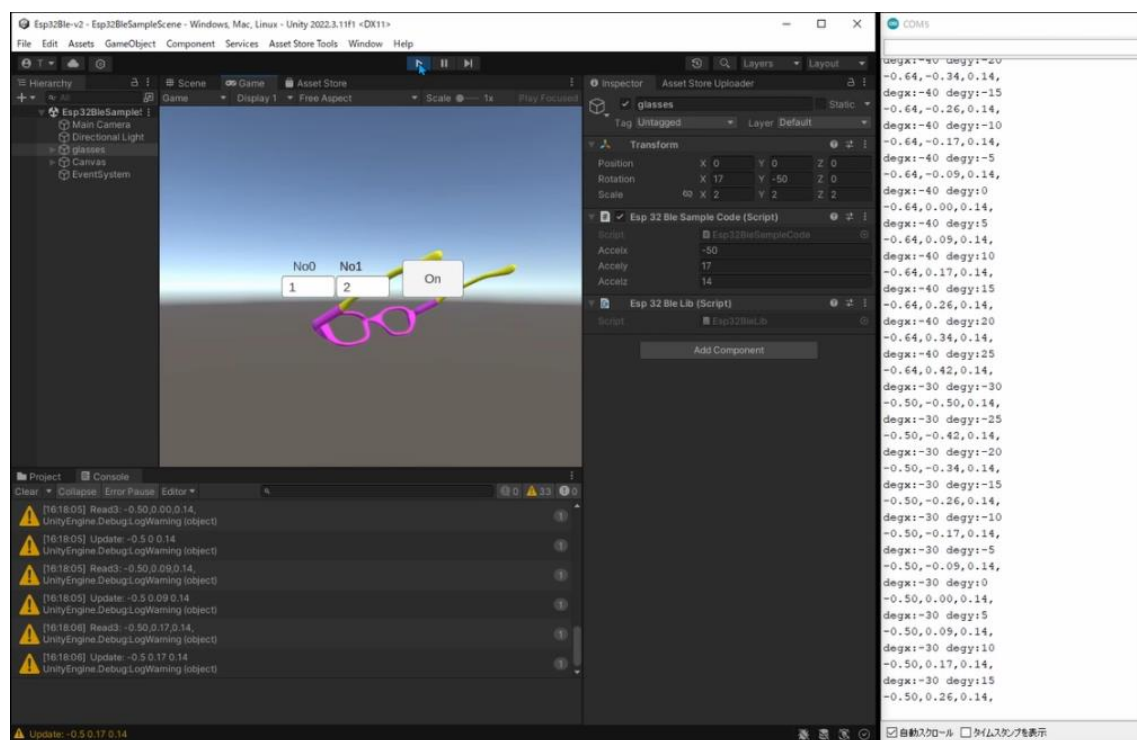


# Esp32Ble Quick Start

A Unity 3D object 'glasses' in the Demo scene tilts using acceleration data of the Esp32. The ESP32 generates 3D Object position data 'x, y, z' using the SIN function. The Esp32 sends that data from the Esp32 to Unity through Windows PC using the BLE (Bluetooth low energy) interface.

When clicking a Unity 3D object 'Button' in the Demo scene, Unity sends the numerical data obtained from the Unity 3D object 'Inputfield' to the Esp32. Unity works at Windows PC and sends the numerical data to the Esp32 using the BLE (Bluetooth low energy) interface.

The following shows the capture screen when the Demo scene runs.



## [Demo scene](#)

This Quick Start explains how to display the Demo scene on Unity using Esp32Ble. This Quick Start also includes some steps for downloading and setting up a package in Unity Project, and writing the Esp32 code.

