The Electroencephalogram Control System

Contributors: Kyle Zeller, Shakhar Dasgupta, & Christopher Wells

Main Goal

- Control a robot with an electroencephalogram (EEG) by directly analyzing the signals in your brain and identifying action(s) you were intending to perform

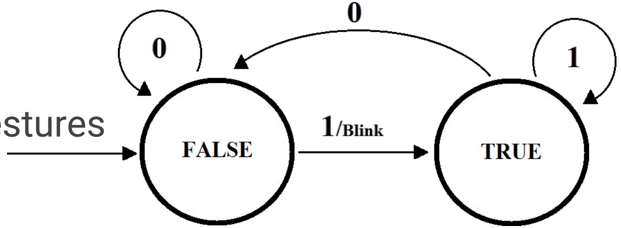
Muse EEG Headband

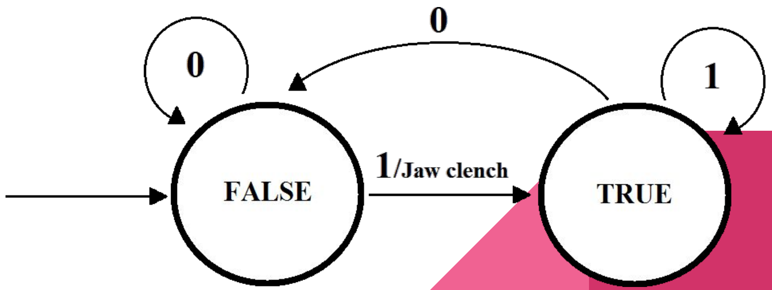
- Created by Interaxon
- 7 dry electrode sensors
- Alpha, Beta, Theta, Gamma, Delta waves
- Open Sound Control (OSC) packets are sent from the muse headset to the computer

Signal Analysis

- Blink : (10 Hz)
- Jaw Clench : (10 Hz)
- Concentration : (Gamma waves : 25 - 100 Hz)
- Java Library (w/ GitHub link)
    - https://github.com/SUNY-Oswego-MUSE-Project/muse-gestures

```
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/elements/beta_absolute ffff  -0.03 -0.23 0.54 0.19
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/elements/delta_absolute ffff  0.27 -0.07 -0.29 0.23
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/elements/gamma_absolute ffff  -0.23 -0.37 0.50 -0.18
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/elements/theta_absolute ffff  0.39 -0.20 0.18 0.72
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/elements/alpha_session_score ffff  0.31 0.03 0.60 0.49
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/elements/beta_session_score ffff  0.00 0.46 0.98 0.19
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/elements/delta_session_score ffff  0.02 0.37 0.27 0.00
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/elements/gamma_session_score ffff  0.19 0.57 1.00 0.00
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/elements/theta_session_score ffff  0.34 0.39 0.72 0.87
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/eeg ffff  838.94 857.03 843.87 838.94
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/eeg ffff  842.23 858.68 838.94 842.23
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/eeg ffff  845.52 858.68 850.45 842.23
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/eeg ffff  843.87 857.03 853.74 840.58
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/eeg ffff  842.23 855.39 845.52 840.58
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/eeg ffff  838.94 855.39 835.65 842.23
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/eeg ffff  842.23 855.39 838.94 845.52
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/acc fff  -332.03 960.94 85.94
Playback Time: 2.0s : Sending Data        1456621826.77 /muse/acc fff  -335.94 964.85 89.84
Playback Time: 2.0s : Sending Data        1456621826.78 /muse/acc fff  -339.84 960.94 89.84
Playback Time: 2.0s : Sending Data        1456621826.79 /muse/drlref ff  1645161.25 1645161.25
Playback Time: 2.0s : Sending Data        1456621826.79 /muse/acc fff  -343.75 960.94 85.94
Playback Time: 2.0s : Sending Data        1456621826.82 /muse/eeg ffff  840.58 857.03 838.94 838.94
Playback Time: 2.0s : Sending Data        1456621826.82 /muse/eeg/quantization iiii  1 1 1 1
```





Steps

Step 1: Displaying Values In Text

- Output just the signals we want from the osc stream
- Analyze these signals with boolean logic to test function calls

