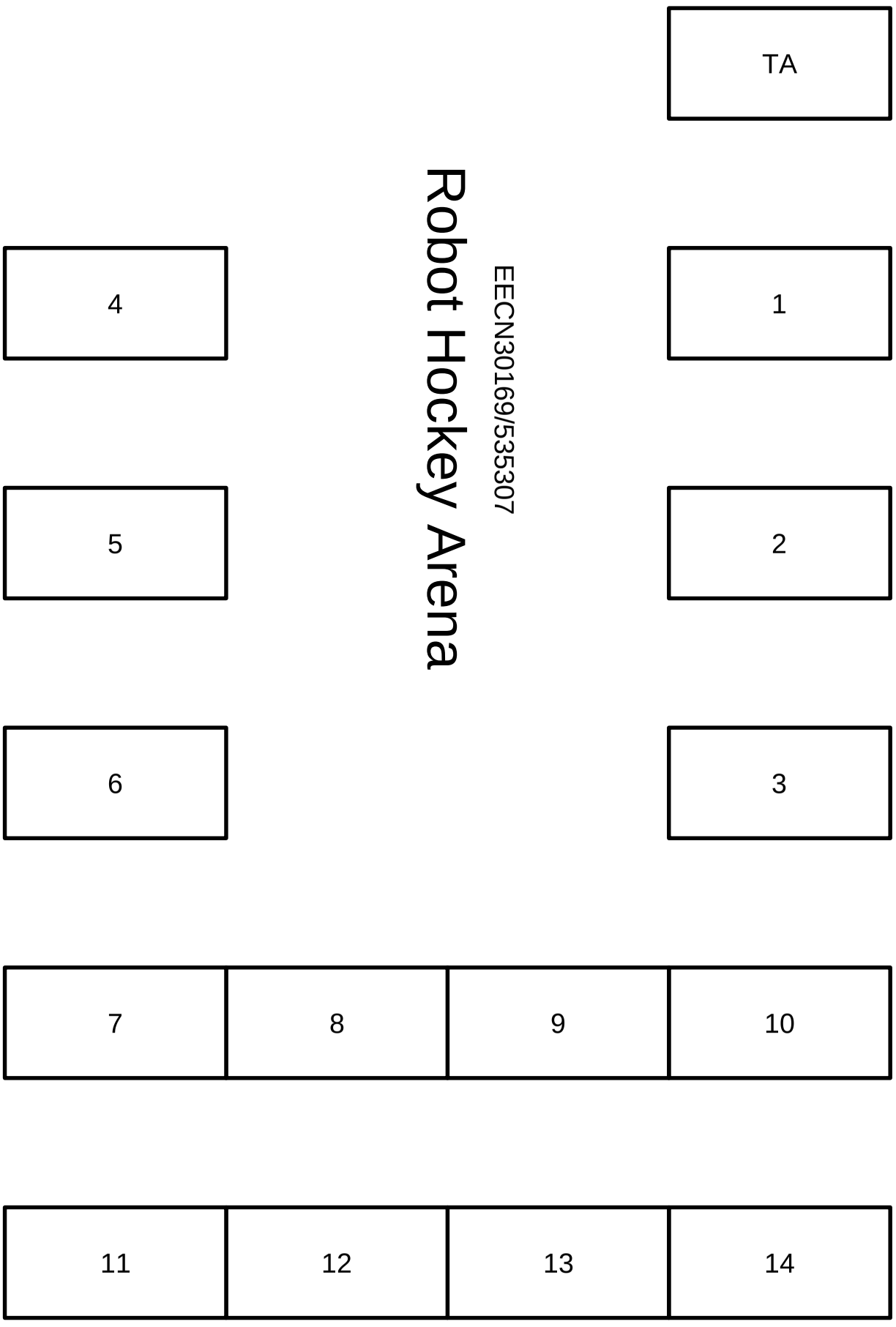
Floor plan EE632



Chap 1
Embedded Computing System with ROS

Date : 09/23/2022

Introduction to ROS

- Introduction to ROS
- ROS file system level
- ROS computation graph level
- Publisher & Subscriber

