

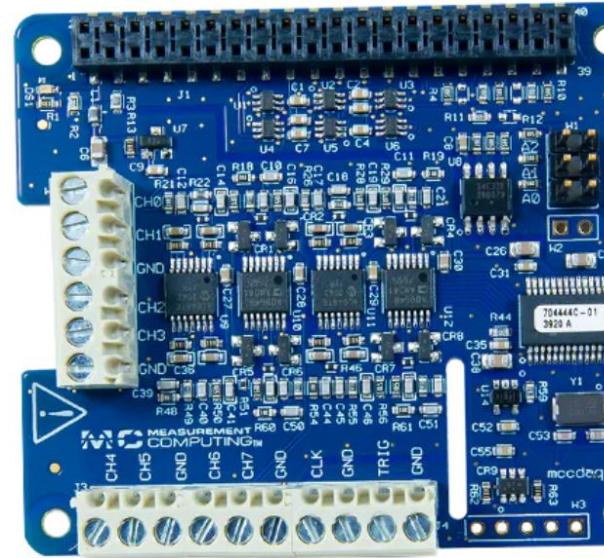
# VOLTAGE MONITOR SYSTEM

BY NG WAY KIAT



Raspberry Pi 5  
(Single Board Computer)

Smart MCC 118/128 Voltage DAQ HAT for Raspberry Pi



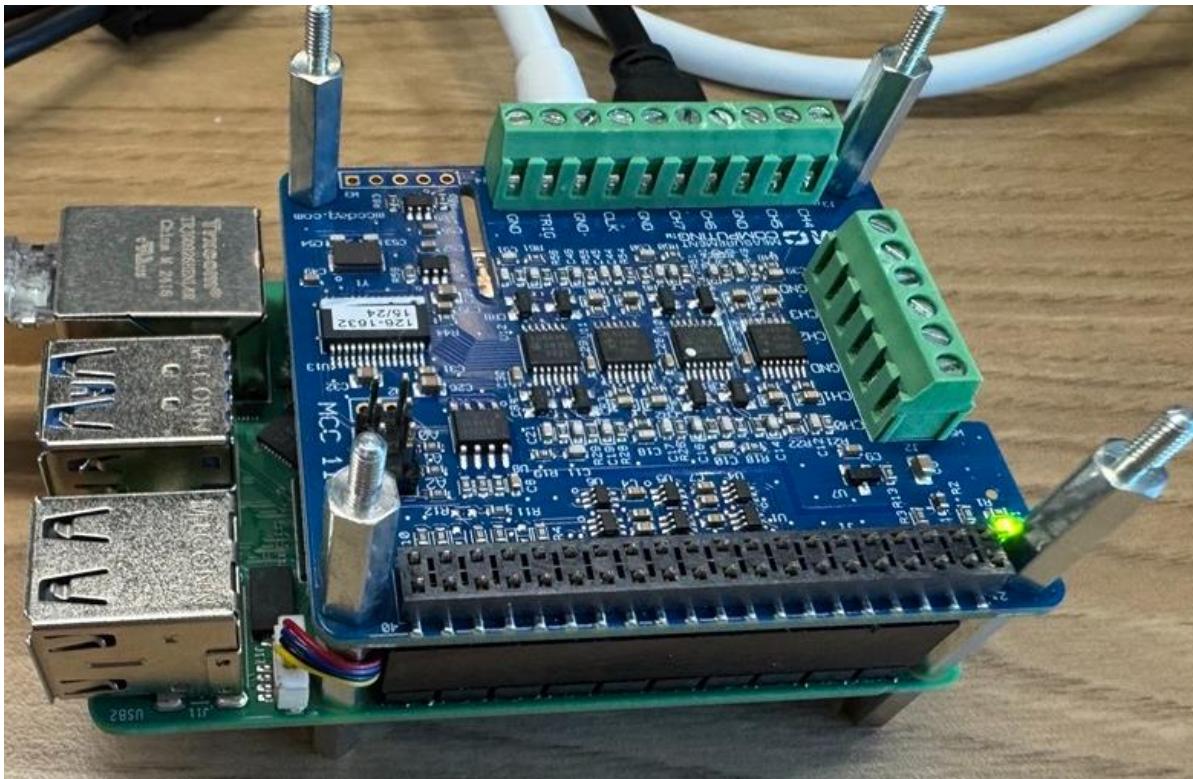
MCC118 Hat  
(ADC)

