

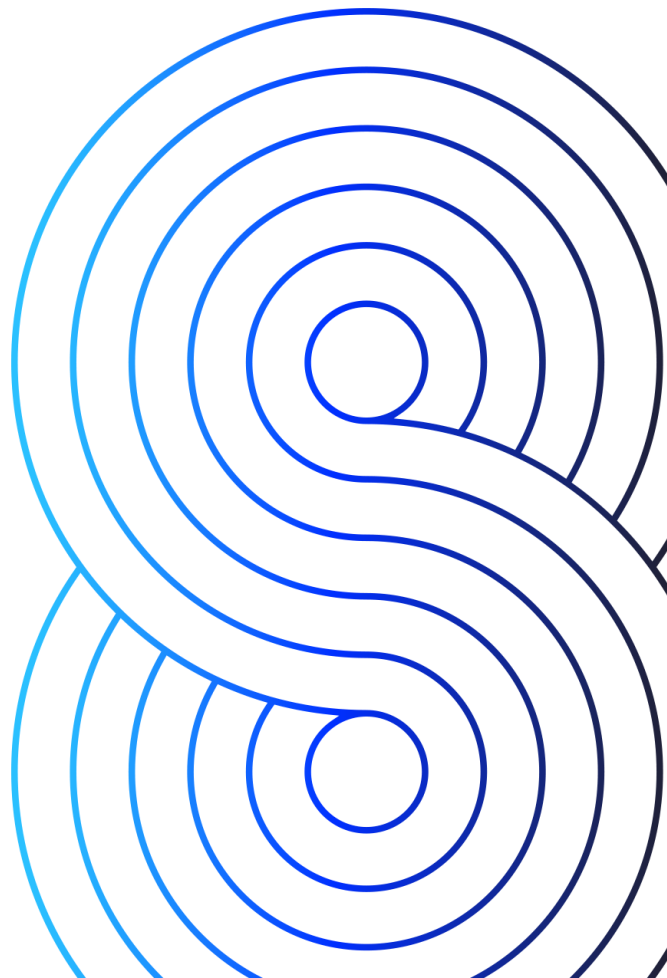
# Tizen.IoTivity.Init

## Configuring Preferences for Starting Tizen IoTivity

SWC | TPL | Eunyoung Lee

26/10/2017

**SOSCON**  
SAMSUNG  
OPEN SOURCE  
CONFERENCE



# Preparation

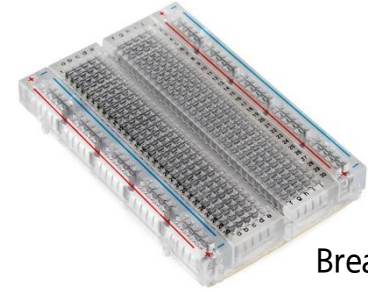
SOSCON



Micro SD Card



Ultrasonic Distance Sensor  
(HC-SR04)



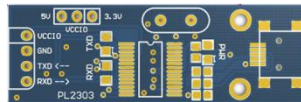
Bread board



Note PC



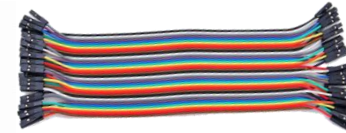
Raspberry Pi 3



Serial Port (PL2303)



Power Charger



Jumper Cable



1k ohm

2k ohm

Resistance

# Installing Tizen Platform

**How to install IoT Headless image on RPi3 (for ubuntu)**



1. Downloads a script that can install the image of tizen platform on the SD card.

- `rpi_sdcard_setup.sh`

```
$ wget https://git.tizen.org/cgit/apps/native/position-finder-client/plain/tool/rpi_sdcard_setup.sh
$ chmod +x rpi_sdcard_setup.sh
```

- You can also go to the location where the script is located and download it.
  - Clones position-finder-client repository with **Git Bash**.

```
$ git clone git://git.tizen.org/apps/native/position-finder-client (Branch : template)
```

- The script(`rpi_sdcard_setup.sh`) is located in the `position-finder-client/tools` directory.

```
$ cd position-finder-client/tools/
```

2. Runs the downloaded script on the SD card.

1. Inserts SD card reader with an SD card into the PC.

2. Runs the script.

```
$ ./rpi_sdcard_setup.sh
```

3. There are three questions when the script runs.

```
#1 : Please type device node of usb [press enter key to use default(sdb)] :
```

```
#2 : Please type binary version of tizen [press enter key to use default(latest)] :
```

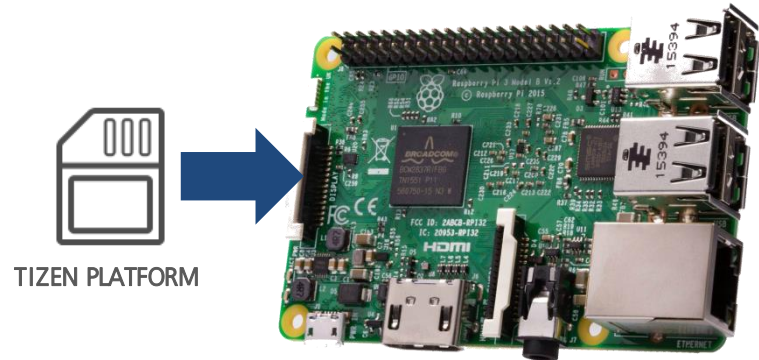
- When you press enter, latest binary is downloaded. If you want another version, please enter the desired version on date. (e.g. 20171020.3)
- Information about the version can be found on <http://download.tizen.org/snapshots/tizen/4.0-unified/>

```
#3 : /dev/sdb will be formatted, Is it OK? [y/n]
```

# What the script(rpi\_sdcard\_setup.sh) does

1. Downloading files needed for image installation
2. Downloading the IoT Headless Image
3. Making partitions of SD Card
4. Fusing boot & platform image to SD Card
5. Copying firmwares for Wi-Fi and Bluetooth

\* *NOTE*: The above should be done on Ubuntu PC.



# 1. Downloading files needed for image installation

## 1. Installing pv package on Ubuntu PC.

```
$ sudo apt-get install pv
```

## 2. Downloading the fusing-script for RPi3.

```
$ wget https://git.tizen.org/cgit/platform/kernel/linux-rpi3/plain/scripts/sd_fusing_rpi3.sh?h=submit/tizen/20170725.223437 --output-document=$download_dir/$script_fusing
```

## 3. Downloading firmwares for Wi-Fi and Bluetooth.

```
$ wget https://github.com/RPi-Distro/firmware-nonfree/raw/master/brcm80211/brcm/brcmfmac43430-sdio.bin  
$ wget https://github.com/RPi-Distro/firmware-nonfree/raw/master/brcm80211/brcm/brcmfmac43430-sdio.txt  
$ wget https://github.com/OpenELEC/misc-firmware/raw/master/firmware/brcm/BCM43430A1.hcd
```

## 2. Downloading the IoT Headless Image

- Kernel & Module Image

Access <http://download.tizen.org/snapshots/tizen/unified/latest/images/standard/iot-boot-arm64-rpi3/>  
Download **tizen-unified\_2017xxxx.x\_iot-boot-arm64-rpi3.tar.gz**

- Tizen Platform Image

Access <http://download.tizen.org/snapshots/tizen/unified/latest/images/standard/iot-headless-2parts-armv7l-rpi3/>  
Download **tizen-unified\_2017xxxx.x\_iot-headless-2parts-armv7l-rpi3.tar.gz**

## 3. Making partitions of SD Card

1. Check the device node of SD card. (ex. sdx : x is alphabet)
2. Make partitions using the fusing-script.\*

```
$ sudo ./sd_fusing_rpi3.sh -d /dev/sdb --format
```

Device: /dev/sdb

/dev/sdb will be formatted, Is it OK? [y/n]

y

-----  
Start /dev/sdb format

=====

Label	dev	size
-------	-----	------

=====

boot	/dev/sdb1	64 MB
------	-----------	-------

rootfs	/dev/sdb2	3072 MB
--------	-----------	---------

system-data	/dev/sdb3	512 MB
-------------	-----------	--------

[Extend]	/dev/sdb4	
----------	-----------	--

user	/dev/sdb5	26848 MB
------	-----------	----------

modules	/dev/sdb6	20 MB
---------	-----------	-------

ramdisk	/dev/sdb7	8 MB
---------	-----------	------

Remove partition table...