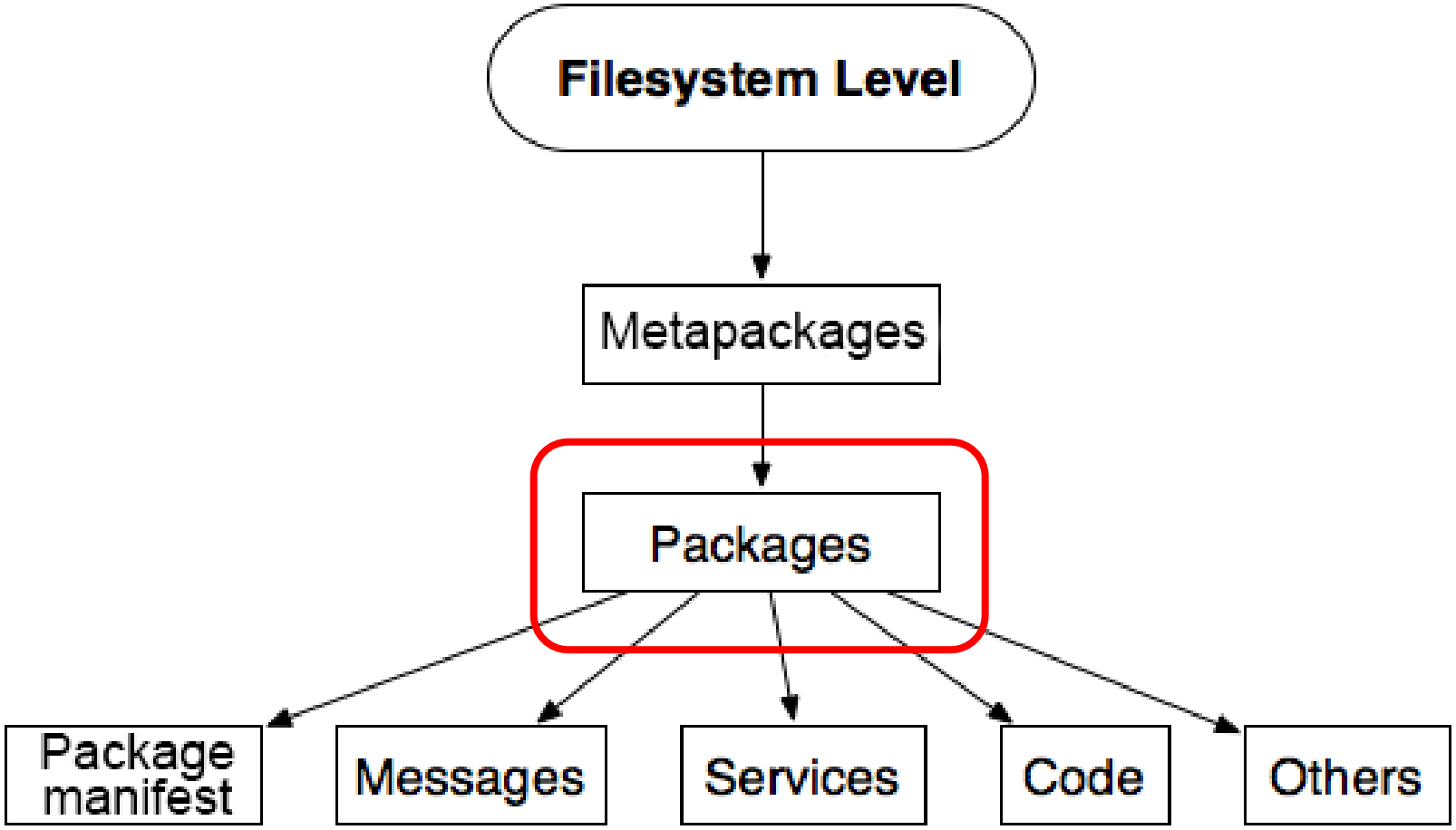
Note

- Site: http://www.ros.org/
- Documentation: http://wiki.ros.org/
- Reference: Mastering ROS for Robotics Programming

Introduction to ROS

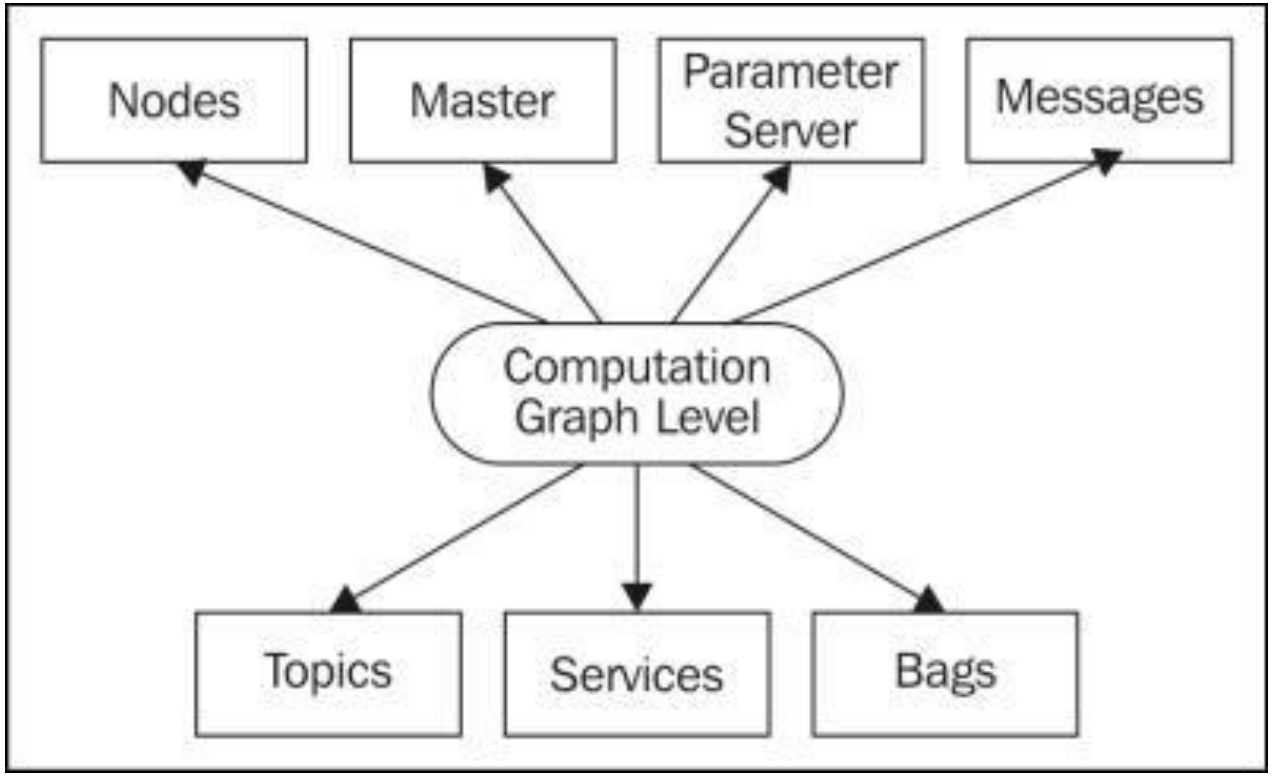
- ROS (Robot Operating System) is an open-source, meta-operating system for robot.
- Features
  - Code reuse (exec. nodes, grouped in packages)
  - Distributed, modular design (scalable)
  - Language independent (C++, Python, …)
  - Easy testing (ready-to-use)
  - Vibrant community & collaborative environment

