Connecting OpenViBE to Arduino.

- Camille Jeunet -
  - PhD in Cognitive Sciences, Univ. Bordeaux / Inria -
- 0.0 - Bad News -

WHICH PROGRAMMING LANGUAGE DO EELS PREFER?

SEA SHARP
- 0.1 - Good News x 3 -
- I - Context -

What is the impact of the user’s profile?  
What is the impact of the training protocol?
- I - Context -

What is the impact of the user’s profile?

What is the impact of the training protocol?
- I - Context -
"Ride a bike"

"Good"

"Bad"

"Stop"
Around 17% of the users did not manage to learn.
- I - Context -

What is the impact of the user’s profile?

What is the impact of the training protocol?
What is the impact of the user’s profile?

What is the impact of the training protocol?

Feedback
What is the impact of the training protocol?

What is the impact of the user’s profile?

Feedback
What is the impact of the training protocol?

What is the impact of the user’s profile?

Feedback

Tactile Feedback
[Control-Display Mapping]
- I - Context -

XP #1 - Vibrotactile feedback in a multitasking context.
**- I - Context -**

**XP #1** - Vibrotactile feedback in a multitasking context.
- I - Context -

XP #1 - Vibrotactile feedback in a multitasking context.

Experiment 2: Comparing visual and tactile feedback

- Motor-imagery in a gaming scenario
- Evaluating the effectiveness of tactile feedback
What is the impact of the user’s profile?

What is the impact of the training protocol?

Feedback

Tactile Feedback
[Control-Display Mapping]
What is the impact of the user’s profile?

What is the impact of the training protocol?

Feedback

Tactile Feedback [Control-Display Mapping]
What is the impact of the user’s profile?

What is the impact of the training protocol?

What is Tactile Feedback [Control-Display Mapping]?

How Improves the Sense of Agency?
- I - Context -

XP #2 - Impact of the Sense of Agency on MI-BCI Performance
What is the impact of the user’s profile?

What is the impact of the training protocol?

- Feedback
- Tactile Feedback [Control-Display Mapping]
- Improves the Sense of Agency?
Connecting OpenViBE to Arduino
Connecting OpenViBE to Arduino
Connecting OpenViBE to Arduino

- openvibe > share > openvibe > scenarios
  > bci-examples > motor-imagery-CSP
Connecting OpenViBE to Arduino

- Acquisition
- CSP filters
- Time based epoching
- Power
- Normalisation
- Classifier

OpenViBE

LSL Export
Stream[0] = classif output
Stream[1] = stimulations

Classifier processor

Configure LSL Export (Gipsa) settings
Stream name: OpenViBE Stream
Stream type: EEG
Override settings with configuration file
Load...  Save...  Default  Rétablir  Annuler  Appliquer
Connecting OpenViBE to Arduino
- II - Connecting OpenViBE to Arduino -

1. Import some packages – Add pylsl to known path

```python
import sys; sys.path.append('./pylsl')
from pylsl import StreamInlet, resolve_stream
import serial
import time
```

2. Open a serial port (windows)

```python
serialPort = serial.Serial('COM14', 9600, timeout=1)
```

3. Connection to Arduino

4. Communication with OpenViBE
   4.1 Resolve an EEG stream

```python
streams = resolve_stream('type', 'EEG')
```

   4.2 Create a new inlet to read from the stream

```python
inlet = StreamInlet(streams[0])
```
5. Action!

while Connected2Arduino:
    get sample (classif_output + stimulation)
    if stimulation == left
        instruction = left
    elif stimulation == right
        instruction = right
    if stimulation == feedback_continuous
        motor2activate = f(classif_output, instruction)
        serialPort.write('motor2activate')
    elif stimulation == endOfTrial
        serialPort.write('stopMotors')
Connecting OpenViBE to Arduino

```python
while ConnectedToArduino:
    getSample(classif_output + stimulation)
    if stimulation == left:
        instruction = left
    elif stimulation == right:
        instruction = right
    if stimulation == feedback_continuous:
        motor = f(classif_output, instruction)
        serialPort.write('motor2activate')
    elif stimulation == endOfTrial:
        serialPort.write('stopMotors')
```

- Define stimulation codes:

```
#define OVTG_GDF_SSSEP 0x133 // 307
#define OVTG_GDF_Start_Of_Trial 0x300 // 768
#define OVTG_GDF_Left 0x301 // 769
#define OVTG_GDF_Right 0x302 // 770
#define OVTG_GDF_Foot 0x303 // 771
#define OVTG_GDF_Tongue 0x304 // 772
#define OVTG_GDF_class5 0x305 // 773
#define OVTG_GDF_Down 0x306 // 774
#define OVTG_GDF_class7 0x307 // 775
#define OVTG_GDF_class8 0x308 // 776
#define OVTG_GDF_class9 0x309 // 777
#define OVTG_GDF_class10 0x30A // 778
#define OVTG_GDF_class11 0x30B // 779
#define OVTG_GDF_Up 0x30C // 780
#define OVTG_GDF_Feedback_Continuous 0x30D // 781
#define OVTG_GDF_Feedback_Discrete 0x30E // 782
#define OVTG_GDF_Cue_Unknown_Undefined 0x30F // 783
#define OVTG_GDF_Beep 0x311 // 785
#define OVTG_GDF_Cross_On_Screen 0x312 // 786
#define OVTG_GDF_Flashing_Light 0x313 // 787
// SPECIALY ADDED BY YR
#define OVTG_GDF_End_Of_Trial 0x320 // 800
```
5. Action!

while Connected2Arduino:
    get sample (classif_output + stimulation)
    if stimulation == left
        instruction = left
    elif stimulation == right
        instruction = right
    if stimulation == feedback_continuous
        motor2activate = f(classif_output, instruction)
        serialPort.write('motor2activate')
    elif stimulation == endOfTrial
        serialPort.write('stopMotors')
- II - Connecting OpenViBE to Arduino -
Arduino script

get the sample from the serial port
if sample == motor2activate
    activate motor2activate
elif sample == stopMotors
    stop the motors
- II - Connecting OpenViBE to Arduino -
II - Connecting OpenViBE to Arduino -
- Camille Jeunet -
- camille.Jeunet@inria.fr -
- camillejeunet.wordpress.com -
- https://team.inria.fr/potioc/ -
- https://twitter.com/potioc_team -

- Thank You! -
Thank You!