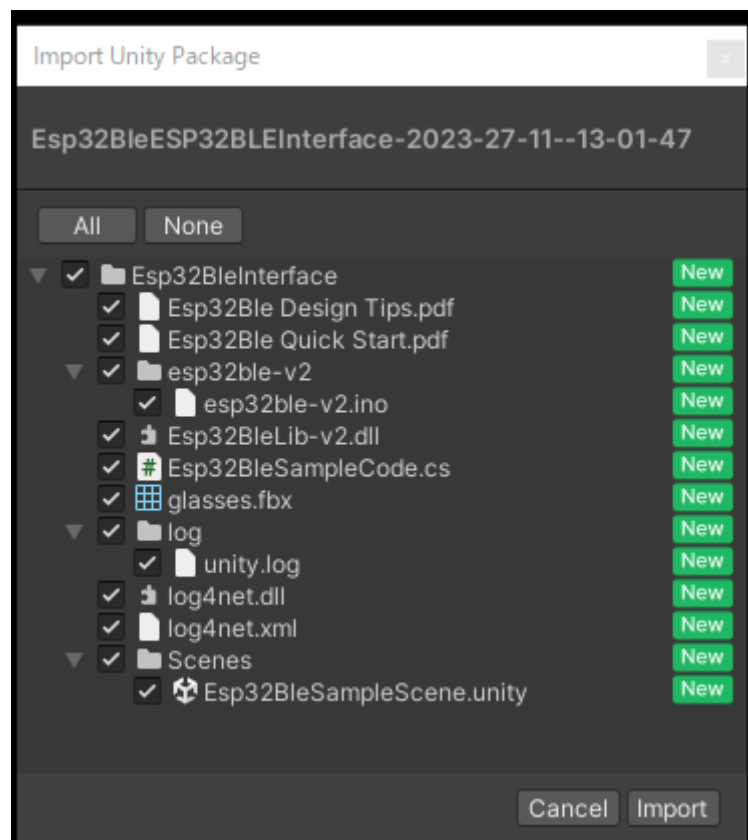
Step-1 Download and set up a package in Unity Project

Step-2 Write Esp32 code using Arduino IDE

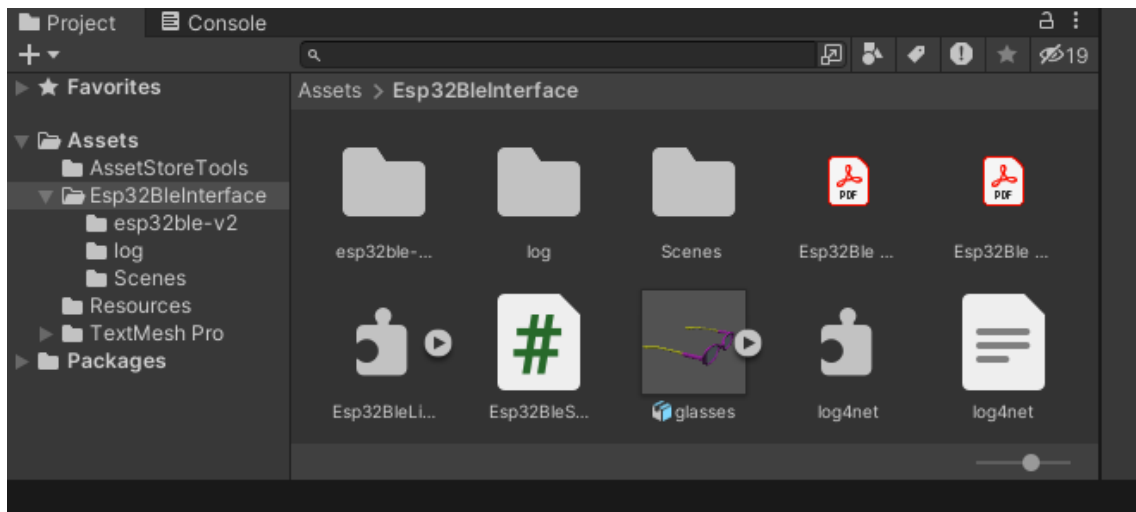
Step-3 Run Demo scene

## Step-1 Download and set up a package in Unity Project

(1) Download Esp32Ble from the Asset Store, and click on the Import button.