ROS file system level

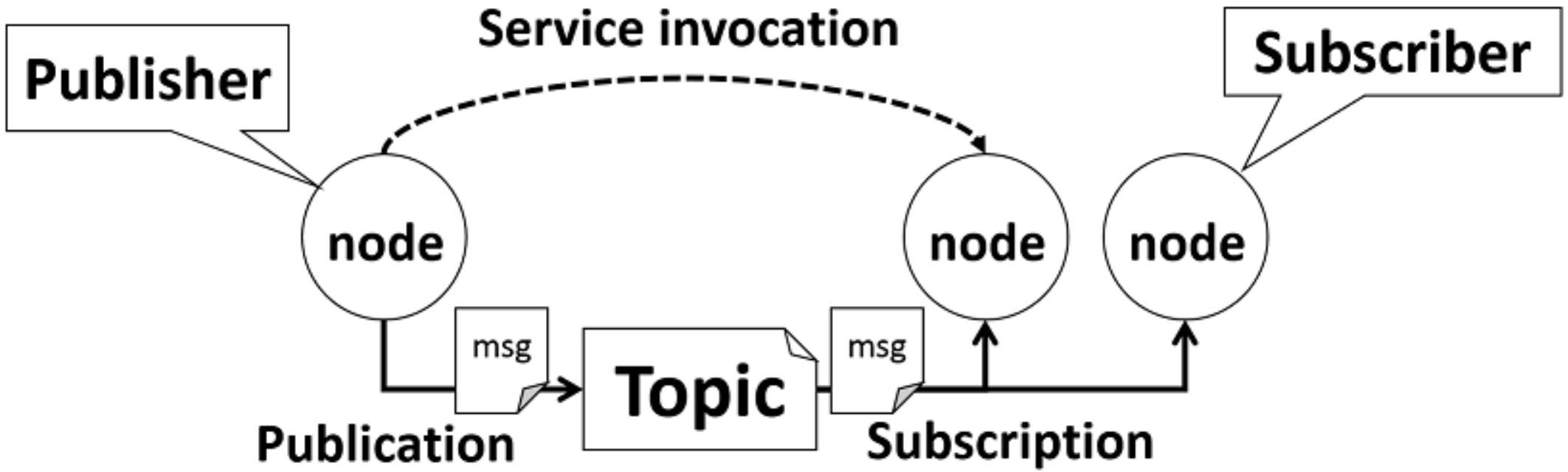


ROS computation graph level

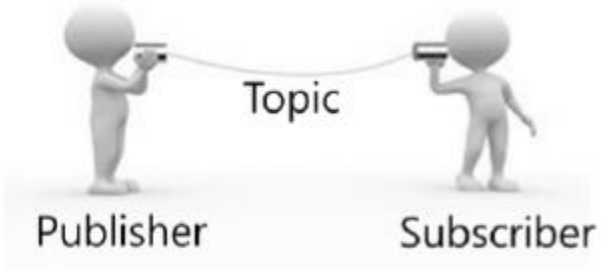
- Nodes
- Master
- Messages
- Topics

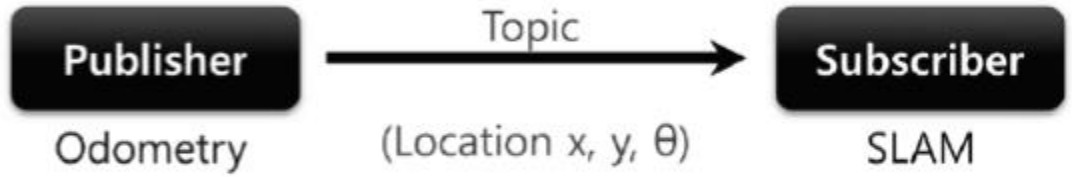


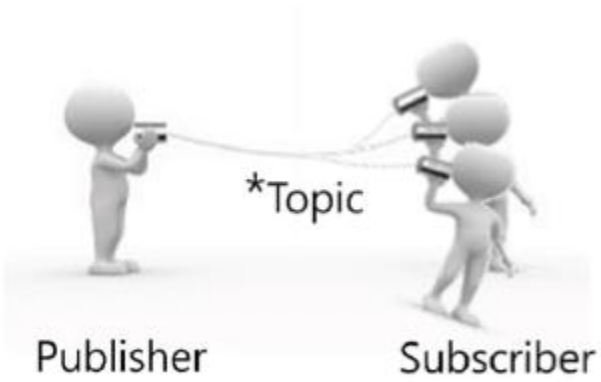
ROS nodes communication

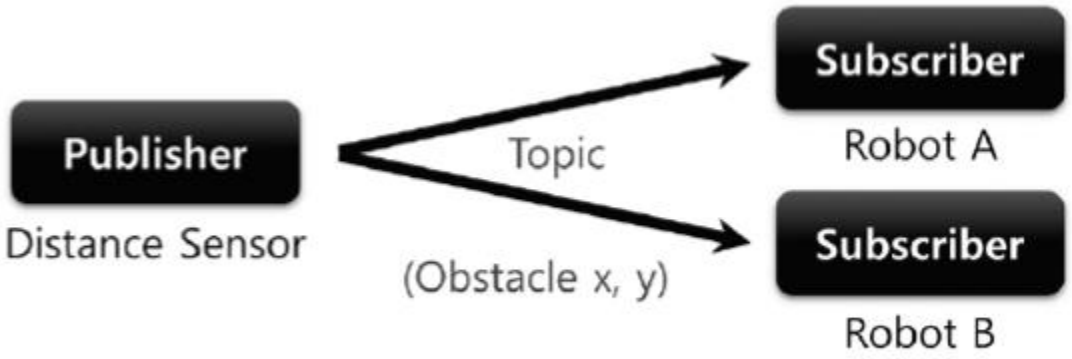


ROS nodes communication









Environment Setting

- PC setup
- Install Ubuntu mate 18.04
- Install ROS Melodic
- Remote Control Setting

Wi-Fi

- SSID：iscilab621-new2.4
- PWD：iscilab621

Install Ubuntu mate 18.04

- Download Ubuntu mate 18.04 from https://releases.ubuntu-mate.org/archived/18.04/

