Control Electronics with your Brain using NextMind

Created by Collin Cunningham

https://learn.adafruit.com/control-electronics-with-your-brain-using-nextmind

Last updated on 2022-01-28 11:55:45 AM EST
Table of Contents

Overview 3
- What you’ll need 3

Nextmind Setup 4

Unity Project 5
- Duplicate Scenes 5
- Add scripts 6
- Set Triggered Calls 7
- Edit project settings 8
- Edit Scene Reference 8

Hardware 8
- Wiring 8
- Add pointer 9

Arduino 10

Arduino IDE Setup 10

Using with Arduino IDE 13
- Install SAMD Support 13
- Install Adafruit SAMD 14
- Install Drivers (Windows 7 & 8 Only) 15
- Blink 17
- Successful Upload 18
- Compilation Issues 19
- Manually bootloading 20
- Ubuntu & Linux Issue Fix 20

Arduino Code 20
- Program the Feather 21
- Reposition pointer 22

Usage 23
Overview

Use neural signals from your brain’s visual cortex to control electronics. This guide will show you how to use the Nextmind Dev Kit, Unity game development software, and an Adafruit Feather microcontroller board to control a servo motor with your mind & visual focus. This example project translates area of visual focus into the physical position of a servo with attached pointer, but it is just that – an example. You’ll likely think of a million more awesome applications after experimenting a bit.

The Nextmind Dev Kit offers an unusually reliable method for converting neural activity to user input. The device’s sensors measure tiny electrical signals from the visual cortex located at the back of the brain. So, it essentially reads what you’re visually focussing on. This may seem less magical compared to other brain-computer interfaces, but it seems to make the Nextmind much more practical, dependable than other brain-computer interfaces I’ve experimented with.

What you’ll need

- Nextmind Dev Kit (https://adafruit.it/Q6C)
- Windows or Mac computer w Bluetooth
- "Craft stick" / tongue depressor (optional)

1 x Feather M4 Express
Feather M0, QT Py, others should work as well

1 x Micro Servo
SG92R

1 x Micro USB Cable
Connect Feather to your computer
<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 x Hook-Up Wire</td>
<td>Set of 6 spools</td>
<td><a href="https://www.adafruit.com/product/1311">https://www.adafruit.com/product/1311</a></td>
</tr>
<tr>
<td>1 x Half-size Breadboard</td>
<td>For Feather to servo connections</td>
<td><a href="https://www.adafruit.com/product/64">https://www.adafruit.com/product/64</a></td>
</tr>
<tr>
<td>1 x Wire Strippers</td>
<td>For makin' jumpers</td>
<td><a href="https://www.adafruit.com/product/527">https://www.adafruit.com/product/527</a></td>
</tr>
<tr>
<td>1 x Pocket Screwdriver</td>
<td>Install pointer on servo</td>
<td><a href="https://www.adafruit.com/product/3284">https://www.adafruit.com/product/3284</a></td>
</tr>
<tr>
<td>1 x Panavise Jr.</td>
<td>Optional way to hold servo during use</td>
<td><a href="https://www.adafruit.com/product/151">https://www.adafruit.com/product/151</a></td>
</tr>
</tbody>
</table>

## Nextmind Setup

Follow the Nextmind Quickstart guide ([https://adafruit.it/Q6D](https://adafruit.it/Q6D)) to install the Nextmind Manager app and get acquainted with the connection & calibration process.

Once you’ve got the Nextmind up & running, move on to Tutorial #1: Setting up the Environment ([https://adafruit.it/Q6E](https://adafruit.it/Q6E)) (skip the XR Setup step). This section will show you how to install the correct version of Unity.
Unity Project

Our goal in Unity is to output a serial message to Feather over USB, indicating which cube was activated via focusing with the NextMind.

To get our project up & running quickly, you can duplicate and edit example scenes included in the NextMind SDK package.

Duplicate Scenes

In your Unity project's Project pane, navigate to Assets -> NextMindSDK -> Examples folder. Copy the scene titled Hub, paste it into Assets -> Scenes, and change its name to Hub_edit.

