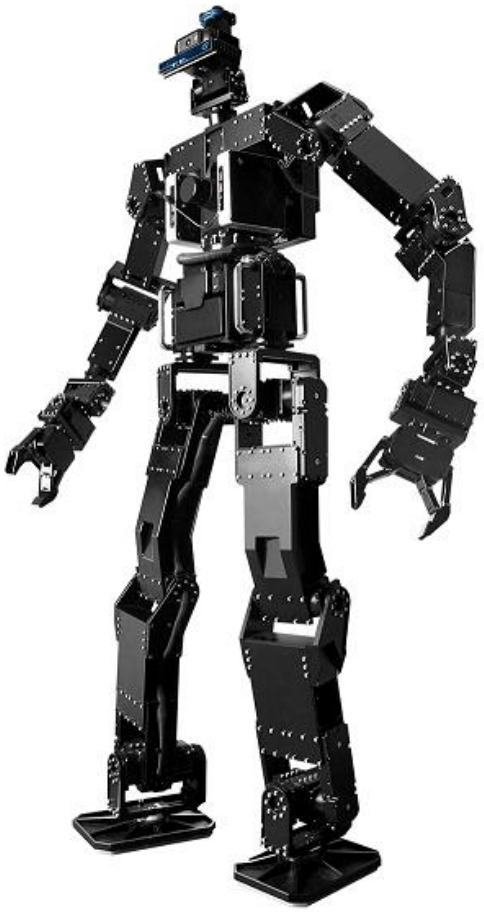


THORMANG3 Tutorial

Quick Start

Table of Contents

1. Introduction

Congratulations on purchasing THORMANG.
THORMANG will lead you to the robot world with virtually unlimited research possibilities and opportunities.
Read this guide thoroughly before getting with THORMANG.

1. What is THORMANG3

- THOR (Tactical Hazardous Operations Robot) is an affordable, full size humanoid robot platform with advanced computational power, sophisticated sensors, high payload capacity, and dynamic motion abilities to enable various exciting researches and educational activities.

2. Package Contents

1.	Fully Assembled THORMANG Robot	X 1
2.	Battery Packs (included in robot)	X 1
3.	Battery Charger	X 1
4.	Wrench Set	X 1
5.	Screwdrivers	X 2
6.	Spare Cables	
7.	Spare Nuts and Bolts	
8.	USB	X 1
9.	Quick Start Manual	X 1
10.	Pelican Case	X 3
11.	Carabiner and rope	X 2
12.	Lift	X 1
13.	Net hub(D-Link DIR-806A)	X 1