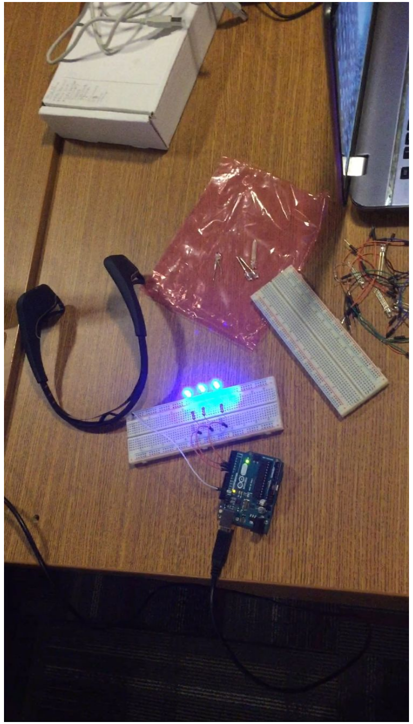
```
0
0
1
JAW NOT CLENCHED
0
1
JAW CLENCHED
0
1
JAW NOT CLENCHED
0
1
JAW CLENCHED
0
1
JAW NOT CLENCHED
0
1
JAW CLENCHED
0
1
```

Step 2: Final State Machines

- Implement the "blink" & "jaw clench" FSM's in the code
- Use the blink FSM to toggle the state of a lightbulb from on to off
- Use the jaw clench FSM to toggle the state of a lego man's facial expression from jaw not clenched to jaw clench

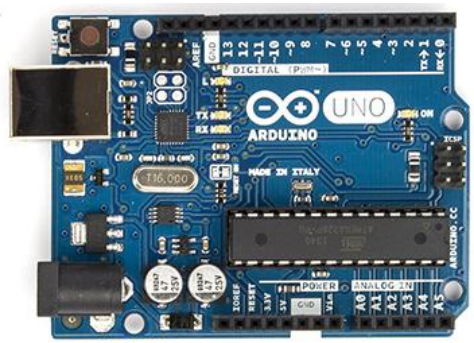
Step 3: Using Hardware

- Write a program in C for the arduino uno to wait for commands to come in serially to toggle the state of leds



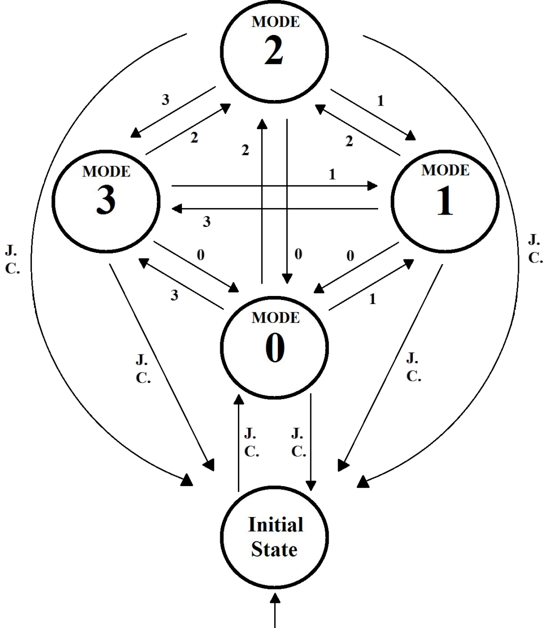
Step 4: Wireless Communication

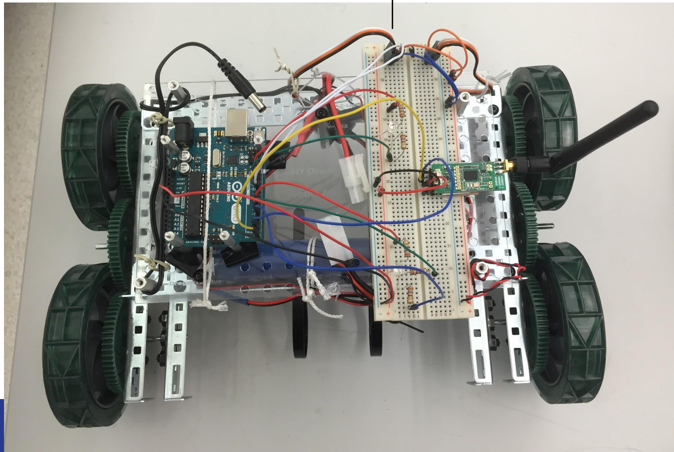
- Interface wireless serial communication between the hardware (arduino uno) & the computer using radio modules



