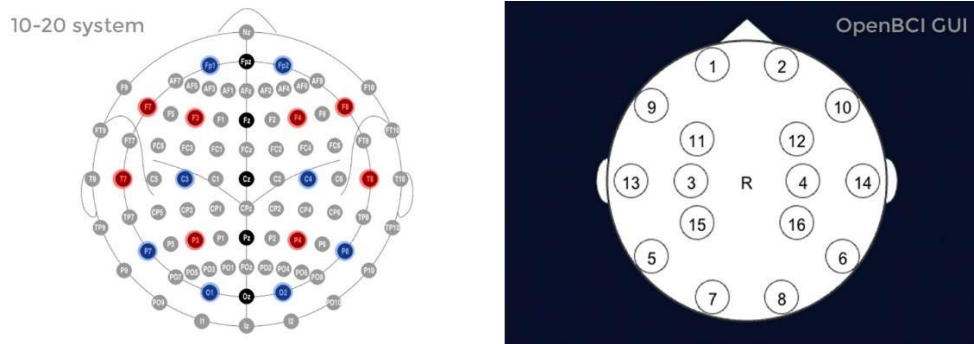
## 1.4 Software & Hardware

Per crear una BCI cal disposar d'un aparell per recollir les dades (hardware) i d'un conjunt de programes pel seu posterior processament (software). Per tal de crear una BCI primer s'ha de generar un classificador que funcioni amb dades recollides prèviament o de manera offline. Un cop s'ha aconseguit generar un classificador fiable pel set de dades guardat, aquest ja es pot provar en viu o de manera online. En el cas d'aquest projecte s'utilitza sempre el mateix hardware i software per l'obtenció de dades i el seu processament. El hardware emprat per aquest projecte és l'Ultracortex Mark IV i la placa Cyton de la companyia OpenBCI mostrats a la figura 6.



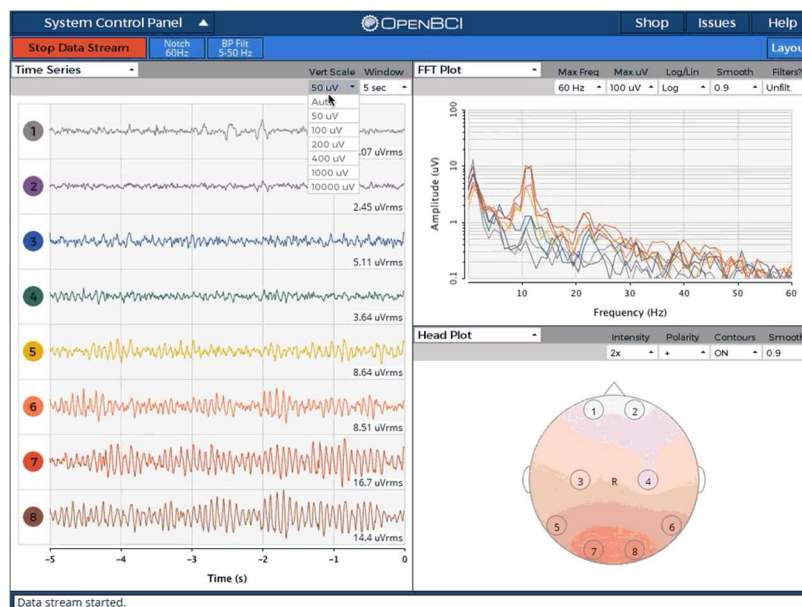
**Figura 6.** Casc d'EEG Ultracortex Mark IV a l'esquerre i placa Cyton a la dreta. (*OpenBCI Ultracortex Mark IV Headset (Left); Cyton Biosensing Board (Right) | Download Scientific Diagram, n.d.*)

Aquest casc d'EEG té molts avantatges respecte a altres cascs disponibles al mercat. Les més importants són el cost, ja que és molt econòmic comparat amb altres cascs del mercat, l'accessibilitat a informació, codi, i software proporcionat per OpenBCI i la seva comunitat, i la seva facilitat de muntatge i versatilitat d'ús (*Ultracortex Mark IV · OpenBCI Documentation, n.d.*). Tot l'accés a codi i informació és gratuït. Ja que els SSVEP generen la resposta a la zona occipital del cervell només seran estudiades les dades provinents dels elèctrodes situats en aquesta zona, que com es pot veure a la figura 7 són els elèctrodes O1 i O2 (7 i 8). Aquests dos elèctrodes recullen les modulacions en l'EEG. Posteriorment seran detectades pel classificador. La placa Cyton només permet la interpretació d'un sistema d'EEG de 8 canals, però com només es precisen els dos elèctrodes occipitals no és un factor limitant pel projecte.



**Figura 7.** Posició dels elèctrodes al cas d'EEG Ultracortex Mark IV a l'esquerra i la seva representació amb el software d'OpenBCI. (*Ultracortex Mark IV · OpenBCI Documentation, n.d.*)

El software emprat per crear el data set per generar el classificador i trobar el processament correcte ha estat el software proporcionat per OpenBCI. És anomenat OpenBCI\_GUI. Aquesta interfície permet veure molta informació important en viu i guardar les dades recollides en les sessions d'estudi. A la figura 8 podem veure com és aquesta interfície. Les 3 característiques que importen per saber si el senyal obtingut és el desitjada són; la gràfica en funció del temps dels canals desitjats en una finestra de 5 segons, la representació de la mateixa finestra en el domini de la freqüència, i la representació de la intensitat del senyal al cap a cada elèctrode. El software també proporciona altres aplicacions per diverses funcions, com la creació de Lab Streaming Layers.



**Figura 8.** Interfície de l'OpenBCI\_GUI. (*An Arduino-Based Brain-Computer Interface - Popular Electronics, n.d.*)

Un cop assegurat que les dades recollides han estat correctes amb l'OpenBCI\_GUI sortim de la sessió. Les dades són guardades sense cap processament excepte l'amplificació del senyal i la resta de la mitja dels elèctrodes de referència col·locats a les orelles (Lopez-Calderon & Luck, 2014). Malauradament la versió actual del software no permet el tractament de les dades en viu. Un cop obtingut el fitxer que guarda les dades en format CSV aquest pot ser processat per obtenir diversos resultats. Per tal de tenir un control total de les dades, entendre per quins processos ha de passar l'EEG per tal de ser visualitzat, i posteriorment extreure característiques d'aquest s'utilitzarà MatLab 2020a (*MATLAB - El Lenguaje Del Cálculo Técnico - MATLAB & Simulink*, n.d.). Actualment existeixen molts programes creats per MatLab com podrien ser EEGLab o BCILab per tal de visualitzar les dades de manera offline (EEGLab i BCILab) i online (BCILab), però al venir predefinides algunes etapes del processament de dades no és possible de tenir tot el control sobre les dades.

MatLab ofereix un llenguatge intuïtiu i versàtil. MatLab proporciona funcions predefinides per a una àmplia varietat de processos, des de la creació d'estímuls amb PsychToolBox (PTB) fins a tractament i processament de senyals. Aquesta última llibreria de funcions de MatLab ens permet crear filtres digitals, filtrar dades, o transformar al domini de la freqüència entre altres. La diferència entre processar les dades amb MatLab des de zero o emprar un software com BCILab és que amb MatLab podem programar un camí de processament i veure com afecta cada pas a les dades i l'impacte que aquest sobre el set de dades. Per altra banda, BCILab proporciona una interfície que amb un clic i escollint uns quants paràmetres et permet visualitzar l'EEG. Respecte a la creació dels estímuls PTB és la llibreria que s'ha emprat perquè permet crear de zero i dóna un control total de la pantalla on apareixen els estímuls.

MatLab permet fer el processament de les dades d'una manera eficaç, però de moment només s'ha adreçat el tòpic offline. Per tal de poder processar dades en viu a MatLab només hi ha una opció perquè OpenBCI no ha desenvolupat cap llibreria o aplicacions en MatLab per llegir les dades procedents del casc pel port sèrie. La solució és utilitzar un canal Lab Streaming Layer (LSL) que serà rebut per una petició d'un programa a MatLab. Per tal de crear la LSL i enviar les dades a MatLab cal tenir la interfície OpenBCI\_GUI i utilitzar el repositori de GitHub que proporciona OpenBCI (*Lab Streaming Layer (LSL) · OpenBCI Documentation*, n.d.). Un cop rebudes, les dades són processades amb MatLab.