1


2


3


4


5


6


7


8


9

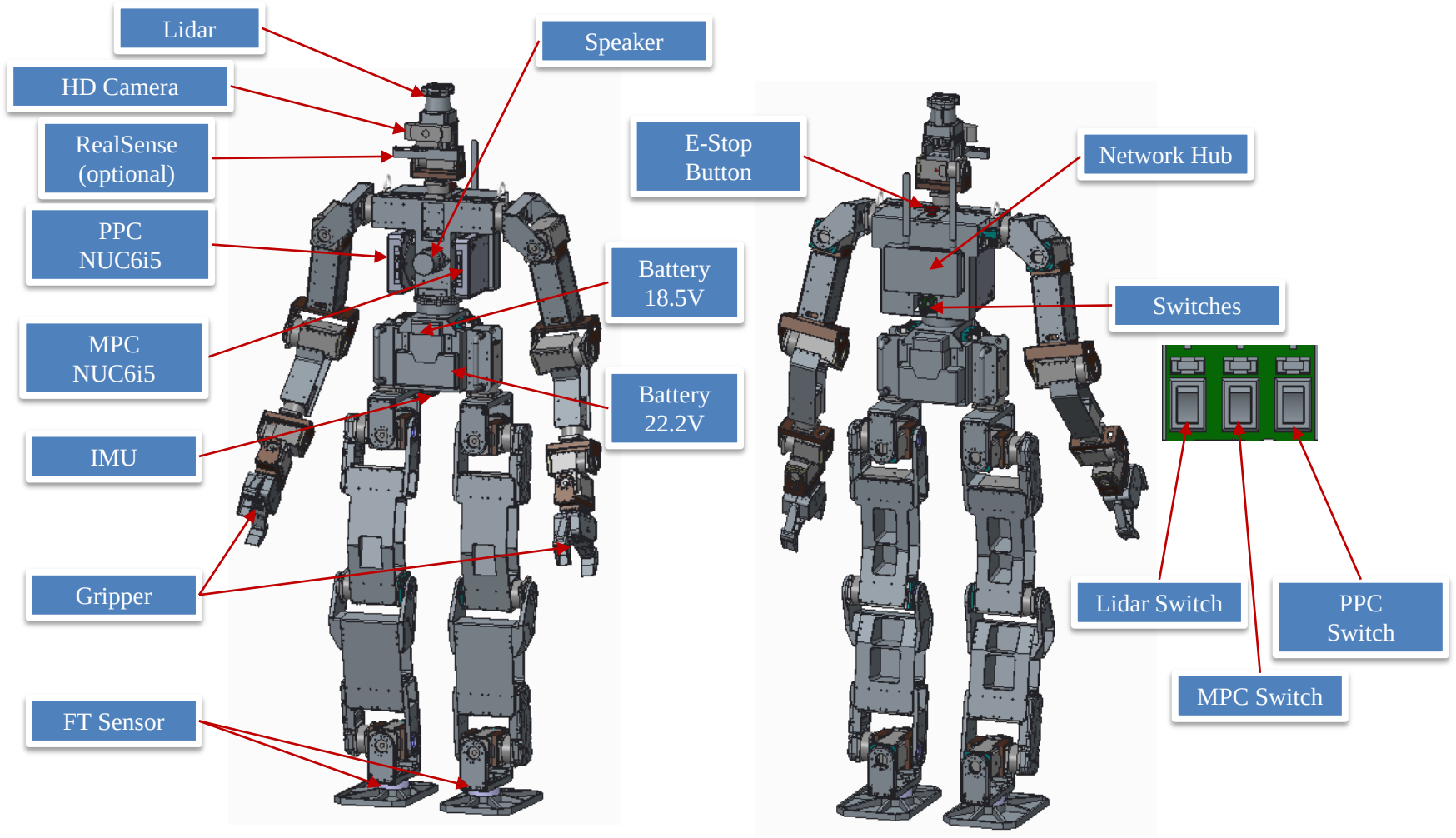

10


11

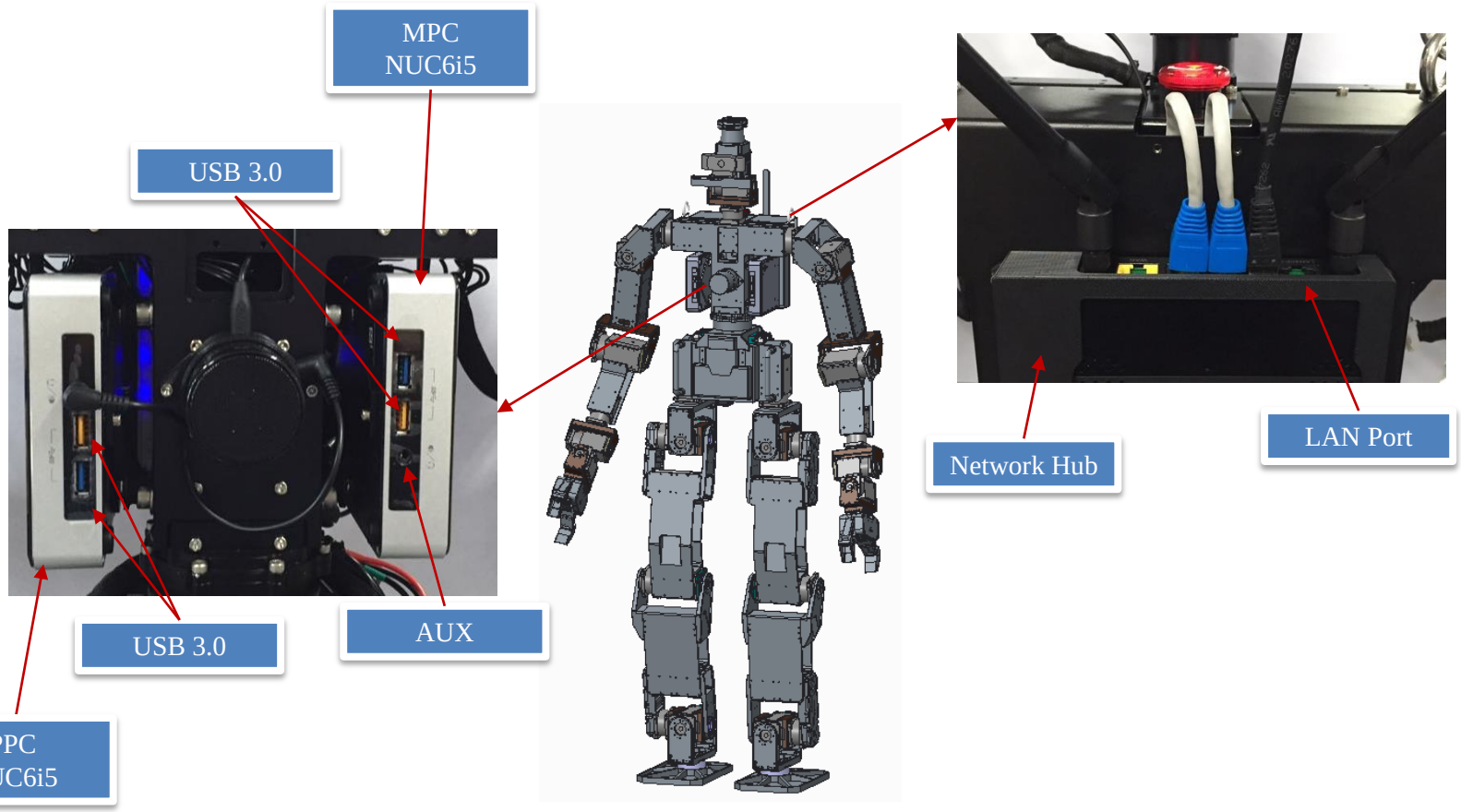

12


13


3. Robot Layout (1)



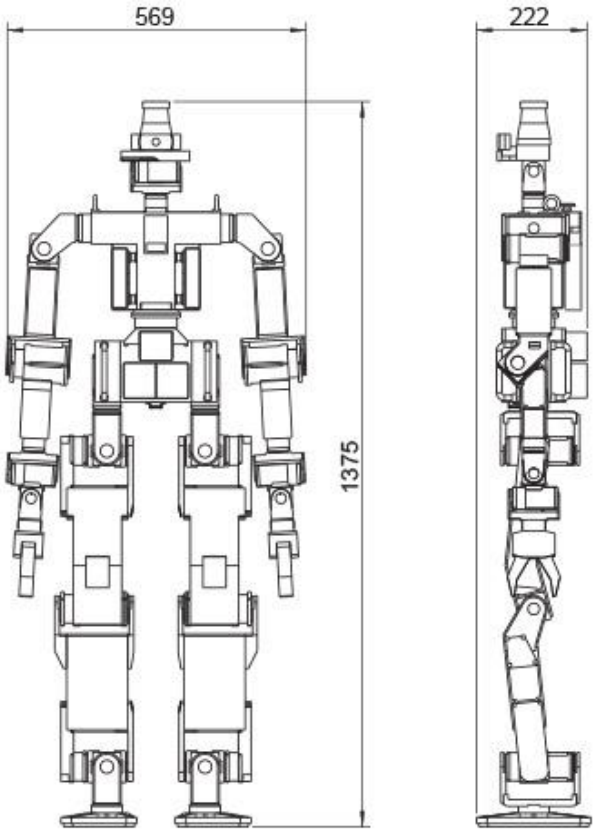
3. Robot Layout (2)



4. Specifications

H/W

- Weight : 42 kg
- Height : 137.5
- DoF : 29
- Actuator : Dynamixel PRO 200W x 10 / 100W x 11 / 20W x 8
- Computor : Intel® NUC with Intel® Core™ i5 Processor x 2
  (RAM 8G / M.2 SSD)
- Wireless router : Dlink DIR-806A x 1
- Sensor
  - Logitech C920 HD Camera x 1
  - Intel RealSense (Option) x 1
  - Hokuyo UTM-30LX-EW (Option) x 1
  - F/T Sensor : Ati Mini58-SI-2800-120 x 2
  - IMU : MicroStrain 3DM-GX4-25 x 1
- Battery : LIPO 22.2V 22,000mA & 18.5V 11,000mA

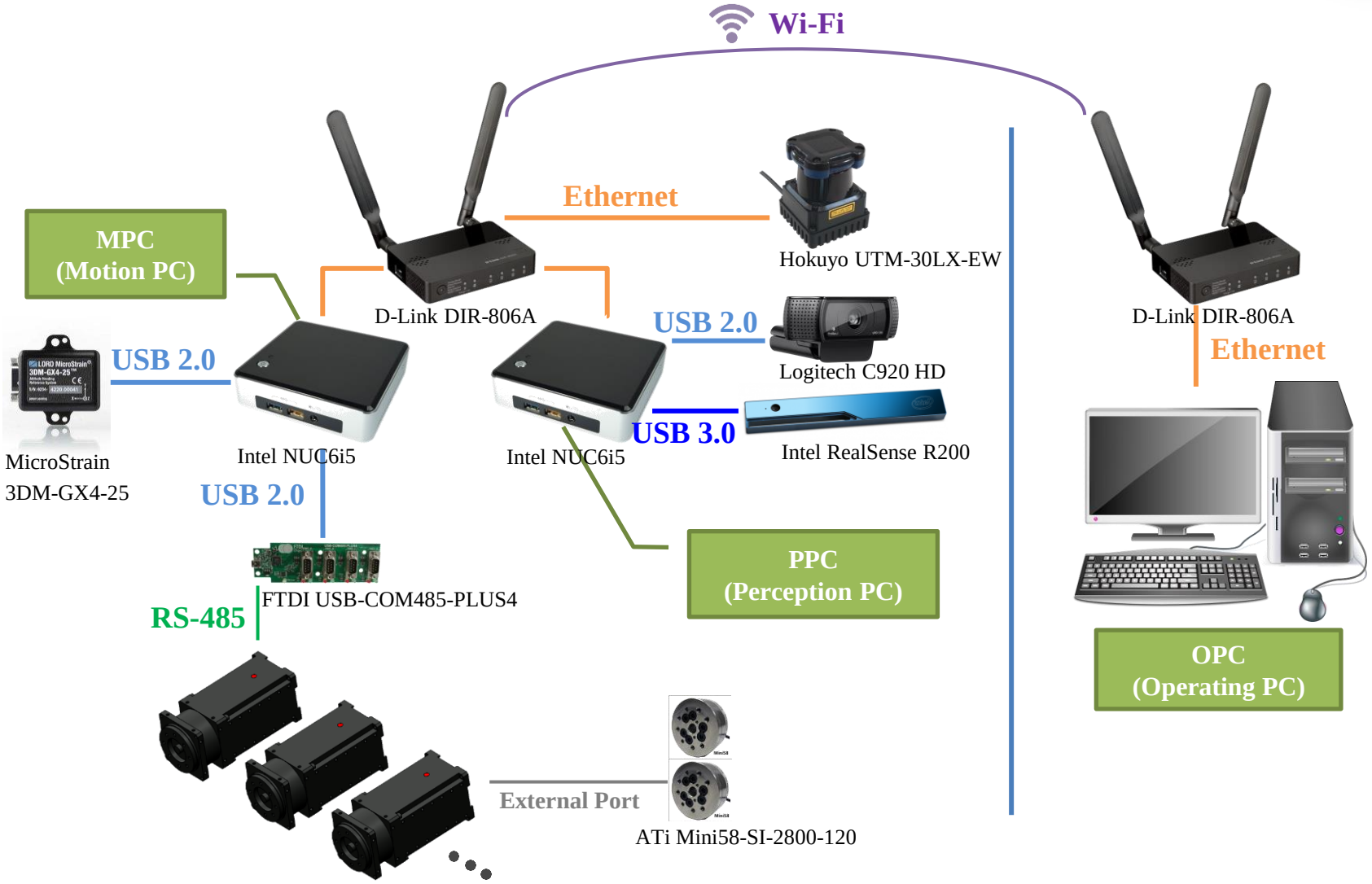


S/W

- OS : Linux (64bit)
- SW : ROS
- Compiler : GNU Project C and C++ Compiler
- Programming Language : C++

2. Getting started

1. System Configuration



2. Unpacking (1)

1. Open the case and remove the sponge.

[Red dot] indicates the location of the robot head.

1

2. Unpack items
Items: Shock absorbing sponge, Charger, Consumables box

2

2. Unpacking (2)

3. Close the case. The head should face up.

[Red dot] indicates the location of the robot head.

3

4. Open the box and hang the carabiners on the lift

4

2. Unpacking (3)

5

5. Connect i-bolts to the carabiners.
   Take out the upper body out of the case.
   [Yellow circles] indicates the location of i-bolts.