- You can choose arm64 version or armhf version by yourself

  - Reference to ：

  https://ubuntu-mate.org/raspberry-pi/download/

Note

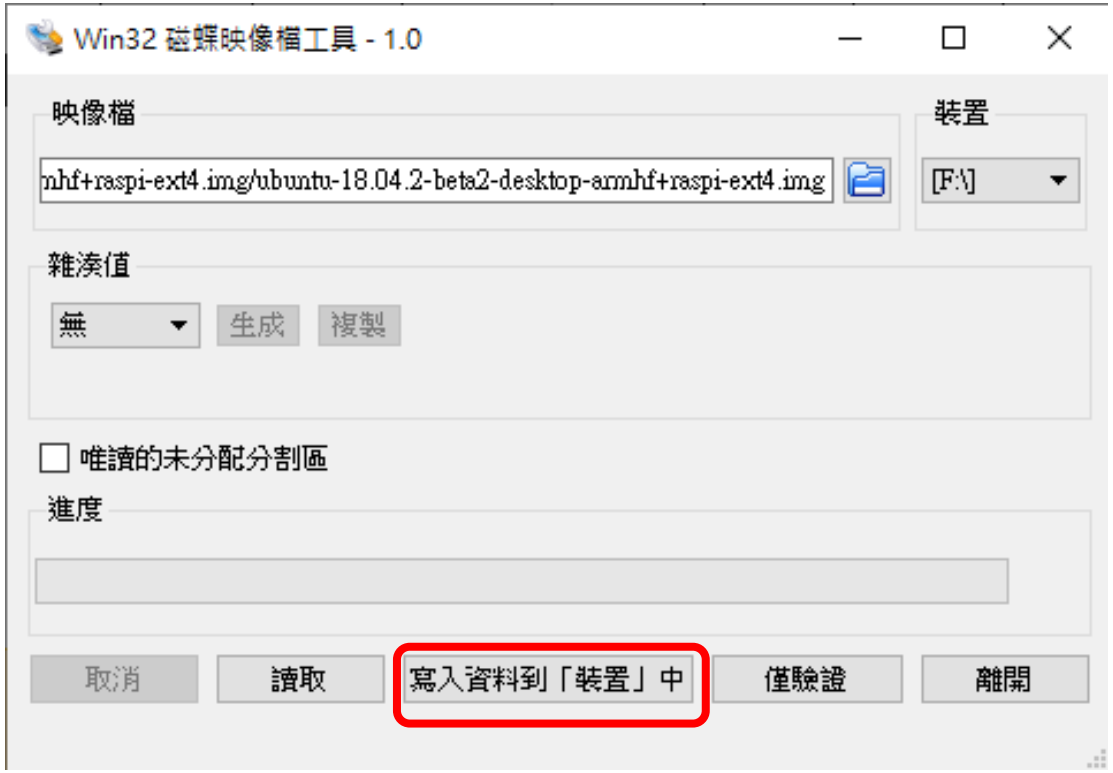
---

ubuntu-mate-18.04.2-beta1-desktop-arm64+raspi3-ext4.img
ubuntu-18.04.2-beta2-desktop-armhf+raspi-ext4.img

are already zipped and uploaded to New e3, please check the file

Install Ubuntu mate 18.04

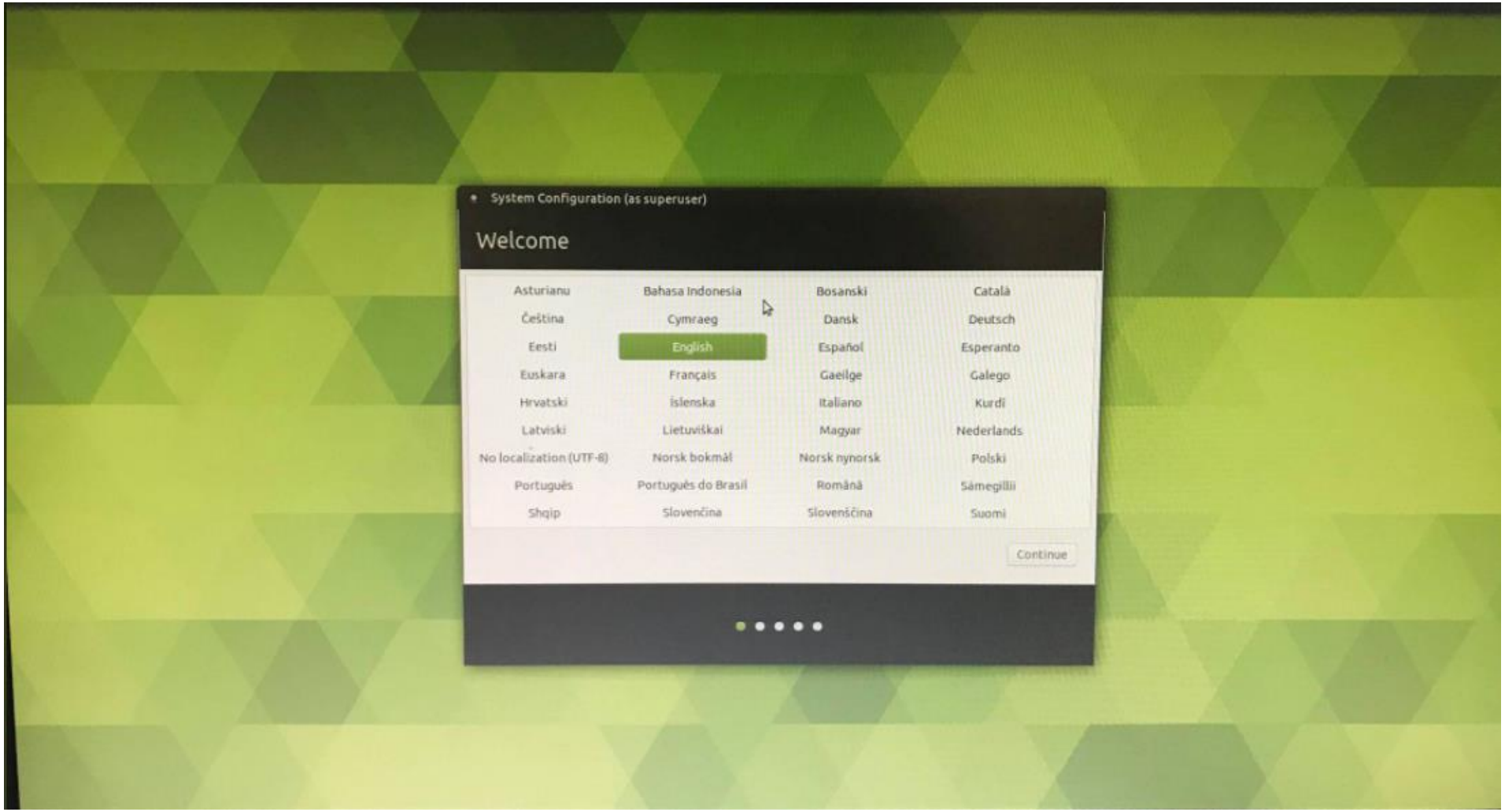
- Make image SD card by using GNOME Disk or Win32 Disk