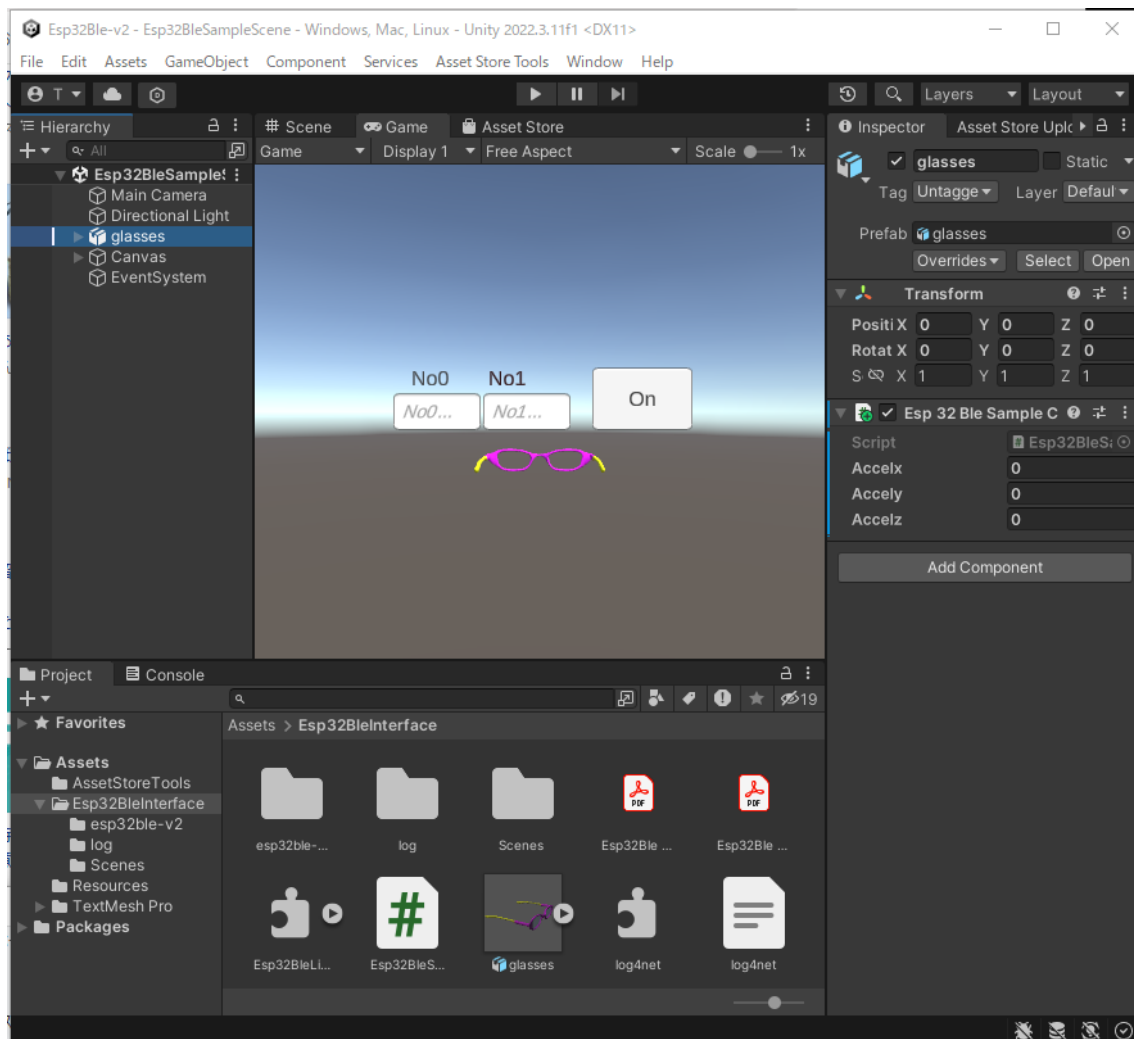
(2) The package is imported under the Assets folder in your Unity project.



(3) Download the external plugin 'MicroBleConnect-v2' below and set it into the 'Assets/Esp32BleInterface' folder.