16+0 records in

16+0 records out

8192 bytes (8.2 kB) copied, 0.22686 s, 36.1 kB/s

Checking that no-one is using this disk right now ...

OK

Disk /dev/sdb: 30528 cylinders, 64 heads, 32 sectors/track

sfdisk: ERROR: sector 0 does not have an msdos signature

/dev/sdb: unrecognized partition table type

Old situation:

No partitions found

New situation:

Units = mebibytes of 1048576 bytes, blocks of 1024 bytes, counting from 0

Device	Boot	Start	End	MiB	#blocks	Id	System
/dev/sdb1	*	4	67	64	65536	e	W95 FAT16 (LBA)
/dev/sdb2		68	3139	3072	3145728	83	Linux
/dev/sdb3		3140	3651	512	524288	83	Linux
/dev/sdb4		3652	30527	26876	27521024	5	Extended
/dev/sdb5		3652+	30499	26848	27492351+	83	Linux
/dev/sdb6		30500+	30519	20	20479+	83	Linux
/dev/sdb7		30520+	30527	8	8191+	83	Linux

Successfully wrote the new partition table

Re-reading the partition table ...

If you created or changed a DOS partition, /dev/foo7, say, then use dd(1)

to zero the first 512 bytes: dd if=/dev/zero of=/dev/foo7 bs=512 count=1

(See fdisk(8).)

mkfs.fat 3.0.26 (2014-03-07)

mkfs.fat: warning - lowercase labels might not work properly with DOS or Windows

End /dev/sdb format

-----

## 4. Fusing boot & platform image to SD Card

1. Write boot image to SD Card using the fusing-script.\*

```
$ sudo ./sd_fusing_rpi3.sh -d /dev/sdb -b tizen-unified_2017xxxx.x_iot-boot-arm64-rpi3.tar.gz
```

2. Write IoT Headless image to SD Card using the fusing-script.\*

```
$ sudo ./sd_fusing_rpi3.sh -d /dev/sdb -b tizen-unified_2017xxxx.x_iot-headless-2parts-armv7l-rpi3.tar.gz
```

## 5. Copying firmwares for Wi-Fi and Bluetooth

Copy firmwares for Wi-Fi and Bluetooth to SD Card

```
$ mkdir rootfs
$ sudo mount /dev/sdb2 rootfs
$ sudo mkdir -p rootfs/usr/etc/bluetooth
$ sudo cp BCM43430A1.hcd rootfs/usr/etc/bluetooth
$ sudo mkdir -p rootfs/usr/lib/firmware/brcm
$ sudo cp brcmfmac43430-sdio.* rootfs/usr/lib/firmware/brcm
$ sync
$ sudo umount rootfs
$ rmdir rootfs
```

Raspberry Pi3 downloader, version 0.1

Device: /dev/sdb  
Fusing binary:  
tizen-unified\_20170814.1\_common-boot-arm64-rpi3.tar.gz

-----  
Skip /dev/sdb format  
-----  
modules.img  
boot.img

[Fusing boot.img]  
umount: /dev/sdb1: not mounted  
65536+0 records in81MB/s] [=====>] 97% ETA 0:00:00  
65536+0 records out  
33554432 bytes (34 MB) copied, 4.09045 s, 8.2 MB/s  
32MB 0:00:04 [ 7.8MB/s] [=====>] 100%  
65536+0 records in  
65536+0 records out  
33554432 bytes (34 MB) copied, 6.93111 s, 4.8 MB/s  
resize2fs 1.42.9 (4-Feb-2014)  
resize2fs: Bad magic number in super-block while trying to open /dev/sdb1  
Couldn't find valid filesystem superblock.

[Fusing modules.img]  
umount: /dev/sdb6: not mounted  
29864+0 records in  
29864+0 records out  
15290368 bytes (15 MB) copied, 0.0488528 s, 313 MB/s  
14.6MB 0:00:00 [ 299MB/s] [=====>] 100%  
29864+0 records in  
29864+0 records out  
15290368 bytes (15 MB) copied, 1.44288 s, 10.6 MB/s  
resize2fs 1.42.9 (4-Feb-2014)  
Resizing the filesystem on /dev/sdb6 to 5119 (4k) blocks.  
The filesystem on /dev/sdb6 is now 5119 blocks long.

Screen when boot image is installed.\*

```
Device: /dev/sdb
Fusing binary:
tizen-unified_20170814.1_common-iot_core-2parts-armv7l-rpi3.tar.gz
```

```
-----
Skip /dev/sdb format
-----
```

```
ramdisk.img
rootfs.img
system-data.img
```

#### [Fusing ramdisk.img]

```
umount: /dev/sdb7: not mounted
14336+0 records in
14336+0 records out
7340032 bytes (7.3 MB) copied, 0.0141529 s, 519 MB/s
 7MB 0:00:00 [ 495MB/s] [=====>] 100%
14336+0 records in
14336+0 records out
7340032 bytes (7.3 MB) copied, 0.616807 s, 11.9 MB/s
resize2fs 1.42.9 (4-Feb-2014)
Resizing the filesystem on /dev/sdb7 to 8188 (1k) blocks.
The filesystem on /dev/sdb7 is now 8188 blocks long.
```

#### [Fusing rootfs.img]

```
umount: /dev/sdb2: not mounted
297704+0 records in0kB/s] [=====>] 93% ETA 0:00:00
297704+0 records out
152424448 bytes (152 MB) copied, 6.17764 s, 24.7 MB/s
145MB 0:00:06 [23.5MB/s] [=====>] 100%
0+2193 records in
0+2193 records out
152424448 bytes (152 MB) copied, 13.5183 s, 11.3 MB/s
resize2fs 1.42.9 (4-Feb-2014)
Resizing the filesystem on /dev/sdb2 to 786432 (4k) blocks.
The filesystem on /dev/sdb2 is now 786432 blocks long.
```

#### [Fusing system-data.img]

```
umount: /dev/sdb3: not mounted
130952+0 records in
130952+0 records out
67047424 bytes (67 MB) copied, 0.780652 s, 85.9 MB/s
63.9MB 0:00:00 [81.8MB/s] [=====>] 100%
0+919 records in
0+919 records out
67047424 bytes (67 MB) copied, 5.95975 s, 11.3 MB/s
resize2fs 1.42.9 (4-Feb-2014)
Resizing the filesystem on /dev/sdb3 to 131072 (4k) blocks.
The filesystem on /dev/sdb3 is now 131072 blocks long.
```

Screen when IoT Headless Image is installed.\*

# Booting the RPi3

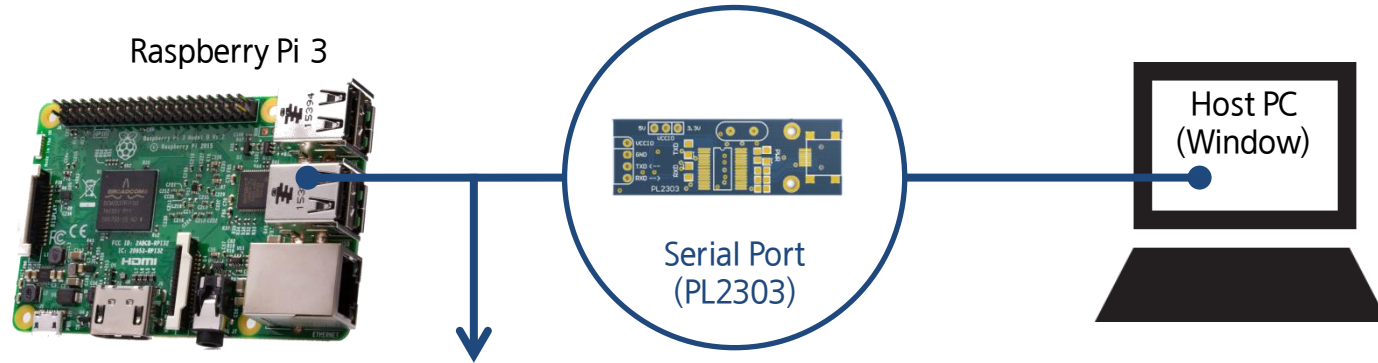
Connecting a Host PC to RPi3 via UART



1. Inserting SD Card to RPi3.
2. Connecting a Host PC to RPi3 via UART Board.
3. Connecting power to RPi3.
4. Entering user ID and password