## 2. Objectius

Aquest és el primer projecte intentant crear una BCI de principi a fi a la Universitat de Vic amb el casc OpenBCI. Per aquest motiu hi ha moltes problemàtiques a l'hora de generar objectius. Hi ha molts objectius del projecte que tenen a veure amb la comprovació i verificació de l'adquisició de senyals i la correcta generació dels estímuls pels SSVEP a l'EEG. Per fer una BCI cal passar per 5 etapes ben definides: adquisició del senyal, preprocessament, extracció de característiques, classificació, i la interfície de control (Khalid et al., n.d.). Els objectius estan definits per aquestes. També venen determinats per la diferència entre el processament online i offline. Les dades primer s'han de tractar offline i posteriorment emprar les eines obtingudes d'aquest processament per aplicar-les en viu. Al començar sense cap treball previ amb aquest hardware, la cerca d'informació per resoldre problemes que han anat sorgint durant el projecte ha estat tediosa. Els propòsits del projecte són:

- **Utilitzar el casc Ultracortex Mark IV i familiaritzar-se amb el software OpenBCI\_GUI.** Aquesta etapa és necessària per facilitar l'èxit dels següents objectius.
- **Generar els estímuls a les freqüències desitjades que són: 6.67 Hz, 7.5 Hz, 8.57 Hz, 10 Hz amb PsychToolBox de MatLab.** Per comprovar que els estímuls estiguin ben generats s'haurà de comprovar amb una sessió de prova.
- **Generar un data set per cada una de les freqüències amb el casc d'EEG.** Només serà necessari recollir els senyals dels elèctrodes desitjats: O1 i O2. Un cop assegurat que els estímuls generen SSVEP en el senyal d'EEG es realitzen 5 sessions on es recullen 40s per a cada estímulo i la base:
  - La primera sessió recull una senyal base on l'usuari ha de romandre mirant a una pantalla en negre durant 40 s.
  - Les següent sessió consisteix en recollir les dades de l'usuari mirant a l'estímul de 6.67 Hz en períodes alterns. L'usuari mira al centre de la pantalla negra durant 8s i 8s a l'estímul. Aquest procés es repeteix per les

frequències de 7.5 Hz, 8.57 Hz, i 10 Hz, generant un total de 5 sessions amb la base.

- **Dividir els data sets anteriors en finestres de 4s i visualització d'aquestes.** En aquesta etapa ja es defineix un camí de processament per les dades per a la seva visualització de manera offline. En aquest punt hi ha 5 sets de dades de 40s dividits en finestres de 4s, amb una superposició de 3s, i amb dos elèctrodes per set.
- **Generar un vector de característiques per les finestres del senyal base i de les modulades pels estímuls.** Per aquesta etapa s'apliquen filtres digitals per reduir el soroll del senyal i la transformada ràpida de Fourier (FFT) a les dades. S'analitza el senyal base per obtenir un vector de característiques. Aquest vector s'utilitza com a referència pels altres vectors característics dels EEG amb els SSVEP. Després s'etiqueta manualment cada finestra dels conjunts de dades finals depenent del moment de cada sessió al que pertanyen.
- **Definir un classificador que distingeixi entre les finestres de 4s depenent de si l'usuari estava mirant a un estímulo.** La creació del classificador consistirà en analitzar el vector de característiques de les finestres de 4s obtingut mitjançant la FFT. Depenent de la amplitud del senyal en les freqüències desitjades el classificador ha de donar una classe com ha resultat que ha de ser 0 quan l'usuari mirava al centre de la pantalla, o 6, 7, 8, i 10 quan l'usuari mirava als estímuls corresponents.
- **Comprovar la precisió del classificador.** Durant aquest procés cal tenir en compte que al haver-hi una superposició entre les finestres de 3s és possible que la precisió es vegi afectada ja que hi pot haver vectors característics de moments en que l'usuari miri 2s al centre de la pantalla i 2s a l'estímul.
- **Generar un canal de comunicació LSL.** Un cop el classificador ha donat resultats raonables es prova amb senyals en viu. Abans cal generar una via de comunicació estable entre el casc d'OpenBCI i MatLab. S'utilitza l'OpenBCI\_GUI per inicialitzar una LSL i recollir les dades amb MatLab.
- **Comprovació del senyal rebut per LSL.** En aquest procés es guarda el senyal rebuda per LSL en finestres de 4s que consisteixen en 4s mirant al centre de la

pantalla i 8s mirant a l'estímul de 7.5 Hz. Es repeteix pels altres estímuls. Un cop guardats els senyals procedents dels elèctrodes O1 i O2 s'aplica el mateix procediment que als sets de dades previs.

- **Generar la nostra BCI amb el nostre classificador.** Es defineix un camí de processament de les dades en finestres de 4s per les senyals rebudes per LSL. El classificador obté la classe a la qual pertany aquesta finestra.

### 3. Metodologia

Sent aquest el primer projecte BCI a la Universitat de Vic cal deixar reflectit d'una manera clara el procediment seguit per a dur a terme les primeres 4 etapes per aconseguir crear una BCI. Com s'ha esmentat, aquestes són: adquisició del senyal, preprocessament, extracció de característiques, i classificació. Facilitant el desenvolupament de futurs projectes relacionats. Respecte l'última etapa, generació d'una interfície d'acció, es proporcionen idees d'implementació i de possibles col·laboracions amb altres projectes.

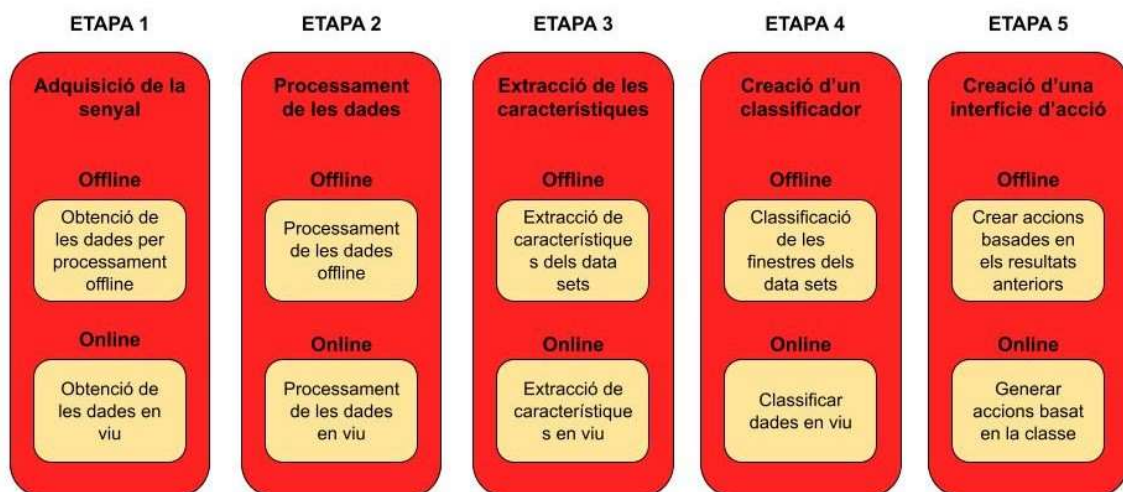
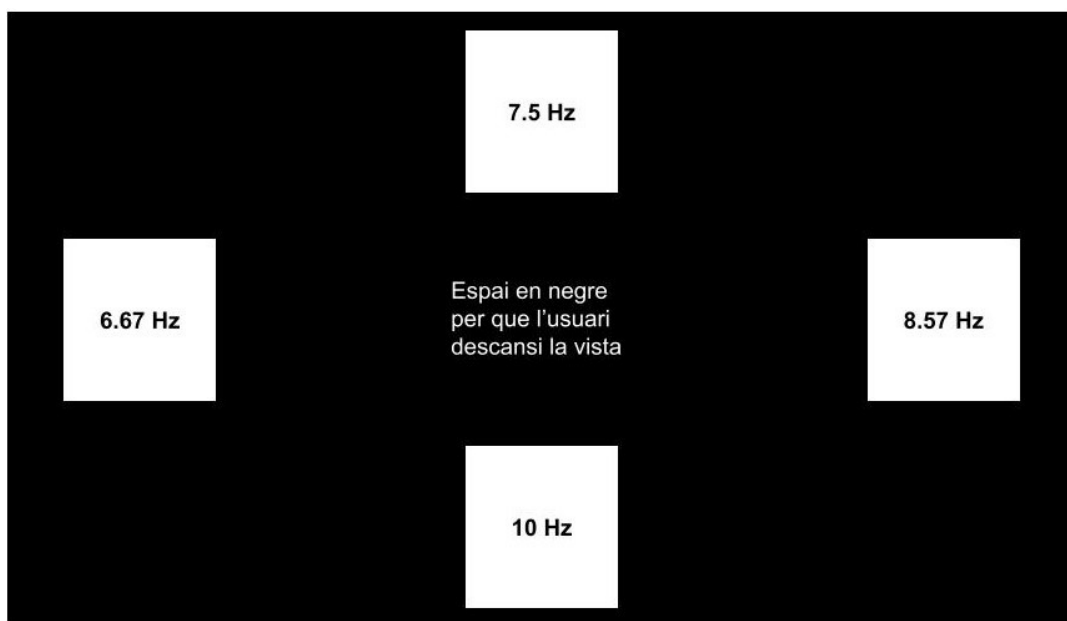


Figura 9. Esquema de les etapes a seguir durant el projecte.