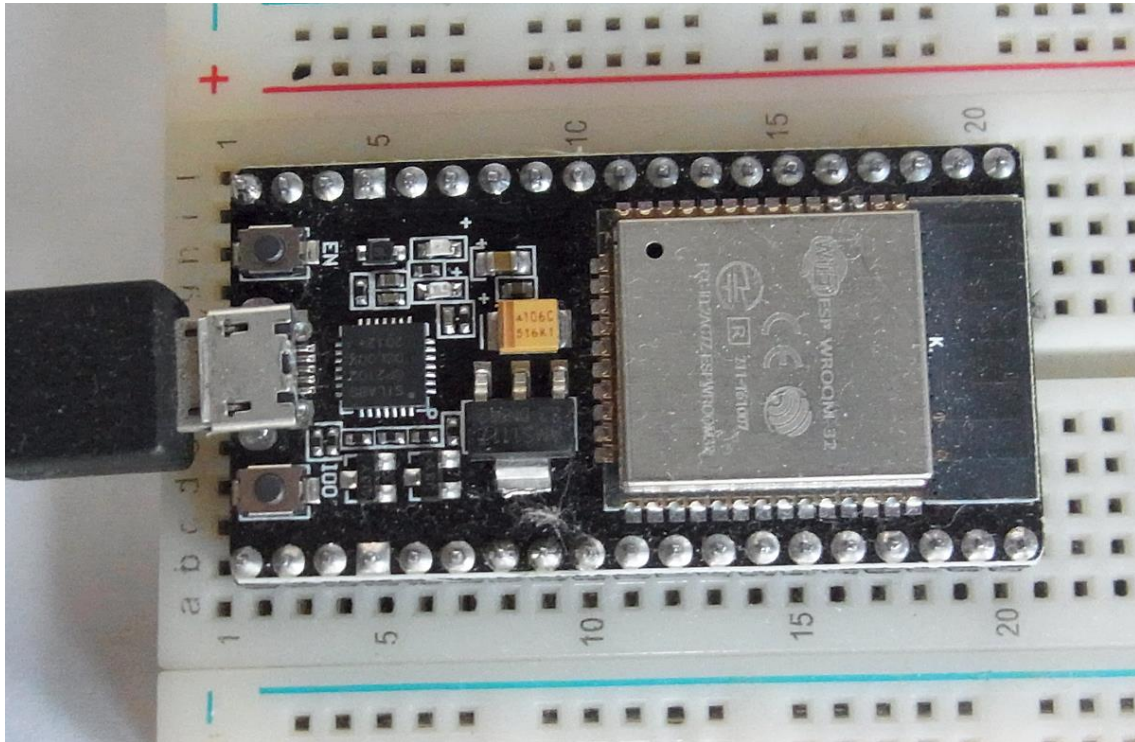
- [external plugin 'Esp32BleConnect-v2'](#)

(4) Click the Project tab, choose 'Assets' > 'Esp32BleInterface' > 'Scenes' folder of the left side menu, double click on the Demo scene 'Esp32BleSampleScene' in the Project window.



## Step-2 Write Esp32 code using Arduino IDE

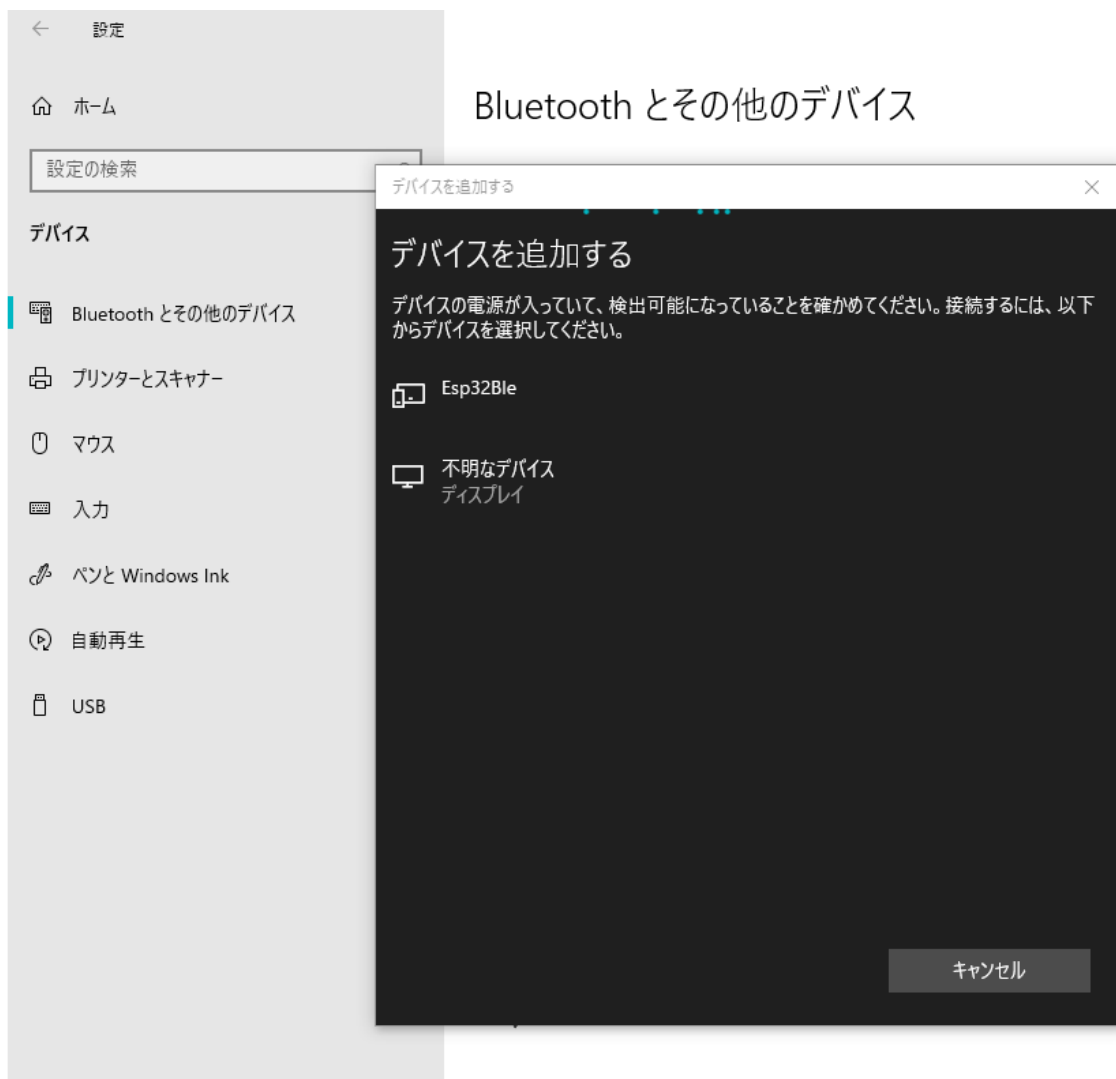
(1) Connect the Esp32 by Micro USB cable to Windows PC, then build a software development environment for Esp32Ble with the Arduino IDE according to [Arduino IDE for Esp32Ble](#).