Do the same with:

- Assets/NextMindSDK/Examples/Calibration/Calibration scene
- & Assets/NextMindSDK/Examples/SDKDiscovery/SDKDiscovery scene

Your Scenes folder should look like the image below:

Mac users note – if you’re unable to locate the Nextmind SDK file (NextMindSDK_Core.unitypackage), download the Windows installer and extract it from there.
Finally, open the SDKDiscovery-edit scene and delete all child objects of the ScenePanels object.

Create a new child object of ScenePanels and name it Neurotags

Copy the CubeTag prefab from Assets/NextMindSDK/Core/Runtime/Prefabs/NeuroTags/ and paste it as a child of Neurotags a total of three times.

Rename those newly pasted CubeTag objects to CubeTagA, CubeTagB, CubeTagC

Your scene hierarchy should now look like the above example.

Add scripts

Click the button below to download a zip file containing the following two scripts:

- SerialComms.cs
- TelekinesisStep.cs

https://adafruit.it/Q8e
Unzip the downloaded file.

Add the TelekinesisStep script to the NeuroTags object.

Add the SerialComms script to the ScenePanels object.

Note the `Com_port_number` field in the Serial Comms script. This is the serial port for the Feather – we’ll fill this in later.

Set Triggered Calls

In CubeTagA, under Tracking Events, click the plus button under On Triggered ().

Click and drag the ScenePanels object from the Hierarchy pane and release it over the field labelled None (object).

Click on the pull-down menu labelled No Function and choose Serial Comms -> TriggerFirst

Repeat the process for CubeTagB, choosing TriggerSecond, & CubeTagC, choosing TriggerThird
Edit project settings

Open the Project Settings window and make the following changes:

- Player -> Other Settings -> Active Input Handling – set to Both
- Player -> Other Settings -> API Compatibility – set to .NET 4.x

Edit Scene Reference

Finally, open the Hub-edit scene and locate the object in the scene hierarchy named SDKDiscoveryGroup. Within that object's Button script, edit the target variable of On Click () to read SDKDiscovery-edit.

Now the Discover button in the main menu will load our test scene equipped with serial communication.

Hardware

Wiring

Cut three pieces of jumper wire ~5cm in length and strip ~7mm of insulation from each end.
Use the jumpers to make the following connections between the servo & Feather board:

- Servo yellow wire to Feather pin 5
- Servo orange wire to Feather pin USB
- Servo brown wire to Feather pin GND

Add pointer

The big pointer is optional – a basic servo horn works as well.
Attach one of the servo's included horns by press-fitting it onto the output shaft. If you don’t plan on adding pointer, go ahead and screw the horn in place.

A “craft stick”, aka - tongue depressor, works well for this. Use scissors or flush cutters to trim one end to a point. Use a nail or small drill to make small hole near the opposite end for the mounting screw. Add additional customization as desired. I painted my pointer green because it's the human eye's most favorite color (https://adafru.it/Q8f).

Arduino

Download & install the Arduino IDE and add support for the Feather board by following the steps on the following two pages.

Arduino IDE Setup

The first thing you will need to do is to download the latest release of the Arduino IDE. You will need to be using version 1.8 or higher for this guide

Arduino IDE Download

https://adafru.it/f1P
After you have downloaded and installed the latest version of Arduino IDE, you will need to start the IDE and navigate to the Preferences menu. You can access it from the File menu in Windows or Linux, or the Arduino menu on OS X.

A dialog will pop up just like the one shown below.

We will be adding a URL to the new Additional Boards Manager URLs option. The list of URLs is comma separated, and you will only have to add each URL once. New Adafruit boards and updates to existing boards will automatically be picked up by the Board Manager each time it is opened. The URLs point to index files that the Board Manager uses to build the list of available & installed boards.