Tools

- Linux: GNOME Disk
- Windows: Win32 Disk Imager

Setting Raspberry Pi



Install ROS Melodic

http://wiki.ros.org/melodic/Installation/Ubuntu

1.2 Setup your sources.list

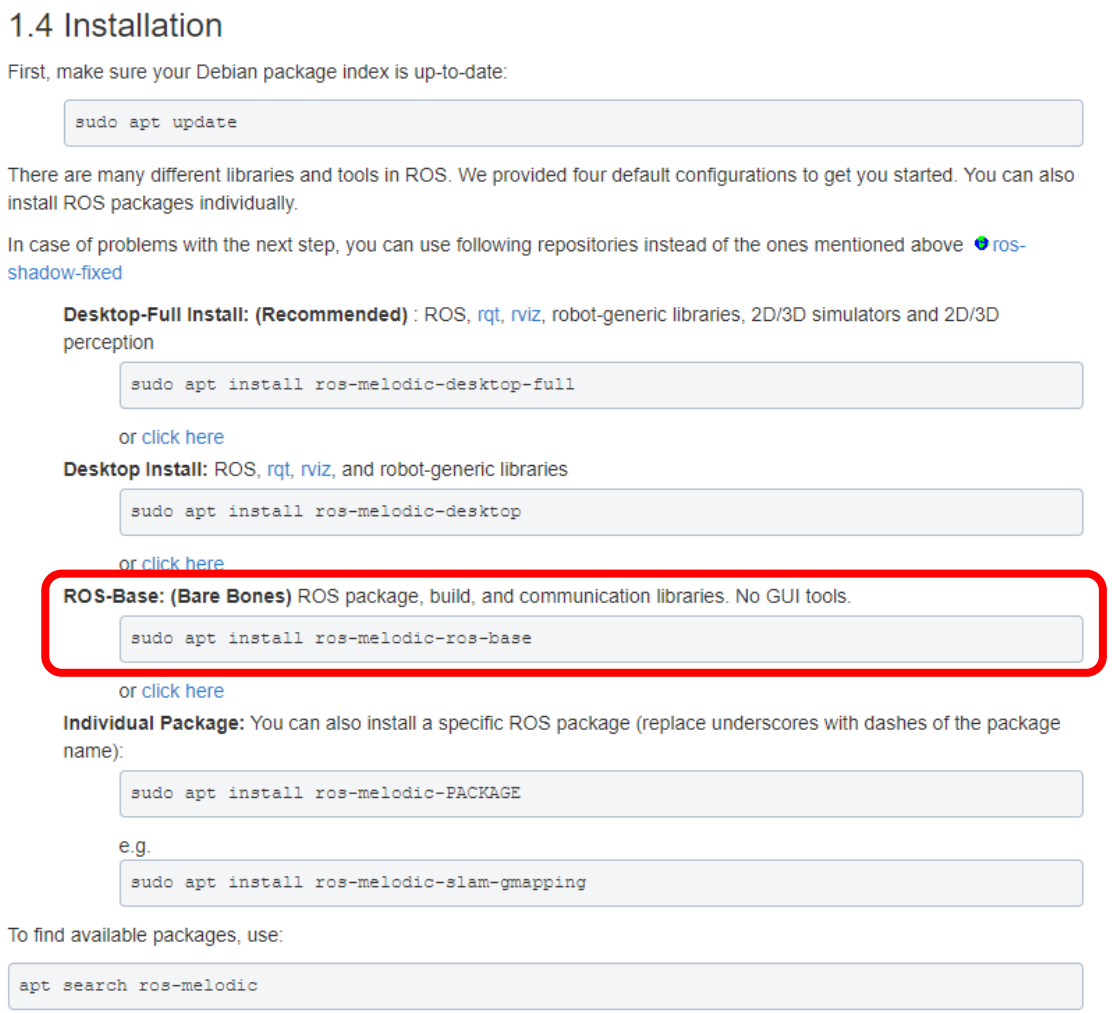
1.3 Set up your keys

1.4 Installation

1.5 Initialize *rosdep*

1.6 Environment setup

1.7 Dependencies for building packages



1.4 Installation

First, make sure your Debian package index is up-to-date:

```
sudo apt update
```

There are many different libraries and tools in ROS. We provided four default configurations to get you started. You can also install ROS packages individually.

In case of problems with the next step, you can use following repositories instead of the ones mentioned above ros-shadow-fixed

Desktop-Full Install: (Recommended) : ROS, rqt, rviz, robot-generic libraries, 2D/3D simulators and 2D/3D perception

```
sudo apt install ros-melodic-desktop-full
```

or click here

Desktop Install: ROS, rqt, rviz, and robot-generic libraries

```
sudo apt install ros-melodic-desktop
```

or click here

ROS-Base: (Bare Bones) ROS package, build, and communication libraries. No GUI tools.

```
sudo apt install ros-melodic-ros-base
```

or click here

Individual Package: You can also install a specific ROS package (replace underscores with dashes of the package name):

```
sudo apt install ros-melodic-PACKAGE
```

e.g.

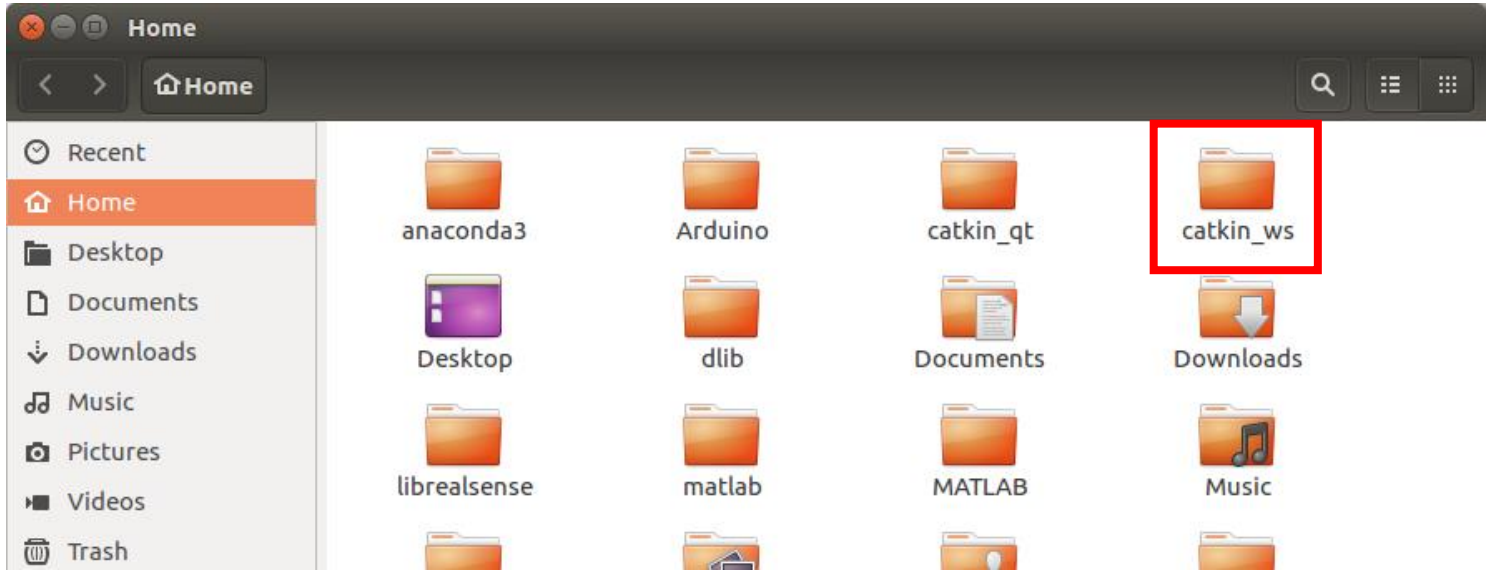
```
sudo apt install ros-melodic-slam-gmapping
```

To find available packages, use:

```
apt search ros-melodic
```

Create ROS Workspace

- A catkin workspace is a folder where you modify, build, and install catkin packages.
- Follow the step from “Create a ROS Workspace” guide
- http://wiki.ros.org/ROS/Tutorials/InstallingandConfiguringROSEnvironment