6

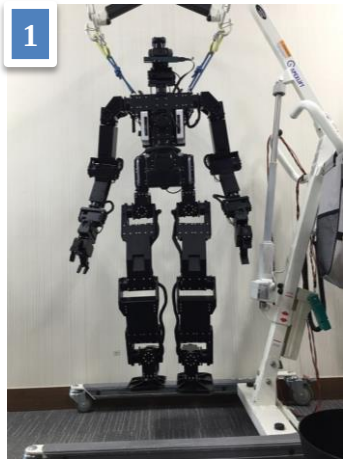
6. The robot will come out naturally via the lift.
   Please pull out the robot legs manually.

2. Unpacking (4)

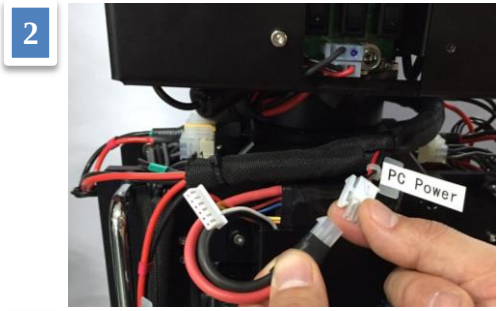
7. Raise the lift up and take out the battery charger box.
   [Yellow square] indicates the location of the battery charger box.

3. Turning on THORMANG3 (1)

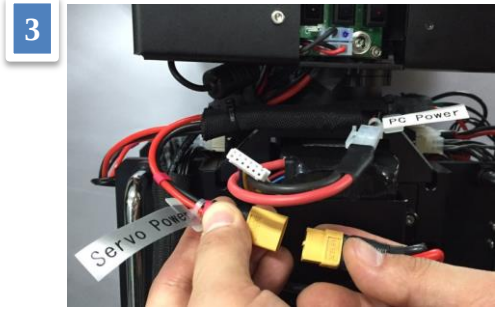
1. Hang THORMANG on lift. Lift THORMANG's feet off from the ground.



2. Using the Power Supply
   b) Connect the power supply (18V, > 10A) to the PC Power cable to provide power to the PCs.
   c) Connect the power supply (24V, >30A) to the Servo Power cable to provide power to Dynamixels.

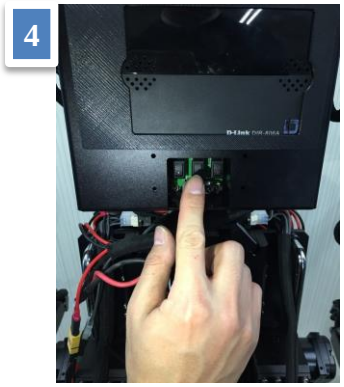


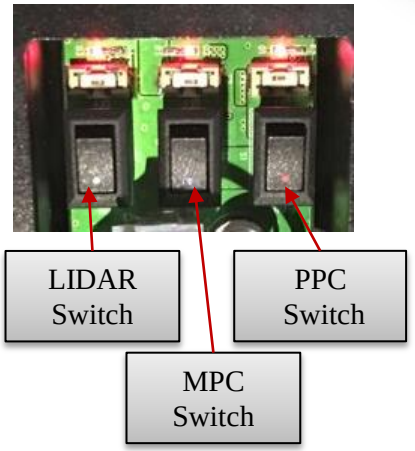
3. Using the Battery Packs
   a) Ensure the batteries are fully charged.
   b) Open the battery compartment door by loosening the thumbscrew. Insert the battery packs. Close the compartment by tightening the thumbscrew.
   c) Connect the 22.2V battery's cable (yellow jack) to Servo Power and connect the 18.5V battery's cable (white terminal) to PC Power.

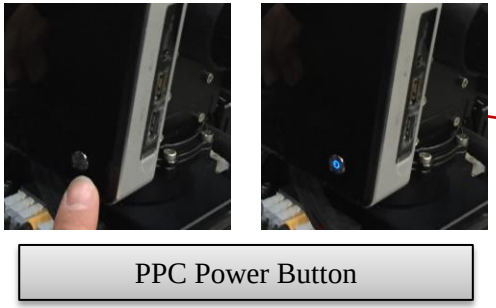


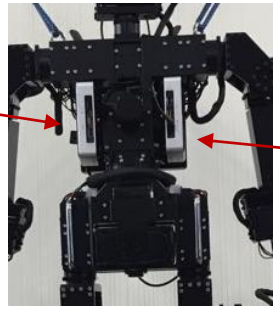
3. Turning on THORMANG3 (2)

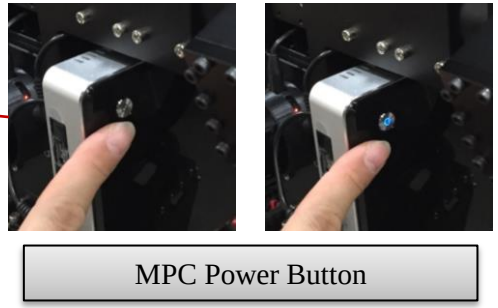
4. First, flip the three switch located on the power board (Lidar, MPC, PPC). Then push the MPC and PPC power buttons.











5. Turn the E-Stop Button.

❖ If the E-Stop Button is pressed, the DXL power is turned off.

4. Running the Basic Program (1)

1. Connecting THORMANG to your PC

On your PC's network setting menu, go to LAN settings and set static IP as follows: 10.17.3.xxx

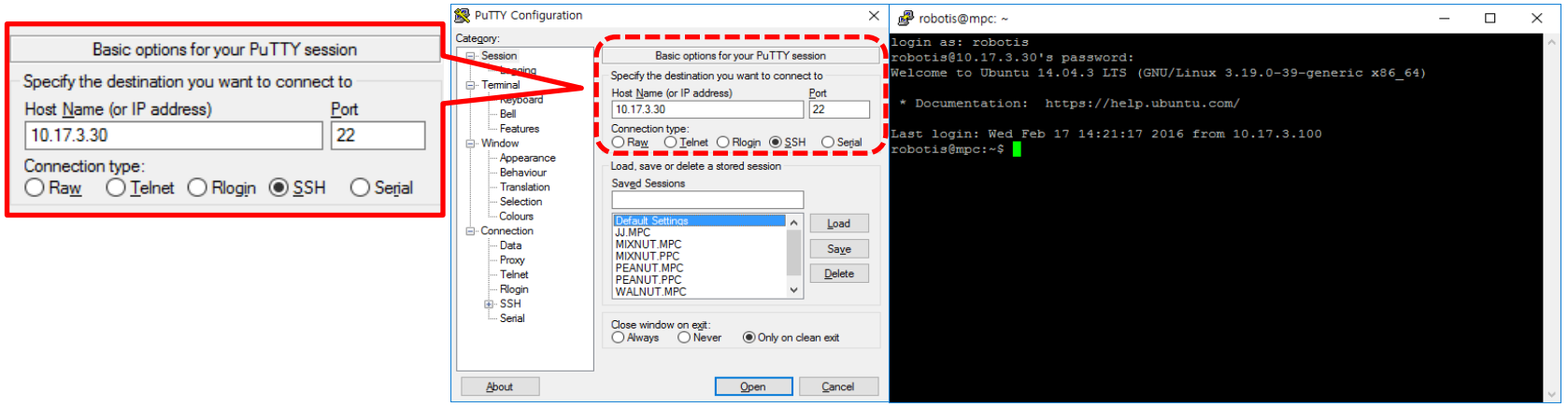
- Network Information

```
1) MPC (Motion PC) IP      : 10.17.3.30
2) PPC (Perception PC) IP  : 10.17.3.35
3) MPC & PPC User Name     : robotis
4) MPC & PPC Password      : 111111
```