## ITEMS REQUIREMENTS

# OBJECTIVE

- To monitor voltage from the Voltage Amplifier based on current architecture
- Receive and Reply the data to the server based on current architecture

# HARDWARE DISPLAY



# WHY RASPBERRY PI 5?

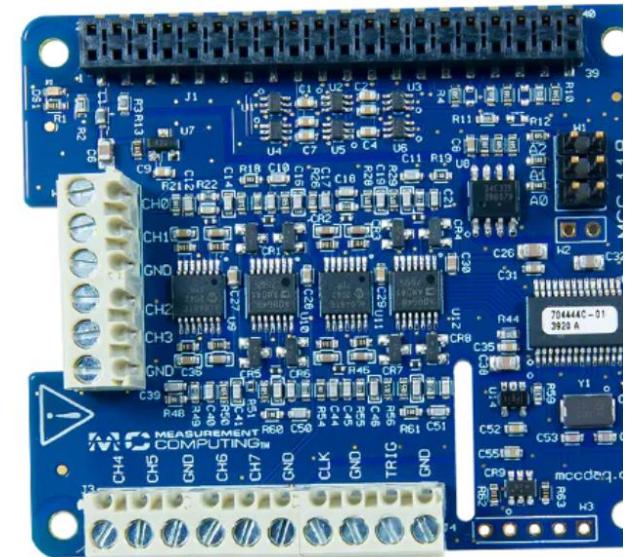
- Microprocessor (act as a full computer)
- Multitasking (able to run and handle multiple program at the same time)
- Costless (function like full computer but it is cheaper)
- Multi Programming Language (Python, C, C++)
- High Computation Performance (Able handle complex computation in a program)
- Have 40 GPIO pins (Digital pins)
- End of Life Period (January 2036)



# WHY MCC118 HATS?

- ADC convertor to the main board (Raspberry Pi 5 is digital pins)
- Using SPI protocol to communicate to the main board (Raspberry Pi 5)
- Programming able through MCC DAQHATS library (Python, C, C++)
- Have 8 available channel to use to monitor the voltage (ADC convertor)
- The floating voltage will be read once the channel is disconnected

MCC 118/128 Voltage DAQ HAT for Raspberry Pi



# THE DIFFERENCES?

No	Items	Existing Device (By Vendor)	New Device (RaspberryPi5)
1.	Monitor voltage (0V ~ 10V)	Y	Y
2.	Socket Communication (Send & Reply from Server)	Y	Y
3.	Capture the device log	N	Y
4.	File Sharing Network (user able read the file through sharing network)	N	Y
5.	Framework consistency (use across another project)	N	Y
6.	SSH access	N	Y
7.	VNC access	N	Y

## ACCESSORIES (RASPBERRY PI 5)

- USB-C cable with power adapter
- Micro HDMI cable
- Keyboard and Mouse
- MicroSD card reader

# COMMON PROBLEM & SOLUTION (STARTED)

Problem Area	Specific Problem	Solution
Operating System (OS)	Require Operating System Installation	Install Raspberry Pi Imager to install the Linux OS into SD Card from Window PC. <a href="#"><u>(Solution: S1)</u></a>
OS Time	The office network is not allowed to access to outside network. It will cause the time would not synchronize.	Setup the ntp connection and link to BIOSG domain server. <a href="#"><u>(Solution: S2)</u></a>
Interface Configuration	By default, all the interface setting is disable (SSH, VNC, I2C, SPI and so on)	Need manually turn on those interface setting through raspberry configuration. <a href="#"><u>(Solution: S3)</u></a>
IDE (Integrated Development Environment)	Require the specific IDE tool to do programing like VS CODE (Visual Studio Code). VS CODE is free to use for commercial purpose.	Need to install the VS CODE through Linux command application install package. <a href="#"><u>(Solution: S4)</u></a>
OS common update	Installation OS it is does not updated package. It require manual do the upgrade by user.	Manual update action through Linux command. <a href="#"><u>(Solution: S5)</u></a>

# C++ ENVIRONMENT

Library	Use For?	Linux Command
GPIO	Raspberry Pi5 GPIO protocol	<code>sudo apt install libgpiod-dev</code>
MCC DAQHATS	MCC DAQHATS function	<code>sudo apt install git -y</code> <code>cd ~</code> <code>git clone <a href="https://github.com/mccdaq/daqhats.git">https://github.com/mccdaq/daqhats.git</a></code> <code>cd ~/daqhats</code> <code>sudo ./install.sh</code>