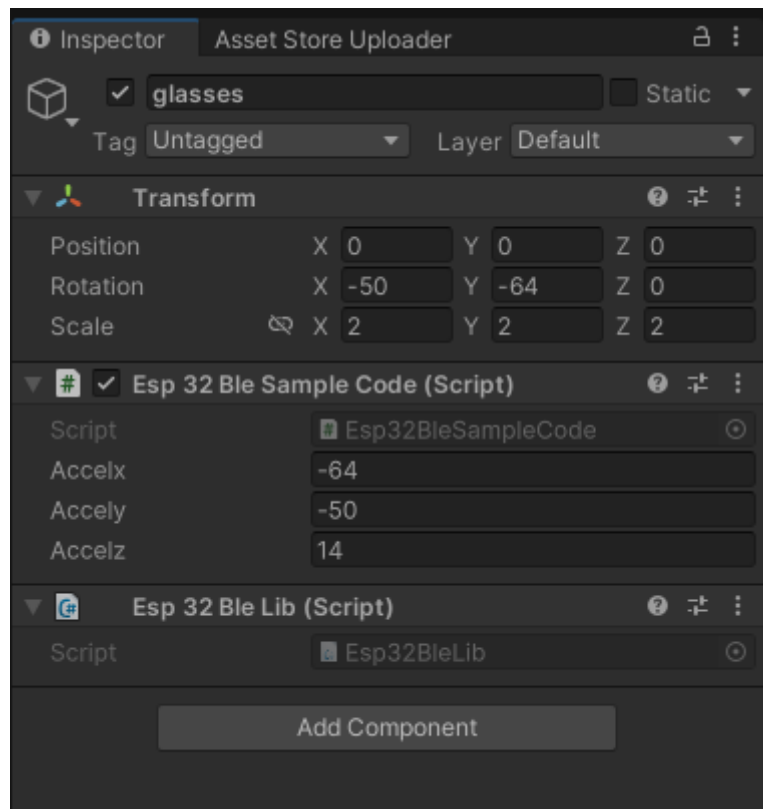
- (2) Compile and write Esp32 code 'esp32ble-v2.ino' of the folder '/Assets/Esp32BleInterface/esp32ble-v2' into the Esp32 using the Arduino IDE.