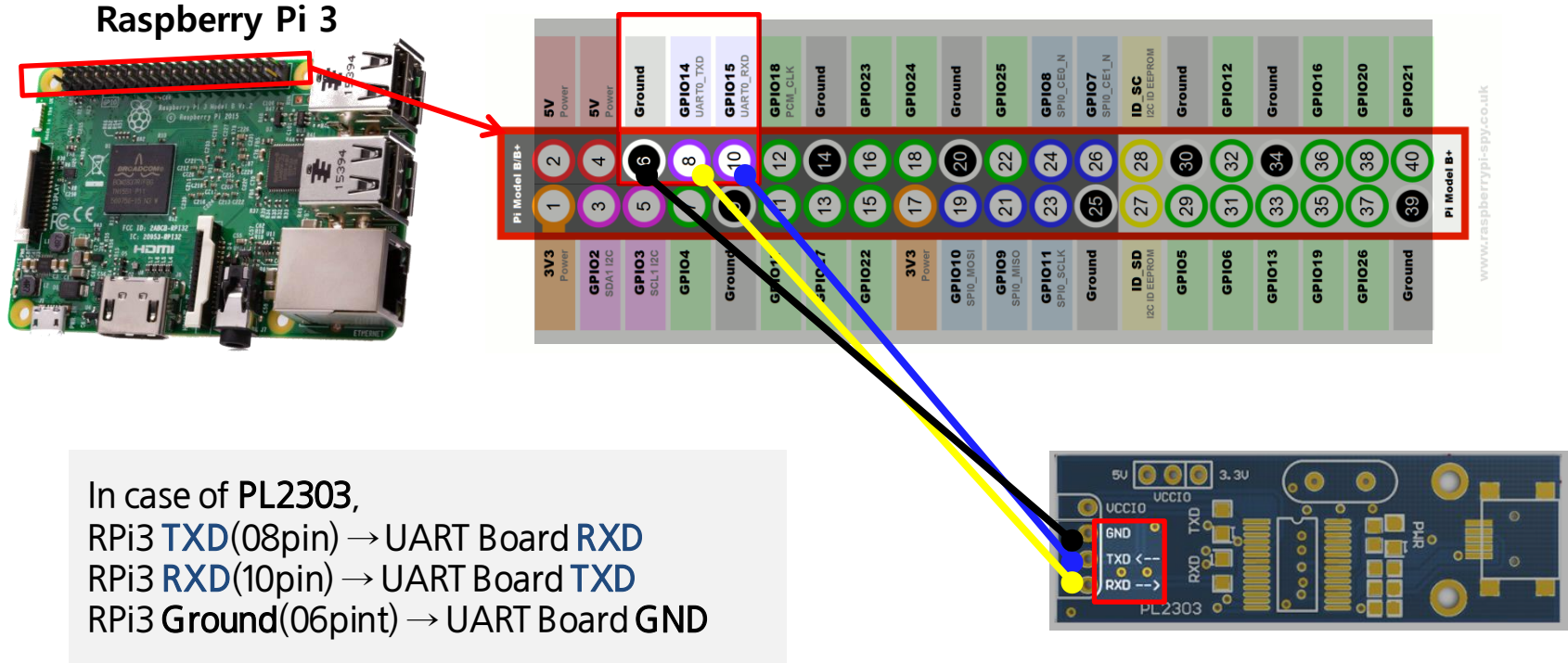
# Booting the RPi3

1. Inserting SD Card to RPi3.
2. Connecting a Host PC to RPi3 via UART Board.



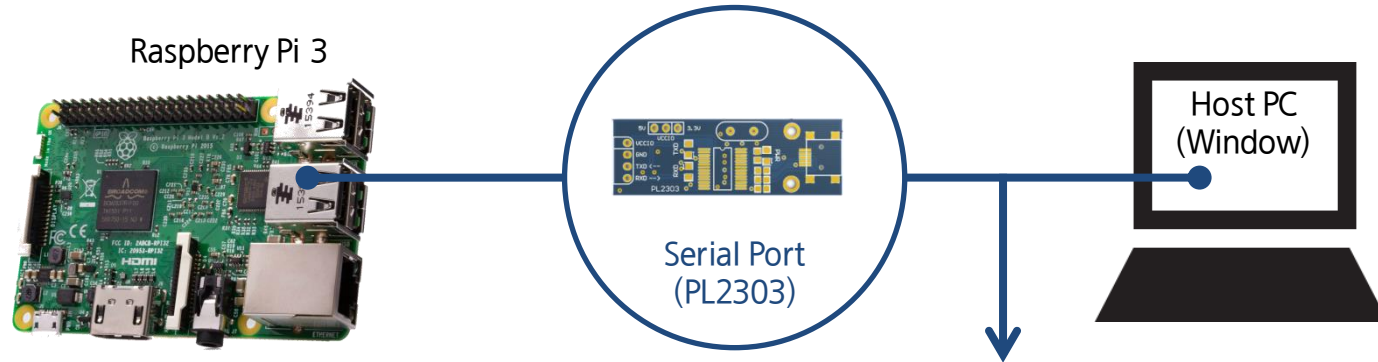
1) Information of connecting to UART Board\*

# 1) Information of connecting to UART Board\*



# Booting the RPi3

1. Inserting SD Card to RPi3.
2. Connecting a Host PC to RPi3 via UART Board.



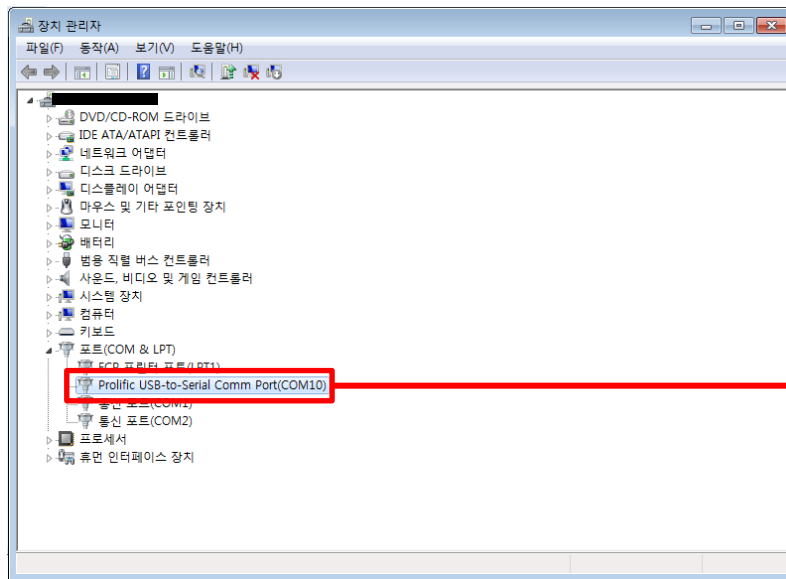
2) Opening a Serial Terminal with Host PC(Windows)\*

Use PuTTY program in practice.

