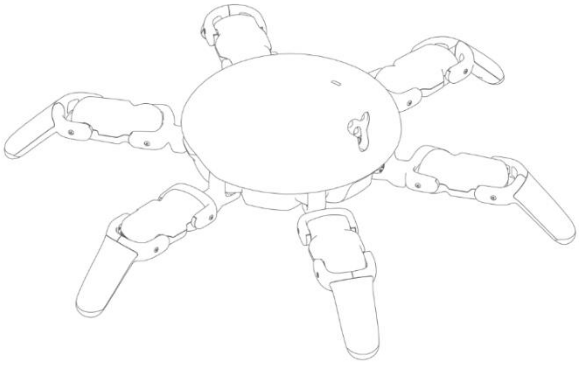
HEXABOT

Manual

by

Sujet Phodapol



Development of Hexabot

The project is to implement a neural control in Hexabot. Neural control is one of the control algorithms that use two oscillators to create a signal. The project can be divided into two main parts: robot and remote. The first task is to write a controller in Golang. The second part is to create a controller UI in HTML.

Full manual: file:///C:/Users/hp/Desktop/HEXA_User_Guide.pdf
One page manual for run: http://bit.ly/NeuralHEXABOT
Video : http://bit.ly/HexaClip

What is neural control?

Neural control is one of the control methods that can use in various applications. In this project, I use this method to control HEXA, six-legged robot. This algorithm tries to mimic the neural network of the organism. It use in two main purposes. The first one is to understand how insects or animals move their body. The researchers tries to create the model of the brain to explain the locomotion of the animals. The second purpose is to control the multi-legged robot using this method. By changing only one parameter, the robot can walk in several different gaits.

For more details:

https://www.frontiersin.org/articles/10.3389/fncir.2013.00012/full?fbclid=IwAR1JcixtAmE8oEPyl83S-e0WhKa0aVO6A_TR7tupA4rtJKTM-_YoJMMcpmw
http://manoonpong.com/paper/2007/Manoonpong-RAS07Submit.pdf