### Step-3 Run Demo scene

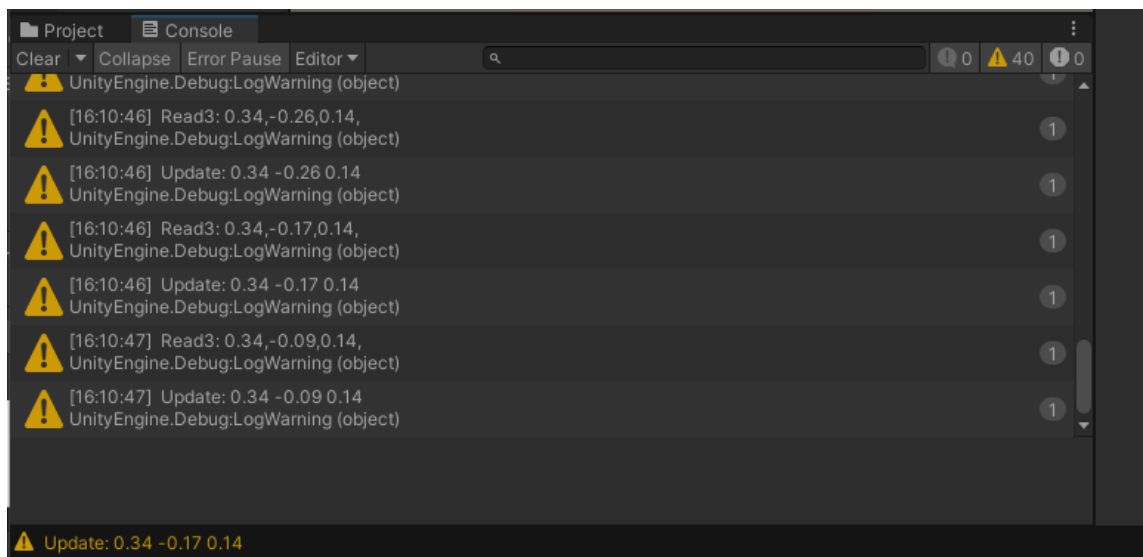
- (1) In Windows PC, go to 'Start', choose 'Settings' > 'Devices' > 'Bluetooth and other devices settings'. Click 'Add Bluetooth or other device' to start searching for the device.
- (2) Like below, select the Bluetooth device 'Esp32Ble' you want to add from the list.



- (3) Click the 'Play' button on Unity. The Unity 3D object 'glasses' tilts in sync with the Esp32 when inclining the Esp32. The 'Accelx, Accely, Accelz' in the 'Esp32Ble Sample Code' of the 'Inspector' tab displays the received data 'x, y, z' from the Esp32 as follows.



(4) The 'console' tab displays the game status as follows.

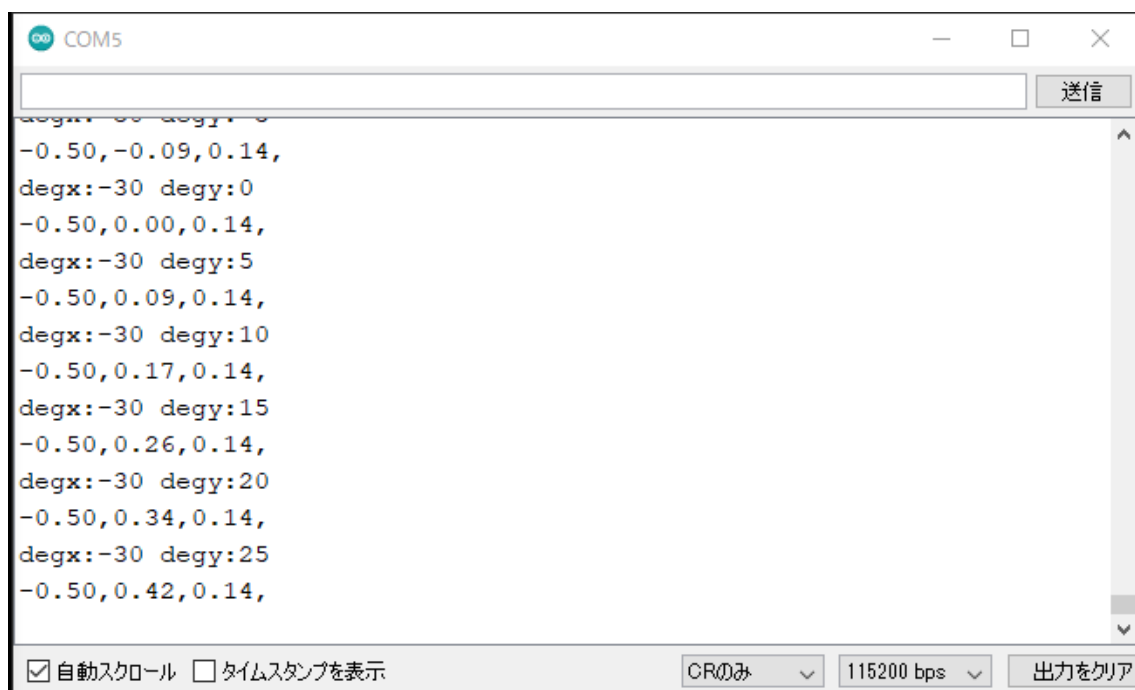


(5) The 'Serial Monitor' of Arduino IDE displays the data "x, y, z" as follows.