## 2) Opening a Serial Terminal with Host PC(Windows)\*

### 1. Install the appropriate drivers.

Download the PL2303 Windows Driver : [http://www.prolific.com.tw/US/ShowProduct.aspx?p\\_id=225&pcid=41](http://www.prolific.com.tw/US/ShowProduct.aspx?p_id=225&pcid=41)



When the serial port is recognized:

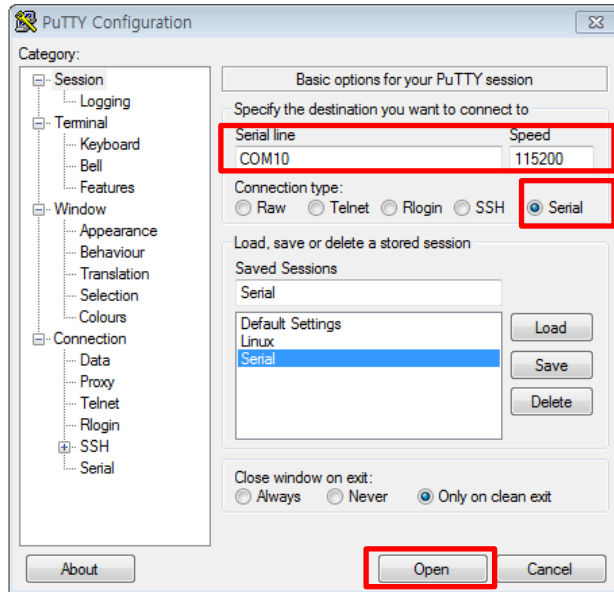
**Prolific USB-to-Serial Comm Port**(COM#)



## 2) Opening a Serial Terminal with Host PC(Windows)\*

### 1. Execute a PuTTY program to open serial terminal.

Download the PuTTY terminal emulator : <http://www.putty.org/>



Configure the PuTTY menu as follows:

- Under Connection type, select **Serial**.
- In the **Serial line** field, enter the **COM#** for your board, such as COM10.

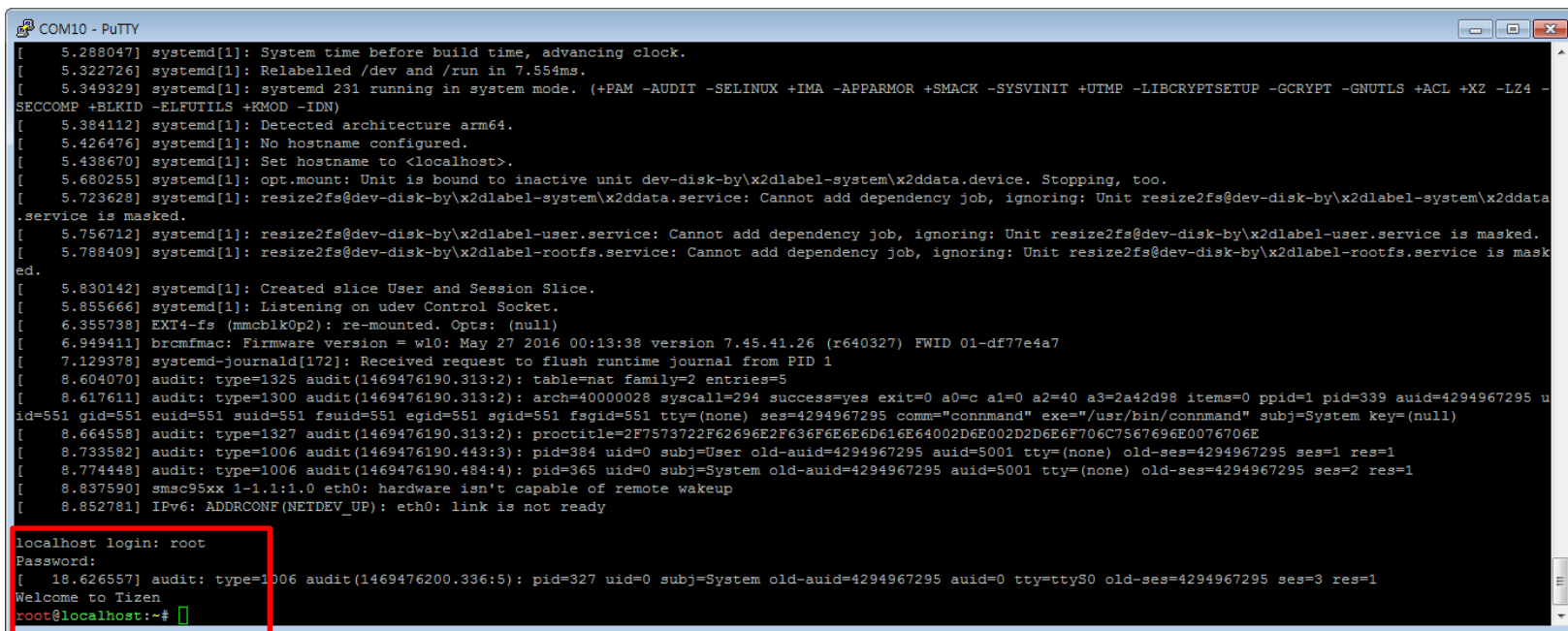
**\* Note :** If you did not identify your COM# when setting up your board, navigate to the Device Manager and check for an entry called **USB Serial Port**. The COM# is displayed next to the USB Serial Port entry.



## 3. Connecting power to RPi3.

## 4. Entering user ID and password

- User id : root / Password : tizen (The password is not shown when you enter it.)



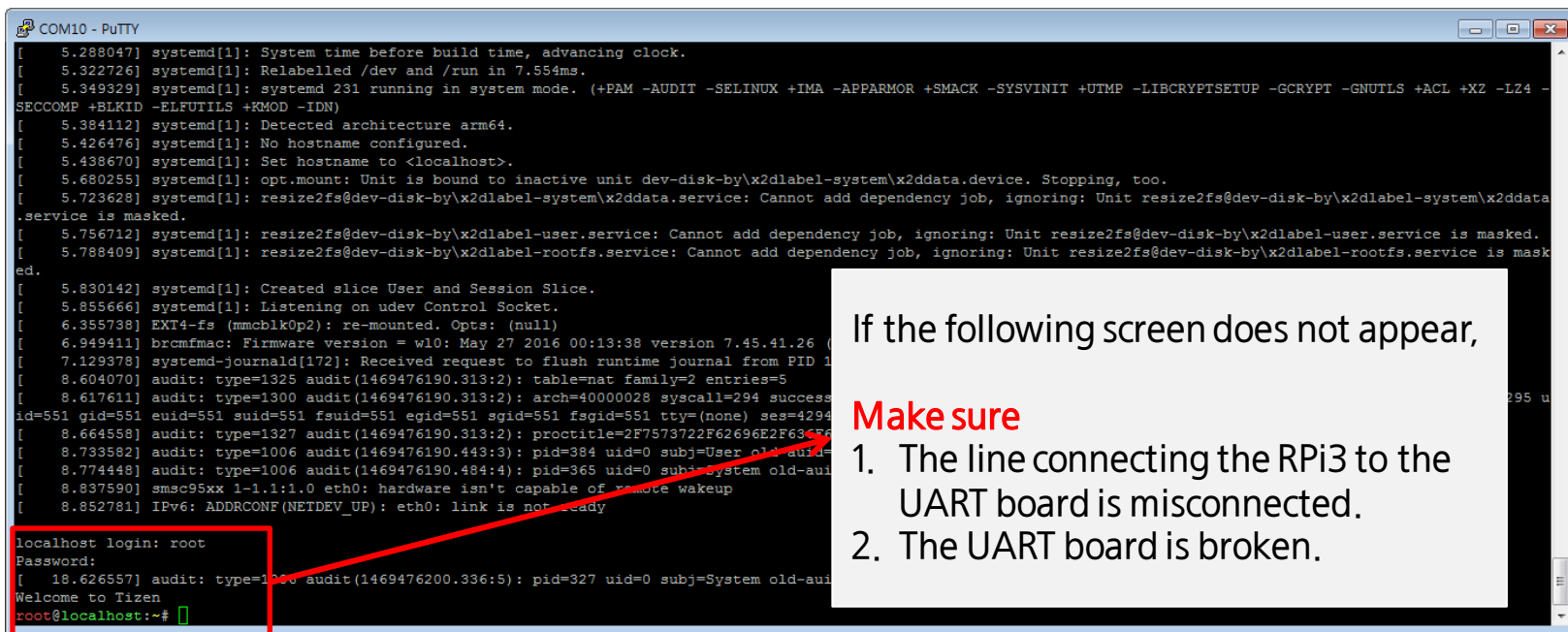
```
COM10 - PuTTY
[ 5.288047] systemd[1]: System time before build time, advancing clock.
[ 5.322726] systemd[1]: Relabelled /dev and /run in 7.554ms.
[ 5.349329] systemd[1]: systemd 231 running in system mode. (+PAM -AUDIT -SELINUX +IMA -APPARMOR +SMACK -SYSVINIT +UTMP -LIBCRYPTSETUP -GCRYPT -GNUTLS +ACL +XZ -LZ4 -
SECCOMP +BLKID -ELFUTILS +KMOD -IDN)
[ 5.384112] systemd[1]: Detected architecture arm64.
[ 5.426476] systemd[1]: No hostname configured.
[ 5.438670] systemd[1]: Set hostname to <localhost>.
[ 5.680255] systemd[1]: opt.mount: Unit is bound to inactive unit dev-disk-by\x2dlabel-system\x2ddata.device. Stopping, too.
[ 5.723628] systemd[1]: resize2fs@dev-disk-by\x2dlabel-system\x2ddata.service: Cannot add dependency job, ignoring: Unit resize2fs@dev-disk-by\x2dlabel-system\x2ddata
.service is masked.
[ 5.756712] systemd[1]: resize2fs@dev-disk-by\x2dlabel-user.service: Cannot add dependency job, ignoring: Unit resize2fs@dev-disk-by\x2dlabel-user.service is masked.
[ 5.788409] systemd[1]: resize2fs@dev-disk-by\x2dlabel-rootfs.service: Cannot add dependency job, ignoring: Unit resize2fs@dev-disk-by\x2dlabel-rootfs.service is mask
ed.
[ 5.830142] systemd[1]: Created slice User and Session Slice.
[ 5.855666] systemd[1]: Listening on udev Control Socket.
[ 6.355738] EXT4-fs (mmcblk0p2): re-mounted. Opts: (null)
[ 6.949411] brcmfmac: Firmware version = wl0: May 27 2016 00:13:38 version 7.45.41.26 (r640327) FWID 01-df77e4a7
[ 7.129378] systemd-journald[172]: Received request to flush runtime journal from PID 1
[ 8.604070] audit: type=1325 audit(1469476190.313:2): table=nat family=2 entries=5
[ 8.617611] audit: type=1300 audit(1469476190.313:2): arch=400000028 syscall=294 success=yes exit=0 a0=c a1=0 a2=40 a3=2a42d98 items=0 ppid=1 pid=339 auid=4294967295 u
id=551 gid=551 euid=551 suid=551 fsuid=551 egid=551 sgid=551 fsgid=551 tty=(none) ses=4294967295 comm="connmand" exe="/usr/bin/connmand" subj=System key=(null)
[ 8.664558] audit: type=1327 audit(1469476190.313:2): proctitle=2F7573722F62696E2F636F6E6D616E64002D6E002D2D6E6F706C7567696E0076706E
[ 8.733582] audit: type=1006 audit(1469476190.443:3): pid=384 uid=0 subj=User old-auid=4294967295 auid=5001 tty=(none) old-ses=4294967295 ses=1 res=1
[ 8.774448] audit: type=1006 audit(1469476190.484:4): pid=365 uid=0 subj=System old-auid=4294967295 auid=5001 tty=(none) old-ses=4294967295 ses=2 res=1
[ 8.837590] smsc95xx 1-1.1:1.0 eth0: hardware isn't capable of remote wakeup
[ 8.852781] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready

localhost login: root
Password:
[ 18.626557] audit: type=1006 audit(1469476200.336:5): pid=327 uid=0 subj=System old-auid=4294967295 auid=0 tty=ttyS0 old-ses=4294967295 ses=3 res=1
Welcome to Tizen
root@localhost:~#
```