4. Running the Basic Program (2)

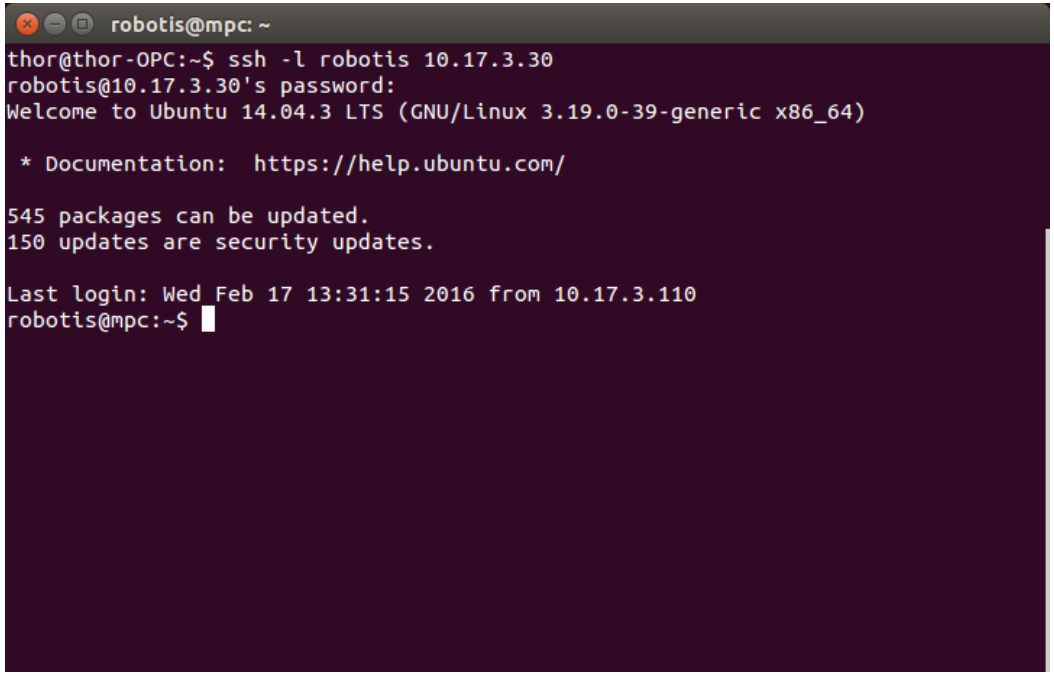
ROBOTIS recommends that users connect via an SSH client

- Example: Windows SSH Client
  1) Execute SSH client program (e.g. PuTTY)
  2) Input the MPC's IP address : **`10.17.3.30`**
  3) Select **SSH** as a connection type and then click Open.
  4) Input the MPC's user name : **`robotis`**
  5) Input the MPC's password : **`111111`**



4. Running the Basic Program (2)

- Example: Ubuntu SSH Client
  1) Open a new terminal window.
  2) Type the following SSH command utilizing the MPC's user name and IP address :
     `$ ssh -l robotis 10.17.3.30`
  3) Input the MPC's password : `111111`

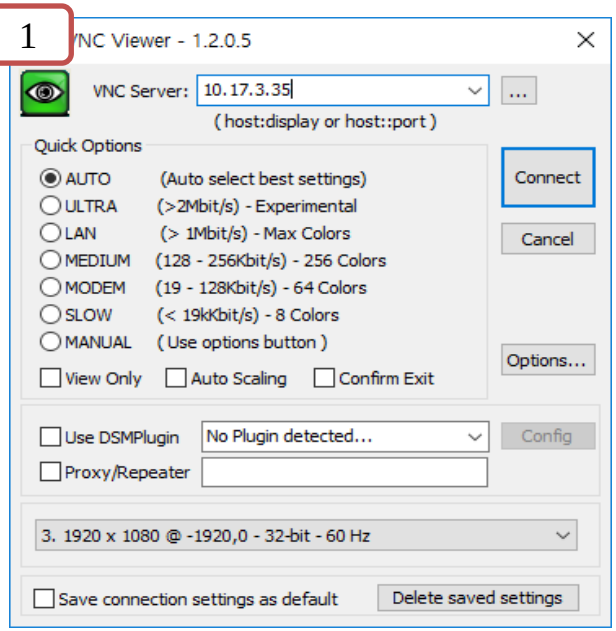


4. Running the Basic Program (3)

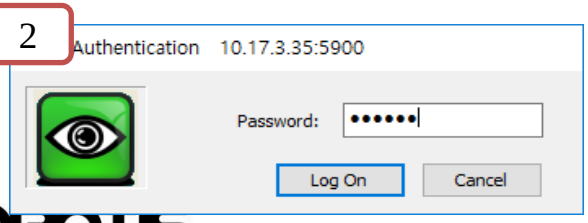
Accessing the MPC via remote desktop may result in slower performance.

- Example: Windows VNC client
  1) Execute VNC client program (e.g. Ultra VNC Viewer)
  2) Input the MPC's IP address : **`10.17.3.30`**
  3) Input the MPC's password : **`111111`**

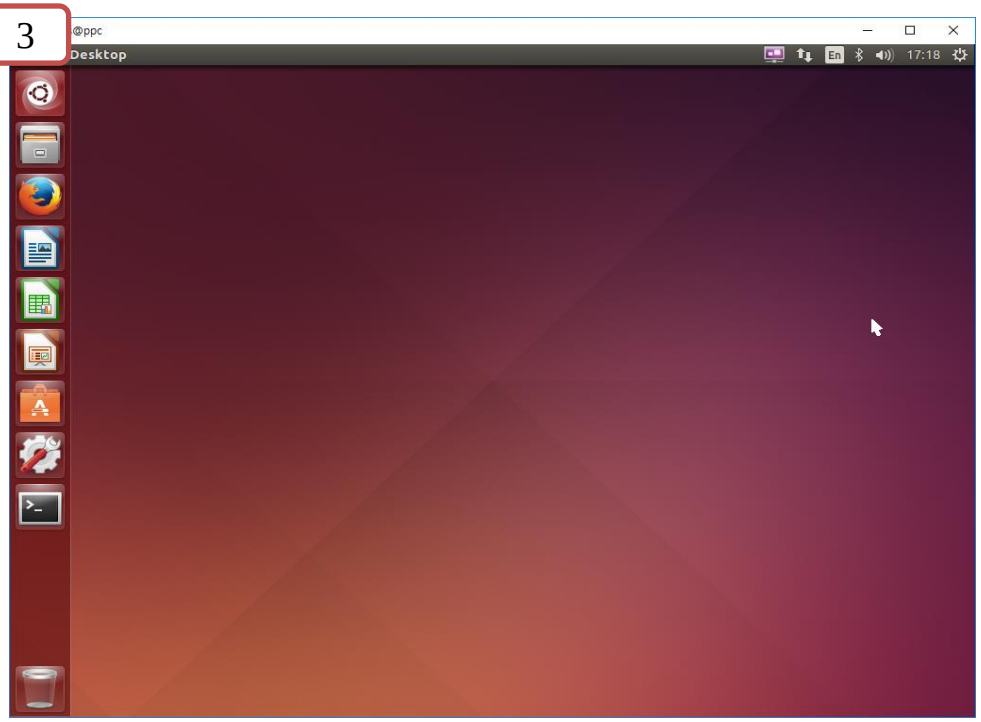
1



2



3



2. Running roscore and THORMANG3 Manager

1. roscore

`roscore` is a collection of nodes and programs that are prerequisites of a ROS-based system. You **must** have `roscore` running in order for ROS nodes to communicate each other. It can be launched using the `roscore` command.

NOTE: If you use roslaunch, it will automatically run roscore if it is not already running.

- Connect to the PPC via SSH client program. (IP: **`10.17.3.35`**)
- `roscore` can be launched using the roscore executable:

```
$ roscore
```

2. THORMANG3 Manager

`thormang3_manager` is a base node using ROBOTIS' framework. `thormang3_manager` must be running before you can run the Simple Demo nodes and before you can check the sensors because they are using `thormang3_manager`.

- Connect to the MPC via SSH client program. (IP: **`10.17.3.30`**)
- To launch THORMANG3 Manager, simply type the following :

```
$ sudo bash
# roslaunch thormang3_manager thormang3_manager.launch
```

3. Running the Simple Demo Programs and Checking the Sensors

1. Manipulation Simple Demo

- Manipulation Control Module allows for two kinds of control :
  - Joint Space Control
  - Task Space Control

- Connect to the PPC via SSH client program. (IP: **`10.17.3.35`**)

- Execute the Manipulation Simple Demo by typing the following command :

```
$ rosrun thormang3_manipulation_demo thormang3_manipulation_demo
```

- These commands should be executed in new terminal.

