EPSON

Setup Manual for
Vibration Measurement System
Using
Raspberry Pi Products

Notice of Document

Evaluation board/kit and development tool important notice

1. This evaluation board/kit or development tool is designed for use for engineering evaluation, demonstration, or development purposes only. Do not use it for other purposes. It is not intended to meet the requirements of design for finished products.
2. This evaluation board/kit or development tool is intended for use by an electronic engineer and is not a consumer product. The user should use it properly and in a safe manner. Seiko Epson does not assume any responsibility or liability of any kind of damage and/or fire caused by the use of it. The user should cease to use it when any abnormal issue occurs even during proper and safe use.
3. The part used for this evaluation board/kit or development tool may be changed without any notice.

NOTICE : PLEASE READ CAREFULLY BELOW BEFORE THE USE OF THIS DOCUMENT

The content of this document is subject to change without notice.

1. This document may not be copied, reproduced, or used for any other purposes, in whole or in part, without the consent of Seiko Epson Corporation("Epson").
2. Before purchasing or using Epson products, please contact with our sales representative for the latest information and be always sure to check the latest information published on Epson's official web sites and sources.
3. Information provided in this document such as application circuits, programs, usage, etc., are for reference purpose only. Please use the application circuits, programs, usage, etc. in the design of your equipment or systems at your own responsibility. Epson makes no guarantees against any infringements or damages to any third parties' intellectual property rights or any other rights resulting from the information. This document does not grant you any licenses, intellectual property rights or any other rights with respect to Epson products owned by Epson or any third parties.
4. Epson is committed to constantly improving quality and reliability, but semiconductor products in general are subject to malfunction and failure. In using Epson products, you shall be responsible for safe design in your products; your hardware, software and systems are designed enough to prevent any harm or damages to life, health or property even if any malfunction or failure might be caused by Epson products. In designing of your products with using Epson products, please be sure to check and comply with the latest information regarding Epson products (this document, specifications, data sheets, manuals, Epson's web site, etc.). When using the information included in the above materials such as product data, chart, technical contents, programs, algorithms and application circuit examples, you shall evaluate your products both in stand-alone basis and within your overall systems. You shall be solely responsible for deciding whether or not to adopt and use Epson products.
5. Epson has prepared this document and programs provided in this document carefully to be accurate and dependable, but Epson does not guarantee that the information and the programs are always accurate and complete. Epson assumes no responsibility for any damages which you incurred by due to misinformation in this document and the programs.
6. No dismantling, analysis, reverse engineering, modification, alteration, adaptation, reproduction, etc., of Epson products is allowed.
7. Epson products have been designed, developed and manufactured to be used in general electronic applications (office equipment, communications equipment, measuring instruments, home electronics, etc.) and applications individually listed in this document ("General Purpose"). Epson products are NOT intended for any use beyond the General Purpose that requires particular/higher quality or reliability in order to refrain from causing any malfunction or failure leading to harm to life, health or serious property damage or severe impact on society, including, but not limited to listed below. Therefore, you are advised to use Epson products only for the General Purpose. Should you desire to buy and use Epson products for the particular purpose other than the General Purpose, Epson makes no warranty and disclaims with respect to Epson products, whether express or implied, including without limitation any implied warranty of merchantability or fitness for any particular purpose.
   [Particular purpose]
   Space equipment (artificial satellites, rockets, etc.)
   Transportation vehicles and their control equipment (automobiles, aircraft, trains, ships, etc.)
   Medical equipment (other than applications individually listed in this document) / Relay equipment to be placed on sea floor
   Power station control equipment / Disaster or crime prevention equipment / Traffic control equipment / Financial equipment
   Other applications requiring similar levels of reliability as the above
8. Epson products listed in this document and our associated technologies shall not be used in any equipment or systems that laws and regulations in Japan or any other countries prohibit to manufacture, use or sell. Furthermore, Epson products and our associated technologies shall not be used for developing military weapons of mass destruction, military purpose use, or any other military applications. If exporting Epson products or our associated technologies, you shall comply with the Foreign Exchange and Foreign Trade Control Act in Japan, Export Administration Regulations in the U.S.A (EAR) and other export-related laws and regulations in Japan and any other countries and follow the required procedures as provided by the relevant laws and regulations.
9. Epson assumes no responsibility for any damages (whether direct or indirect) caused by or in relation with your non-compliance with the terms and conditions in this document.
10. Epson assumes no responsibility for any damages (whether direct or indirect) incurred by any third party that you assign, transfer, loan, etc., Epson products.
11. For more details or other concerns about this document, please contact our sales representative.
12. Company names and product names listed in this document are trademarks or registered trademarks of their respective companies.