# PYTHON ENVIRONMENT

Item	Purpose	Action
Python Environment	To allow python execute in the system	<code>sudo apt install python3 idle3</code> <code>sudo apt install python3-pip</code> <code>sudo apt-get install python3-spidev</code> <code>sudo apt-get install python3-gpiozero</code> <code>sudo apt install python3-flask</code>
Python Web	To allow python web interface	<code>sudo apt install apache2 -y</code>
Python MCC DAQHATS	To use MCC DAQHATS function	<code>cd /</code> <code>sudo python3 -m venv "pyvenv"</code> <code>sudo chmod 777 "pyvenv"</code> <code>sudo /pyvenv/bin/pip install daqhats</code> <code>sudo /pyvenv/bin/pip install dash</code> <code>cd ~/daqhats</code> <code>sudo /pyvenv/bin/python setup.py install</code>

# RESULT

This is device log result during start up.

```
2025-11-10 11:48:59 [INFO] =====
2025-11-10 11:48:59 [INFO] The MCC DAQ HATS Program Started.
2025-11-10 11:48:59 [INFO] The Program Version [1.1.0] MODIFY_DATE [06-NOV-2025]
2025-11-10 11:48:59 [INFO] The remote client connection mapping list as below: Total[2]
2025-11-10 11:48:59 [INFO] 1. Name [MCC_DAQ_HOST1] IP [10.60.144.79] Port [5000] Status [Disconnected]
2025-11-10 11:48:59 [INFO] 2. Name [MCC_DAQ_HOST2] IP [10.60.144.79] Port [6000] Status [Disconnected]
2025-11-10 11:48:59 [INFO] The global variable mapping list as below: Total[3]
2025-11-10 11:48:59 [INFO] 1. Name [HOST_PORT] Value [9999] Description [This is Host Port Number.]
2025-11-10 11:48:59 [INFO] 2. Name [LOG_DAY] Value [7] Description [This is Truncate Log Days.]
2025-11-10 11:48:59 [INFO] 3. Name [MCC118_CHANNEL_FLAG] Value [1] Description [This is enable for the [MCC118_CHANNEL_FLAG] config control.]
2025-11-10 11:48:59 [INFO] The Global Variable [HOST_PORT] : giHostPort=9999
2025-11-10 11:48:59 [INFO] The Global Variable [LOG_DAY] : giLogDays=7
2025-11-10 11:48:59 [INFO] The Global Variable [MCC118_CHANNEL_FLAG] : gbMCC118ConfigFlag=Enable
2025-11-10 11:48:59 [INFO] Found the MCC DAQ HATS in the system.
2025-11-10 11:48:59 [INFO] Start the Truncate Function....
2025-11-10 11:48:59 [INFO] No Log File need to truncate...
2025-11-10 11:48:59 [INFO] MCC HATS Info: Id[322], Address[0], ProductName[MCC 118 Voltage Input HAT], Version[1]
2025-11-10 11:48:59 [INFO] Selected MCC 118 device at address[0]
2025-11-10 11:48:59 [INFO] Success to open a connection to the MCC 118 device at address[0]
2025-11-10 11:48:59 [INFO] The MCC118 Device Information as below:
2025-11-10 11:48:59 [INFO] No. of Channel = 8
2025-11-10 11:48:59 [INFO] Requested scan rate: 1000.00
2025-11-10 11:48:59 [INFO] Actual scan rate: 1000.00
2025-11-10 11:48:59 [INFO] MIN ADC Code = 1369497312
2025-11-10 11:48:59 [INFO] MAX ADC Code = 1369497312
2025-11-10 11:48:59 [INFO] MIN Voltage = 1369497312
2025-11-10 11:48:59 [INFO] MAX Voltage = 1369497312
2025-11-10 11:48:59 [INFO] MIN Range = 1369497312
2025-11-10 11:48:59 [INFO] MAX Range = 1369497312
2025-11-10 11:48:59 [INFO] The initiate value for the MCC118 Structur List as below:
2025-11-10 11:48:59 [INFO] CHN[0], Name:Channel 0, Description:MCC118 DAQ HAT - Channel 0, Status:Enable, Value:0.000000, OriSlope:1.043737, OriOffset:-83.301702
2025-11-10 11:48:59 [INFO] CHN[1], Name:Channel 1, Description:MCC118 DAQ HAT - Channel 1, Status:Enable, Value:0.000000, OriSlope:1.046837, OriOffset:-91.554857
2025-11-10 11:48:59 [INFO] CHN[2], Name:Channel 2, Description:MCC118 DAQ HAT - Channel 2, Status:Enable, Value:0.000000, OriSlope:1.048416, OriOffset:-96.492832
2025-11-10 11:48:59 [INFO] CHN[3], Name:Channel 3, Description:MCC118 DAQ HAT - Channel 3, Status:Enable, Value:0.000000, OriSlope:1.049981, OriOffset:-100.262347
2025-11-10 11:48:59 [INFO] CHN[4], Name:Channel 4, Description:MCC118 DAQ HAT - Channel 4, Status:Enable, Value:0.000000, OriSlope:1.047680, OriOffset:-96.074281
2025-11-10 11:48:59 [INFO] CHN[5], Name:Channel 5, Description:MCC118 DAQ HAT - Channel 5, Status:Enable, Value:0.000000, OriSlope:1.047660, OriOffset:-89.665328
2025-11-10 11:48:59 [INFO] CHN[6], Name:Channel 6, Description:MCC118 DAQ HAT - Channel 6, Status:Enable, Value:0.000000, OriSlope:1.047343, OriOffset:-93.989806
2025-11-10 11:48:59 [INFO] CHN[7], Name:Channel 7, Description:MCC118 DAQ HAT - Channel 7, Status:Enable, Value:0.000000, OriSlope:1.044827, OriOffset:-83.864409
2025-11-10 11:48:59 [INFO] Voltage Result: CH[0] =1.74810 V CH[1] =1.74901 V CH[2] =1.74842 V CH[3] =1.74825 V CH[4] =1.75212 V CH[5] =1.74737 V CH[6] =1.74814 V CH[7] =1.74785 V
2025-11-10 11:48:59 [INFO] Start server socket function. SERVER_PORT [9999]
2025-11-10 11:49:04 [INFO] Voltage Result: CH[0] =1.74810 V CH[1] =1.74901 V CH[2] =1.74842 V CH[3] =1.74825 V CH[4] =1.75212 V CH[5] =1.74737 V CH[6] =1.74814 V CH[7] =1.74785 V
2025-11-10 11:49:09 [INFO] Voltage Result: CH[0] =1.74810 V CH[1] =1.74390 V CH[2] =1.74842 V CH[3] =1.74825 V CH[4] =1.75212 V CH[5] =1.74737 V CH[6] =1.74814 V CH[7] =1.74785 V
2025-11-10 11:49:14 [INFO] Voltage Result: CH[0] =1.74810 V CH[1] =1.74901 V CH[2] =1.74842 V CH[3] =1.74825 V CH[4] =1.75212 V CH[5] =1.74225 V CH[6] =1.74814 V CH[7] =1.74785 V
```