1) **Initialization 1** : go to initial pose (from Base Module)
```
$ rostopic pub -1 /robotis/manipulation_demo/command std_msgs/String “ini_pose”
```
2) **Initialization 2** : set Manipulation Control Module
```
$ rostopic pub -1 /robotis/manipulation_demo/command std_msgs/String “set_mode”
```
3) **demo 3** : go to manipulation base pose – **Joint Space Control**
```
$ rostopic pub -1 /robotis/manipulation_demo/command std_msgs/String “base_pose”
```
4) **demo 4** : move right arm (torso joint is not included) – **Task Space Control**
```
$ rostopic pub -1 /robotis/manipulation_demo/command std_msgs/String “right_arm”
```
5) **demo 5** : move left arm (torso joint is not included) – **Task Space Control**
```
$ rostopic pub -1 /robotis/manipulation_demo/command std_msgs/String “left_arm”
```

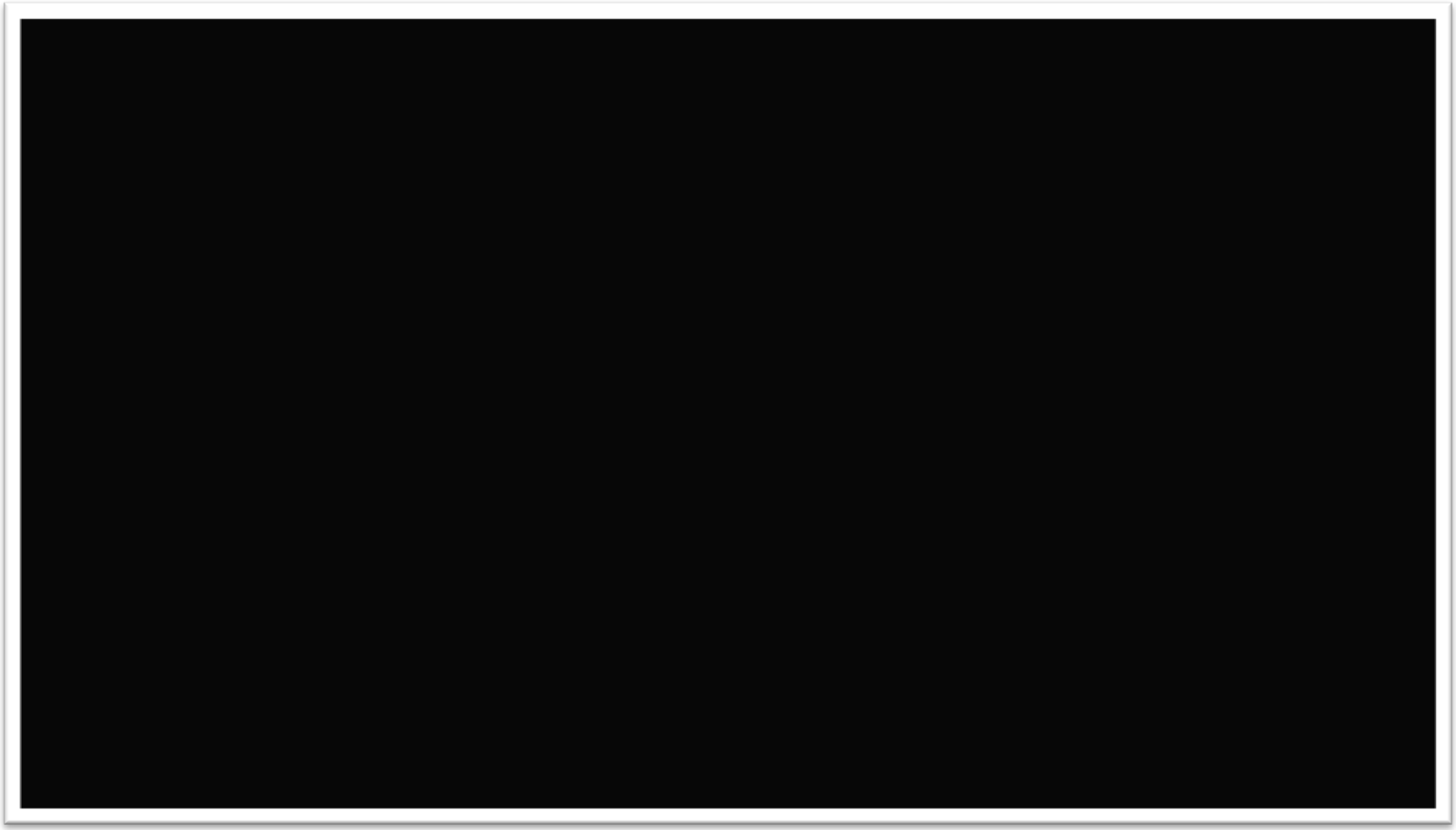
4. Running the Basic Program (5)

On the terminal window, you can check a successful demo execution by comparing your screen output to the following screenshot:

```
thor@SIM-PC:~$ rosrun thormang3_manipulation_demo thormang3_manipulation_demo
[ INFO] [1456273852.518346294]: Robotis Thormang3 Manipulation Simple Demo
[ INFO] [1456273858.108976598, 1877.168000000]: demo 1: go to initial pose
[ INFO] [1456273929.708964751, 1946.535000000]: demo 2: set manipulation control mode
[ INFO] [1456273935.924793739, 1952.625000000]: demo 3: go to manipulation base pose
[ INFO] [1456273943.555563594, 1960.293000000]: demo 4: move right arm
[ INFO] [1456273949.728911625, 1966.374000000]: demo 5: move left arm
```

4. Running the Basic Program (6)

Manipulation Simple Demo Video



4. Running the Basic Program (7)

2. Walking Simple Demo

- Walking Module provides two kinds of functions
  - One step forward/backward walking
  - Balance ON/OFF : All balance parameter will be loaded from "`data/balance_param.yaml`"

- Connect to the PPC via SSH client program. (IP: **`10.17.3.35`**)

- Execute the Walking Simple Demo by typing the following command :

```
$ rosrun thormang3_walking_demo thormang3_walking_demo
```

- These commands should be executed in new terminal.

1) **Initialization 1** : go to initial pose (from Base Module)

```
$ rostopic pub -1 /robotis/walking_demo/command std_msgs/String "ini_pose"
```

2) **Initialization 2** : set Walking Control Module

```
$ rostopic pub -1 /robotis/walking_demo/command std_msgs/String "set_mode"
```

3) **demo 3** : make balance algorithm enable/disable – **Balance ON / OFF**

```
$ rostopic pub -1 /robotis/walking_demo/command std_msgs/String "balance_on"
$ rostopic pub -1 /robotis/walking_demo/command std_msgs/String "balance_off"
```

4) **demo 4** : walk forward – **One step forward walking**

```
$ rostopic pub -1 /robotis/walking_demo/command std_msgs/String "forward"
```

5) **demo 5** : walk backward – **One step backward walking**

```
$ rostopic pub -1 /robotis/walking_demo/command std_msgs/String "backward"
```

4. Running the Basic Program (8)

On the terminal window, you can check a successful demo execution by comparing your screen output to the following screenshot:

```
robotis@mpc:~$ rosrun thormang3_walking_demo thormang3_walking_demo_node
[ INFO] [1455875066.217024182]: [Demo]  : receive [ini_pose] msg
[ INFO] [1455875066.217090233]: [Demo]  : go to initial pose
[ INFO] [1455875066.217208610]: [Demo]  : please wait 5 seconds
[ INFO] [1455875075.729093836]: [Demo]  : receive [set_mode] msg
[ INFO] [1455875075.729155435]: [Demo]  : set walking control mode
[ INFO] [1455875075.731306347]: [Robot] : Walking_Module_is_enabled
[ INFO] [1455875085.883878337]: [Demo]  : receive [balance_on] msg
[ INFO] [1455875085.883942699]: [Demo]  : balance enable
[ INFO] [1455875085.890794091]: [Demo]  : Succeed to set balance param
[ INFO] [1455875085.890860928]: [Robot] : Balance_Param_Setting_Started
[ INFO] [1455875086.885257816]: [Robot] : Balance_Param_Setting_Finished
[ INFO] [1455875097.554503231]: [Demo]  : receive [forward] msg
[ INFO] [1455875097.554568993]: [Demo]  : forward walking
[ INFO] [1455875097.562789075]: [Demo]  : Succeed to add step data array
[ INFO] [1455875097.563741307]: [Robot] : Walking_Started
[ INFO] [1455875104.660218824]: [Robot] : Walking_Finished
[ INFO] [1455875110.671069684]: [Demo]  : receive [backward] msg
[ INFO] [1455875110.671131309]: [Demo]  : backward walking
[ INFO] [1455875110.674200761]: [Demo]  : Succeed to add step data array
[ INFO] [1455875110.675751408]: [Robot] : Walking_Started
[ INFO] [1455875117.772244468]: [Robot] : Walking_Finished
```

Walking Simple Demo Video

4. Running the Basic Program (10)

3. Check the Sensors

- Run the MPC's Sensors (IMU, FT, Lidar)
  - Type the following commands to run the MPC's sensors :

```
$ sudo bash
# roslaunch thormang3_manager thormang3_manager.launch
```

- How to Check the MPC's Sensors
  - **IMU** : Type the following command and check the output :

```
$ rostopic echo /robotis/sensor/imu/imu
```

```
robotis@mpc:~$ rostopic echo /robotis/sensor/imu/imu
header:
  seq: 218798
  stamp:
    secs: 1456227058
    nsecs: 657393447
  frame_id: imu
orientation:
  x: 0.551555931568
  y: 0.831686019897
  z: -0.0350182652473
  w: 0.0534624755383
orientation_covariance: [-1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
angular_velocity:
  x: 0.00529906712472
  y: 0.0256397109479
  z: 0.108881101012
angular_velocity_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
linear_acceleration:
  x: 1.40463167375
  y: 0.0995190255708
  z: 9.7412729802
linear_acceleration_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
```

