Introduction

In this documentation, we describe the structure of the BCI Framework code with explanations for individual functions. We define the process for getting started to use the framework and installation of necessary drivers, libraries, and toolboxes. BCI Framework contains the following main folders

- Communications
- CslDaqLib
- GeneralFramework
- FeatureExtraction
- Parameters
- Presentation
- SignalProcessing
- Libraries

System Requirements

- Windows 7 (32 bit),
- MATLAB 2012 a (32 bit)
- Inpout library (32 bit)
- gTec MATLAB API 3.11 (Win 32)
- gTec g.USBamp driver 3.10 (Win 32)
- Psychophysics MATLAB toolbox version 3 (32 bit)

Installation Steps

The required software, drivers and libraries are located in the FTP server under Projects/BCISoftwareDriversLibraries. Please download them and follow the steps.

1. Install Matlab 32 bit (You can download this from online Mathworks account)

2. Open Matlab and run the following command

- mbuild -setup
- Press y and enter
- Press 1 and enter
- Press y and enter

3. Install Slik Subversion 32 bit

4. Close Matlab

5. Create a new folder in C drive called "toolbox"

6. Copy DownloadPsychtoolbox.m to C:\toolbox

7.

- Run Matlab as Administrator and run the following commands
    - cd C:\toolbox\
    - DownloadPsychtoolbox('C:\toolbox')
- Press enter

8. Use CD (recommended) or do the following for the installation of the PCI Parallel Port Card

- Plug in the Express Card
- Right click to My Computer
- Select Properties
- Select Device Manager
- In other devices right click onto PCI Parallel Port Device
- Select update driver
- Select Browse my computer for driver software
- Select the directory with the driver (Step 8)

9. Copy inpout32 to C:\toolbox

10. Close Matlab and run Matlab as Administrator. In Matlab

- Select File -> Set Path
- Click Add with Subfolders
- Select C:\toolbox\inpout32 and Press Ok
- Press Save.
- Press Close

11. In Matlab run the following command

- mex -setup
- Press y and enter
- Press 1 and enter
- Press y and enter

12. Install gTec drivers and Matlab API

- Press Windows Start button
- Write the following to search box

- cmd
- Right click to cmd result comes up and select Run as Administrator
- In the command window cd to the Directory of Step 11 (gTec)\gTec\g.USBamp_Driver_3.10_Win32
- Run the following command
- msiexec.exe /i gUSBampDriver.msi
- Install Matlab API from "Step 11 (gTec)\gTec\MatlabAPI_3.11_Win32\gUSBampMLAPI.msi"

Getting Started

Devices and Cables Needed:

- Amplifiers
- Electrodes
- Gamma box (for active electrodes)
- Express card
- Parallel port to digital I/O cable
- Amplifier to USB cable
- Synch cable (if there are more than two amplifiers)
- Power supply and amplifier to accumulator cable (if running using battery)
- Power plug-in cable for amplifier (if not using the battery to run the amplifiers)

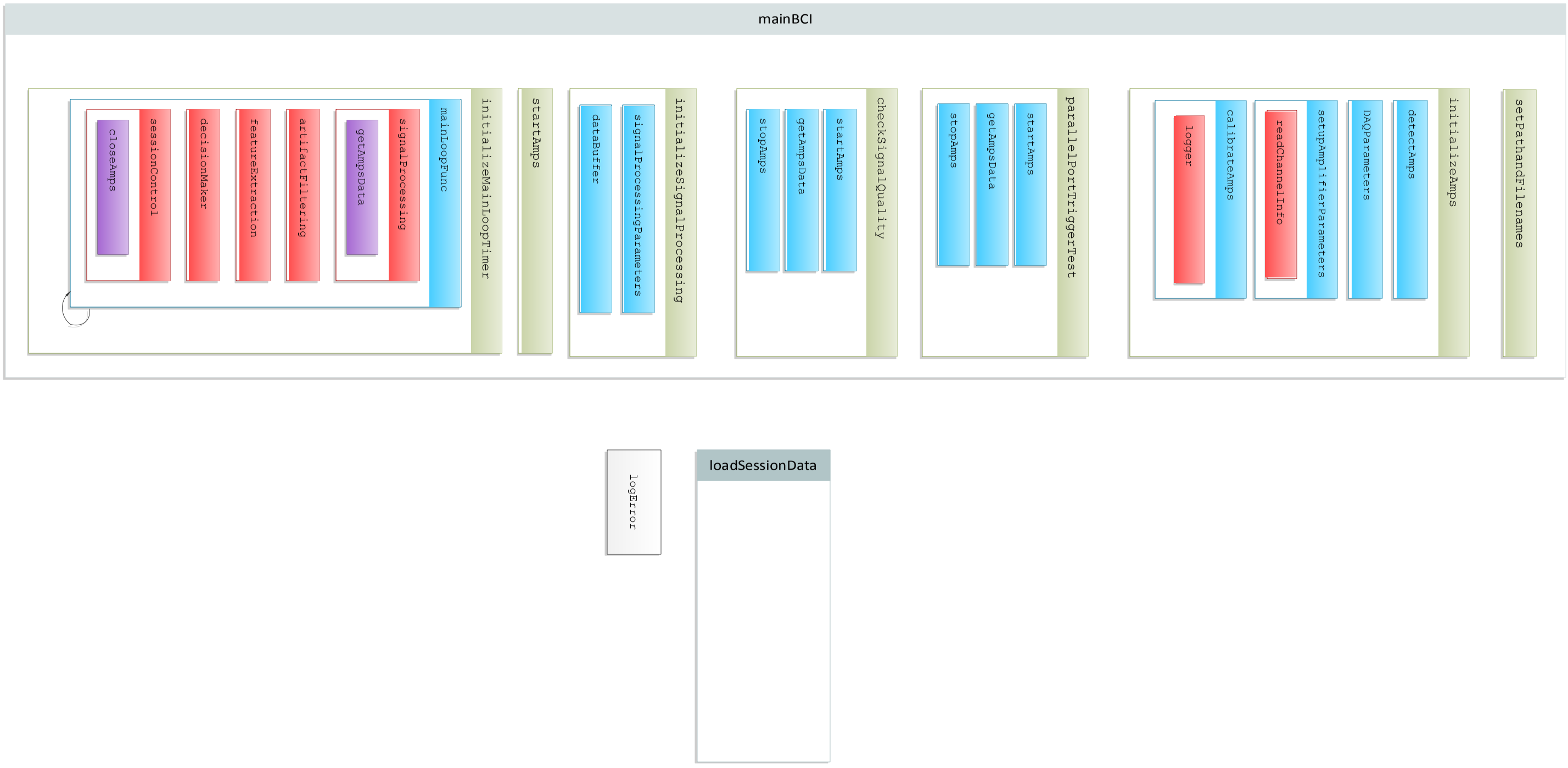
Connections:

- Connect the amplifiers to the computer using amplifier to USB cables
- Connect the amplifiers to each other if there are more than one amplifiers using synch cable (if there are two amplifiers choose one of the amplifiers as the master and the other as the slave, connect the synch cable between the synch-out port of the master and synch-in port of the slave. If there are four amplifiers; use three synch cables; choose one of the amplifiers as the master and the others as slaves; connect two of the synch cables between the synch-in and synch-out ports of the three slaves, and connect the last cable between the synch-out port of the master and the synch-in port of one of the slaves)
- If a battery is used connect the battery to the amplifiers using the amplifier to accumulator cable; otherwise, use the power plug-in cable
- Connect the express card to the parallel port using the parallel port to digital I/O cable
- Connect the electrodes to the amplifier, use the gamma box if the electrodes are active

Running the Framework Code

- Open MATLAB and browse to the BCI Framework folder
- Edit the MainBCI.m to enter the Subject ID and ProjectID and SessionID
- Browse to Parameters folder and edit channels.csv; there are 4 amplifiers and 64 channels listed in this file; all the channels which are not connected should be marked as NC; the type of the connected channels should be written under the Channel Type column; the electrode scalp location should be written under the Electrode Location column
- Edit DAQParameters.m and assign values to the elements of DAQStruct structure (see below for the description of DAQStruct elements)
- Edit signalProcessingParameters.m and assign values to the elements of mainBuffer structure (see below for the description of mainBuffer elements)
- Run MainBCI on MATLAB

BCI Framework Flowchart



Structures used in the BCI framework

- **amplifierStruct** : A structure that contains information about the amplifiers connected to the system. The following are the elements of this structure.
    - *numberOfAmplifiers* - An integer value demonstrating the number of amplifiers connected to the system.
    - *ai* - A vector of analog input objects (created using the data acquisition toolbox) corresponding to the amplifiers connected to the system.
    - *masterIndex* - An integer value showing the index of the master amplifier.
    - *slaveIndex* - A vector of integers showing the indices of the slave amplifiers.
    - *numberOfChannels* - A vector of integers showing the number of channels connecting to each amplifier.
    - *triggerIndex* - An integer showing the index of the trigger channel.
    - *totalNumberOfChannels* - An integer showing the total number of channels connected to all the amplifiers.
    - *channelBeginIndeces* - A vector of integers showing the beginning indices of the channels for each amplifier, that is the index of the first channel for each amplifier.
    - *channelEndIndeces* - A vector of integers showing the ending indices of the channels for each amplifier, that is the index of the last channel for each amplifier.

- **daqStruct** : A structure that contains user defined parameters to be used in the data acquisition process. The following are the elements of this structure.
    - *fs* - sampling frequency
    - *ampFilterNdx* - Filter index for using built-in bandpass amplifier filter
    - *notchFiltexNdx* - Filter index for using built-in notch amplifier filter
    - *ampBufferLengthSec* - Amplifier buffer length in seconds.
    - *calibrationOn* - Amplifier calibration enable flag

- **ampChannelList** : A structure that contains the information about the channels connected to the amplifiers. The members of this structure are the followings.
    - *channelIndices* - A list that contains the indices of channels connected to each amplifier (integer).
    - *electrodeLocations* - A list that contains the physiological locations of the electrodes that are connected to amplifiers (string).
    - *channelType*- A list that contains the type of the channels connected to the amplifiers (EEG, EMG, etc.)

- **frontendFilter** - the structure containing the frontend filter information. It should have the following fields,
    - *groupDelay* - shift to be introduced for triggers in samples
    - *Den* - denominator coefficients of the filter. This should be 1 for FIR filters.
    - *Num* - numerator coefficients of the filter.

- **mainBuffer**: A structure that contains the parameters for the initial signal processing after fetching the data. The processing consists of bandpass filtering and buffering. The following fields are set as parameters,
    - *bufferingMethod* - Buffering type ('linear'/'circular')
    - *bufferDurationSec* - Length of the buffer in seconds
    - *frontendFilteringFlag* - Enables (1)/disables (0) the frontend filtering before buffering