3. Connecting power to RPi3.

4. Entering user ID and password

- User id : root / Password : tizen (The password is not shown when you enter it.)



```
COM10 - PuTTY
[ 5.288047] systemd[1]: System time before build time, advancing clock.
[ 5.322726] systemd[1]: Relabelled /dev and /run in 7.554ms.
[ 5.349329] systemd[1]: systemd 231 running in system mode. (+PAM -AUDIT -SELINUX +IMA -APPARMOR +SMACK -SYSVINIT +UTMP -LIBCRYPTSETUP -GCRYPT -GNUTLS +ACL +XZ -LZ4 -
SECCOMP +BLKID -ELFUTILS +KMOD -IDN)
[ 5.384112] systemd[1]: Detected architecture arm64.
[ 5.426476] systemd[1]: No hostname configured.
[ 5.438670] systemd[1]: Set hostname to <localhost>.
[ 5.680255] systemd[1]: opt.mount: Unit is bound to inactive unit dev-disk-by\x2dlabel-system\x2ddata.device. Stopping, too.
[ 5.723628] systemd[1]: resize2fs@dev-disk-by\x2dlabel-system\x2ddata.service: Cannot add dependency job, ignoring: Unit resize2fs@dev-disk-by\x2dlabel-system\x2ddata
.service is masked.
[ 5.756712] systemd[1]: resize2fs@dev-disk-by\x2dlabel-user.service: Cannot add dependency job, ignoring: Unit resize2fs@dev-disk-by\x2dlabel-user.service is masked.
[ 5.788409] systemd[1]: resize2fs@dev-disk-by\x2dlabel-rootfs.service: Cannot add dependency job, ignoring: Unit resize2fs@dev-disk-by\x2dlabel-rootfs.service is mask
ed.
[ 5.830142] systemd[1]: Created slice User and Session Slice.
[ 5.855666] systemd[1]: Listening on udev Control Socket.
[ 6.355738] EXT4-fs (mmcblk0p2): re-mounted. Opts: (null)
[ 6.949411] brcmfmac: Firmware version = w10: May 27 2016 00:13:38 version 7.45.41.26
[ 7.129378] systemd-journald[172]: Received request to flush runtime journal from PID 1
[ 8.604070] audit: type=1325 audit(1469476190.313:2): table=nat family=2 entries=5
[ 8.617611] audit: type=1300 audit(1469476190.313:2): arch=40000028 syscall=294 success
id=551 gid=551 euid=551 suid=551 fsuid=551 egid=551 sgid=551 fsgid=551 tty=(none) ses=4294
[ 8.664558] audit: type=1327 audit(1469476190.313:2): proctitle=2F7573722F62696E2F6338
[ 8.733582] audit: type=1006 audit(1469476190.443:3): pid=384 uid=0 subj=User old-auid=
[ 8.774448] audit: type=1006 audit(1469476190.484:4): pid=365 uid=0 subj=system old-auid=
[ 8.837590] smsc95xx 1-1.1:1.0 eth0: hardware isn't capable of remote wakeup
[ 8.852781] IPv6: ADDRCONF(NETDEV_UP): eth0: link is not ready

localhost login: root
Password:
[ 18.626557] audit: type=1300 audit(1469476200.336:5): pid=327 uid=0 subj=System old-auid=
Welcome to Tizen
root@localhost:~#
```

If the following screen does not appear,

**Make sure**

1. The line connecting the RPi3 to the UART board is misconnected.
2. The UART board is broken.

# **Building a Tizen project**

- **Cloning Tizen repositories**
- **Building the project with Tizen Studio**



# Prerequisite

## 1. Downloading the Git.

<https://www.git-scm.com/downloads>

## 2. Installing the Tizen Studio.

<https://developer.tizen.org/development/tizen-studio/download>



## Template source code

- position-finder-server

Git path : apps/native/position-finder-server  
Branch : template

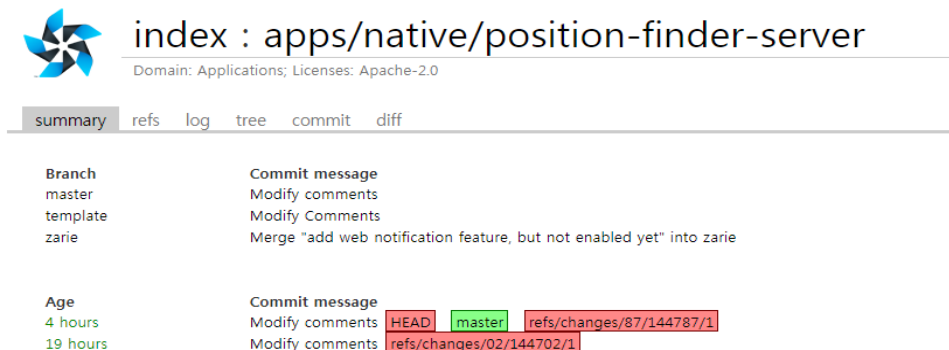
- position-finder-client

Git path : apps/native/position-finder-client

- 1) Access the Tizen Project GIT Repositories.

