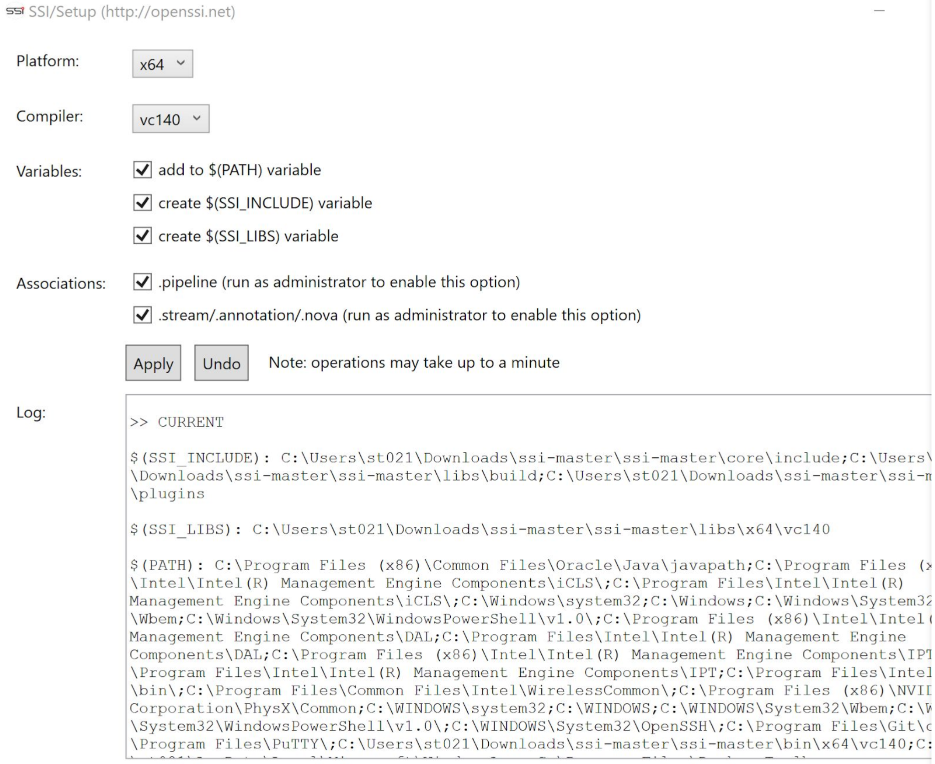
# Windows Install

Install Link: https://github.com/hcmlab/ssi
Guide: https://rawgit.com/hcmlab/ssi/master/docs/index.html#installation

- Download SSI Zip
- Extract
- Run setup.exe as an administrator, check all of the boxes , click apply, then done
- 

SSI/Setup (http://openssi.net)

Platform: x64

Compiler: vc140

Variables:
☑ add to $(PATH) variable
☑ create $(SSI_INCLUDE) variable
☑ create $(SSI_LIBS) variable

Associations:
☑ .pipeline (run as administrator to enable this option)
☑ .stream/.annotation/.nova (run as administrator to enable this option)

Note: operations may take up to a minute

Log:

```
>> CURRENT

$(SSI_INCLUDE): C:\Users\st021\Downloads\ssi-master\ssi-master\core\include;C:\Users\
\Downloads\ssi-master\ssi-master\libs\build;C:\Users\st021\Downloads\ssi-master\ssi-m
\plugins

$(SSI_LIBS): C:\Users\st021\Downloads\ssi-master\ssi-master\libs\x64\vc140

$(PATH): C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\Program Files (x
\Intel\Intel(R) Management Engine Components\iCLS\;C:\Program Files\Intel\Intel(R)
Management Engine Components\iCLS\;C:\Windows\system32;C:\Windows;C:\Windows\System32
\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Program Files (x86)\Intel\Intel(
Management Engine Components\DAL;C:\Program Files\Intel\Intel(R) Management Engine
Components\DAL;C:\Program Files (x86)\Intel\Intel(R) Management Engine Components\IPT
\Program Files\Intel\Intel(R) Management Engine Components\IPT;C:\Program Files\Intel
\bin\;C:\Program Files\Common Files\Intel\WirelessCommon\;C:\Program Files (x86)\NVID
Corporation\PhysX\Common;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\W
\System32\WindowsPowerShell\v1.0\;C:\WINDOWS\System32\OpenSSH\;C:\Program Files\Git\c
\Program Files\PuTTY\;C:\Users\st021\Downloads\ssi-master\ssi-master\bin\x64\vc140;C:
```

# Testing SSI Master

- Pipes ->>
  - double click audio.pipeline and select own microphone it will record
  - Double click the usb, it is your webcamera
- Recordings do not auto save to folder
- To save outputs; you must add an additional consumer
- Set overwrite to false when collecting data. Otherwise the file will write over itself. Each file will be set sequentially. You could add a timestamp potentially.

| | |
|---|---|
| Audio | <consumer create="WavWriter" path="audio.wav" overwrite="true"><br><input pin="audio" frame="512"/><br></consumer> |

|  |  |
|---|---|
| Video | <consumer create="CameraWriter" option="camerawriter" overwrite="true" path="camera.avi"><br>        <input pin="video" frame="1" delta="0"/><br>    </consumer> |

Ran both Audio and Video Clients

## Synchronization

To turn a pipeline into a server add the line:

```
<framework sync="true" slisten="false" sport="1111" sdialog="true"/>
```

To setup a client add:

```
<framework sync="true" slisten="true" sport="1111"/>
```

These are pasted underneath <pipeline ssi-v=”1”>
Open all clients before running any of the server options.

## Modification

When modifying you may need to add a register to use certain functions. For examples to manipulate signals you need to load “ssisignal”

I tried to modify the camera_write pipeline but image isn’t supported for the limits sliders

------------

Empatica documentation

To run

1. Open the E4 streaming server on a windows computer
2. Login using the developers code associated with empaticas
3. Pick an empatica
4. Make sure the empatica usb dongle is attached
5. If E4 streaming server says discovering, hold down the little button on top left and that should connect, this is visually indicated by a blinking Blue light
6. Click start discovery

7. Once it appears in the streaming server click connect
8. Locate the DeviceID of the device (country name) that you have
9. Insert this ID into the deviceid = line of the appropriate empatica_processing_mw.py file
    a. For up to 3 empatica that you are collecting data from, a separate empatica_processing_mw.py file can be made with each device's deviceID
    b. If the deviceID isn't in the table run the following code in command line:
        i.

```
import socket
import time
import pylsl
import os
from subprocess import Popen, PIPE
serverAddress = '127.0.0.1'
serverPort = 9999
bufferSize = 4096
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.settimeout(3)
s.connect((serverAddress, serverPort))
s.send(b'device_discover_list 2 | BC3864 Empatica_E4
allowed\r\n')
```

    c. To modify the pipeline, for multiple sensors, create a sensor with each respective python file.
10. To run the pipeline open the command line, cd to the correct folder, and enter `empatica_multiprocessing.pipeline > test_gym.txt`
11. Pipe the command output to a file to catch errors etc for the next run
12. Each empatica will create an output text file.

Installation files

- SetupEmpatica_2.0.3.5119_driver_dotnet
- Python 3.73
- SetupEmpaticaBLEServer_1.0.1.4930