# RESULT CONT'S

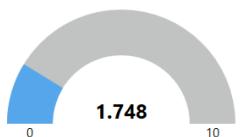
This is device log result during client connection interrupt and receive & reply to the client side.

```
2025-11-10 14:24:39 [INFO] Server Socket Client Connected. IP [10.60.241.105] PORT [51381]
2025-11-10 14:24:39 [INFO] Server Socket Data Receive from Clint_IP[10.60.241.105], Client_Port[51381]. RecData: A8888!:|Check all channel
2025-11-10 14:24:39 [INFO] SendData: [0=1.748;1=1.749;2=1.748;3=1.753;4=1.752;5=1.747;6=1.748;7=1.743;]
2025-11-10 14:24:39 [WARNING] Client Disconnected. Client_IP[10.60.241.105], Client_Port[51381]
2025-11-10 14:24:42 [INFO] Server Socket Client Connected. IP [10.60.241.105] PORT [64282]
2025-11-10 14:24:42 [INFO] Server Socket Data Receive from Clint_IP[10.60.241.105], Client_Port[64282]. RecData: A8888!:|Check all channel
2025-11-10 14:24:42 [INFO] SendData: [0=1.748;1=1.749;2=1.748;3=1.748;4=1.752;5=1.742;6=1.748;7=1.748;]
2025-11-10 14:24:42 [WARNING] Client Disconnected. Client_IP[10.60.241.105], Client_Port[64282]
```