<https://git.tizen.org/>

- 2) Searching for **position-finder-server**.



The screenshot shows the GitHub-style interface for the Tizen repository 'apps/native/position-finder-server'. It includes the Tizen logo, the repository name, and the domain/licenses. Below this is a navigation bar with tabs: 'summary' (selected), 'refs', 'log', 'tree', 'commit', and 'diff'. The main content area is divided into two columns. The left column lists branches: 'master', 'template', and 'zarie'. The right column shows commit messages. The first commit message is 'Merge "add web notification feature, but not enabled yet" into zarie'. Below this, there is a section for 'Age' and 'Commit message'. The 'Age' section shows '4 hours' and '19 hours'. The 'Commit message' section shows 'Modify comments' with a red box around 'HEAD', a green box around 'master', and a red box around 'refs/changes/87/144787/1'. Another commit message shows 'Modify comments' with a red box around 'refs/changes/02/144702/1'.

- 3) How to clone the repository : apps/native/position-finder-server.\*
- 4) Running a Git Bash and cloning the Tizen repository.\*
- 5) How to change the branch of the project.\*
- 6) Changing the branch of the Tizen Project.\*

### 3) How to clone the repository : apps/native/position-finder-server\*

```
$ git clone <Git address of the repository>
```

```
$ git clone git://git.tizen.org/apps/native/position-finder-server
```

<https://git.tizen.org/>



index : apps/native/position-finder-server

Domain: Applications; Licenses: Apache-2.0

summary refs log tree commit diff

#### Branch

master  
template  
zarie

#### Commit message

Modify comments  
Modify Comments  
Merge "add web notification feature, but not enabled yet" into zarie

#### Age

4 hours  
19 hours  
19 hours  
2 days  
7 days  
7 days  
7 days  
10 days  
2017-08-03  
2017-08-01  
[...]

#### Commit message

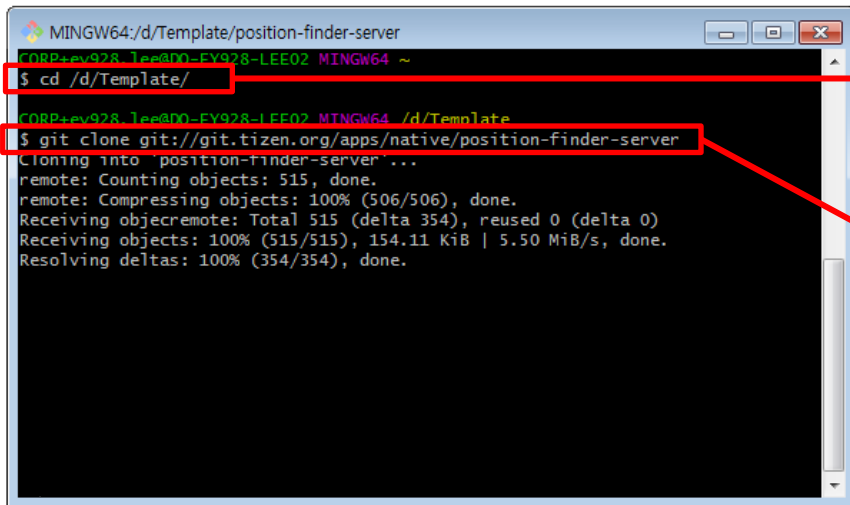
Modify comments HEAD master refs/changes/87/144787/1  
Modify comments refs/changes/02/144702/1 refs/changes/91/144691/1  
CBOR file in res dir and auto-naming in spec refs/changes/91/144691/1  
Modify comments about connectivity API refs/changes/04/144204/1  
Add API for LED resource to check human behavior refs/changes/53/143753/1  
Add comments for sensor and connectivity API refs/changes/98/143698/1  
Trivial modification refs/changes/42/143642/2  
Copy CBOR file from res dir to data dir for protecting crash refs/changes/89/143089/1  
Remove presence API - not working, maybe it'll be deprecated refs/changes/52/142152/1  
Merge "Add connectivity APIs for various data types"

#### Clone

<https://git.tizen.org/cgi/apps/native/position-finder-server>  
<git://git.tizen.org/apps/native/position-finder-server>

## 4) Running a Git Bash and cloning the Tizen repository\*

Git bash



The screenshot shows a Git Bash terminal window with the title bar 'MINGW64:/d/Template/position-finder-server'. The terminal prompt is 'CORP+ev928.1ee8DD-FY928-LEE02 MINGW64 ~'. The first command entered is '\$ cd /d/Template/' which is highlighted with a red box. The second command entered is '\$ git clone git://git.tizen.org/apps/native/position-finder-server' which is also highlighted with a red box. The output of the clone command is visible below the prompt.

```
MINGW64:/d/Template/position-finder-server
CORP+ev928.1ee8DD-FY928-LEE02 MINGW64 ~
$ cd /d/Template/
CORP+ev928.1ee8DD-FY928-LEE02 MINGW64 /d/Template
$ git clone git://git.tizen.org/apps/native/position-finder-server
Cloning into 'position-finder-server'...
remote: Counting objects: 515, done.
remote: Compressing objects: 100% (506/506), done.
Receiving objecremote: Total 515 (delta 354), reused 0 (delta 0)
Receiving objects: 100% (515/515), 154.11 KiB | 5.50 MiB/s, done.
Resolving deltas: 100% (354/354), done.
```

\$ cd <THE PATH>

Enters the path to download the source code.

\$ git clone <Git address of the repository>

Downloads the repository on your local PC using the *git clone* command.

ex) \$ git clone

git://git.tizen.org/apps/native/position-finder-server

## 5) How to change the branch of the project\*

```
$ git checkout <BRANCH NAME>
```

```
$ git checkout template
```

<https://git.tizen.org/>



index : apps/native/position-finder-server

Domain: Applications; Licenses: Apache-2.0

summary refs log tree commit diff

Branch  
master  
template  
zarie

Commit message

Modify comments

Modify Comments

Merge "add web notification feature, but not enabled yet" into zarie

Age

4 hours

19 hours

19 hours

2 days

7 days

7 days

7 days

10 days

2017-08-03

2017-08-01

[...]

Commit message

Modify comments HEAD master refs/changes/87/144787/1

Modify comments refs/changes/02/144702/1

CBOR file in res dir and auto-naming in spec refs/changes/91/144691/1

Modify comments about connectivity API refs/changes/04/144204/1

Add API for LED resource to check human behavior refs/changes/53/143753/1

Add comments for sensor and connectivity API refs/changes/98/143698/1

Trivial modification refs/changes/42/143642/2

Copy CBOR file from res dir to data dir for protecting crash refs/changes/89/143089/1

Remove presence API - not working, maybe it'll be deprecated refs/changes/52/142152/1

Merge "Add connectivity APIs for various data types"

Clone

<https://git.tizen.org/cgit/apps/native/position-finder-server>

<git://git.tizen.org/apps/native/position-finder-server>

## 6) Changing the branch of the Tizen Project\*

Git bash

```
MINGW64:/d/Template/position-finder-server
CORP+ey928.lee@DD-EY928-LEE02 MINGW64 ~
$ cd /d/Template/

CORP+ey928.lee@DD-EY928-LEE02 MINGW64 /d/Template
$ git clone git://git.tizen.org/apps/native/position-finder-server
Cloning into 'position-finder-server'...
remote: Counting objects: 515, done.
remote: Compressing objects: 100% (506/506), done.
Receiving objecremote: Total 515 (delta 354), reused 0 (delta 0)
Receiving objects: 100% (515/515), 154.11 KiB | 5.50 MiB/s, done.
Resolving deltas: 100% (354/354), done.

CORP+ey928.lee@DD-EY928-LEE02 MINGW64 /d/Template
$ cd position-finder-server/

CORP+ey928.lee@DD-EY928-LEE02 MINGW64 /d/Template/position-finder-server
$ git checkout template
Switched to a new branch 'template'
Branch template set up to track remote branch template from origin.

CORP+ey928.lee@DD-EY928-LEE02 MINGW64 /d/Template/position-finder-server
$ |
```