### 3.1 Generació dels estímuls

L'etapa de generació d'estímuls ha de ser explicada per facilitar futurs estudis. En aquest projecte s'ha emprat PsychToolBox (PTB) de MatLab per a la creació dels estímuls pel control que atorga al programador sobre la pantalla. PTB és un conjunt de funcions de Matlab emprades per a la recerca en els camps de la visió i neurociència. PTB permet generar d'una manera fàcil estímuls visuals o auditius i tenir un control molt acurat d'aquests (*Psychtoolbox-3 - Overview*, n.d.). Aquesta llibreria està disponible de manera gratuïta a la pàgina de PTB, on es troba un manual de descàrrega (*Psychtoolbox-3 - Download, Installation, and Update*, n.d.).

Un cop completats tots els passos indicats a la web apareix una carpeta descarregada amb exemples comentats de com fer servir el programa i totes les funcions que permeten que PTB funcioni. Un cop descarregat el software i havent-se familiaritzat amb ell es comença a programar per tal d'aconseguir generar una pantalla amb els estímuls visual desitjats. La intenció és aconseguir tenir quatre quadrats pampalluguejant als cantons de la pantalla, i deixar suficient espai negre en el mig perquè l'observador pugui redirigir la seva atenció cap aquest.



**Figura 10.** Disposició de la pantalla creada per la generació de SSVEP al senyal d'EEG.



### 3.2 Obtenció de les dades pel processament offline

Un cop familiaritzat amb el casc, el corresponent software, i generats els estímuls es realitzaran les 5 sessions per a la recollida de dades. Aquestes dades seran analitzades per comprovar que els SSVEP han estat generats correctament posteriorment. Per a la realització de les sessions es seguiran les instruccions proporcionades per OpenBCI (*Cyton Getting Started Guide · OpenBCI Documentation*, n.d.).

<b>Etapes</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>Total</b>
<b>Sessió Base</b>	2s	8s	8s*	8s	8s*	8s	2s	44s
<b>Sessió 7 Hz</b>	2s	8s	8s*	8s	8s*	8s	2s	44s
<b>Sessió 8 Hz</b>	2s	8s	8s*	8s	8s*	8s	2s	44s
<b>Sessió 9 Hz</b>	2s	8s	8s*	8s	8s*	8s	2s	44s
<b>Sessió 10 Hz</b>	2s	8s	8s*	8s	8s*	8s	2s	44s

**Taula 2.** Format de les sessions. \*L'observador a de mirar l'estímul corresponent.

Un cop finalitzades les sessions les dades d'EEG es guarden a l'ordinador de manera automàtica en format CSV. La placa Cyton consta de 8 canals per a l'adquisició de senyals d'EEG, però també recull altres paràmetres calculats amb l'OpenBCI\_GUI. Alguns exemples d'aquests paràmetres són l'acceleració, o l'estampa de temps (*The OpenBCI GUI · OpenBCI Documentation*, n.d.). El format del fitxer de sortida és representat a la taula 2. Les primeres 4 línies contenen la informació del títol de les dades, la freqüència de mostreig (Fs), 250 Hz, el número de canals, 8, i la placa emprada, Cyton.

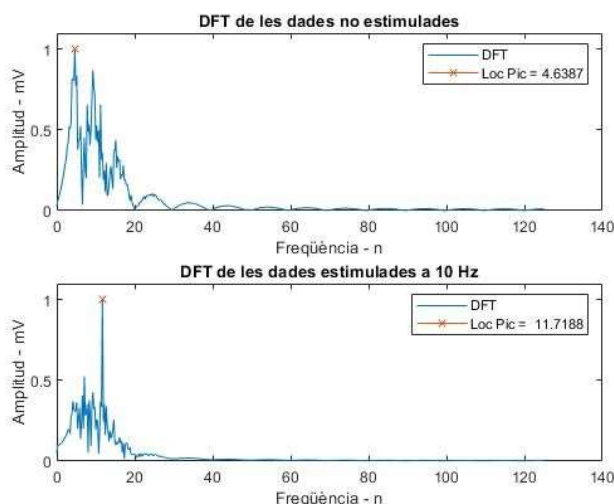


Sample Index	EXG Channel 0	...	EXG Channel 7	Accel Channel 1/3	Other	...	Other	Analog Channel 1/3	Timestamp	Timestamp (Formatted)
0.0	1161149.8125	...	967837.125	0.1900000125169754	192.0	...	208.0	0.0	1.621437249410129E9	2021-05-19 17:14:09.410
1.0	1142916.75	...	848481.46875	0.1900000125169754	192.0	...	0.0	0.0	1.6214372494103124E9	2021-05-19 17:14:09.410
2.0	1115564.0625	...	764611.5	0.1900000125169754	192.0	...	0.0	0.0	1.6214372494103837E9	2021-05-19 17:14:09.410
3.0	1117069.875	...	847441.875	0.1900000125169754	192.0	...	0.0	0.0	1.6214372494104545E9	2021-05-19 17:14:09.410

**Taula 3.** Format del fitxer de sortida.

A l'analitzar el document ens adonem que les 4 primeres línies poden ser saltades en llegir el fitxer. La informació rellevant per a la detecció dels SSVEP al senyal d'EEG està compresa a la zona occipital. En el nostre casc només tenim dos elèctrodes en aquesta part, O1 i O2. Aquests dos canals estan situats a les columnes 8 i 9 que corresponen als EXG Channel 6 i EXG Channel 7. Totes les altres columnes contenen informació no rellevant per a la detecció d'EEG. A l'Annex: I es troba adjuntat el codi de la funció ReadData.m.

Aquesta funció proporciona una matriu de sortida de  $N \times 2$  on  $N$  és igual número de mostres i el número de canals que contenen el senyal d'EEG dels elèctrodes O1 i O2. Tenint en compte que les sessions són de 40 segons i que la  $F_s$  és 250 Hz, s'espera tenir una matriu de 10 000 x 2 aproximadament. Per assegurar tenir els 40 s rellevants de la sessió dintre de les dades en començar i acabar la sessió deixem 2 s on continuem rebent dades. Aquests dos segons inicials i finals són extrets de les dades dins de la mateixa funció. Un cop s'extreuen les dades podem fer una anàlisi espectral per comprovar que els SSVEP són presents a l'EEG com es pot observar a la figura 12.



**Figura 11.** Comprovació de la generació dels SSVEP als sets de dades de 10 Hz i el base. Es pot observar un pic al segon gràfic a 11.7188 Hz, indicant que s'ha generat una resposta a l'estímul.

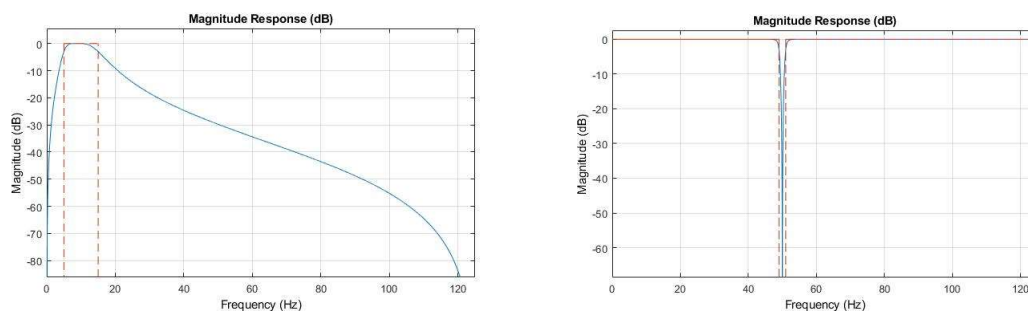
### 3.3 Processament de les dades offline

Les matrius de dades contenen els 40 s de les 5 sessions generades. Els primers segons i últims no estan perquè no contenen informació important. El processament de les dades tindrà tres parts: divisió de les sessions en finestres de 4 segons, creació de filtres digitals per a extreure la part rellevant, i posterior ús (Puce & Hämäläinen, 2017). S'ha escollit finestres de 4s perquè en finestres de menys segons és possible que no es vegin reflectides les modulacions a l'EEG generades pels estímuls (Norcia et al., 2015).

Per dividir les dades en finestres de 4 s s'utilitza la funció `DividirData.m`. El codi de la funció està a l'Annex: I. El format de sortida de la funció és una matriu que conté  $1000 \times M$  valors.  $M$  és el número de finestres de 4 s amb una superposició de 3 s multiplicat per 2, ja que a la primera columna hi ha els valors de l'elèctrode O1 i la segona la de l'elèctrode O2. Les dades d'entrada contenen les primeres 10 000 mostres per assegurar la mateixa longitud dels sets de dades, tenint en compte que les últimes no contindran cap SSVEP a l'EEG pel disseny de les sessions. Coneixent aquesta longitud es dedueix que la matriu de sortida serà de  $1000 \times 74$  (37 finestres de 4s dels elèctrodes O1 i O2.).