4. Running the Basic Program (11)

- How to Check the MPC's Sensors
  - **FT Sensor** : Type the following commands and check the output :

```
$ rostopic echo /robotis/sensor/ft_right_foot/raw
$ rostopic echo /robotis/sensor/ft_left_foot/raw
```

```
robotis@mpc:~$ rostopic echo /robotis/sensor/ft_right_foot/raw
header:
  seq: 110484
  stamp:
    secs: 1456227469
    nsecs: 577939051
  frame_id: r_leg_foot_link
wrench:
  force:
    x: 5175.13464253
    y: 12796.9383106
    z: 9230.67538064
  torque:
    x: -134.298186423
    y: -234.033554692
    z: -267.399062596
```

```
robotis@mpc:~$ rostopic echo /robotis/sensor/ft_left_foot/raw
header:
  seq: 113669
  stamp:
    secs: 1456227506
    nsecs: 383800602
  frame_id: l_leg_foot_link
wrench:
  force:
    x: 9699.47838375
    y: 5305.99705173
    z: 10538.1527615
  torque:
    x: -243.133327243
    y: -325.658292323
    z: -202.514837491
```

4. Running the Basic Program (12)

- How to Check the MPC's Sensors
  - **Lidar** : Type the following command and check the output :
    `$ rostopic echo /robotis/sensor/scan --noarr`

```
robotis@mpc:~$ rostopic echo /robotis/sensor/scan --noarr
header:
  seq: 403902
  stamp:
    secs: 1456227605
    nsecs: 320116247
  frame_id: lidar_link
angle_min: -2.35619449615
angle_max: 2.35619449615
angle_increment: 0.00436332309619
time_increment: 1.73611151695e-05
scan_time: 0.0250000003725
range_min: 0.0230000000447
range_max: 60.0
---
header:
  seq: 403903
  stamp:
    secs: 1456227605
    nsecs: 350245326
  frame_id: lidar_link
angle_min: -2.35619449615
angle_max: 2.35619449615
angle_increment: 0.00436332309619
time_increment: 1.73611151695e-05
scan_time: 0.0250000003725
range_min: 0.0230000000447
range_max: 60.0
```

4. Running the Basic Program (13)

- Run the PPC's Sensors (Web Camera (HD Camera), Depth Camera (RealSense))
  - Type the following command to run PPC's sensors :
    ```
    $ roslaunch thormang3_sensors thormang3_sensors.launch
    ```

- How to Check the PPC's Sensors
  - **Web Camera** : Type the following command and check the output.
    ```
    $ rostopic echo /robotis/sensor/camera/image_raw --noarr
    ```

```
robotis@DDC:~$ rostopic echo /robotis/sensor/camera/image_raw --noarr
header:
  seq: 23
  stamp:
    secs: 1456228631
    nsecs: 803098057
  frame_id: head_p_link
height: 480
width: 640
encoding: rgb8
is_bigendian: 0
step: 1920
---
header:
  seq: 24
  stamp:
    secs: 1456228631
    nsecs: 903091604
  frame_id: head_p_link
height: 480
width: 640
encoding: rgb8
is_bigendian: 0
step: 1920
```

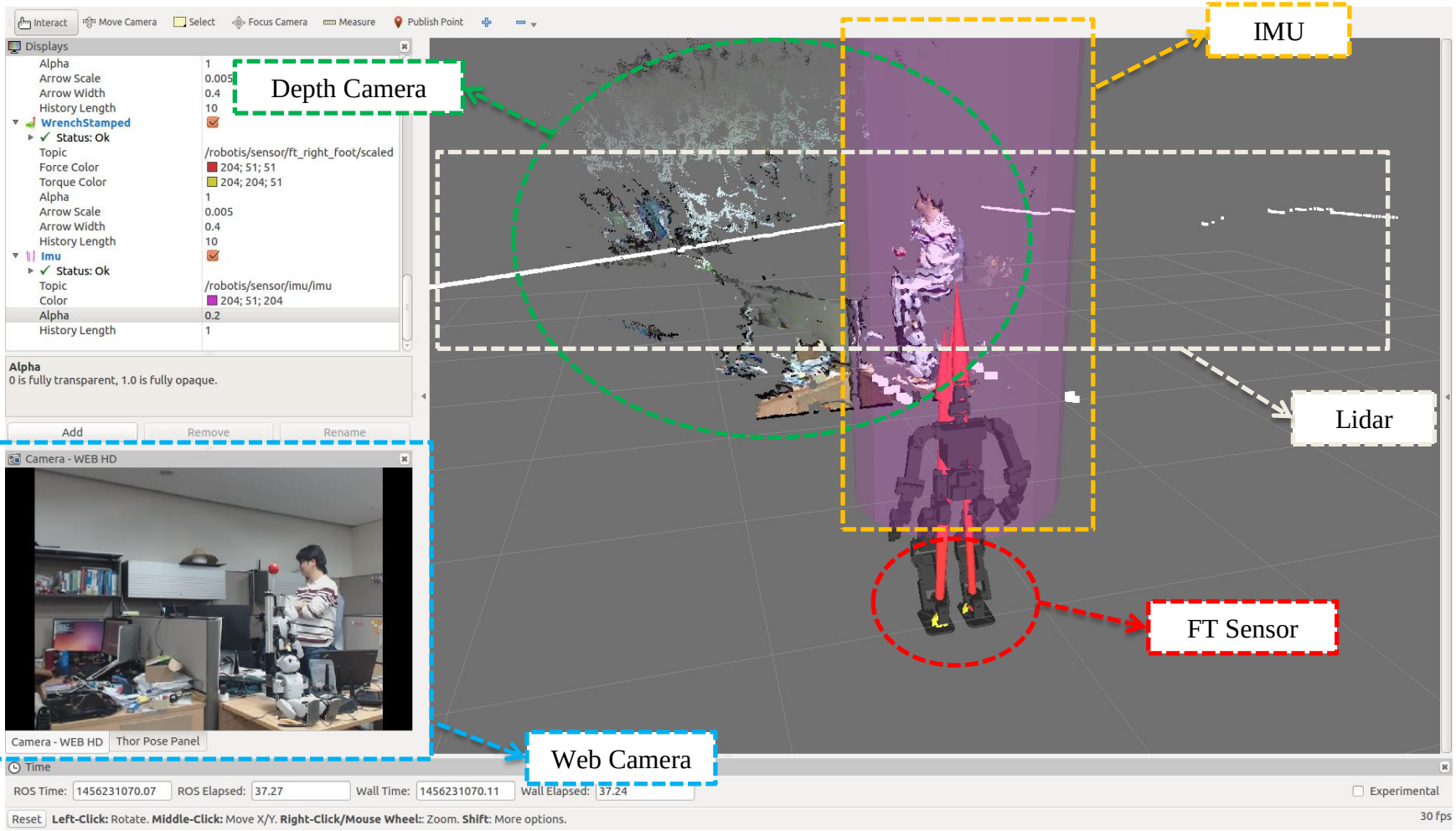
4. Running the Basic Program (14)

- How to Check the PPC's Sensors
  - **Depth Camera**: Type the following command and check the output.
    `$ rostopic echo /realsense/depth_registered/points --noarr`

```
robotis@mpc:~$ rostopic echo /realsense/depth_registered/points --noarr
header:
  seq: 0
  stamp:
    secs: 1489127967
    nsecs: 361551208
  frame_id: realsense_rgb_optical_frame
height: 480
width: 640
is_bigendian: False
point_step: 32
row_step: 20480
is_dense: False
---
header:
  seq: 23
  stamp:
    secs: 1489127968
    nsecs: 128217875
  frame_id: realsense_rgb_optical_frame
height: 480
width: 640
is_bigendian: False
point_step: 32
```

4. Running the Basic Program (15)

- How to Check the THORMANG's Sensors in the GUI
  - **Refer to Demo or Vision Presentations**



5. Turning THORMANG3 Off

1. **Hang THORMANG on lift. Lift THORMANG's feet off the ground.**

   - **First, shut down the MPC and the PPC by typing the following commands.**
     - MPC :

     ```
     $ sudo poweroff
     ```

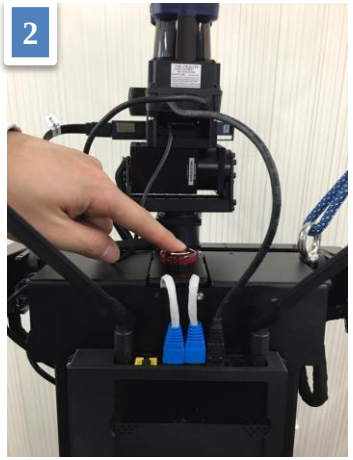
     - PPC :