To find the most up to date list of URLs you can add, you can visit the list of third party board URLs on the Arduino IDE wiki (https://adafruit.it/f7U). We will only need to add one URL to the IDE in this example, but you can add multiple URLs by separating
them with commas. Copy and paste the link below into the Additional Boards Manager URLs option in the Arduino IDE preferences.

https://adafruit.github.io/arduino-board-index/package_adafruit_index.json

Here's a short description of each of the Adafruit supplied packages that will be available in the Board Manager when you add the URL:

- **Adafruit AVR Boards** - Includes support for Flora, Gemma, Feather 32u4, ItsyBitsy 32u4, Trinket, & Trinket Pro.
- **Adafruit SAMD Boards** - Includes support for Feather M0 and M4, Metro M0 and M4, ItsyBitsy M0 and M4, Circuit Playground Express, Gemma M0 and Trinket M0
- **Arduino Leonardo & Micro MIDI-USB** - This adds MIDI over USB support for the Flora, Feather 32u4, Micro and Leonardo using the [arcore project](https://adafru.it/eSI).

If you have multiple boards you want to support, say ESP8266 and Adafruit, have both URLs in the text box separated by a comma (,)

Once done click OK to save the new preference settings. Next we will look at installing boards with the Board Manager.

Now continue to the next step to actually install the board support package!
Using with Arduino IDE

The Feather/Metro/Gemma/QTPy/Trinket M0 and M4 use an ATSAMD21 or ATSAMD51 chip, and you can pretty easily get it working with the Arduino IDE. Most libraries (including the popular ones like NeoPixels and display) will work with the M0 and M4, especially devices & sensors that use I2C or SPI.

Now that you have added the appropriate URLs to the Arduino IDE preferences in the previous page, you can open the Boards Manager by navigating to the Tools->Board menu.

Once the Board Manager opens, click on the category drop down menu on the top left hand side of the window and select All. You will then be able to select and install the boards supplied by the URLs added to the preferences.

Remember you need SETUP the Arduino IDE to support our board packages - see the previous page on how to add adafruit's URL to the preferences

Install SAMD Support

First up, install the latest Arduino SAMD Boards (version 1.6.11 or later)

You can type Arduino SAMD in the top search bar, then when you see the entry, click I nstall
Install Adafruit SAMD

Next you can install the Adafruit SAMD package to add the board file definitions.

Make sure you have Type All selected to the left of the Filter your search... box.

You can type Adafruit SAMD in the top search bar, then when you see the entry, click Install.

Even though in theory you don't need to - I recommend rebooting the IDE.

Quit and reopen the Arduino IDE to ensure that all of the boards are properly installed. You should now be able to select and upload to the new boards listed in the Tools->Board menu.
Select the matching board, the current options are:

- Feather M0 (for use with any Feather M0 other than the Express)
- Feather M0 Express
- Metro M0 Express
- Circuit Playground Express
- Gemma M0
- Trinket M0
- QT Py M0
- ItsyBitsy M0
- Hallowing M0
- Crickit M0 (this is for direct programming of the Crickit, which is probably not what you want! For advanced hacking only)
- Metro M4 Express
- Grand Central M4 Express
- ItsyBitsy M4 Express
- Feather M4 Express
- Trellis M4 Express
- PyPortal M4
- PyPortal M4 Titano
- PyBadge M4 Express
- Metro M4 Airlift Lite
- PyGamer M4 Express
- MONSTER M4SK
- Hallowing M4
- MatrixPortal M4
- BLM Badge

Install Drivers (Windows 7 & 8 Only)

When you plug in the board, you'll need to possibly install a driver
Click below to download our Driver Installer

Download Latest Adafruit Drivers package

https://adafruit.it/mb8

Download and run the installer

Run the installer! Since we bundle the SiLabs and FTDI drivers as well, you'll need to click through the license

Select which drivers you want to install, the defaults will set you up with just about every Adafruit board!
Click Install to do the installin'

Blink

Now you can upload your first blink sketch!

Plug in the M0 or M4 board, and wait for it to be recognized by the OS (just takes a few seconds). It will create a serial/COM port, you can now select it from the dropdown, it'll even be 'indicated' as Trinket/Gemma/Metro/Feather/ItsyBitsy/Trellis!
Please note, the QT Py and Trellis M4 Express are two of our very few boards that does not have an onboard pin 13 LED so you can follow this section to practice uploading but you wont see an LED blink!