Obtinguts els sets de dades es generen filtres digitals en el domini de la freqüència per tal d'obtenir com a resultat la part del senyal desitjada. Ja que les freqüències d'estímul han estat 6.67, 7.5, 8.57, i 10 Hz només ens caldria interpretar el rang de freqüències de 6.5 a 10 Hz. Però depenent de la persona que observa els SSVEP, aquests poden mostrar-se a freqüències pròximes. Per exemple l'estímul de 10 Hz podria generar un SSVEP a la freqüència 10.75 Hz (Norcia et al., 2015). Per aquest motiu l'interval de freqüències que volem conservar del senyal original és de 5 a 15 Hz. Encara que el rang de freqüències a analitzar està per sota de 50 Hz és necessari treure el soroll corresponent a la xarxa elèctrica del senyal d'EEG situat a 50 Hz a Europa. A altres països aquest filtre ha de ser a la banda de 60 Hz. Per aquest motiu cal un filtre passa bandes d'entre 6 Hz i 15 Hz i un filtre elimina banda a 50 Hz.. Aquests filtres han de ser dissenyats com a Infinit Impulse Response (IIR) ja que són millor alhora de preservar l'amplitud del senyal. Els filtres IIR utilitzen molt poca capacitat computacional de l'ordinador i memòria, per aquest motiu son emprats en el processament en viu de dades (Oshana, 2006).

Per altra banda, els filtres tipus Finit Response Impulse podria ser l'opció a escollir si no fos per què estan molt limitats per la velocitat d'aplicació. Per aquest motiu no solen ser utilitzats quan hi pot haver processament en viu (Oshana, 2006). Reduint també l'ordre dels filtres i l'habitual nombre de filtres emprats pel filtratge d'EEG de 3 a 2 (Puce & Hämäläinen, 2017), es redueix molt la capacitat funcional que requereix l'ordinador processant les dades. Hi ha diversos tipus de filtres digitals passa bandes, però el que produeix la resposta més estable és el butterworth (*Diseñar Filtros Digitales - MATLAB Designfilt*, n.d.).



**Figura 12.** Filtre butterworth passa bandes entre 5 i 15 Hz d'ordre 4 (esquerra) i filtre butterworth elimina banda a 50 Hz (dreta).

### 3.4 Extracció de característiques

El mètode més emprat en l'extracció de característiques per SSVEP és la Fast Fourier Transform, FFT. Aquesta FFT computa la Discrete Fourier Transform, DFT, de la seqüència de dades i per tant converteix el senyal del domini temporal al freqüencial. Els SSVEP es veuen reflectides en el domini de la freqüència després d'aplicar la FFT (Vargic, n.d.). La funció ExtCaract.m adjuntada a l'Annex: I obté la FFT de les finestres de 4 s que se li enviïn. Per a la localització dels SSVEP a la DFT del senyal només cal observar l'amplitud de les mateixes freqüències a la que és estimulat l'observador. Un cop extreta la DFT és normalitzada per què els valors d'amplitud de les finestres de 4 s estiguin dintre del rang de 0 a 1.

Per a una optimització i precisió més gran de la FFT amb MatLab, cal tenir en compte el zero padding i que la funció de MatLab pel càlcul de la FFT funciona millor amb potències del número 2. El zero padding consisteix a afegir 0 al final del senyal per tenir més mostres i incrementar el nombre de grups emprats per al càlcul de la FFT, incrementant d'aquesta manera la resolució del senyal en el domini de la freqüència (Peng-Wie Hsia et al., n.d.). Per optimitzar el rendiment del codi el zero padding serà utilitzat fins a generar un vector d'entrada a la funció de la FFT amb una mida de 1024, ja que sabem que el nostre vector té originàriament 1 000 mostres, 250 per segon.

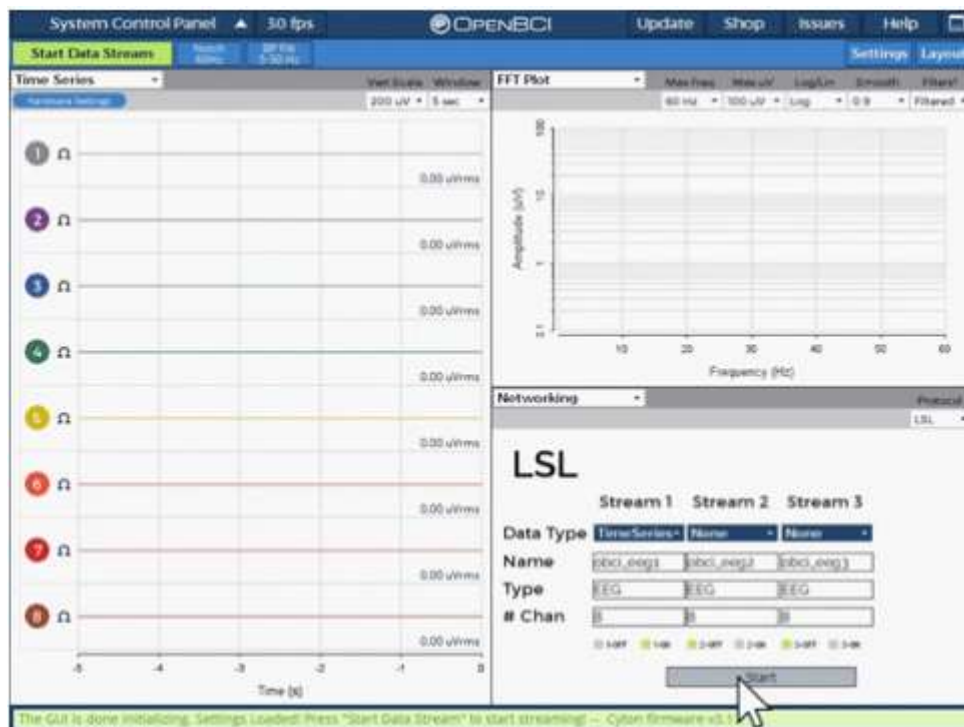
### 3.5 Classificador

Una mateixa persona sempre que és exposada a un estímul de la mateixa freqüència genera un pic com a resposta al seu EEG al domini transformat. Per aquest motiu el nostre classificador es basarà només a determinar la freqüència en la qual trobem l'estímul generat. Si l'estímul està dintre d'uns dels rangs especificats, se li assignarà una classe. Això comporta un inconvenient, cada cop que canvia l'usuari s'han de canviar les freqüències a observar del senyal d'EEG. Per tal d'obtenir una classe hem de donar un rang de freqüències diferent per a cada classe dependent de l'observador de l'estímul. El codi amb tot el procediment està situat a l'Annex: II.

Diferents persones mirant a l'estímul de 8.57 Hz poden generar SSVEP amb diferències de més d'1 Hz (Norcia et al., 2015; Puce & Hämäläinen, 2017). Per tal d'obtenir el rang adequat s'ha de realitzar un estudi previ a l'usuari del cas per tal de localitzar-lo. En cas d'obviar el cas previ i simplement determinar un rang de freqüències estàndard, per exemple de 8 a 9 Hz per a la classe que pertany a l'estímul de 8.57 Hz, la precisió del nostre classificador es veuria bastant afectada (Vargic, n.d.). La sortida del classificador és 0, 6, 7, 8, o 10 on 0 vol dir que cap SSVEP és generat i 7, 8, 9, 10 que concorden amb els corresponents estímuls. El codi del classificador està situat a l'Annex: I sota el nom Classificador.m.

### 3.6 Obtenció de les dades en viu

Open BCI no proporciona una connexió directa entre MatLab i la placa Cyton. El que sí que ofereix és comunicació via LSL. El mètode utilitzat per generar una LSL és un dels proporcionats per OpenBCI (*Lab Streaming Layer (LSL) · OpenBCI Documentation*, n.d.). Aquest requereix només l'OpenBCI\_GUI i MatLab. El software proporcionat té diferents aplicacions. Entre elles es troben la visualització de les dades en viu, la representació de les dades en el domini del temps, o l'aplicació de xarxes (*GUI Widget Guide · OpenBCI Documentation*, n.d.). Per a generar el canal LSL només cal seleccionar l'aplicació de treball en xarxes anomenada networking a l'OpenBCI\_GUI, escollir LSL com a mètode de comunicació i començar la sessió.



**Figura 13.** OpenBCI\_GUI amb l'aplicació de xarxes al quadrant dret inferior. (*Lab Streaming Layer (LSL) · OpenBCI Documentation*, n.d.)

El format a enviar de les dades pot ser tant l'EEG, com la DFT, o l'acceleració XYZ. Un cop iniciada la sessió les dades són enviades per LSL esperant ser rebudes. OpenBCI proporciona una llibreria de GitHub que ofereix exemples de com rebre les dades a MatLab. El codi utilitzat és el que ells proporcionen al repositori de GitHub (*GitHub - Labstreaminglayer/Liblsl-Matlab: Matlab Bindings for Liblsl*, n.d.).

### **3.7 Processament de les dades en viu**

Pel processament de les dades en viu s'utilitza només un dels filtres utilitzats a l'apartat de processament offline. Aquest serà el filtre de 5 Hz a 15 Hz. Les dades rebudes via LSL a MatLab ja venen amb un preprocessament previ. També poden ser rebudes en cru. El tractament previ consisteix en un filtre elimina banda a 50 Hz i un altre filtre passa bandes a l'elecció de l'usuari. Les opcions de les quals es disposen són: 1-100 Hz, 5-50 Hz, o 7-13 Hz. El filtre 7-13 Hz seria una bona opció si no fos perquè una de les freqüències estímul és 6.67 Hz. El codi per al processament de les dades està contingut al programa principal de MatLab, situat a l'Annex: III.