2022.08

Trademark

- Raspberry Pi is a trademark of Raspberry Pi Foundation.
- Microsoft and Windows are trademarks of the Microsoft group of companies.
- EPSON is a registered trademark of Seiko Epson Corporation.
- Other product names are trademarks or registered trademarks of the respective companies.

Table of contents

Revision History

Rev. No.	Rev. Date	Page	Rev. Contents
20240315	2024/3/15	ALL	First edition
20240927	2024/9/27	---	Revisions Corresponding to the Release of MSG002-001a_v1.1.0
		ALL	• Minor corrections to the description, addition of notes, and updates/additions to illustrations.
		8, 21, 27	• The default storage folder for measurement data is now set to use the internal SD card, with the option to use an external USB memory.
		28	• Added setup procedures for the Raspberry Pi Zero 2 W.
20250331	2025/3/31	---	• Updated for MSG002-001a_v1.2.0 release
		6	• Added related documents
		7, 10, 11, 12, 20	• Added support for Raspberry Pi 5; changed OS to Bookworm; added Pi 5-specific notes
		12, 18	• Updated Raspberry Pi Imager screenshots to English version; updated Windows screenshots to Windows 11 English version
		14, 17	• Added OS customization via Raspberry Pi Imager
		18, 27	• Updated network setup instructions due to OS change
		27	• Added MQTT broker setup instructions due to new message transmission feature
		29	• Added PoE power supply instructions
20250731	2025/7/31	---	• Updated for MSG002-001a_v1.3.0 release
		6, 7, 11, 25	• Added support for M-A370AD
		7	• Added system overview diagram
		10	• Added note recommending RTC usage
		12	• Updated OS version for SD card installation
		17, 18, 22, 23, 27	• Simplified manual by assuming standard usernames and IP addresses
		24	• Added service configuration to delay startup until NTP time sync completes (when RTC not used)
		28	• Updated service file adjustment method for USB memory configuration
		ALL	• Added missing steps and supplemental notes on execution results

1. Related Documents

- “Operation Manual for Vibration Measurement System Using Raspberry Pi Products” Rev.20250731
- “Monitoring Application for Vibration Measurement System – User’s Guide” Rev.20250731
- “M-A370AD Datasheet”
- “M-A352AD Datasheet”
- “M-A342VD Datasheet”
- “M-A552AR Datasheet”
- “M-A542VR Datasheet”

2. Introduction

This manual explains the setup of a system for recording measurement data from Seiko Epson's accelerometers M-A370AD, M-A352AD, M-A552AR, and vibration sensors M-A342VD, M-A542VR on a Raspberry Pi.

The Raspberry Pi models covered are Raspberry Pi 4B and Raspberry Pi 5. The OS used is Raspberry Pi OS based on Debian 12 (bookworm), and Python version 3.11 is used.

The connection methods differ between the USB-connected M-A370AD, M-A352AD, M-A342VD and the RS422-connected M-A552AR, M-A542VR, so the hardware setup is explained for each. However, the software running on the Raspberry Pi is common to all.

Remote PC connection and internet connection can be made via wired LAN or wireless LAN. This manual assumes the following configurations:

- **Wired LAN**: Connect Raspberry Pi and PC directly with a LAN cable and configure via remote access from the PC.
- **Wireless LAN**: Connect Raspberry Pi to a broadband Wi-Fi router for internet access.

In the Appendix, optional setup configurations are explained:

- Setup of MQTT broker for system status messages
- Remote connection via broadband Wi-Fi router
- Saving measurement data to external USB memory
- Power supply via PoE
- Measurement using Raspberry Pi Zero 2 W

Status messages sent to the MQTT broker can be viewed from a PC using a web browser via the “Vibration Measurement System Monitoring Application.”