     ```
     $ sudo poweroff
     ```

     - **NOTE:** If P/W is required, type `111111`

2. **Press the E-Stop Button to turn off the DXL's power.**

3. **After confirming that you have shut down all PCs, toggle three switches on the power board.**



6. Charging the Battery (1)

1. **Connect all cables to the battery charger**
   a) Channel A has the yellow jack
   b) Connect jack a white terminal to channel

2. **Connect the power supply's AC adapter to outlet**

3. **Match the settings with the image to the right by pressing the buttons on the charger.**

1

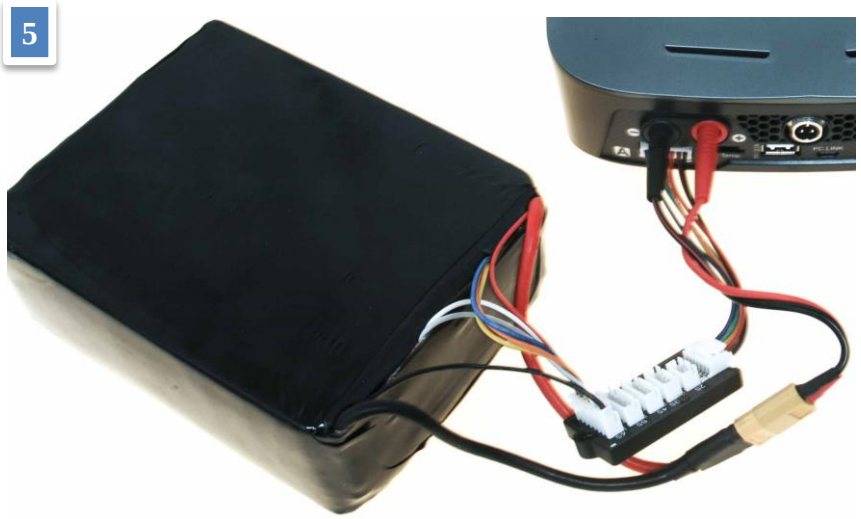
2

3

6. Charging the Battery (2)

4. Connect the charging cable's male connector (white terminal) to the battery's female connector. Then, connect the battery's balance cable to the 5-cell balance socket (6 pins).

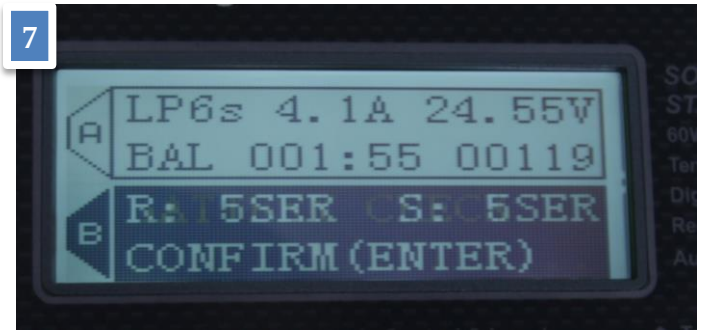
5. Connect the charging cable's male connector (yellow jack) to the battery's female connector. Then, connect the battery's balance cable to the 6-cell balance socket (7 pins).



6. Charging the Battery (3)

6. **Hold down the Enter button in channel A. Press once more after confirming the cell.**

7. **Repeat step 6 for channel B.**



8. **Melody will be played when the battery is fully charged.**

❖**Each channel can be charged independently.**

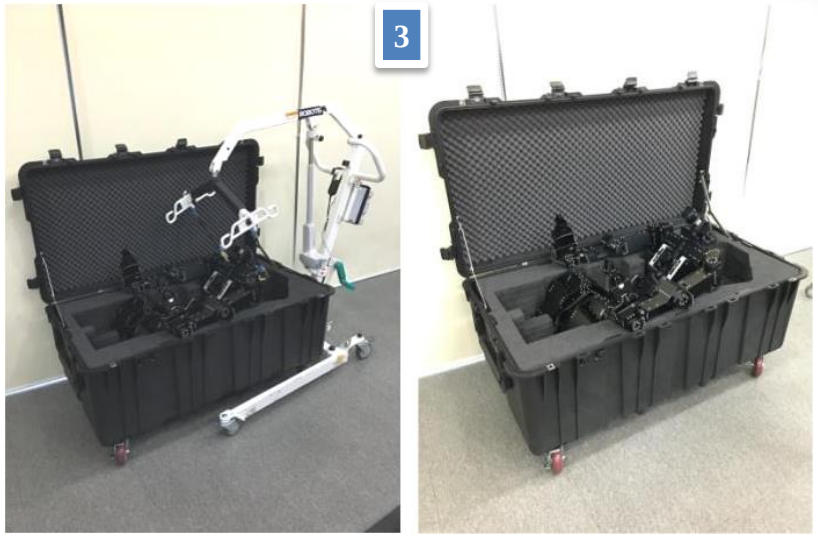
7. Repacking (1)

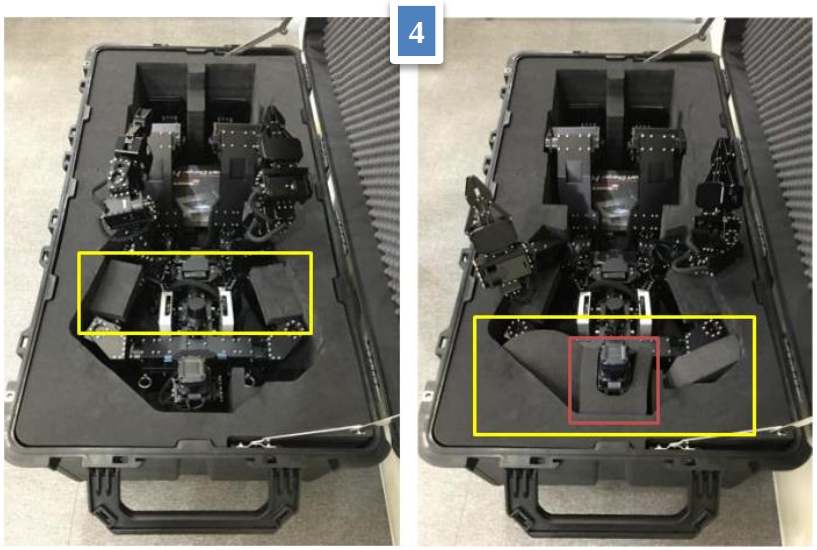
1. Put the battery charger box into the case.