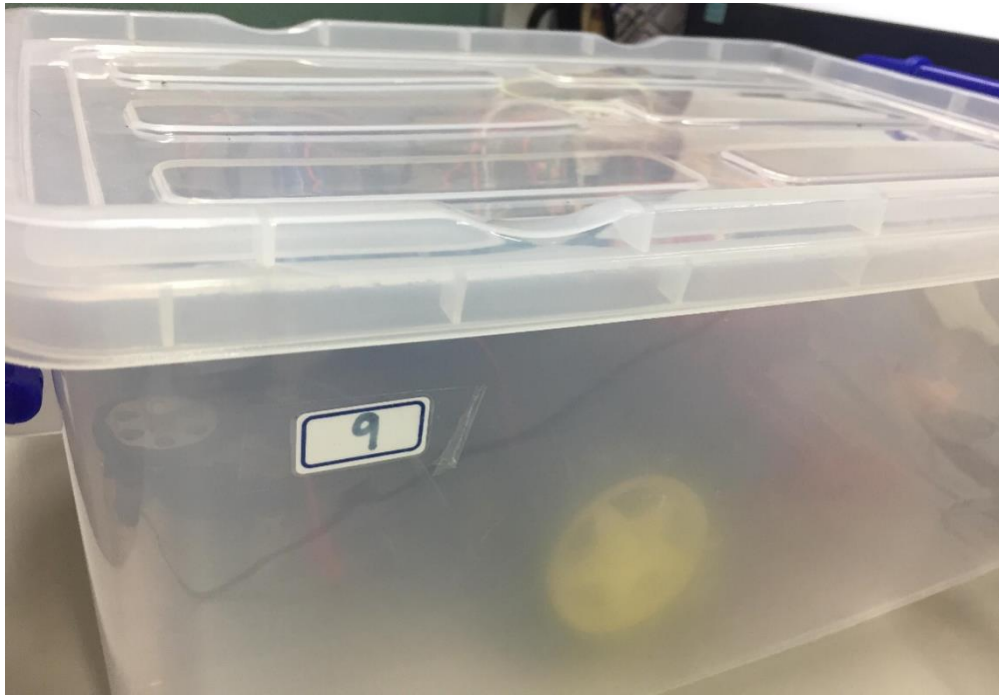
Checkpoint #1

Due : 10/07/2022

Outline

- Materials Check
- Checkpoint #1
- Lab632 opening hours

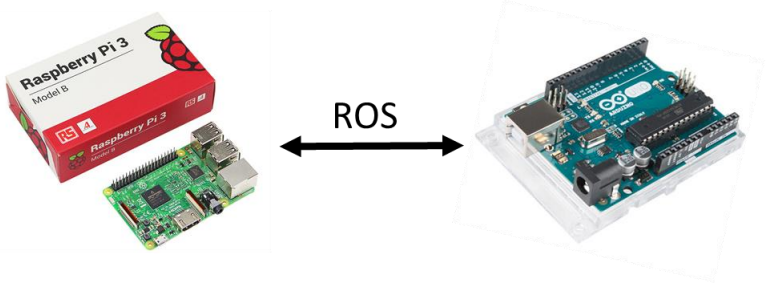
Materials Check



Checkpoint#1 Material List			
1	Raspberry Pi	4	USB A to B
2	Arduino UNO	5	USB A to micro B
3	16G SD card	6	Power Bank
Team _____			

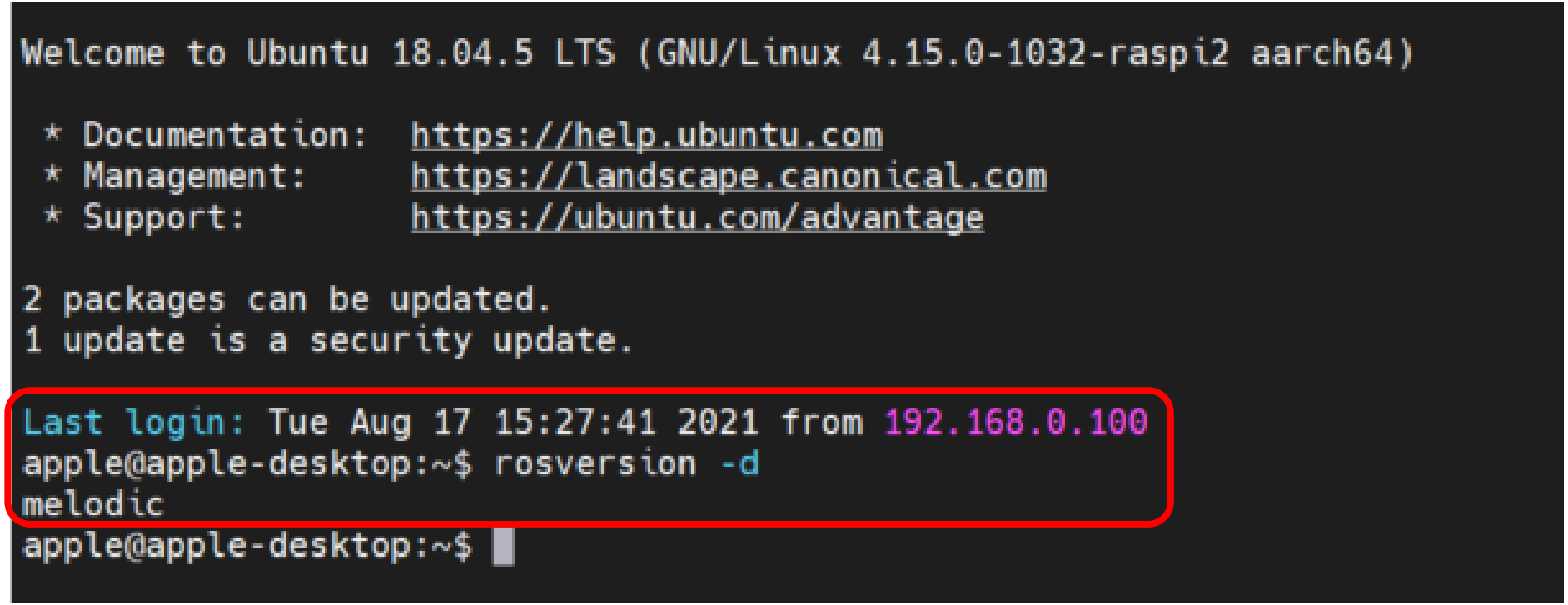
Checkpoint#1

- Purpose: setting the development environment for robot and using ROS system.
- Task1: Setting up the environment and remote control the RPi
  1. Install Ubuntu mate 18.04 in Raspberry Pi
  2. Install ROS Melodic in Raspberry Pi
  3. Using the Raspberry Pi by remote control it
- Task2: Using ROS to communicate
  1. Communicate between Raspberry Pi and Arduino by using ROS