Neural Control for Hexabot

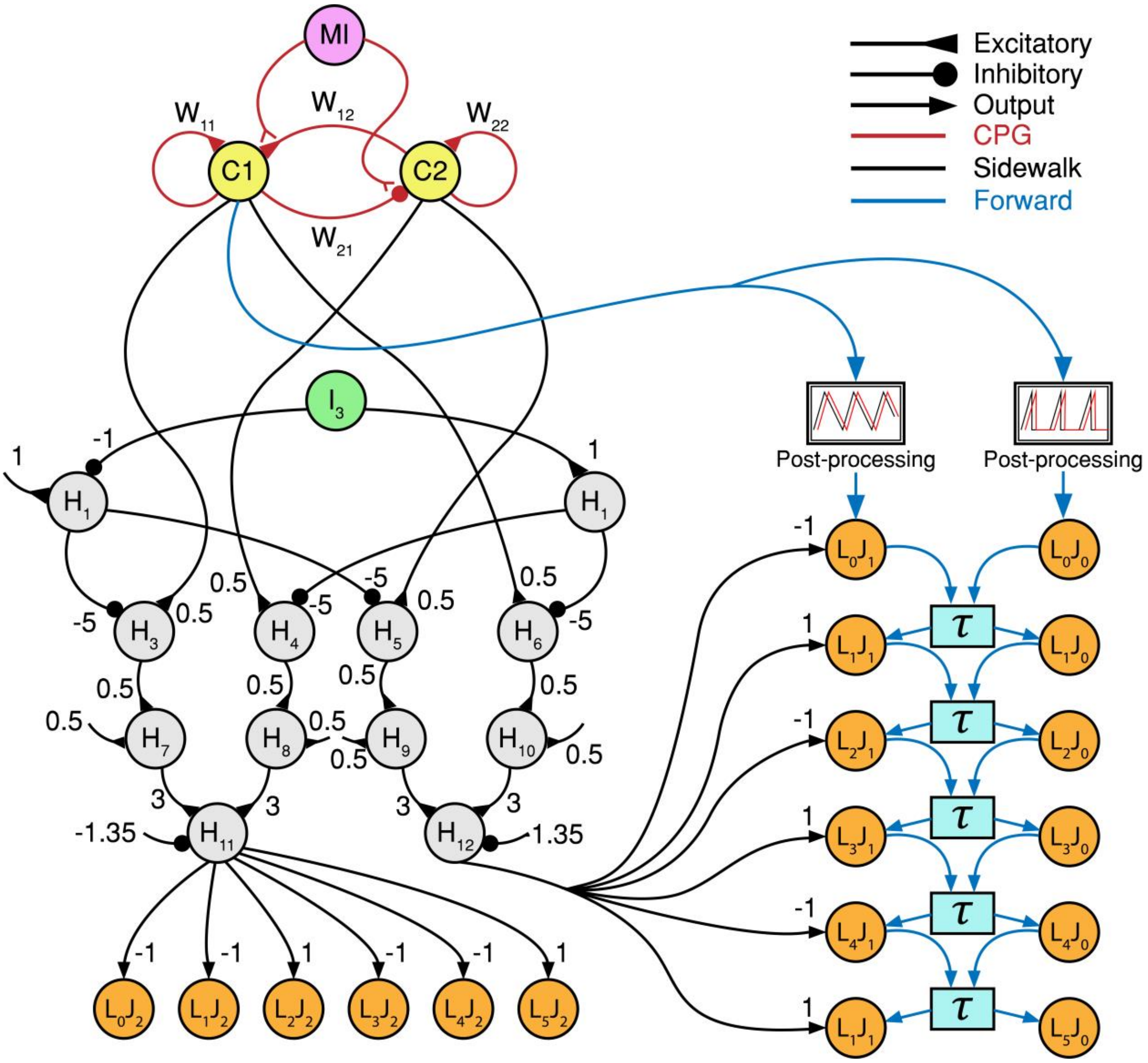
There are two main control in locomotion of Hexa. The first one is walking forward. The other is walking sideway. I use switch case to change between two option (It will be better, if all of the controllers is a single network).