**\$ cd position-finder-server**  
Enter the project.

**\$ git checkout <BRANCH NAME>**  
Change the branch of the project using the *git checkout* command.

ex) **git checkout template**

## 1. Importing the cloned project into Tizen Studio.\*

- a. *File* > Select *Import...*
- b. *Tizen* > Select *Tizen Project* > Next
- c. *Select Directory / File* > Select *Root directory*
- d. *Location* > Browse > Select the path of the cloned project(*...wposition-finder-server*) > Next
- e. Finish

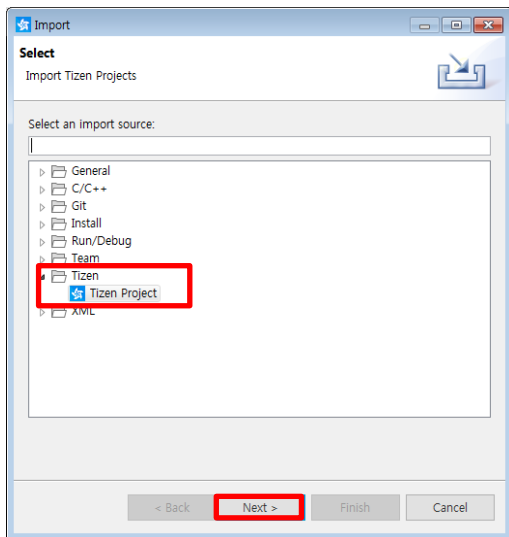
## 2. Building the project

Project > Build Project (F10)

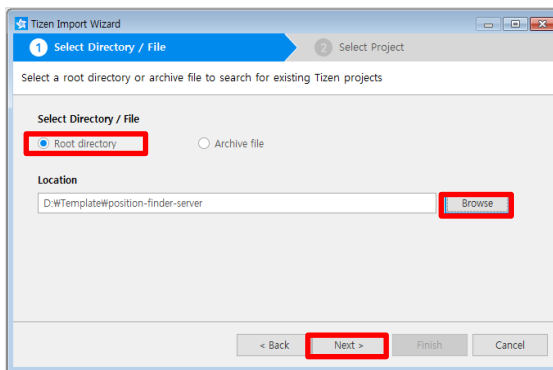
# Import the project into Tizen Studio.\*

- a. *File* > Select *Import...*
- b. *Tizen* > Select *Tizen Project* > Next\*
- c. *Select Directory / File* > Select *Root directory*\*
- d. *Location* > Browse > Select the path of the cloned project(*...Wposition-finder-server*) > Next\*
- e. Finish

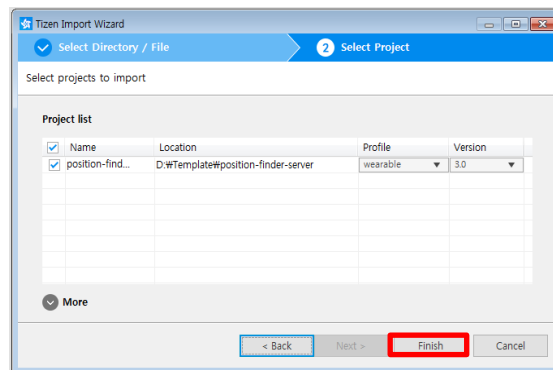
1-b.



1-c.



1-d.



## 1. Import the project into Tizen Studio.\*

- a. *File* > Select *Import...*
- b. *Tizen* > Select *Tizen Project* > Next
- c. *Select Directory / File* > Select *Root directory*
- d. *Location* > Browse > Select the path of the cloned project(*...wposition-finder-server*) > Next
- e. Finish

## 2. Build the project

Project > Build Project (F10)

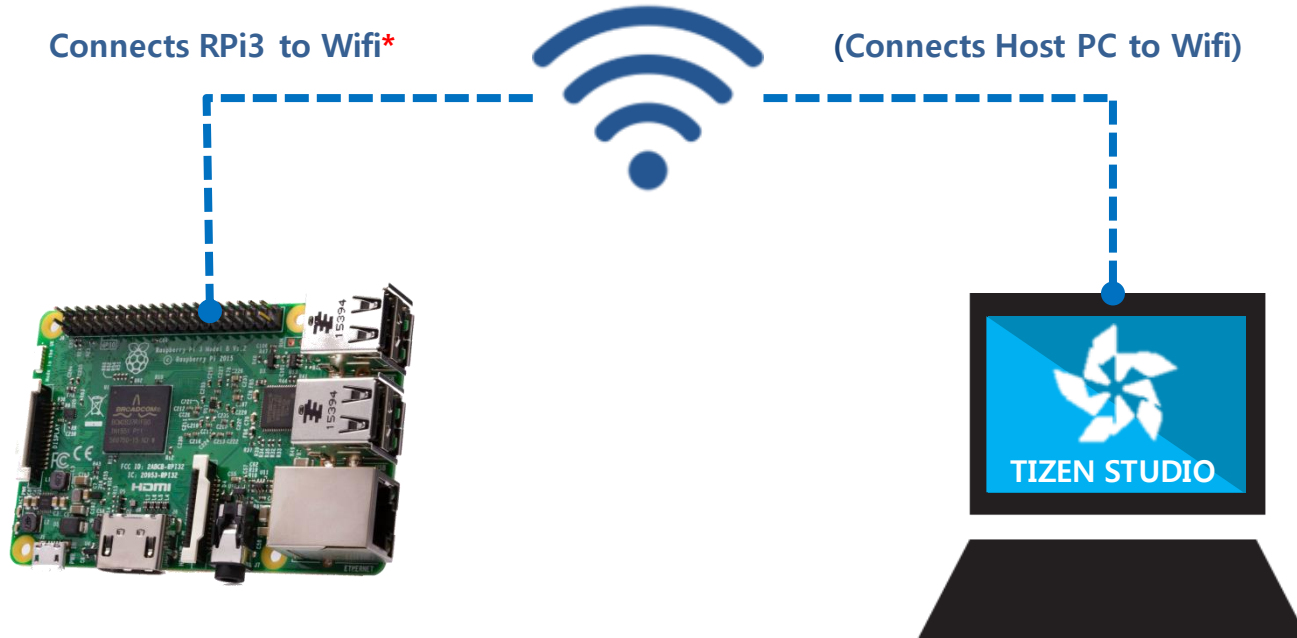
# Running the tizen project on RPi3

- Connecting RPi3 to the network
- Connecting RPi3 to Tizen Studio



# Connecting RPi3 to the network

SOSCON



## Connects RPi3 to Wifi \*

Raspberry Pi 3

1. Connects a Host PC to RPi3 via UART.
2. Opens the serial terminal of RPi3.
3. Connects Wifi with the **wifi\_test** command.

```
$ wifi_test
```

Follow the steps below:

*(Input only numbers.)*

1. Wi-Fi Init and set callbacks
3. Activate Wi-Fi device
- c. Connect (Write AP name/password)
6. Get connection state (Verify the network is connected)
0. Exit

**\* Note :** You should not misspell a AP name and password when you enter it. If you get typo, you have to run it again from the beginning.

```
Network Connection API Test App

Options..
1  - Wi-Fi init and set callbacks
2  - Wi-Fi deinit(unset callbacks automatically)
3  - Activate Wi-Fi device
4  - Deactivate Wi-Fi device
5  - Is Wi-Fi activated?
6  - Get connection state
7  - Get MAC address
8  - Get Wi-Fi interface name
9  - Scan request
a  - Get Connected AP
b  - Get AP list
c  - Connect
d  - Disconnect
e  - Connect by wps pbc
f  - Forget an AP
g  - Set & connect EAP
h  - Set IP method type
i  - Set Proxy method type
j  - Get Ap info
k  - Connect Specific AP
l  - Load configuration
m  - Save configuration
n  - Remove configuration
o  - Set configuration proxy and hidden
p  - Set EAP configuration
q  - TDLS TearDown
r  - TDLS Get Connected Peer
s  - Connect to Hidden AP
0  - Exit
ENTER - Show options menu.....
Operation succeeded!
```