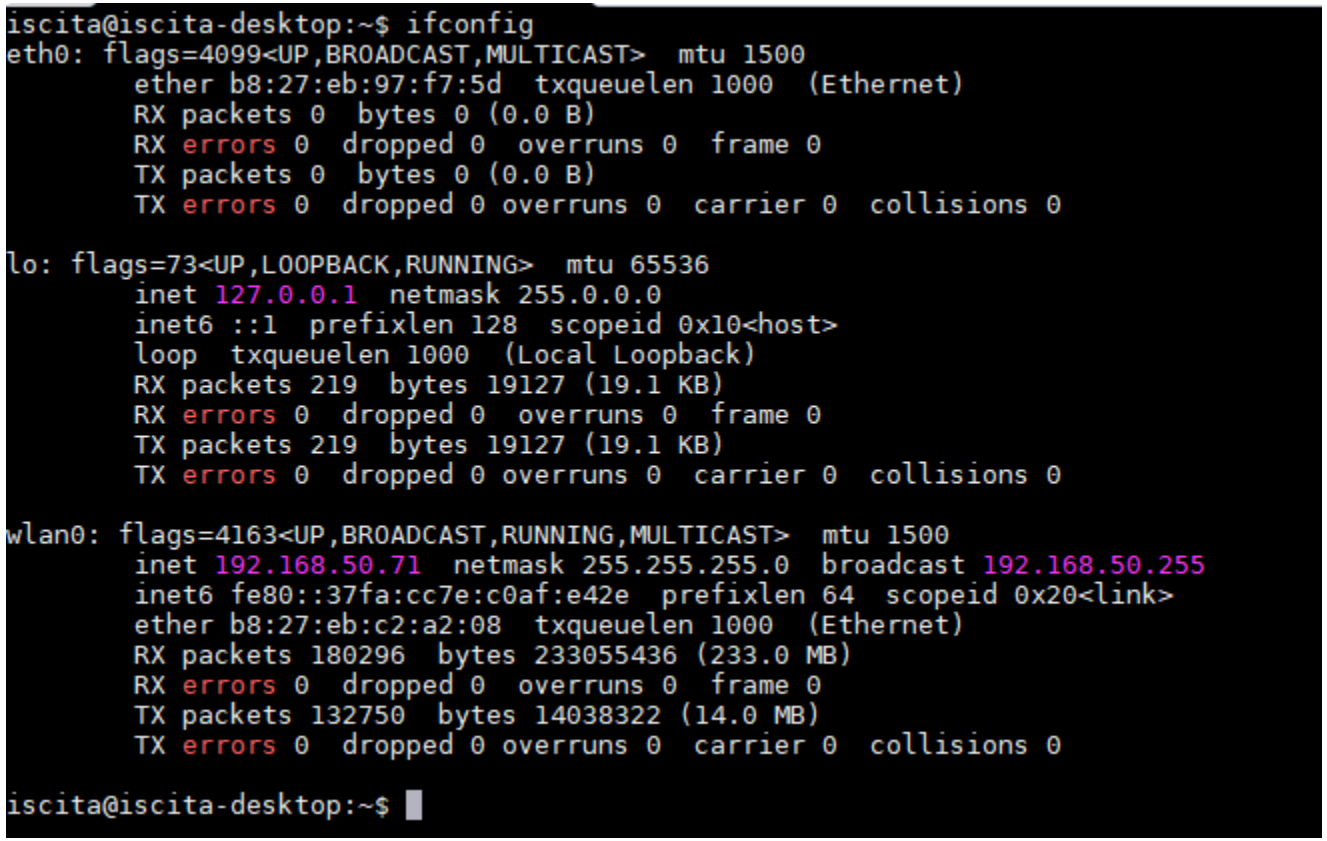
Checkpoint#1 – Task 1

Use ssh command to remote connect to Raspberry Pi from your PC and use rosversion command to show that you already install ROS melodic on Raspberry Pi. (30%)



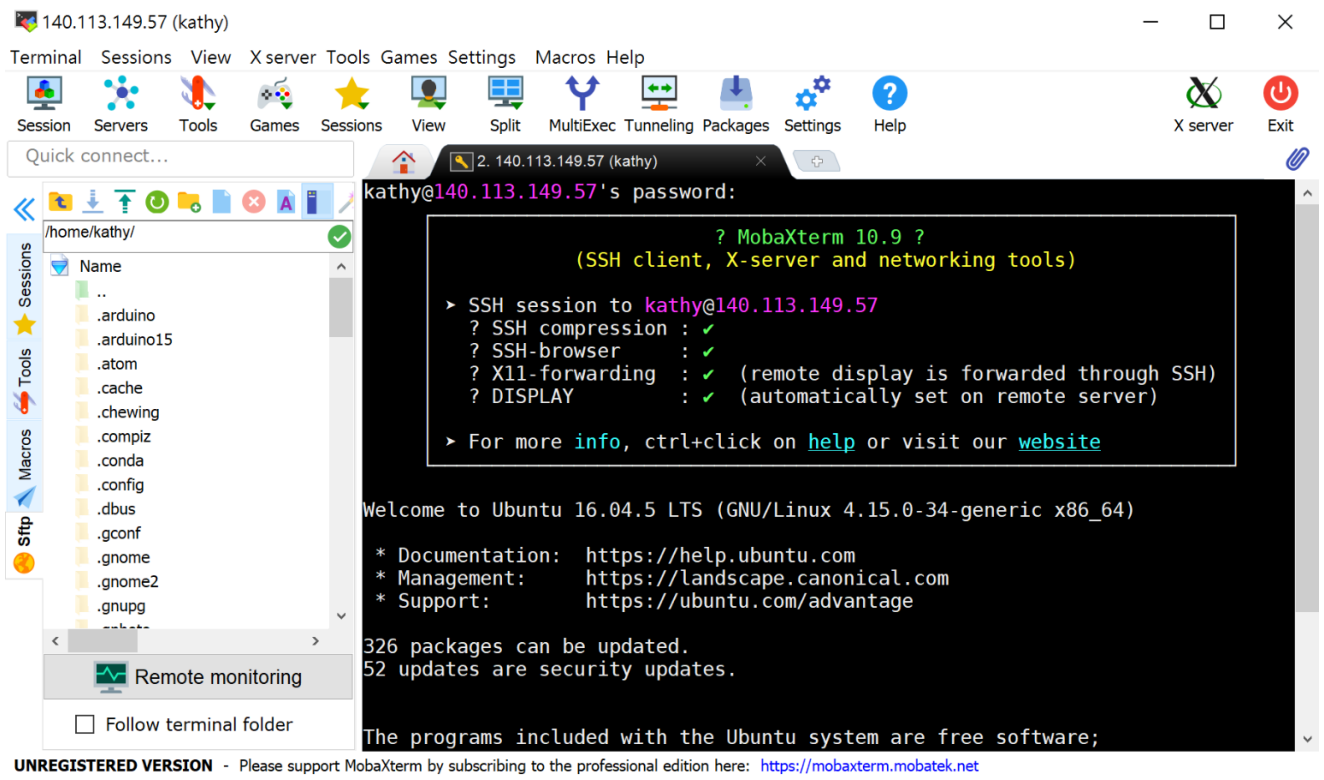
Setting ssh between RPi and PC

- You should fix RasperryPi's IP address, and install *ssh* on RasperryPi and PC(If you're using Linux OS).

