

-----INSTALLATION-----

PCAN:

Download latest version -

<http://www.peak-system.com/fileadmin/media/linux/index.htm>

Follow the install instructions. The current version at the time of this writing is Release Version 6.11. The installation will look for "version.h" in the /usr/src/linux folder. In Ubuntu this folder does not exist so go to the folder and create a symbolic link to redirect:

```
cd /usr/src
sudo ln -s linux-headers-2.6.xx-yy-generic linux
```

(replace xx and yy with your kernel version that you find in this folder). Then change directory where you downloaded the linux driver and follow the driver installation instructions:

```
cd ~/peak-linux-driver-x.y
make clean
make NET=NO
sudo make install
```

Note that setting NET=NO compiles the driver WITHOUT netdev configuration which is totally confusing in the installation instructions. Also make sure you have g++ installed on Ubuntu before the compilation. After compiling the sources and the following binary installation on your target computer it's enough to type in

```
sudo /sbin/modprobe pcan
```

Python:

- Install python2.6-all (particularly python2.6-dev)
- Install IPython

C++:

- Install g++ 4.4.3 or greater

Visualization

- Install GLUT packages

Other required libraries:

-----Running the Simulation-----

Hardware Setup:

- 1) Connect the PCAN bus between the CKBot modules and the workstation
 - Check that it is connected properly with: `/sbin/modprobe pcn`
- 2) Plug in Phidget board
 - Ensure that the resources cradles have 6V input

Software Setup for simulator:

- 1) In `../factoryfloor/ccl-ff/` input: `source init_env`
- 2) In `../factoryfloor/ccl-ff/ff`, set the location of the Hardware in Test in "DynamicFFTB.ccl" using the global 'HIT'
 - Ex: `HIT := 0`, is the bottom left tile of the FFTB.
 - Ex: `HIT := 1`, is the tile north of the bottom left tile.
- 3) In `../factoryfloor/ccl-ff/ff/pytalk`, run receiver `./receiver`
 - Interface between the simulator (DynamicFFTB.ccl), and the CKBot control in Python (FFTB_cntrl.py)
 - Requires a FIFO called "fifo" in the folder.
- 4) In `../factoryfloor/ccl-ff/ff`, run the simulator `"ccli DynamicFFTB.ccl"`

To test path planning outside of simulator:

- 1) Go to `../factoryfloor/ccl-ff/ff/pytalk`, run IPython
- 2) Input: `execfile("FFTB_cntrl.py")`
- 3) Input: `c = Cluster()`
 - Initializes python modules for control

4) Input: c.populate()

- Connect to the modules themselves, checks the 'heartbeats' of the CKBot modules
- Check to see that all modules are seen by the workstation with: c.at.<tab>, the modules should be listed

Ex: NxCO

NxD5

etc.

5) Run.

a) Individually move each servo with c.at.<module>.set_pos(<num>)

where: <module> is ID Tag, Ex: NxCO

<num> is a position value between -9000 and 9000.

b) Run the python scripts

- neutral_state() -- moves to a default position
- retrieve_truss() -- script for truss retrieval
- retrieve_node() -- script for node retrieval (spelled incorrectly for simple tab completion)
- place_node() -- script for node placement
- place_truss(1/2) -- input (1) places trussX, (2) places trussY
- place_support_truss() -- script for trussZ placement
- pass_truss()
- pass_node()
- *- go_to_sleep() -- shuts off control of CKBots

-- > logical.py is used by FFTB_cntrl.py, and is required if a new path planning script is created