[ Not connect to Unity ]



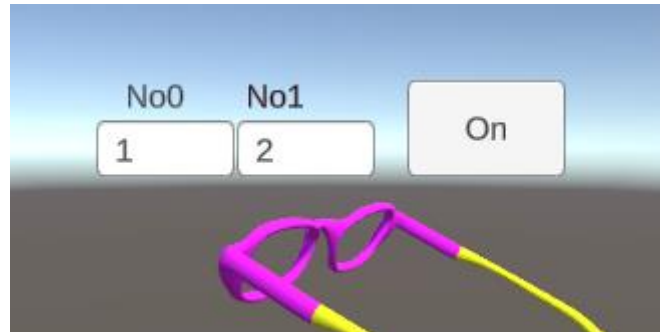
[ Connect to Unity and send acceleration data ]



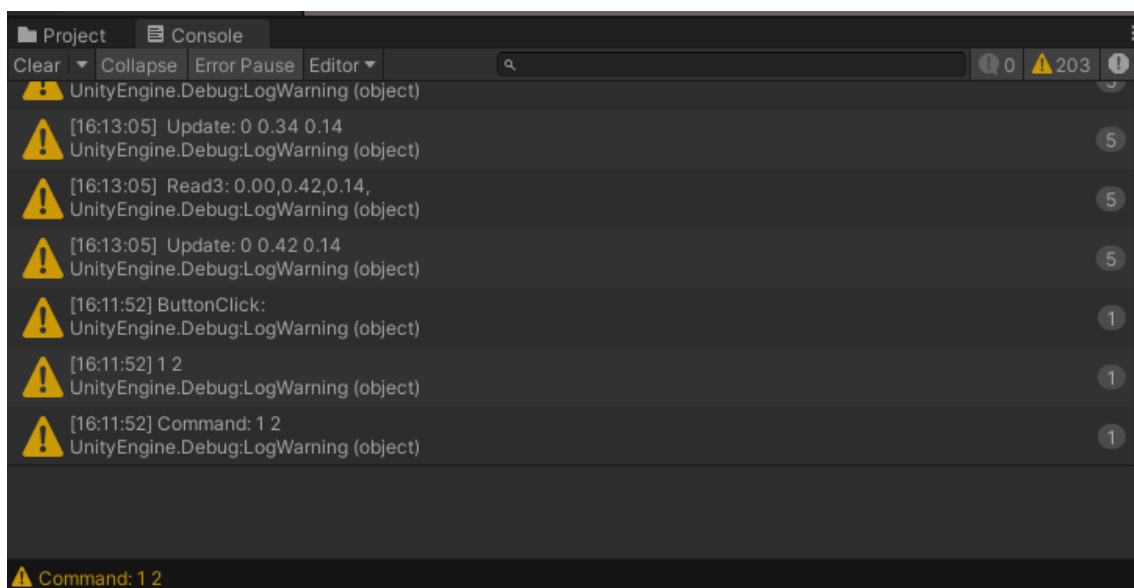
(6) Set No0 / No1 into Inputfield 'No0' / Inputfield 'No1' in the game screen, then



click the Button 'On' to send to the Esp32. Show Inputfield 'No0' / Inputfield 'No1' as follows.



(7) The 'console' tab displays the game status at clicking Button 'On' as follows.



(8) The 'Serial Monitor' in Arduino IDE displays the No0/No1 data as follows.

COM5

送信

```
-0.64,-0.26,0.14,  
degx:-40 degy:-10  
-0.64,-0.17,0.14,  
degx:-40 degy:-5  
-0.64,-0.09,0.14,  
degx:-40 degy:0  
-0.64,0.00,0.14,  
***** receive data! *****  
1  
2  
degx:-40 degy:5  
-0.64,0.09,0.14,  
degx:-40 degy:10  
-0.64,0.17,0.14,  
degx:-40 degy:15
```

☐ 自動スクロール ☐ タイムスタンプを表示

CRのみ

115200 bps

出力をクリア