Les dades començaran a ser processades quan hagin passat els primers 4 segons. Després d'aquest temps es processaran 4 segons per cada segon que avanci, és a dir, 3 segons de superposició. Cada finestra de 4 segons és tractada mitjançant la funció emprada en el processament offline. Aquesta realitza la FFT de les dades d'EEG amb zero padding per obtenir un vector característic o DFT.

### **3.8 Classificador en viu**

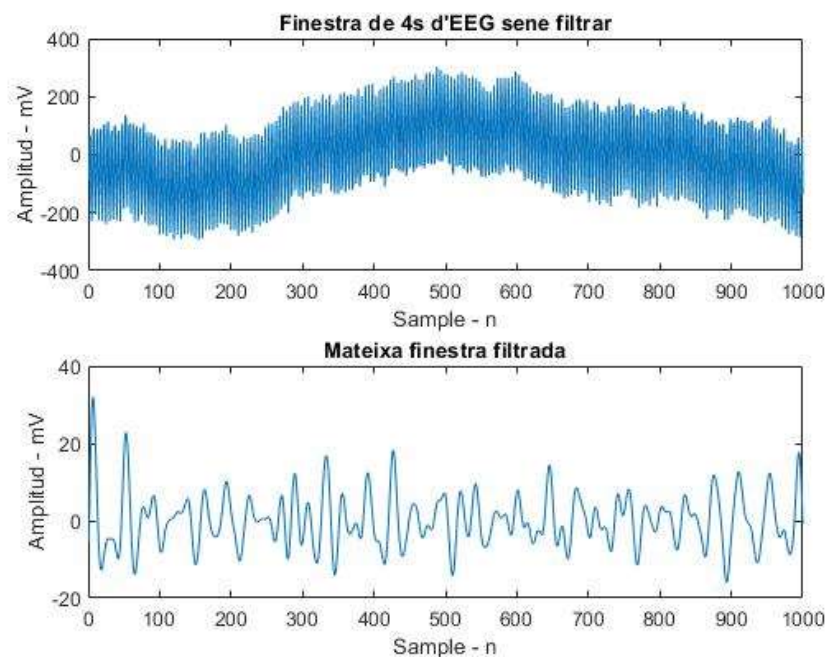
Un cop obtingut el vector característic es pot començar l'etapa de la classificació en viu. El procediment i el classificador és el mateix que en l'apartat offline. Per a cada freqüència estímul es definirà un rang de freqüències. El classificador busca el pic més alt al vector característic. La classe assignada dependrà de la localització d'aquest pic. Per a la seva verificació i càlcul de la precisió l'usuari haurà de mirar els 4 estímuls de 6.67 Hz a 10 Hz en ordre ascendent 3 vegades. Després els valors obtinguts pel classificador i els resultats obtinguts son comparats per obtenir la precisió.

## 4. Resultats

Els resultats de les primeres etapes no es poden reflectir en paper. Un exemple seria la generació d'estímuls, ja que tot i haver funcionat no es pot mostrar en aquest apartat. Malgrat això, els resultats obtinguts en la classificació offline i online demostren que els apartats anteriors han generat un resultat.

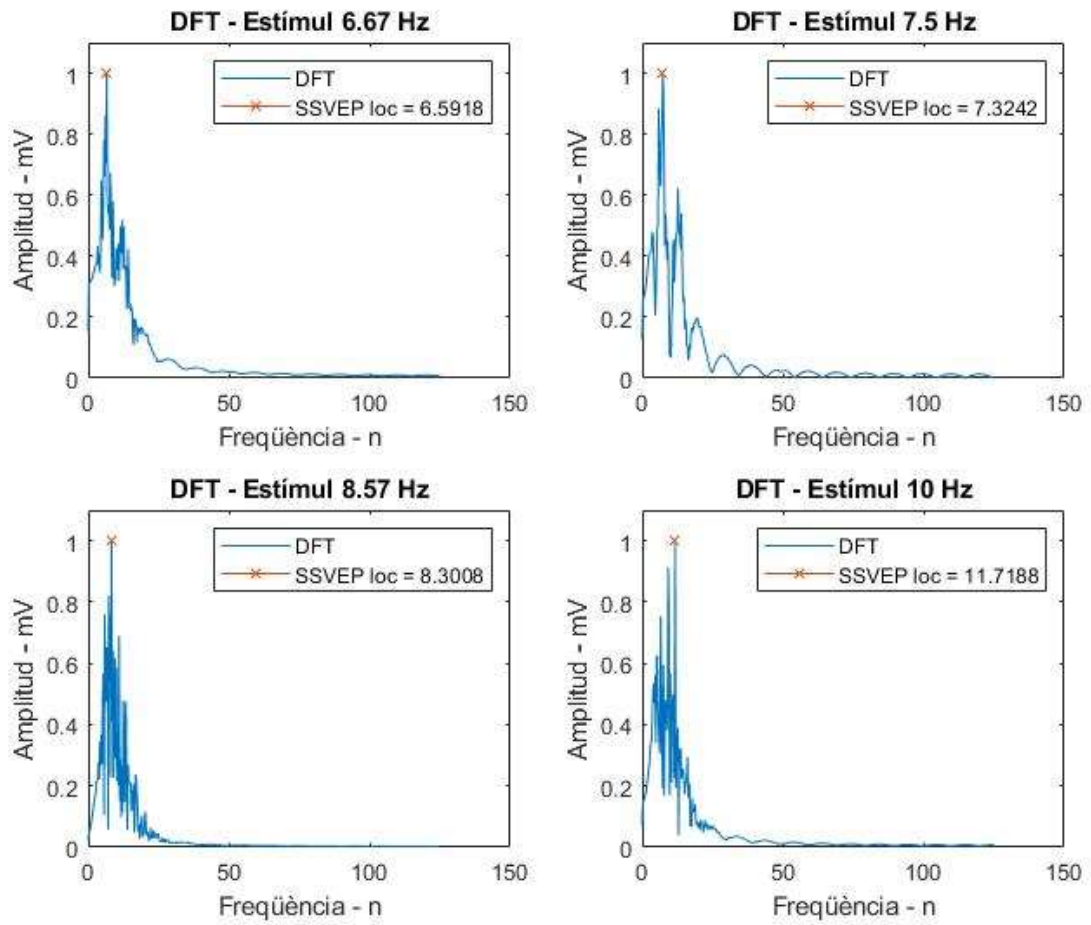
### 4.1 Classificació offline

Per a l'obtenció de resultats de la classificació offline, ha estat necessari el compliment de passos previs. La generació d'estímuls s'ha aconseguit. L'obtenció de dades offline amb OpenBCI\_GUI ha sigut exitosa. El resultat del processament de les dades offline en una finestra de 4 s es pot veure a la figura 13. L'obtenció de la DFT o del vector característic es pot veure a la figura 14. Les finestres han estat escollides de manera aleatòria entre els sets de dades base i 8.57 Hz. A la figura 15 es poden veure les localitzacions dels pics per a cada estímul. Els intervals per la classificació s'han extret d'aquests valors. Aquest intervals són: 6.3-6.8 Hz per la classe 6, 7.2-7.7 Hz per la classe 7, 8.2-8.8 Hz per la classe 8, i 11.4-11.9 per la classe 10. La precisió del classificador i les etiquetes a la que correspondria cada finestra de cada data set són representades a la figura 17.

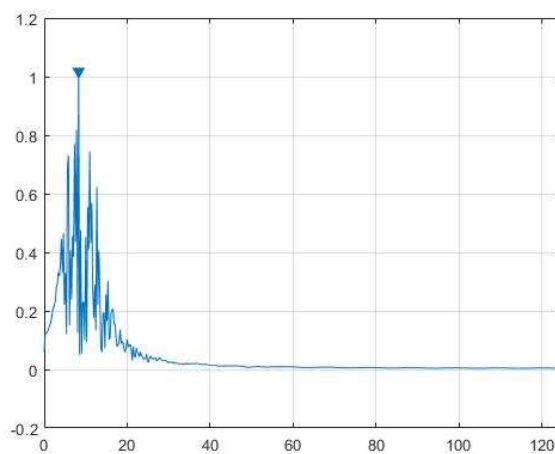


**Figura 14.** Finestra de 4s del data set base previ al processament (dreta) i a posterior (esquerra).





**Figura 15.** DFT de finestres estimulades i la corresponent localització dels SSVEP per a cada estímul.



**Figura 16.** Exemple de localització d'un pic amb el classificador entre la freqüències pertanyent a la classe 8, estímul de 8.57 Hz.



