**Preliminary Experiments**

First Mission – System Calibration

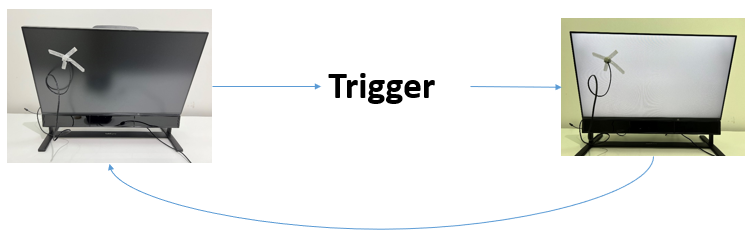
The system included a computer sending triggers via EGG system ant-neuro eego-mylab and another one recording them from it. The first goals were calibrating the system, getting familiarize with the recording equipment, and getting a better understanding of Lab streaming layer (LSL) python package and interfaces of data collection. To complete this goal, after collecting some data and understanding how to analyze it, we conducted a photodiode experiment.

**Photodiode Experiment**

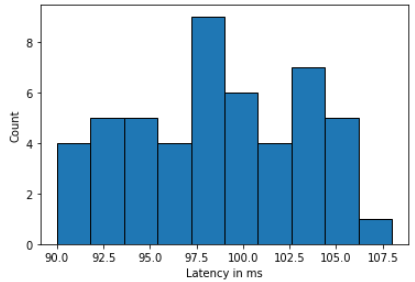
Main goal: calculate the average latency of the system, between sending a trigger and recording it as input.

We taped a photodiode on a computer screen and ran a Python script that sends a trigger through the EEG and turns the screen white for 200ms, multiple times. Then, using the second computer, we recorded the EEG data. We collected, preprocessed, and labeled the time series data from EEG.

We measured the latency between the time the trigger was sent and the time the photodiode recorded a significant change in brightness. We conducted this experiment twice: while recording 2 channels – photodiode and triggers, and while recording multiple channels – photodiode, triggers and 64 EEG electrodes.



Results: the system is robust, the amount of recorded channels has no cost in terms of latency, the average in both cases is about 10ms. This finding must be taken as consideration when running real-time experiences; in case the time difference between components of the experiment is shorter than 10ms, the results obtained will be wrong. The results below are presented in seconds.

2 channels

* Mean – 0.098
* Standard deviation – 0.005
* Minimum – 0.089
* Maximum – 0.107

Chart, histogram

Description automatically generated66 channels

* Mean – 0.097
* Standard deviation – 0.005
* Minimum – 0.087
* Maximum – 0.110

Chart, bar chart, histogram

Description automatically generated

2 channels

Chart, histogram

Description automatically generated

64 channels

Second Mission – Eye data classifier

The goals are collecting, preprocessing, and labeling time-series eye data, using it to train and validate a classification ML algorithm. Later, use this classifier to decode eye data and predict what the subject is looking at.

**Eye Cues Experiment**

We created a dataset of 240 black photos with a white dot in it. The dots’ locations were divided to 4 labels: up, down, left, and right. In each category we sampled the locations from a uniform distribution in range that match the category.

The experiment held 240 trials. In each trial we presented fixation cue followed by a photo from the dataset, in a random order and a trigger was sent with each fixation (and dot-photo???) presented. The subjects were told to look at the dot when it appears, back on the fixation cue and so on. We recorded the eye tracker data and analyzed it.

Analyzing:

We divided the time-series data from each trial into 10 time frames to find out when the eye data is more informative. In each time frame we calculated the average