Walking Forward

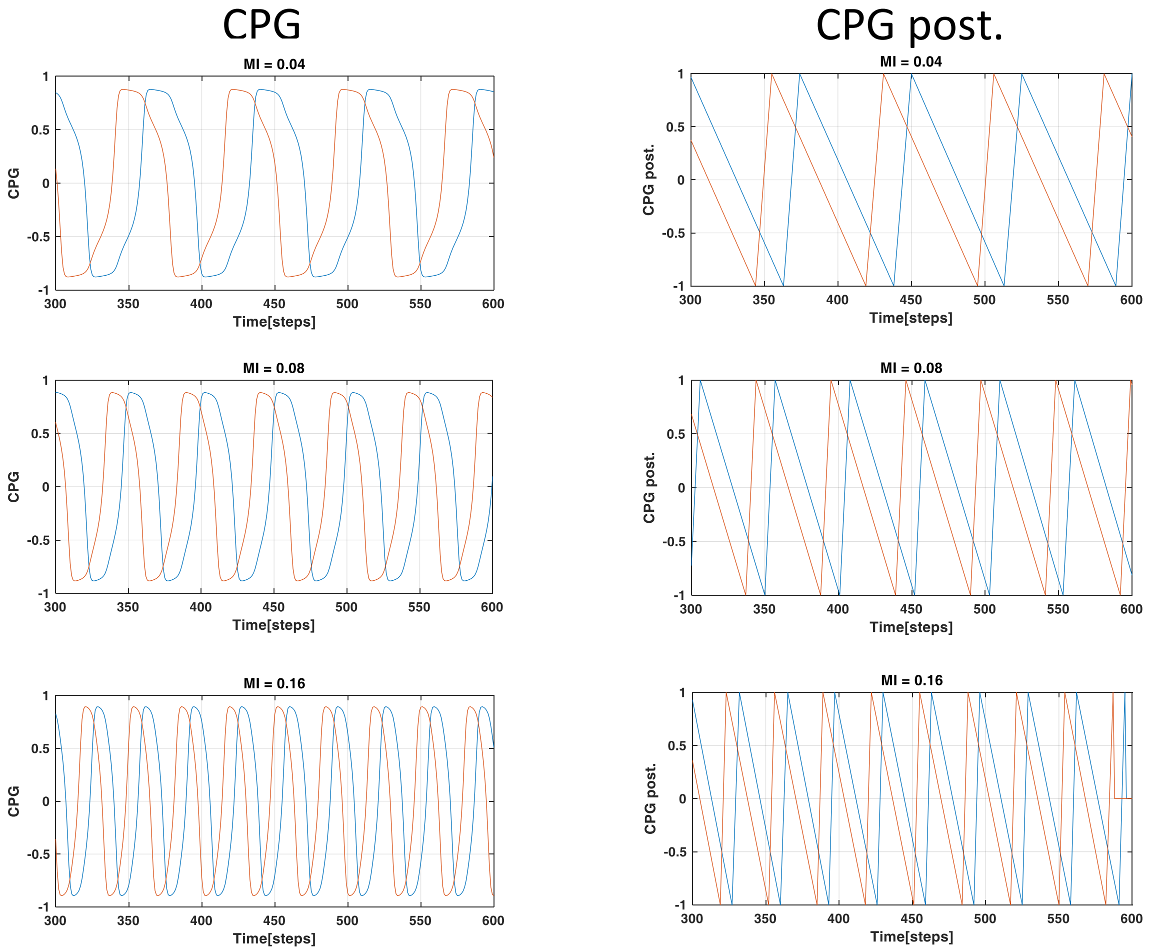
In this algorithm, I use CPG to generate signals and process them to sawtooth wave. The advantages of sawtooth wave are that you can adjust the duration of stance and swing phase. Moreover, when you change MI, if you use CPG, it will go wrong in high MI. Then, the sawtooth signals are sent to command each joint. The delay lines are used in order to change the gait depended on the values of MI. Hexabot can walk in five different gaits: slow wave, fast wave, tetrapod, caterpillar and tripod.