	Classe otorgada a la finestra de 4s corresponent pel classificador																																							Precisió a la sessió		
Segon de sessió	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	%				
Sessió base	0	0	0	0	0	6	0	0	8	0	0	0	0	7	0	0	0	7	0	6	0	0	0	8	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	81.08
Sessió 6.67 Hz	0	0	0	0	0	6	6	7	6	6	0	6	0	0	7	0	0	0	0	0	6	6	8	6	6	6	8	6	0	6	0	0	0	0	6	0	0	6	0	0	78.38	
Sessió 7.5 Hz	0	0	6	0	0	7	6	7	7	6	7	7	7	0	6	6	6	0	0	0	0	7	0	7	7	0	7	6	6	0	0	0	6	0	0	0	0	0	0	70.27		
Sessió 8.57 Hz	0	0	0	0	0	0	8	8	8	8	0	8	8	0	0	0	6	0	0	8	8	8	8	0	8	8	0	8	0	0	6	0	0	0	0	0	0	0	0	86.49		
Sessió 10 Hz	0	0	0	0	10	0	10	10	10	10	0	10	0	0	0	8	0	0	0	0	0	10	10	0	10	10	8	10	10	0	0	0	0	0	7	7	0	0	81.08			

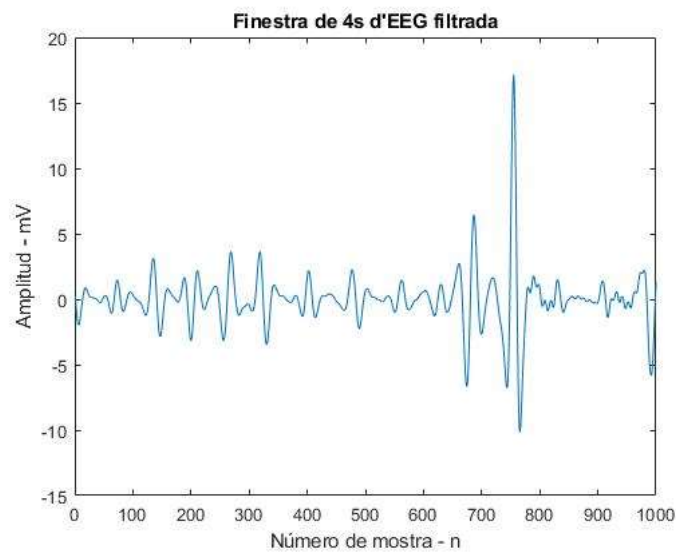
**Taula 4.** Resultats de la classificació offline per cada finestra de 4s i precisió del classificador a cada data set.

	Classe a la que pertany cada finestra de 4s																																						
Segon de sessió	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40		
Sessió base	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Sessió 6.67 Hz	0	0	0	0	0	6	6	6	6	6	6	6	6	0	0	0	0	0	0	0	6	6	6	6	6	6	6	6	6	0	0	0	0	0	0	0	0	0	
Sessió 7.5 Hz	0	0	0	0	0	7	7	7	7	7	7	7	7	0	0	0	0	0	0	0	7	7	7	7	7	7	7	7	7	0	0	0	0	0	0	0	0	0	
Sessió 8.57 Hz	0	0	0	0	0	8	8	8	8	8	8	8	8	0	0	0	0	0	0	0	8	8	8	8	8	8	8	8	8	0	0	0	0	0	0	0	0	0	
Sessió 10 Hz	0	0	0	0	0	10	10	10	10	10	10	10	10	0	0	0	0	0	0	0	10	10	10	10	10	10	10	10	10	0	0	0	0	0	0	0	0	0	

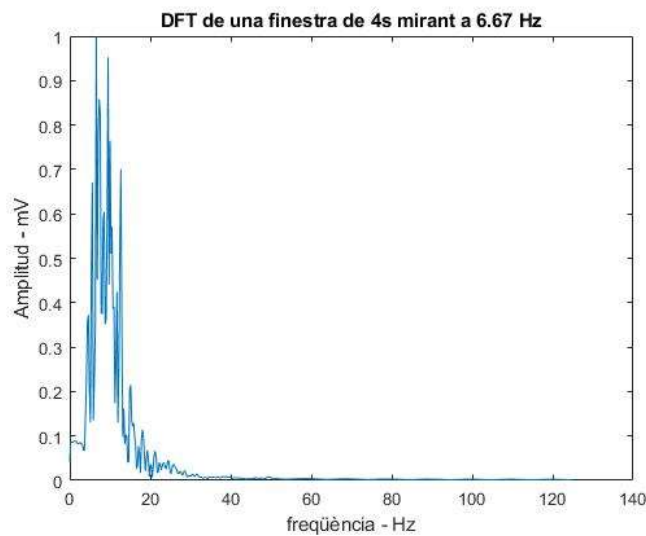
**Taula 5.** Etiquetes corresponents a cada finestra de 4s per tots els data sets.

## 4.2 Classificació en línia

Els resultats de la classificació en línia s'obtidran a partir d'un patró d'observació. Un cop acabat es procedirà a comparar els resultats esperats amb els obtinguts pel processament en viu. Per a obtenir aquests resultats s'han assolit altres parts del projecte com l'obtenció de dades en viu, el tractament d'aquestes, i l'extracció del vector característic. A la figura 17 s'observa un senyal aconseguida en viu filtrada. A la figura 18 és pot observar el vector característic d'aquest senyal. La taula 6 representa la precisió de la classificació en viu.



**Figura 17.** Finestra de 4s obtinguda en viu i processada.



**Figura 18.** DFT d'una finestra de 4s obtinguda en viu.

	Segon de sessió																											
	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Precisió
<b>Classe Real</b>	0	0	6	6	6	6	0	0	7	7	7	7	0	0	8	8	8	8	0	0	10	10	10	10	0	0	0	X
<b>Classe Obtinguda 1r Intent</b>	0	6	6	7	0	6	0	0	0	7	7	8	7	0	8	8	8	8	8	0	0	10	10	10	0	6	0	21/30 - 70%
<b>Classe Obtinguda 2n Intent</b>	0	0	6	6	7	6	6	0	0	7	7	0	0	0	7	8	8	8	0	10	8	8	8	10	10	0	0	20/30 66.67%
<b>Classe Obtinguda 3r Intent</b>	6	6	6	6	6	0	0	0	7	8	6	7	7	0	8	8	8	7	0	0	0	0	10	10	0	0	0	21/30 - 70%

**Taula 6.** Resultats de la classificació en viu per cada finestra de 4s i precisió del classificador a cada intent.

## 5. Discussió

A causa de les dificultats que han sorgit durant el transcurs del projecte, aquest s'ha vist obligat a adaptar-se a mesura que ha avançat. Tot i ser la primera BCI feta a la universitat de Vic s'han obtingut resultats bastant equiparables amb estudis previs que implementen la FFT com a mètode d'extracció de característiques (Vargic, n.d.). Altres mètodes, com la transformada de Wavelet, generen millors resultats, però tenen altres desavantatges com l'increment del poder computacional necessari per a la seva implementació (Al-Fahoum & Al-Fraihat, 2014). La precisió del classificador limita el seu ús per a funcions on la precisió sigui un factor determinant, però obre les portes a altres funcions i futures millores.

### 5.1 Limitacions

Aquest projecte és la primera interfície cervell-ordinador creada per un estudiant a la Universitat de Vic. Això ha estat un factor molt limitant, ja que no es partia de cap punt de referència. El primer problema és com generar un estímul amb freqüències estables i que puguin ser utilitzades en un estudi per a la detecció dels SSVEP amb MatLab. L'única opció per complir aquests requisits és la llibreria PTB. Aquesta llibreria és molt versàtil, però només permet actualitzar la pantalla sencera. Això comporta que la representació d'un estímul sigui molt fàcil, però la representació de més d'un estímul està limitada a la velocitat d'actualització de l'ordinador utilitzat. Aquest factor ha estat tan limitant que va fer canviar les freqüències estímul de 7, 8, 9, i 10 a les que s'han utilitzat en aquest projecte.

Tot i que OpenBCI proporciona molta documentació respecte a l'ús del casc d'EEG, la familiarització amb el software i hardware va ser molt tediosa. El casc d'EEG és molt incòmode de col·locar i qualsevol moviment, per petit que sigui, genera molt soroll. Per contra és molt fiable i es poden obtenir molt bons resultats. El problema més gran ha estat l'obtenció de les dades en cru a MatLab i la comprovació que aquestes siguin correctes. El senyal d'EEG és molt irregular i no tothom la genera igual. La detecció de la qualitat de la senyal i del seu processament no és fàcil i requereix entrenament previ en el tractament de dades d'EEG.

La versió d'OpenBCI\_GUI que s'estava utilitzant prèviament sofreix un error a l'enviar les dades via LSL. La detecció d'aquest error és complicat. Això suposa que primer s'enviïn les dades en viu per Python, però que es perdi molta informació i no es puguin detectar els SSVEP. La versió actualitzada de fa alguns mesos d'OpenBCI\_GUI té aquest error solucionat i proporciona una millora en l'enviament de dades. El rendiment de l'ordinador és un altre factor que afecta el processament de les dades en viu. Si el processament de les dades és molt complexa i el codi no està ben optimitzat hi ha informació que no és rebuda a MatLab. El codi pel processament de dades i la seva posterior classificació s'ha vist afectat per la poca capacitat computacional de l'ordinador utilitzat.