## Connects RPi3 to Wifi \*

Raspberry Pi 3

### 4. Checks the IP address of the connected Wi-Fi on RPi3.

```
$ ifconfig
```

```
root@localhost:~# ifconfig
eth0: flags=--28669<UP,BROADCAST,MULTICAST,DYNAMIC> mtu 1500

    RX packets 0  bytes 0 (0.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 0  bytes 0 (0.0 B)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536

    loop txqueuelen 1  (Local Loopback)
    RX packets 0  bytes 0 (0.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 0  bytes 0 (0.0 B)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

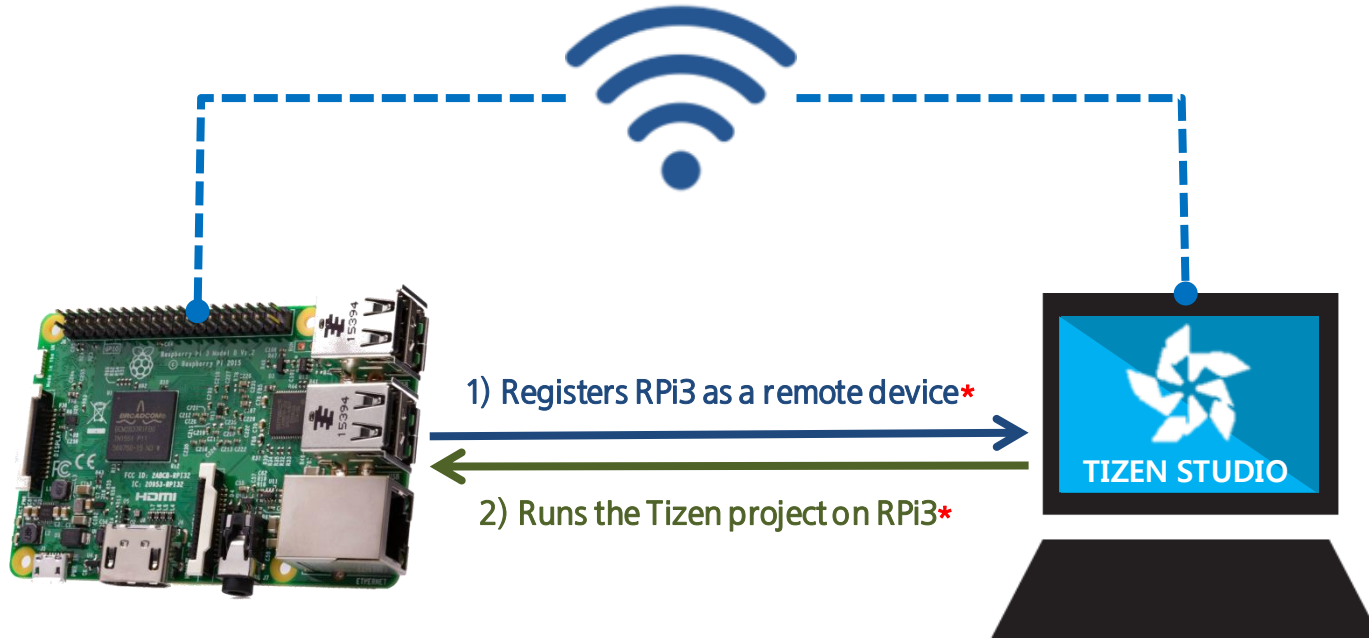
wlan0 flags=--28605<UP,BROADCAST,RUNNING,MULTICAST,DYNAMIC> mtu 1500
    inet 192.168.0.48 netmask 255.255.255.0 broadcast 192.168.0.255

    RX packets 59  bytes 3590 (3.5 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 41  bytes 5912 (5.7 KiB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@localhost:~#
```

# Connecting RPi3 to Tizen Studio

SOSCON



# 1) Registers RPi3 as a remote device\*

\* **Note** : Ipaddress of RPi3 (Refer to [Connects RPi3 to Wifi](#)\*)

Tizen Studio

```
wlan0: flags=--28605<UP,BROADCAST,RUNNING,MULTICAST,DYNAMIC> mtu 1500  
inet 192.168.0.48 netmask 255.255.255.0 broadcast 192.168.0.255
```

## 1. Adds RPi3 as a remote device.

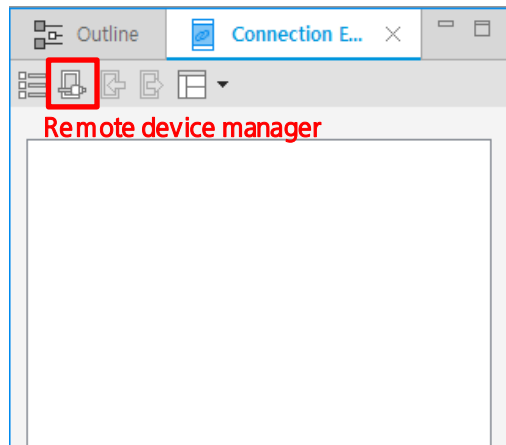
- Connection Explorer* > *Remote Device Manager*
- Scan** (scanning remote devices)

## 2. Sets the connection state to ON.

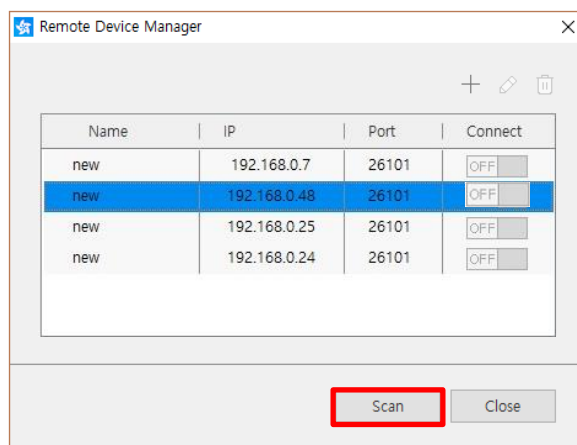
If the **connection fails**,

- Check if Wifi is disconnected or not.
- Your RPi3 and PC are on the same local network.

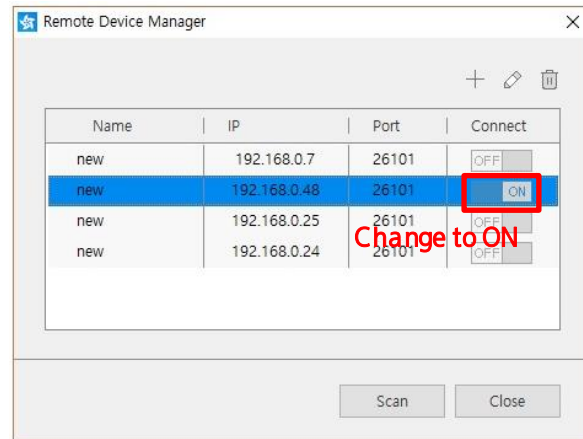
1-a.



1-b.



2.

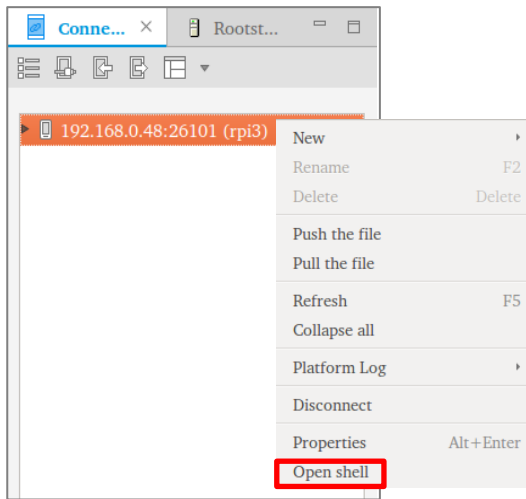


## 2) Runs the Tizen project on RPi3\*

### 1. Running the project on RPi3.

Run > Run As > ...

### 2. How to open the shell.



```
sh-3.2$ su
Password:
bash-3.2#
```

```
$ su
Password : tizen
```

\* **NOTE:** The password is not shown when you enter it.

### 3. How to view logs.

Execute *dlogutil* in the opened shell.

```
$ dlogutil <LOG_TAG> ex) dlogutil TT
```

# THANK YOU

**SOSCON**  
SAMSUNG  
OPEN SOURCE  
CONFERENCE