Walking Sideway

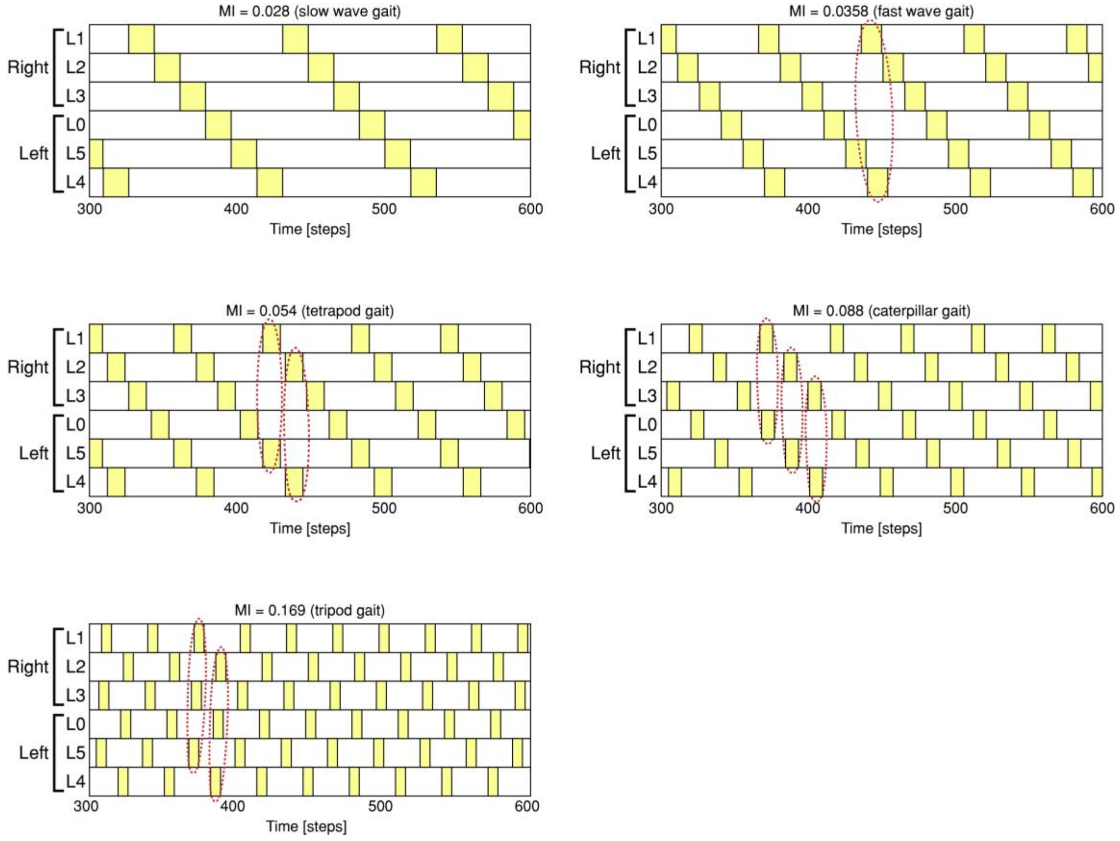
In this locomotion, I create a PSN, Phase Switching Network, in order to shift the phase by 90 degree. Since, I use the CPG signal to control the joints, it can walk only in tripod gait. I3 is used to change the line, so the lead and the lag signals will be switched. To improve this controller, I should make something like I3 rather than using switch case. If I do like that, I will have only one network.



Signals



Gaits



Requirement

Computer

1. Linux operating system (Ubuntu 16.04). I recommend that using dual boot is better than VM (virtual machine) since you will not have troubles with the connection.
2. Download your favourite editor. For me, I use VScode.
3. Install MIND SDK link: https://documentation.vincross.com/Introduction/mind.html

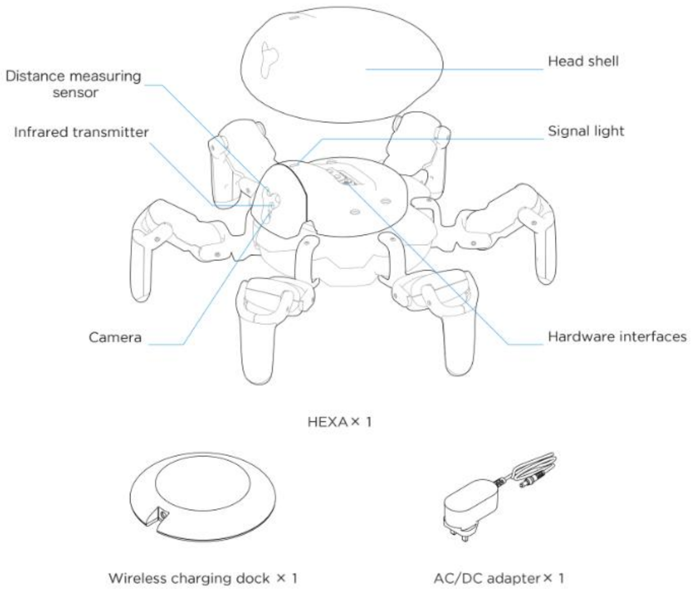
Iphone/Ipad

1. Download Vincross Hexa app from the app store or google play. In the application, you can monitor the batteries, connection status and manual control the robot.
2. Sign up the vincross account and connect to hexa.
3. Check the process by trying to control Hexa manually.