## 5.2 Possibles aplicacions i millores

A causa ser el primer disseny d'una BCI amb el casc d'OpenBCI a la Universitat de Vic encara hi ha molt recorregut de millora. La generació d'estímuls i el processament de les dades ja ha quedat establert per a futurs estudis. L'extracció de característiques del senyal és modificable. Per culpa de l'ordinador utilitzat s'ha hagut d'emprar el mètode de la FFT per obtenir el vector característic. Es recomana canviar aquest mètode per obtenir millors resultats i perdre menys resolució en el domini de la freqüència del senyal. Una recomanació seria el mètode d'autoregressió. La resolució freqüencial d'aquest mètode és molt superior a la FFT. En canviar el processament de les dades també cal crear un classificador nou. També és recomanable deixar més diferència entre les freqüències estímuls, ja que molt errors són causats per la poca distància entre aquestes. Per altra banda la precisió obtinguda pot generar interfícies d'acció en les que un error no és greu, com el moviment del ratolí de l'ordinador.

Per altres aplicacions, com la creació d'una cadira de rodes que es mogui amb el paradigma SSVEP, la precisió ha d'augmentar. Actualment hi ha projectes a la Universitat de Vic que es poden combinar amb una BCI. Un braç robòtic industrial, una mà robòtica, o un dron controlat amb el cervell són possibles aplicacions futures per BCI. Per a la realització d'aquestes funcions són necessàries millores en la BCI proposades en aquest projecte. La col·laboració entre disciplines pot generar un projecte amb molt de recorregut de futur.

## 6. Conclusions

Per culpa de les dificultats que han sorgit durant el transcurs del projecte, aquest s'ha vist obligat a modificar els objectius a mesura que ha anat avançant. Els estímuls són generats de manera satisfactòria. OpenBCI proporciona hardware i software a l'abast de tothom. És econòmic i fiable. Les dades s'obtenen de manera estable. La comprovació de la qualitat del senyal ha estat un èxit. Un cop familiaritzat amb l'ambient i introduït a la comunitat, l'ús del casc d'EEG es fa molt intuïtiu. El fòrum de la companyia és molt actiu, i els dubtes són resolts ràpidament. El processament de les dades i la classificació de les dades offline s'ha assolit de manera satisfactòria.

L'obtenció de les dades en viu és simple ara que es pot utilitzar el software d'OpenBCI. El processament en viu s'ha realitzat amb èxit, igual que la classificació. Es recomana canviar el mètode d'obtenció de característiques i de classificació per obtenir una millor precisió, i la realització d'una interfície d'acció per a futurs projectes. Independentment dels problemes que han sorgit s'han obtingut resultats rellevants que obren les portes a millores que poden acabar en investigacions i col·laboracions entre diferents disciplines.

## 7. Referències

- Al-Fahoum, A. S., & Al-Fraihat, A. A. (2014). Methods of EEG Signal Features Extraction Using Linear Analysis in Frequency and Time-Frequency Domains. *ISRN Neuroscience, 2014*, 1–7.  
<https://doi.org/10.1155/2014/730218>
- An Arduino-based Brain-Computer Interface - Popular Electronics*. (n.d.). Retrieved May 26, 2021, from <https://popularelectronics.technicacuriosa.com/2017/03/09/open-sourcing-the-brain-computer-interface/>
- Basic Science of EEG | Epilepsy Foundation*. (n.d.). Retrieved May 24, 2021, from <https://www.epilepsy.com/living-epilepsy/epilepsy-and/professional-health-care-providers/diagnosis-treatment/basic-science-egg>
- Biasiucci, A., Franceschiello, B., & Murray, M. M. (2019). Electroencephalography. In *Current Biology* (Vol. 29, Issue 3, pp. R80–R85). Cell Press. <https://doi.org/10.1016/j.cub.2018.11.052>
- Botero, P. A. (2013). DISCAPACIDAD Y ESTILOS DE AFRONTAMIENTO: UNA REVISIÓN TEÓRICA DISABILITY AND COPING STYLES: A THEORETICAL REVIEW. *Revista Vanguardia Psicológica, 3*(2).
- Brain-Computer Interfaces / Technologies / The MoreGrasp project*. (n.d.). Retrieved May 25, 2021, from <http://www.moregrasp.eu/technologies/Brain-Computer-Interfaces>
- Cyton Getting Started Guide · OpenBCI Documentation*. (n.d.). Retrieved May 28, 2021, from <https://docs.openbci.com/docs/01GettingStarted/01-Boards/CytonGS>
- Diseñar filtros digitales - MATLAB designfilt*. (n.d.). Retrieved May 29, 2021, from <https://es.mathworks.com/help/signal/ref/designfilt.html>
- Fernando Nicolas-Alonso, L., & Gomez-Gil, J. (2012). Brain Computer Interfaces, a Review. *Sensors, 12*, 1211–1279. <https://doi.org/10.3390/s120201211>
- Galloway, N. R. (1990). Human Brain Electrophysiology: Evoked Potentials and Evoked Magnetic Fields in Science and Medicine. *The British Journal of Ophthalmology, 74*(4), 255.  
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1042082/>
- GitHub - bobvo23/flicker\_stimulator for SSVEP*. (n.d.). Retrieved May 31, 2021, from [https://github.com/bobvo23/flicker\\_stimulator](https://github.com/bobvo23/flicker_stimulator)
- GitHub - labstreaminglayer/liblsl-Matlab: Matlab bindings for liblsl*. (n.d.). Retrieved May 30, 2021, from <https://github.com/labstreaminglayer/liblsl-Matlab>
- GUI Widget Guide · OpenBCI Documentation*. (n.d.). Retrieved May 31, 2021, from <https://docs.openbci.com/docs/06Software/01-OpenBCISoftware/GUIWidgets>
- Khalid, M. B., Rao, I., Rizwan-I-Haque, I., Munir, S., & Tahir, F. (n.d.). *TOWARDS A BRAIN COMPUTER INTERFACE USING WAVELET TRANSFORM WITH AVERAGED AND TIME SEGMENTED ADAPTED WAVELETS*.
- Kiret Dhindsa. (2017). *Number of BCI publications per year*. [https://www.researchgate.net/figure/The-number-of-BCI-publications-per-year-obtained-by-searching-for-braincomputer\\_fig2\\_323969239](https://www.researchgate.net/figure/The-number-of-BCI-publications-per-year-obtained-by-searching-for-braincomputer_fig2_323969239)

- Lab Streaming Layer (LSL) · OpenBCI Documentation.* (n.d.). Retrieved May 26, 2021, from <https://docs.openbci.com/docs/06Software/02-CompatibleThirdPartySoftware/LSL>
- Lopez-Calderon, J., & Luck, S. J. (2014). ERPLAB: An open-source toolbox for the analysis of event-related potentials. *Frontiers in Human Neuroscience*, 8(1 APR). <https://doi.org/10.3389/fnhum.2014.00213>
- MATLAB - El lenguaje del cálculo técnico - MATLAB & Simulink.* (n.d.). Retrieved May 26, 2021, from <https://es.mathworks.com/products/matlab.html>
- Neuron action potentials: The creation of a brain signal (article) | Khan Academy.* (n.d.). Retrieved May 24, 2021, from <https://www.khanacademy.org/test-prep/mcat/organ-systems/neuron-membrane-potentials/a/neuron-action-potentials-the-creation-of-a-brain-signal>
- Norcia, A. M., Gregory Appelbaum, L., Ales, J. M., Cottureau, B. R., & Rossion, B. (2015). The steady-state visual evoked potential in vision research: A review. *Journal of Vision*, 15(6), 1–46. <https://doi.org/10.1167/15.6.4>
- OpenBCI Ultracortex Mark IV headset (left); Cyton Biosensing board (right) | Download Scientific Diagram.* (n.d.). Retrieved May 26, 2021, from [https://www.researchgate.net/figure/OpenBCI-Ultracortex-Mark-IV-headset-left-Cyton-Biosensing-board-right\\_fig2\\_328539159](https://www.researchgate.net/figure/OpenBCI-Ultracortex-Mark-IV-headset-left-Cyton-Biosensing-board-right_fig2_328539159)
- Oshana, R. (2006). Overview of Digital Signal Processing Algorithms. In *DSP Software Development Techniques for Embedded and Real-Time Systems* (pp. 59–121). Elsevier. <https://doi.org/10.1016/b978-075067759-2/50006-5>
- Peng-Wie Hsia, A., Joel, S. E., Hsia, P.-W., & Professor, A. (n.d.). *Zero-padded fast Fourier transform reveals new dominant frequency pattern in optical maps during ventricular fibrillation.*
- Psychtoolbox-3 - Download, Installation, and Update.* (n.d.). Retrieved May 27, 2021, from <http://psychtoolbox.org/download.html>
- Psychtoolbox-3 - Overview.* (n.d.). Retrieved May 27, 2021, from <http://psychtoolbox.org/>
- Puce, A., & Hämäläinen, M. S. (2017). A review of issues related to data acquisition and analysis in EEG/MEG studies. In *Brain Sciences* (Vol. 7, Issue 6). MDPI AG. <https://doi.org/10.3390/brainsci7060058>
- Soufneyestani, M., Dowling, D., & Khan, A. (2020). Electroencephalography (EEG) technology applications and available devices. In *Applied Sciences (Switzerland)* (Vol. 10, Issue 21, pp. 1–23). MDPI AG. <https://doi.org/10.3390/app10217453>
- The OpenBCI GUI · OpenBCI Documentation.* (n.d.). Retrieved May 28, 2021, from <https://docs.openbci.com/docs/06Software/01-OpenBCISoftware/GUIDocs>
- Ultracortex Mark IV · OpenBCI Documentation.* (n.d.). Retrieved May 26, 2021, from <https://docs.openbci.com/docs/04AddOns/01-Headwear/MarkIV>
- van Atteveldt, N., Janssen, T. W. P., & Davidesco, I. (n.d.). *MEASURING BRAIN WAVES IN THE CLASSROOM.*
- Vargic, R. (n.d.). *Review of simple SSVEP evaluation FFT based algorithms.*