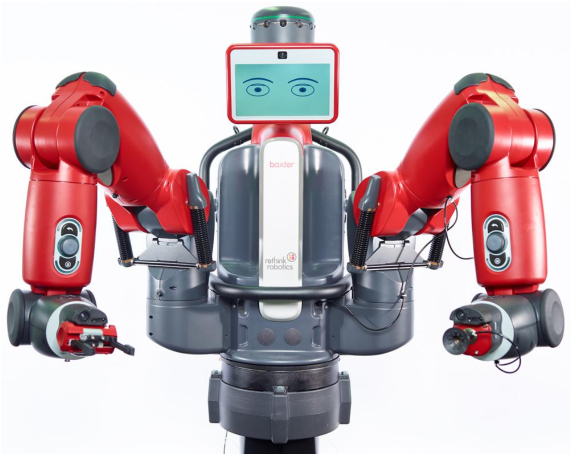
Step 5: Robotic Car

- Rewrite the C program to control the speed of the servo motors on the robotic car as well as the state of 2 RGB LEDs
- Implement the FSM for motor control
- The speed of the robotic car is controlled by how much the user is able to concentrate on something
- The state of the robot (on : off) is now controlled by the user clenching their jaw





Objective Status & Future Plans



- Objective : “COMPLETE”
- FUTURE PLANS :
  - ❑ Perform more signal analysis for other [raw] signals (gamma & beta) to create other inputs to control devices
  - ❑ Add more steering capabilities (left, right, backwards)
  - ❑ Control Baxter :
    - ➢ Arms, Hands
    - ➢ Onboard Sensors & Camera
    - ➢ Digital Display

Sources

- Robot. Digital image. Web. 5 Apr. 2016. <http://www2.vobs.at/ball-online/Topics/TOYS/toys_s8.jpg>.
- EEG. Digital image. Web. 4 Apr. 2016. <http://tehnot.com/wp-content/uploads/2015/07/nejro.jpeg>.
- MUSE. Digital image. Web. 4 Apr. 2016. <http://michaelapollo.com/wp-content/uploads/2015/08/Muse.jpg>.
- Light-off. Digital image. Web. 4 Apr. 2016. <http://static.wixstatic.com/media/2a1842_8b92974844c54f19955eb98ddd20f9df.png/v1/fill/w_114,h_169,al_c,usm_0.66_1.00_0.01/2a1842_8b92974844c54f19955eb98ddd20f9df.png>.
- Light-on. Digital image. Web. 4 Apr. 2016. <http://www.sebastien-laframboise.com/wp-content/uploads/twiz-light-bulb-lit.png>.
- Jaw-clench. Digital image. Web. 4 Apr. 2016. <http://cdn2.mommyish.com/wp-content/uploads/2013/06/lego_minifig__head_male_black_raised_eyebrows__angry_open_mouth__white_pupils_pattern_3626bpb533yellow.jpg>.
- No-jaw-clench. Digital image. Web. 4 Apr. 2016. <http://thumbs2.ebaystatic.com/m/m_5tYOBjfAOE8URDKMRLUQg/140.jpg>.
- Arduino. Digital image. Web. 4 Apr. 2016. <http://k32.kn3.net/taringa/C/0/8/7/0/4/eduard10000/FFE.jpg>.
- Baxter. Digital image. Web. 4 Apr. 2016. <https://lh3.googleusercontent.com/yXQG6hz8nOV-CP4q0ddJSVaclZI2w2LOEL5Zuck4r3Q13urpMAaBKZmrKzt0f9xGFsqL-oLbzY3E6e42RQA1iKOmgUypszp0Awx0noaH3W9JN3IWS9D0X0gFGS1_-kplPAUPUC3_>.

LIVE DEMO

Questions?