IP Address: 10.60.144.84 Port: 9999

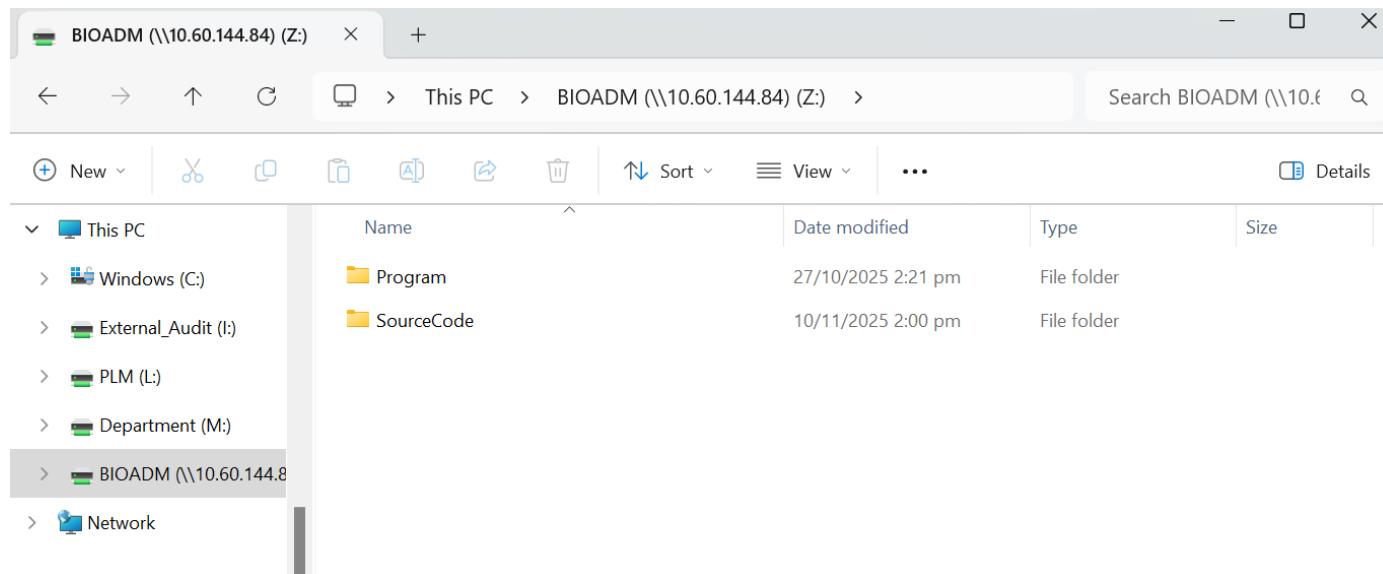
Channel 0:	1.748
Channel 1:	1.744
Channel 2:	1.748
Channel 3:	1.748

Channel 4:	1.747
Channel 5:	1.742
Channel 6:	1.748
Channel 7:	1.743



# RESULT CONT'S

This is SAMBA file sharing.



## SAMBA Directory Access List:

//10.60.144.84/BIO	piusr:piusr
//10.60.144.84/BIOADM	piadm:piadm

# RESULT CONT'S

The List as below:

- rc-local.service
- pyweb.service
- nmbd.service
- smbd.service

```
pi@raspberrypi:/ $ systemctl list-units --type=service --state=running
 _UNIT           LOAD ACTIVE SUB   DESCRIPTION
accounts-daemon.service loaded active running Accounts Service
apache2.service      loaded active running The Apache HTTP Server
avahi-daemon.service loaded active running Avahi mDNS/DNS-SD Stack
bluetooth.service    loaded active running Bluetooth service
cron.service         loaded active running Regular background program processing daemon
cups-browsed.service loaded active running Make remote CUPS printers available locally
cups.service          loaded active running CUPS Scheduler
dbus.service          loaded active running D-Bus System Message Bus
getty@tty1.service    loaded active running Getty on tty1
lightdm.service       loaded active running Light Display Manager
ModemManager.service loaded active running Modem Manager
NetworkManager.service loaded active running Network Manager
nmbd.service          loaded active running Samba NMB Daemon
polkit.service        loaded active running Authorization Manager
pyweb.service         loaded active running my server
rc-local.service      loaded active running /etc/rc.local Compatibility
rtkit-daemon.service loaded active running RealtimeKit Scheduling Policy Service
serial-getty@ttyAMA0.service loaded active running Serial Getty on ttyAMA0
smbd.service          loaded active running Samba SMB Daemon
ssh.service            loaded active running OpenBSD Secure Shell server
systemd-journald.service loaded active running Journal Service
systemd-logind.service loaded active running User Login Management
systemd-timesyncd.service loaded active running Network Time Synchronization
systemd-udevd.service loaded active running Rule-based Manager for Device Events and Files
triggerhappy.service  loaded active running triggerhappy global hotkey daemon
udisks2.service        loaded active running Disk Manager
user@1000.service     loaded active running User Manager for UID 1000
wayvnc-control.service loaded active running VNC Control Service
wayvnc.service         loaded active running VNC Server
wpa_supplicant.service loaded active running WPA supplicant

LOAD  = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB   = The low-level unit activation state, values depend on unit type.
30 loaded units listed.
```