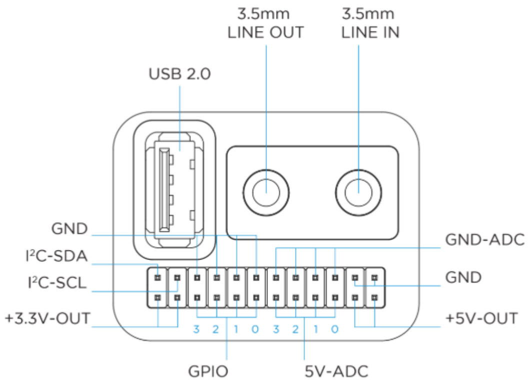
Hardware

For more details: https://documentation.vincross.com/Introduction/hardware.html

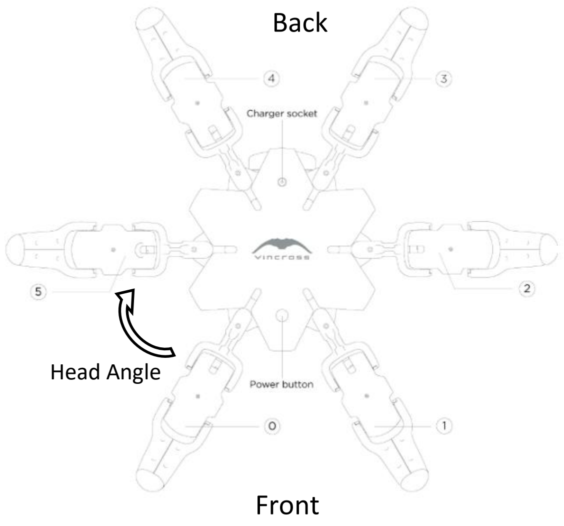
HEXA Inventory

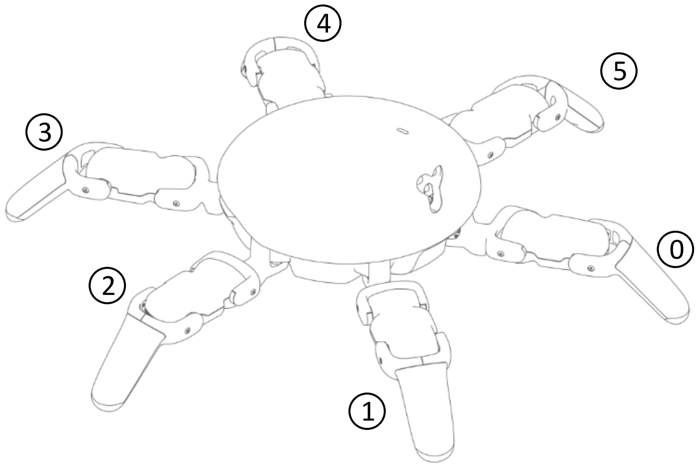


Hardware Interfaces

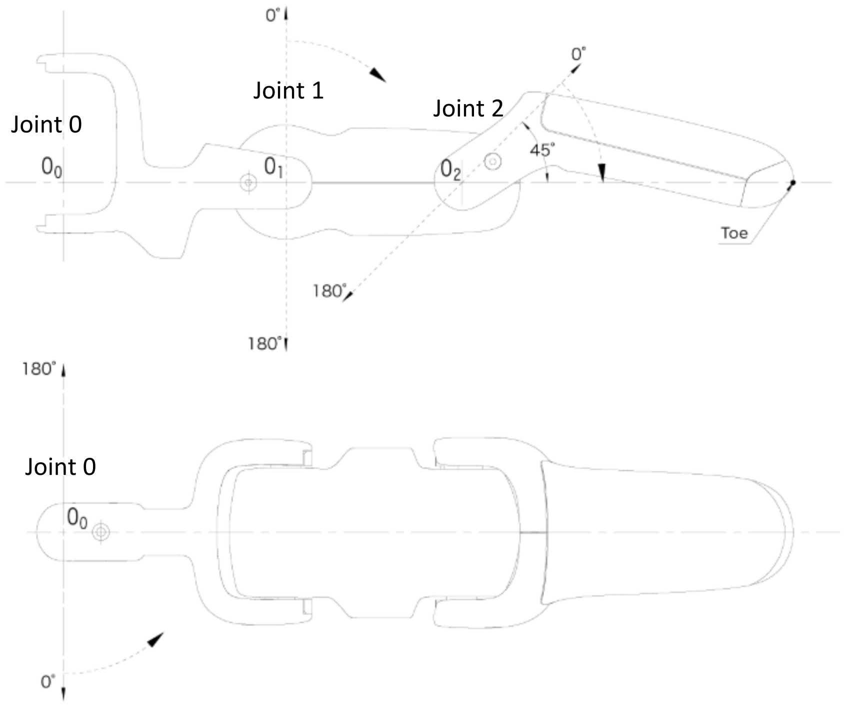


HEXA’s Legs





Range of Motion



Range in degree

Joint 0 (35 – 145)
Joint 1 (10 – 170)
Joint 2 (10 – 160)
Head Angel (0 – 359) *zero at power button

Skill

For more details: https://documentation.vincross.com/Development/yourfirstskill.html

1. Create developer ID and connect your ID with HEXA
2. Install MIND SDK
   For more detail: https://documentation.vincross.com/Development/macandlinux.html
3. Create new skill (In Hexa, the algorithm for robot is called skill)

```
$ mind init <yourskill>
$ cd <yourskill directory>
*for some reasons I need to add "sudo"
```

– Source code: https://github.com/pSujet/Hexa.git

Robot part

- Go to robot folder->src you will found Go script (*yourskill.go*).
- Copy the script and replace it in your go script, then change every package name to your skill name

Remote part

- Go to remote folder you will found html script (*Index.html*).
- Copy the script and replace it in your html script, then change every package name to your skill name

4. Build and package
   Before running the robot, you need to build and pack the skill.

```
$ cd <your skill directory>
$ mind build   # Will build an ARM binary into robot/skill
$ mind pack    # Will pack the Skill as an mpk file.