# Annex: I

```
function data = ReadData(path,session)
fid = fopen(strcat(path,session), 'r');
n = 0;
tline = fgetl(fid);
tline = fgetl(fid);
tline = fgetl(fid);
tline = fgetl(fid);
tline = fgetl(fid);
while ischar(tline)
    tline = fgetl(fid);
    n = n+1;
    if tline~-1
        EEG = strsplit(tline,',' );
        EEG_Data(n,:) = EEG(:,8:9);
    end
end
fclose(fid);
data = cellfun(@str2num,EEG_Data);
data = data(500:length(data)-500,:);
end
```

```
function dataOut = DividirData(data)
c = 1;
for n = 751:250:10000-250
    v1 = data(n-750:n+249,1);
    v2 = data(n-750:n+249,2);
    dataOut(1:1000,c) = v1;
    dataOut(1:1000,c+1) = v2;
    c = c+2;
end
end
```

```
function classe = Classificador(v,f7,f8,f9,f10,f)
classe = 0;
[pic,loc] = findpeaks(v,f,"NPeaks",1,"SortStr","descend");
classe(loc>f7(1) & loc < f7(2)) = 7;
classe(loc>f8(1) & loc < f8(2)) = 8;
classe(loc>f9(1) & loc < f9(2)) = 9;
classe(loc>f10(1) & loc < f10(2)) = 10;
end
```

# Annex: II

```
clc
clear
close all
```

```
%Aaron Baseline
```

```
path = 'C:\Users\aaroon\Documents\OpenBCI_GUI\Recordings\OpenBCISession_Aaron_Sessio_Base\';
session = 'OpenBCI-RAW-2021-06-01_12-19-14.txt';
data = ReadData(path,session);
```

```
%Aaron 6.67 Hz
```

```
path = 'C:\Users\aaroon\Documents\OpenBCI_GUI\Recordings\OpenBCISession_Aaron_Sessio_6Hz\';
session = 'OpenBCI-RAW-2021-06-01_12-14-00.txt';
data6 = ReadData(path,session);
```

```
%Aaron 7.5 Hz
```

```
path = 'C:\Users\aaroon\Documents\OpenBCI_GUI\Recordings\OpenBCISession_Aaron_Sessio_7Hz_2\';
session = 'OpenBCI-RAW-2021-06-01_13-25-53.txt';
data7 = ReadData(path,session);
```

```
%Aaron 8.57 Hz
```

```
path = 'C:\Users\aaroon\Documents\OpenBCI_GUI\Recordings\OpenBCISession_Aaron_Sessio_8Hz\';
session = 'OpenBCI-RAW-2021-04-30_12-43-20.txt';
data8 = ReadData(path,session);
```

```
%Aaron 10 Hz
```

```
path = 'C:\Users\aaroon\Documents\OpenBCI_GUI\Recordings\OpenBCISession_Aaron_Sessio_10Hz\';
session = 'OpenBCI-RAW-2021-04-30_12-39-39.txt';
data10 = ReadData(path,session);
```

```
data = DividirData(data);
```

```
data6 = DividirData(data6);
```

```
data7 = DividirData(data7);
```

```
data8 = DividirData(data8);
```

```
data10 = DividirData(data10);
```

```
fmin = 5;
fmax = 15;
Fs = 250;
```

```
bpFilt = designfilt('bandpassiir','FilterOrder',4,...
    'HalfPowerFrequency1',fmin,'HalfPowerFrequency2',fmax, ...
    'DesignMethod','butter','SampleRate',Fs);
```

```
bsFilt = designfilt('bandstopiir','FilterOrder',4,...
    'HalfPowerFrequency1',49,'HalfPowerFrequency2',51, ...
    'DesignMethod','butter','SampleRate',Fs);
```

```
fvtool(bpFilt);
fvtool(bsFilt);
```

```
data = filtfilt(bpFilt, data);
data = filtfilt(bsFilt, data);
```

```
data6 = filtfilt(bpFilt, data6);
data6 = filtfilt(bsFilt, data6);
```

```
data7 = filtfilt(bpFilt, data7);
data7 = filtfilt(bsFilt, data7);
```

```
data8 = filtfilt(bpFilt, data8);
data8 = filtfilt(bsFilt, data8);
```

```
data10 = filtfilt(bpFilt, data10);
data10 = filtfilt(bsFilt, data10);
```

```
L = 1000;
n = 2^nextpow2(L);
freq = Fs*(0:(n/2))/n;
M = 74;
c=1;
```

```
vCar = zeros(round(n/2)+1,37);
vCar6 = zeros(round(n/2)+1,37);
vCar7 = zeros(round(n/2)+1,37);
vCar8 = zeros(round(n/2)+1,37);
vCar10 = zeros(round(n/2)+1,37);
```

```
for k = 1:2:M
    vCar(:,c) = ExtCaract(data(:,k:k+1),n);
    vCar6(:,c) = ExtCaract(data6(:,k:k+1),n);
    vCar7(:,c) = ExtCaract(data7(:,k:k+1),n);
    vCar8(:,c) = ExtCaract(data8(:,k:k+1),n);
    vCar10(:,c) = ExtCaract(data10(:,k:k+1),n);
    c = c+1;
end
```

```
f6 = [6.3 6.8];
f7 = [7.2 7.7];
f8 = [8.3 8.8];
f10 = [11.6 11.8];
```

```
for k = 1:37
    lbl(k) = Classificador(vCar(:,k),f6,f7,f8,f10,freq);
    lbl6(k) = Classificador(vCar6(:,k),f6,f7,f8,f10,freq);
    lbl7(k) = Classificador(vCar7(:,k),f6,f7,f8,f10,freq);
    lbl8(k) = Classificador(vCar8(:,k),f6,f7,f8,f10,freq);
    lbl10(k) = Classificador(vCar10(:,k),f6,f7,f8,f10,freq);
end
```

```
for k = 1:37
    [pks(k),locs(k)] = findpeaks(vCar(:,k),freq,"NPeaks",1,"SortStr","descend");
    [pks6(k),locs6(k)] = findpeaks(vCar6(:,k),freq,"NPeaks",1,"SortStr","descend");
    [pks7(k),locs7(k)] = findpeaks(vCar7(:,k),freq,"NPeaks",1,"SortStr","descend");
    [pks8(k),locs8(k)] = findpeaks(vCar8(:,k),freq,"NPeaks",1,"SortStr","descend");
    [pks10(k),locs10(k)] = findpeaks(vCar10(:,k),freq,"NPeaks",1,"SortStr","descend");
end
```

# Annex: III

```
clc
clear
close all

disp('Carregar llibreria');
lib = lsl_loadlib();

disp("Trobant un stream d'EEG stream");
result = {};
while isempty(result)
    result = lsl_resolve_byprop(lib,'type','EEG');
end

disp('Obrint entrada de dades');
inlet = lsl_inlet(result{1});

L = 1000;
n = 2^nextpow2(L);
Fs = 250;
freq = Fs*(0:(n/2))/n;
EEG = zeros(1000,2);
k = 1;
f6 = [6.3 6.8];
f7 = [7.2 7.7];
f8 = [8.3 8.8];
f10 = [9.5 11.8];
fmin = 5;
fmax = 15;
c=1;

bpFilt = designfilt('bandpassiir','FilterOrder',4,...
    'HalfPowerFrequency1',fmin,'HalfPowerFrequency2',fmax, ...
    'DesignMethod','butter','SampleRate',Fs);

disp('Revent dades...');
while true
    [vec,ts] = inlet.pull_sample();
    EEG(k,:) = vec(:,7:8);
    k = k+1;
    if EEG(1000,1)~= 0
        EEG = filtfilt(bpFilt, EEG);
        data(:,c:c+1) = EEG;
        DFT = ExtCaract(EEG,n);
        out(c) = Classificador(DFT,f6,f7,f8,f10,freq);
        k = 751;
        EEG(1:750,:) = EEG(251:1000,:);
        EEG(k:end,:) = 0;
        c=c+1;
    end
end
```