TECHNICAL SKILL

Operating System & Software Technical Skill Sample

# AUTORUN C++ PROGRAM

## 1. Make program full authorize

```
sudo chmod 777 /BIO/Program/MCC_DAQ_HATS/MCC_DAQHATS  
sudo chmod +x /BIO/Program/MCC_DAQ_HATS/MCC_DAQHATS
```

## 2. Create the "rc.local" file.

The rc.local file in Linux is a script that traditionally allowed system administrators to execute custom commands or scripts during the system startup process, specifically after all other standard init scripts had completed and the system had reached a multi-user runlevel.

```
sudo nano /etc/rc.local  
#!/bin/bash  
/BIO/Program/MCC_DAQ_HATS/MCC_DAQHATS &  
exit 0
```

## 3. Make "rc.local" full authorize

```
sudo chmod 777 /etc/rc.local  
sudo chmod +x /etc/rc.local
```

## 4. Create the "rc-local.service" full authorize.

The rc-local.service file in Linux is a systemd unit that provides compatibility for the traditional /etc/rc.local script.

```
sudo nano /etc/systemd/system/rc-local.service  
[Unit]  
Description=/etc/rc.local Compatibility  
ConditionPathExists=/etc/rc.local  
After=network-online.target  
  
[Service]  
Type=forking  
ExecStart=/etc/rc.local start  
TimeoutSec=0  
StandardOutput=tty  
RemainAfterExit=yes  
SysVStartPriority=99  
  
[Install]  
WantedBy=multi-user.target
```

## 5. To enable "rc-local.service" autorun in Linux

```
sudo systemctl enable rc-local.service
```

## 6. To check "rc-local.service" status

```
sudo systemctl status rc-local.service
```

## 7. Manual start the service

```
sudo systemctl start rc-local.service
```

## 8. Manual stop the service

```
sudo systemctl stop rc-local.service
```

# AUTORUN PYTHON WEB

## For Python Program

```
sudo nano /etc/systemd/system/pyweb.service
[Unit]
Description=my server
After=network-online.target

[Service]
ExecStart=/pyvenv/bin/python
/home/pi/daqhats/examples/python/mcc118/web_server/web_server.py

[Install]
WantedBy=multi-user.target
```

# SAMBA

## Install SAMBA

```
sudo apt-get install samba
```

## Create SAMBA user and group

```
sudo adduser piusr  
sudo adduser piadm  
sudo groupadd samba  
sudo groupadd sambaadm  
sudo usermod -a -G samba piusr  
sudo usermod -a -G sambaadm piadm  
sudo smbpasswd -a piusr  
sudo smbpasswd -a piadm
```

## Modify the SAMBA config

```
sudo nano /etc/samba/smb.conf  
[BIO]  
comment = BIO MCC_DAO_HATS Folder Share  
path = /BIO  
browseable = yes  
writable = no  
read only = yes  
guest_ok = yes  
valid users = @samba  
force group = samba
```