*for some reasons I need to add "sudo"
```

Skill Structure

In yourskill.go, you will find several function.

Onstart() – Called when Skill is started.
-> Start CPG + Send data to HEXA
OnClose() – Called when Skill is closed.
-> Stop walking and Relax
OnConnect() – Called when skilled is connected.
-> Start package
OnDisconnect() – Called when skilled is disconnected.
-> Stop package
OnRecvJSON([]byte) – Called when remote sent data.
-> Receive MI
OnRecvString(string) – Called when remote sent a string.
-> Receive command

Run

1. Before you can run the skill, you need to login first.

```
$ mind login yourID@mail.com yourpassword
*for some reasons I need to add “sudo”
```

2. Connect all device, including laptop, Ipad and HEXA in the same network. You need to use WiFi with password.
3. Send the code to HEXA. In the website, they suggest this method.

```
$ mind scan
10.0.0.51 HEXA
10.0.0.76 HEXA4XMZ
$ mind set-default-robot HEXA4XMZ # set a robot
$ mind get-default-robot # Just to be sure
HEXA4XMZ
$ mind run
```

However, I can’t use this method, instead I do like this.

```
$ mind run --ip <your robot ip>
*for some reasons I need to add “sudo”
*you can find robot ip in HEXA app in setting->developer->ip
```

Controller User Interface

You can create controller UI by editing *index.html* in remote folder. You can design every parts of your user interface in this script, including buttons, type of data and UI.

In PC

When you run the program, you will found link http://localhost:7597/. The controller UI appear in this link.

