Using the Heavy compiler to compile Pure Data patches (for Bela)

If you want to use the heavy compiler to compile your PD patches, you will need:

- the heavy compiler repository
- python (should come pre-installed on your board, *tested in 2.7.13*)
- the python package manager
- some packages for python
- an internet connection on your board to download the repository and packages
- make sure your bela image is up to date (everything below is tested on version)

Installation&preparation:

The first step is easy, just connect your board to your pc with the usb-cable, and plug in an ethernet cable to your board. If it doesn't immeadiately work, you may have to reboot, or replug the cable. (You can also connect to the internet through WiFi or share the network of your computer over USB. See “Network Connections” at the end of this article.)

You can connect a terminal window from your computer to the bela by setting up an ssh connection:
On Linux/Mac just open a terminal, and enter:

```
 ssh root@192.168.6.2 (Mac: ssh root@192.168.7.2)
```

For Windows users: I recommend using the free program *Putty,* since it is easy to use and you can save and name your connections. (see links below for instructions & how to download.)

Putty:
download:
https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html
instructions:
https://mediatemple.net/community/products/dv/204404604/using-ssh-in-putty-

notes: In “Host name (or IP adress)”, enter 192.168.6.2
leave port 22 as default. You can enter a name of choice for your connection, so next time you open putty, you can just click it and it will start up a connection with your Bela.

Every time you connect, the terminal wil ask you to: “`login as:`” enter: “`root`”.   If it asks you for a password, and you've not set any, the password will be “`root`” as well. When you are logged in; the Bela logo will appear in the terminal, and you are now logged in as “ `root@bela: ~#` ”. you are now running commands on the Bela.

To check which version of python you have, just enter “`python`” in the command line.
If it is installed; it will show you the version of python, and start it. To quit, press ctrl-d. (note: I am running everything described here in python 2.7.13)

next, you need to make sure you have the python package manager installed. In the command line, enter:

```
apt-get install -y python-pip python-setuptools
```

You can use the heavy compiler either from your pc, or directly from the Bela. I will first explain how to use it directly from the Bela; as this is the most easy way. Make sure your internet connection works, and enter the following code in the terminal window:

```
git clone https://github.com/giuliomoro/hvcc.git
```

The heavy repository is now installed. the last thing you need to install are some packages for python: enum, and jinja2. They are included in the heavy repository. To install, go to the directory where you installed it. To do this, enter:

```
cd hvcc/
```

to install the needed packages, enter:

```
pip install -r requirements.txt
```

Using the heavy compiler

Make sure :

- you are working from the hvcc/ directory (as mentioned above: `cd hvcc/`)
- (on Bela:) your main file is called "main_.pd"
- your pd files only contain supported pd objects (most externals/abstractions and libraries are not supported). Unfortunately the enzienaudio website is not online anymore, so there isn't a list of supported objects you can see. Keep this in mind when continuing with the next step,

Run the following code:

```
python2.7 ~/hvcc/hvcc.py ~/Bela/projects/YourProjectName/_main.pd -o
/tmp/hvtmp -n bela -g c

mkdir -p ~/Bela/projects/yourHeavyProjectName

rsync -av ~/Bela/scripts/hvresources/render.cpp /tmp/hvtmp/c/*
~/Bela/projects/yourHeavyProjectName/

make AT=  -C ~/Bela PROJECT=yourHeavyProjectName COMPILER=gcc run
```

notes: `~/Bela/projects/YourProjectName/_main.pd` refers to the path of the project you want to compile.

`~/Bela/projects/yourHeavyProjectName` refers to the path where you're going to save your new compiled project.

`~/Bela/scripts/hvresources/render.cpp` refers to the path where the render.cpp file for the heavy compiler is (in this case the default path on the bela. You can directly copy this path from here, unless you have to use a custom render.cpp file)

"`~/`" refers to the Bela's root folder.

Network connections

Setting up Network connection over USB on the Bela

https://www.youtube.com/watch?v=fzRVVtGNfj8&feature=youtu.be&t=336

To install some packages, you will need an internet connection on your board. Plug in an ethernet cable, and type the following code in the IDE terminal or any terminal connected to bela:

```
apt install net-tools
apt install iproute2
```

you now have the directory `/sbin/route`

enter the following code:

```
/sbin/route add default gw 192.168.6.1
```

to add your pc as a default network gateway.

(If you ever have to reset this, or want to delete it, delete the default gateway: `/sbin/route del default gw 192.168.6.1`)

Windows users: if you are pinging 192.168.6.1 and it is not returning anything; you should go to settings> firewall & network protection >> advanced settings >> inbound rules >> scroll down to "file and printer sharing" and turn it on. It should work now. (if it doesn't, reset the above steps.)

The first time this immediately worked for me, but when I restart, I sometimes lose connection, unable to set it up again in the same way; but resetting all network settings in windows and then restarting fixes that for me.