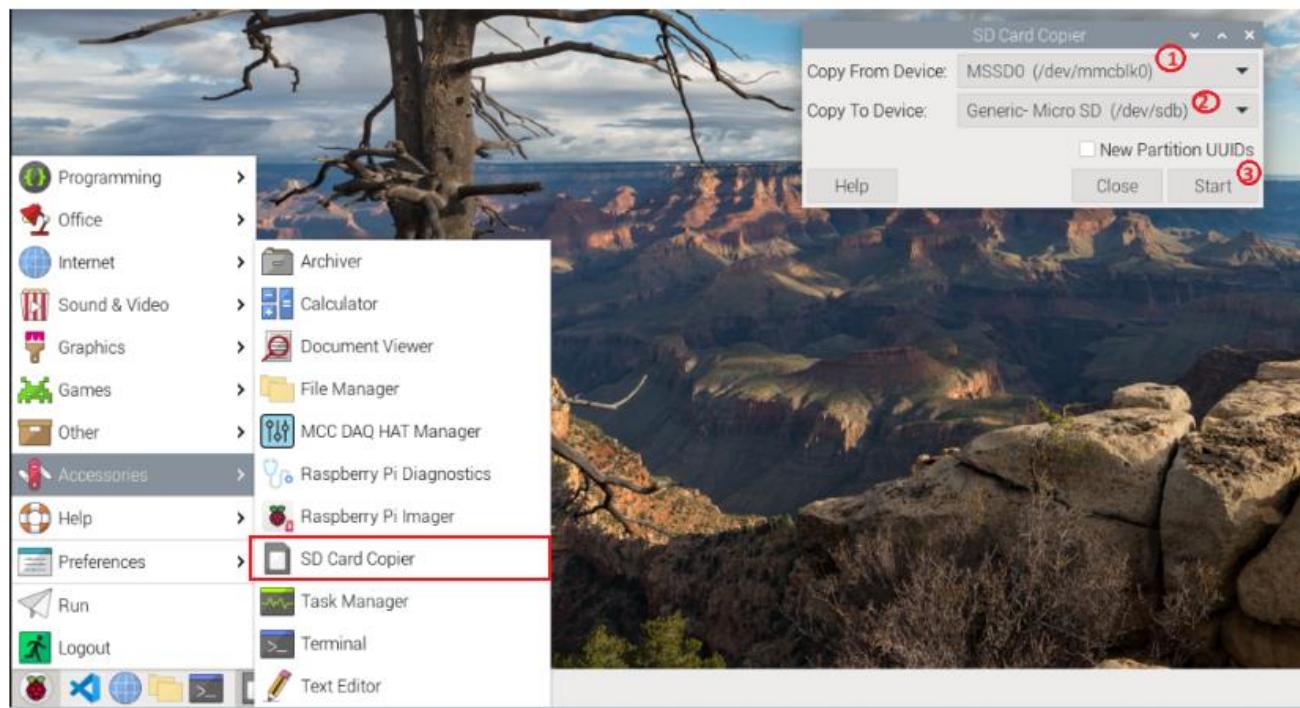
```
[BIOADM]  
comment = BIO Folder Share as full permission  
path = /BIO  
browseable = yes  
writable = yes  
guest_ok = yes  
valid users = @sambaadm  
force group = sambaadm
```

```
sudo chown piusr:piusr /BIO  
sudo chown piadm:piadm /BIO  
sudo setfacl -R -m g:samba:rwx /BIO  
sudo setfacl -R -m g:sambaadm:rwx /BIO
```

```
sudo systemctl enable smbd nmbd  
sudo systemctl restart smbd nmbd
```

# DUPLICATE THE DEVICE TO ANOTHER DEVICE

- From "Accessories" >> "SD Card Copier".
  - o This to chrome the SD Card to another SD Card.
- In SD Card Copier.
  - o Copy From Device "MSSD0"
  - o Copy To Device "Generic-Micro SD"
  - o Click "Start" to execute the action.



# SOLUTION: SI (OPERATING SYSTEM (OS))

- Install Raspberry Pi Image (<https://www.raspberrypi.com/software/>)

## Raspberry Pi Imager

Raspberry Pi Imager is the quick and easy way to install Raspberry Pi OS and other operating systems to a microSD card, ready to use with your Raspberry Pi.

Download and install Raspberry Pi Imager on a computer with an SD card reader. Insert the microSD card you'll use with your Raspberry Pi into the reader and run Raspberry Pi Imager.