2. Pack the robot by placing its feet first.
Please be careful when handling to prevent any damages on the robot.

7. Repacking (2)

3. Bend the robot's head forward.
If the lift doesn't go down, detach the carabiners from the lift.

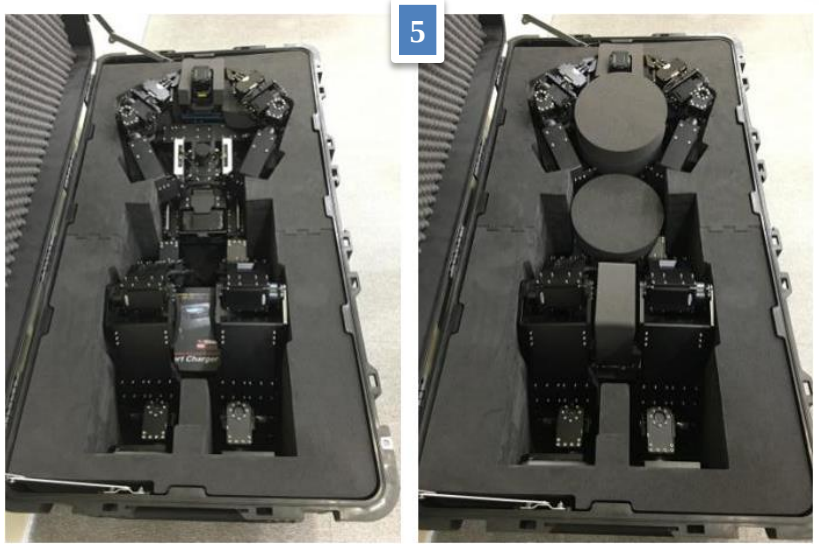




4. Refer to the pictures: Place sponges for the arms and the head protection.

7. Repacking (3)

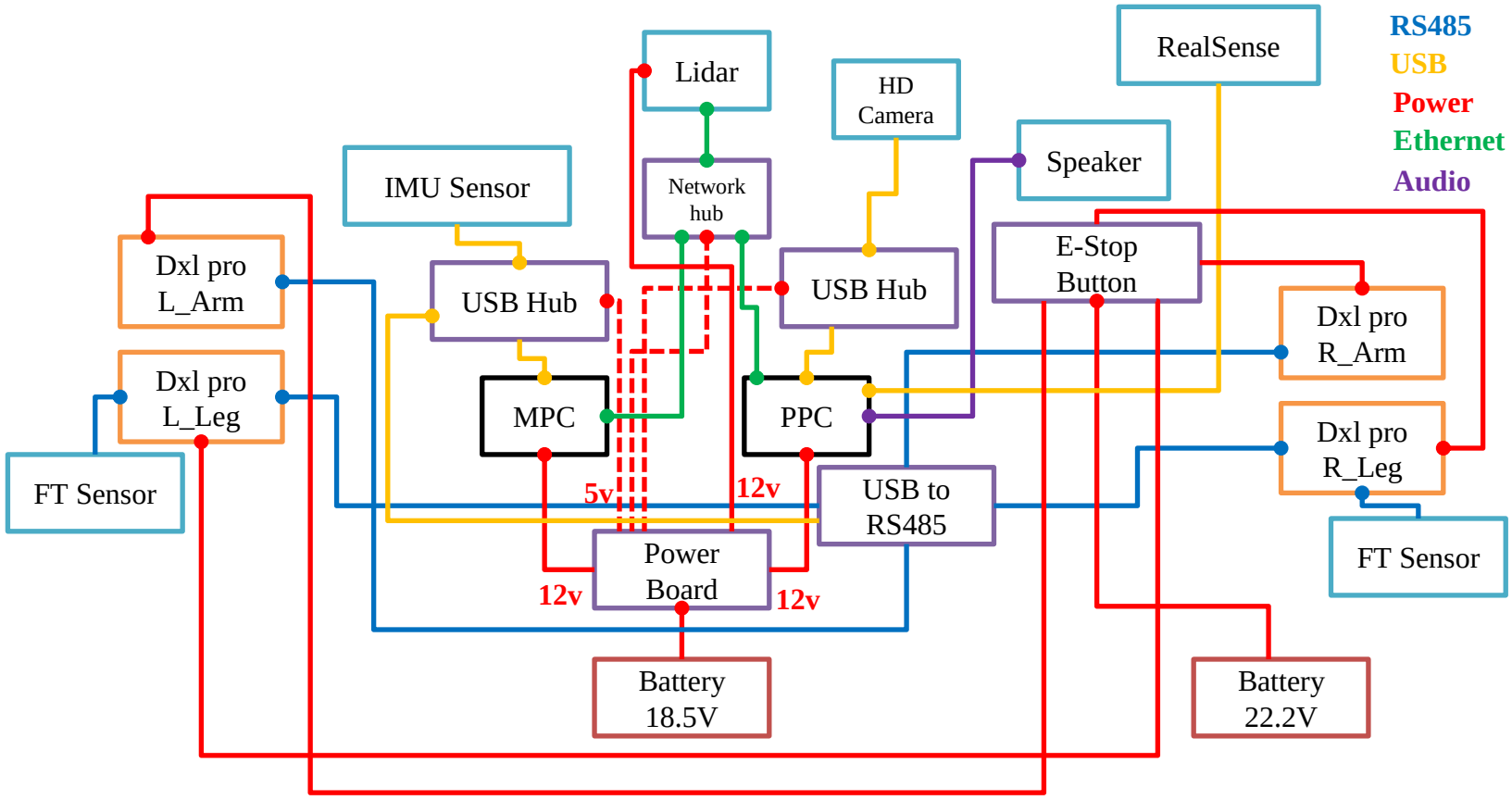
5. Refer to the pictures for robot arm packing.
Place sponges for the upper body and the knee protection.



6. Place the sponge over the robot and close the lid.

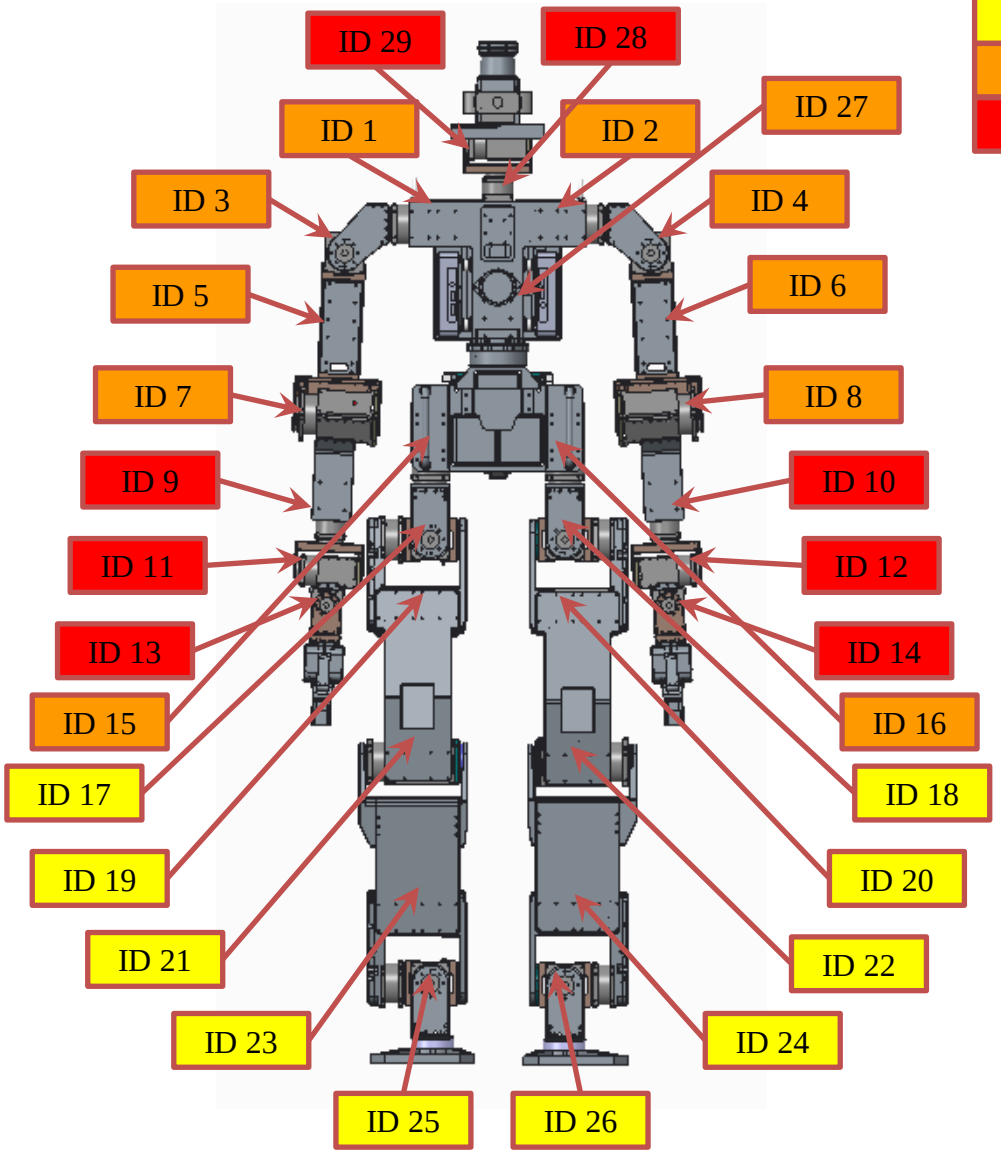
3. Miscellaneous

1. System Block Diagram



2. Joint Name & ID Map

U2D_CH 0	
ID	JointName
27	torso_y
1	r_arm_sh_p1
3	r_arm_sh_r
5	r_arm_sh_p2
7	r_arm_el_y
9	r_arm_wr_r
11	r_arm_wr_y
13	r_arm_wr_p
31	r_arm_grip



200W
100W
20W

U2D_CH 1	
ID	JointName
28	head_y
29	head_p
2	l_arm_sh_p1
4	l_arm_sh_r
6	l_arm_sh_p2
8	l_arm_el_y
10	l_arm_wr_r
12	l_arm_wr_y
14	l_arm_wr_p
30	l_arm_grip

U2D_CH 2	
ID	JointName
15	r_leg_hip_y
17	r_leg_hip_r
19	r_leg_hip_p
21	r_leg_kn_p
23	r_leg_an_p
25	r_leg_an_r

U2D_CH 3	
ID	JointName
16	l_leg_hip_y
18	l_leg_hip_r
20	l_leg_hip_p
22	l_leg_kn_p
24	l_leg_an_p
26	l_leg_an_r