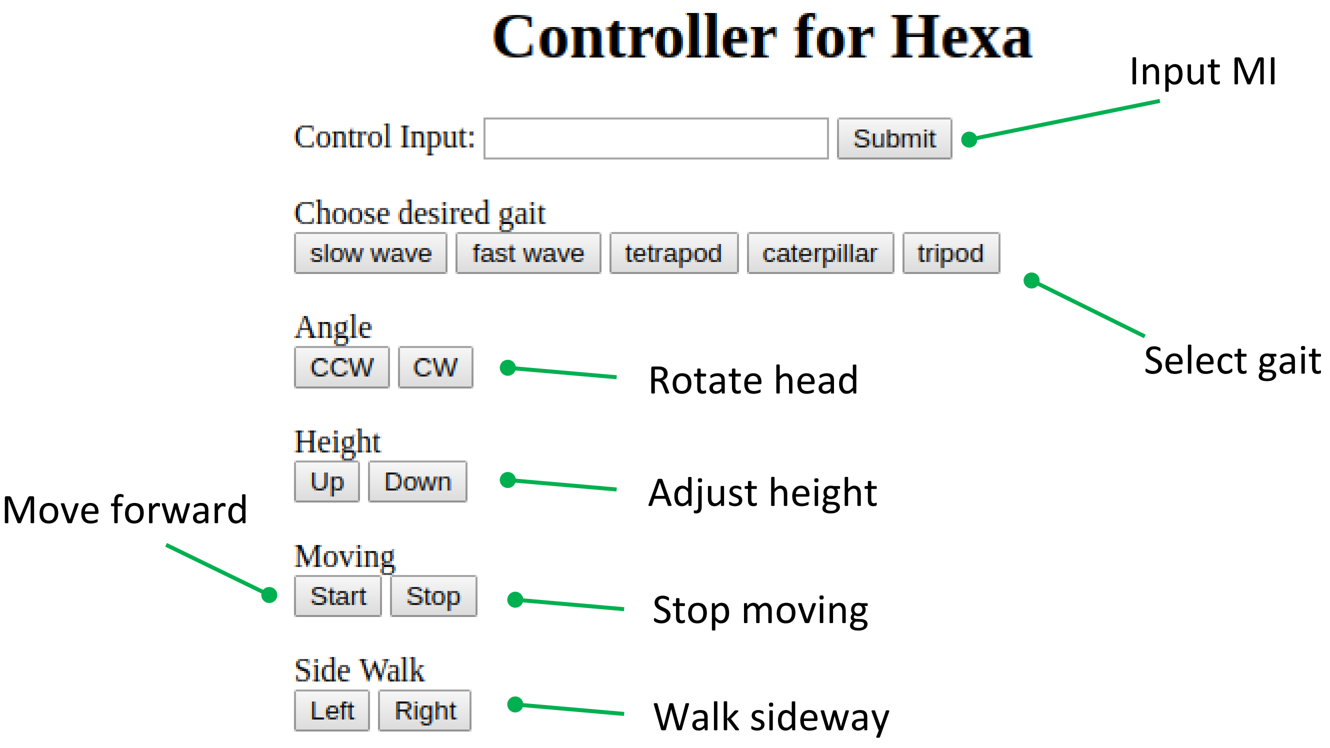
In Ipad/Iphone

You can also control the robot via mobile phone. In the web browser, you have to type laptop’s IP:7597 in order to connect to UI. Moreover, you can find your PC IP from typing this.

```
$ ifconfig
```

In your skill folder, you will find *manifesr.json*. The UI configuration can be defined in this file. For more details: https://documentation.vincross.com/Introduction/manifest.html

Controller



Controller for Hexa

Control Input:

Choose desired gait

Angle

Height

Moving

Side Walk

Input MI

Select gait

Rotate head

Adjust height

Move forward

Stop moving

Walk sideway