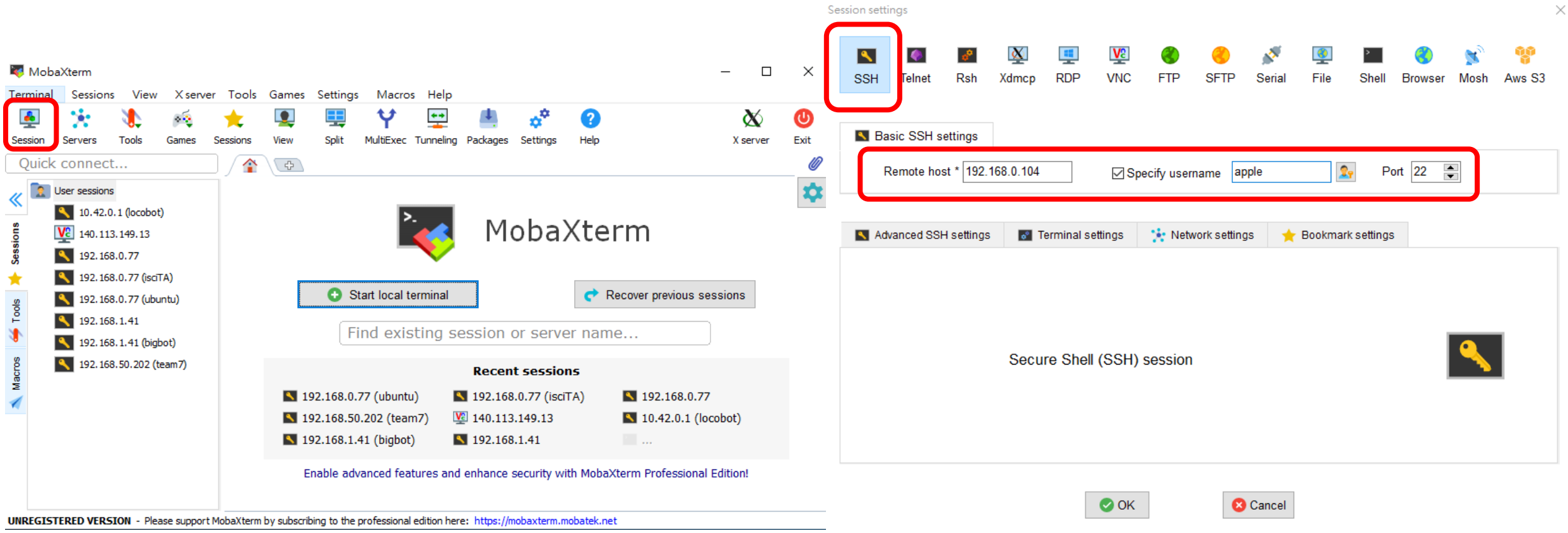


Tools for remote control

- ssh in terminal
- (recommend) MobaXterm



SSH Method - MobaXterm



Checkpoint#1 – Task 2

- Use command and library from rosserial package to create a program which contains publishers and subscribers. It means that your Raspberry Pi should send and receive messages from Arduino. In this program, Arduino should receive a number from Raspberry Pi, multiply it by 2, and then send it back to Raspberry Pi. Show the result from Terminal. (70%)

Checkpoint#1 – Task 2

1

Publish a topic
(a number)

RPi

Arduino

2

Subscribe the topic
(the number)

RPi

Arduino

3

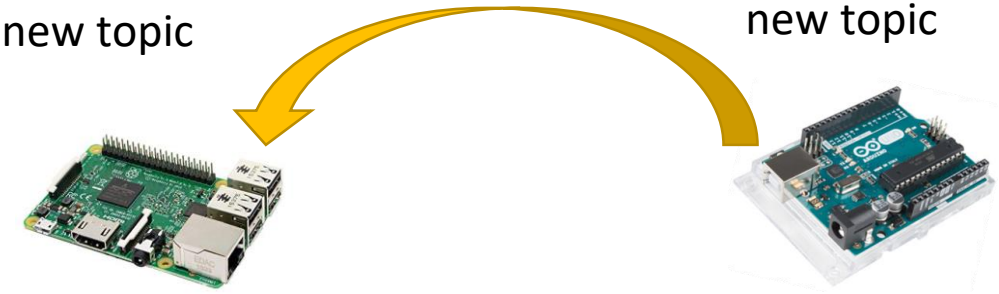
- Multiply the number by 2
- Make it as a new topic

Arduino

4

Subscribe the
new topic

Publish the
new topic



RPi

Arduino

Checkpoint#1 – Task 2

```
auto-starting new master
process[master]: started with pid [6818]
ROS_MASTER_URI=http://192.168.1.119:11311

setting /run_id to 6045c50e-0fb0-11e8-9c99-b827ebaa4d9b
process[rosout-1]: started with pid [6831]
started core service [/rosout]
process[connect_arduino-2]: started with pid [6835]
process[connect-3]: started with pid [6845]
user's input is 1
message from Arduino is 2
user's input is 2
message from Arduino is 4
user's input is 3
message from Arduino is 6
user's input is 4
message from Arduino is 8
user's input is 5
message from Arduino is 10
user's input is 6
message from Arduino is 12
user's input is 7
message from Arduino is 14
user's input is 8
message from Arduino is 16
user's input is 9
message from Arduino is 18
user's input is 10
message from Arduino is 20
```

Result from terminal

Lab632 opening hours

- Every Monday night 6:30 to 9:20 and Wednesday night 6:30 to 9:20 lab632 will open for everyone to practice.

Deadline

- Checkpoint#1 Demo : 10/7
- Checkpoint #1 Report : 10/14