[Download for Windows](#)

[Download for macOS](#)

[Download for Debian or Ubuntu \(x86\\_64\)](#)

To install on Raspberry Pi OS, type  
sudo apt install rpi-imager  
into a terminal window



- Install Raspberry Pi OS (<https://www.raspberrypi.com/software/operating-systems/>)

## Raspberry Pi OS (Legacy) Full

A port of Debian Bookworm with security updates, desktop environment and recommended applications

**Release date** 1 Oct 2025

**System** 64-bit

**Kernel version** 6.12

**Debian version** 12 (bookworm)

**Size** 3,145 MB

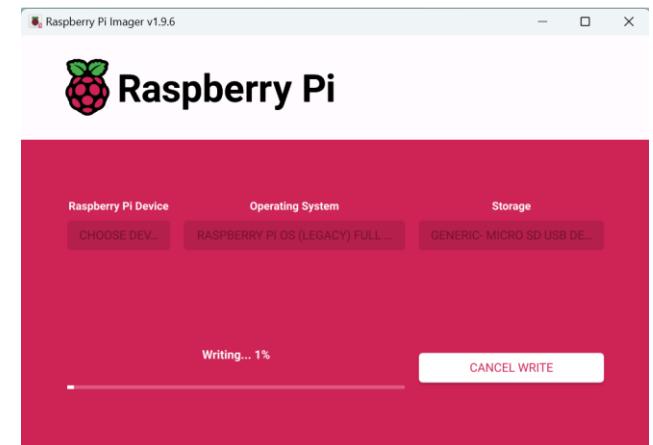
## Download

[Download torrent](#)

[View archive](#)

[View release notes](#)

► [SHA256 file integrity hash](#)



# SOLUTION: S2 (OSTIME)

- **The NTP or local domain server in BIO SG**

- Name: dc-apm65-1.apm.biotronik.int
  - Address: 10.245.0.31

- **Step to Set Up:**

- sudo nano /etc/systemd/timesyncd.conf

```
[Time]
Servers=10.245.0.31
sudo systemctl daemon-reload
sudo timedatectl set-ntp off
sudo timedatectl set-ntp on
sudo timedatectl status
```

- **Result:**

```
pi@raspberrypi:/ $ sudo timedatectl status
          Local time: Tue 2025-11-04 10:23:15 +08
          Universal time: Tue 2025-11-04 02:23:15 UTC
                    RTC time: Tue 2025-11-04 02:23:15
                   Time zone: Asia/Singapore (+08, +0800)
System clock synchronized: yes
          NTP service: active
    RTC in local TZ: no
pi@raspberrypi:/ $
```



# SOLUTION: S3 (INTERFACE CONFIGURATION)

Go To "Preferences" >> "Raspberry Pi Configuration"  
o "Interfaces" >> Turn On All Interfaces.



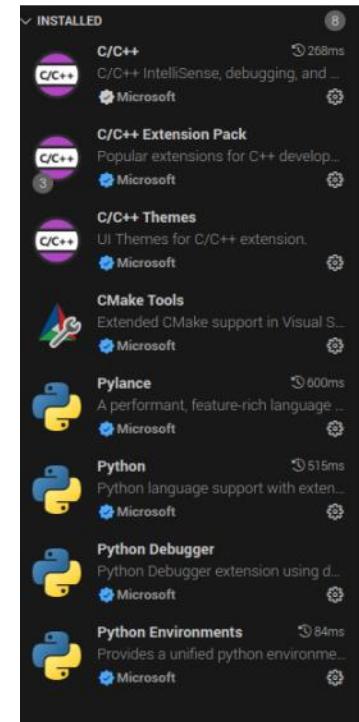
o "Localisation" >> Locale, Timezone, Wireless LAN Country. (SG,Singapore)

# SOLUTION: S4 (IDE)

## Linux Command Installation

- Install Development IDE
  - `sudo apt install code`
  - `sudo apt update`

## VSCODE Extension Package



# SOLUTION: S5 (OS COMMON UPDATE)

- For General Update Require:
  - sudo apt-get upgrade
  - sudo apt update
  - sudo apt full-upgrade
  - sudo reboot



# SUMMARY

- SAMBA folder:
  - For User (Read Only Access):
    - //Ip address/BIO (piusr:piusr)
  - For Admin (Full Access):
    - //Ip address/BIOADM (piadm:piadm)
- Autorun service List:
  - rc-local (C++ program backend autorun)
  - pyweb (python program backend autorun)
  - nmbd (samba service)
  - smbd (samba service)
- Results:
  - Host socket connection (wait & receive client connection)
  - Logging (device log, communication log, data send & receive log, error log)
  - SAMBA file sharing (Allow window OS to access the folder)
  - Auto run service after reboot (C++ program, python web, samba service)



THANK YOU