Now load up the Blink example

```cpp
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);              // wait for a second
}
```

And click upload! That's it, you will be able to see the LED blink rate change as you adapt the delay() calls.

If you are having issues, make sure you selected the matching Board in the menu that matches the hardware you have in your hand.

Successful Upload

If you have a successful upload, you'll get a bunch of red text that tells you that the device was found and it was programmed, verified & reset.
After uploading, you may see a message saying "Disk Not Ejected Properly" about the ...BOOT drive. You can ignore that message: it's an artifact of how the bootloader and uploading work.

Compilation Issues

If you get an alert that looks like

Cannot run program "{runtime.tools.arm-none-eabi-gcc.path}\bin\arm-none-eabi-g++"

Make sure you have installed the Arduino SAMD boards package, you need both Arduino & Adafruit SAMD board packages.
Manually bootloading

If you ever get in a 'weird' spot with the bootloader, or you have uploaded code that crashes and doesn't auto-reboot into the bootloader, click the RST button twice (like a double-click) to get back into the bootloader.

The red LED will pulse and/or RGB LED will be green, so you know that its in bootloader mode.

Once it is in bootloader mode, you can select the newly created COM/Serial port and re-try uploading.

You may need to go back and reselect the 'normal' USB serial port next time you want to use the normal upload.

Ubuntu & Linux Issue Fix

Follow the steps for installing Adafruit's udev rules on this page. (https://adafruit.it/iOE)

Arduino Code

The code that will run on Feather will receive a numerical value over the USB serial connection and rotate the servo shaft an amount corresponding to that value – visually indicating which on-screen cube was activated.
Program the Feather

Connect your Feather to your computer using data + power, known good micro B USB cable.

Open the Arduino IDE, and from the top menu, go to Tools -> Board and ensure Adafruit Feather M4 Express (SAMD51) is selected. Then go to Tools -> Port and select the port corresponding to your Feather board – this is usually the last entry in the list.

Now that you know the Feather's port address, briefly head back into Unity and enter it in the Serial Comms script's com port field as seen above.

Go to File -> New to create a new sketch and delete the default code that appears inside. Click Copy Code on the window below, return to Arduino, paste the code into the sketch and save it.

```cpp
#include <Servo.h>
Servo servo;
int numberRecvd;
String dataString = "";

void setup(){
  Serial.begin(9600);
  pinMode(5, OUTPUT);
  servo.attach(5);
  servo.write(0);
}
```
void loop(){
  if (Serial.available() > 0) {
    dataString = "";
    while (Serial.available() > 0) {
      dataString += char(Serial.read());
      delay(2);
    }
    numberRecvd = dataString.toInt();
    switch (numberRecvd) {
      case 1:
        servo.write(125);
        break;
      case 2:
        servo.write(90);
        break;
      case 3:
        servo.write(55);
        break;
      default:
        servo.write(0);
        break;
    }
    Serial.flush();
    Serial.print("received: ");
    Serial.println(numberRecvd);
  }
  delay(20);
}

Go to Sketch -> Upload to send the project code to your Feather.

Reposition pointer

After the code is uploaded, you should hear the servo activate as it positions to its zero point.
Loosen the screw from the servo’s output shaft and reposition the horn and pointer so that they’re pointing to the right as seen above, with the servo wires feeding out below).

Secure the horn & pointer in place with the screw.

Usage

In the Unity project, open the Hub-edit scene & press the Play button.

Put the NextMind headset on and press the power button. In the Unity player pane, click on Calibration, connect to the NextMind, and complete the entire calibration process.

When calibration is complete, return to the main menu and click Discovery.
You should see the three CubeTag objects you created and edited earlier.

Choose a CubeTag object and focus on it, triggering the green triangle to form at its center.

At this point, the Unity scripts should send a serial message to the Feather indicating which cube was focussed, in turn causing the servo pointer to rotate, indicating that same cube you chose.