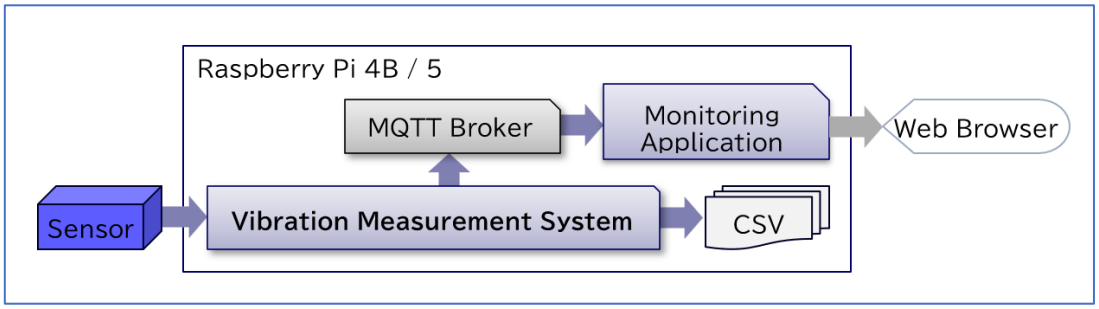

Figure 2-1 Overall System Cofiguration

2.1. Starting and Shutting Down the Raspberry Pi

This manual includes procedures for starting, shutting down, and rebooting the Raspberry Pi. The detailed steps are described below.

2.1.1. Starting the Raspberry Pi

- The Raspberry Pi will boot up when power (AC adapter) is connected to its USB Type-C connector.
- The OS boot process may take some time. If you intend to connect to the Raspberry Pi remotely, please wait for a while after connecting the power. (The green LED ceasing to blink is a good indicator.)
- Ensure that all external devices are connected to the respective connectors of the Raspberry Pi only when the power is turned off.

2.1.2. Shutting Down the Raspberry Pi

- The OS can be shut down by executing the following command in the OS command prompt:
  `sudo shutdown -h now`
  The shutdown process may take some time (the green LED ceasing to blink is a good indicator).
- After the OS has shut down, please turn off the power to the Raspberry Pi.

2.1.3. Rebooting the Raspberry Pi

- The OS can be rebooted by executing the following command in the OS command prompt:
  `sudo reboot`
- The reboot process may take some time (the green LED blinking and turning off twice is a good indicator).

2.2. Editing Files on the Raspberry Pi

There are procedures for editing configuration files and other files on the Raspberry Pi. These manual uses "nano" for file editing. The usage of nano is explained below.

1. When you type `nano “filename”` to edit the desired file, the screen shown below will open.

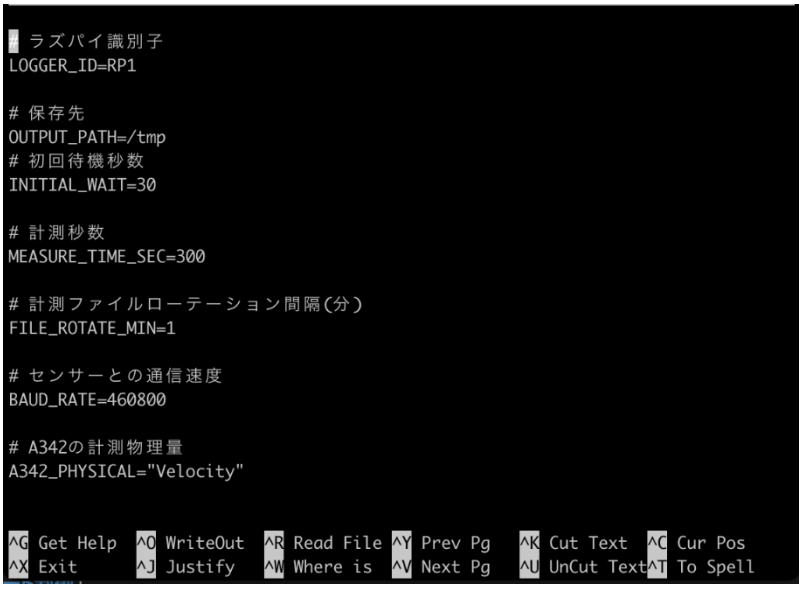

Figure 2-2

2. Use the keyboard to edit the file. Please note that the mouse cannot be used.
3. To save the file, press `Ctrl+X` The screen will change as shown below, then press `y`.

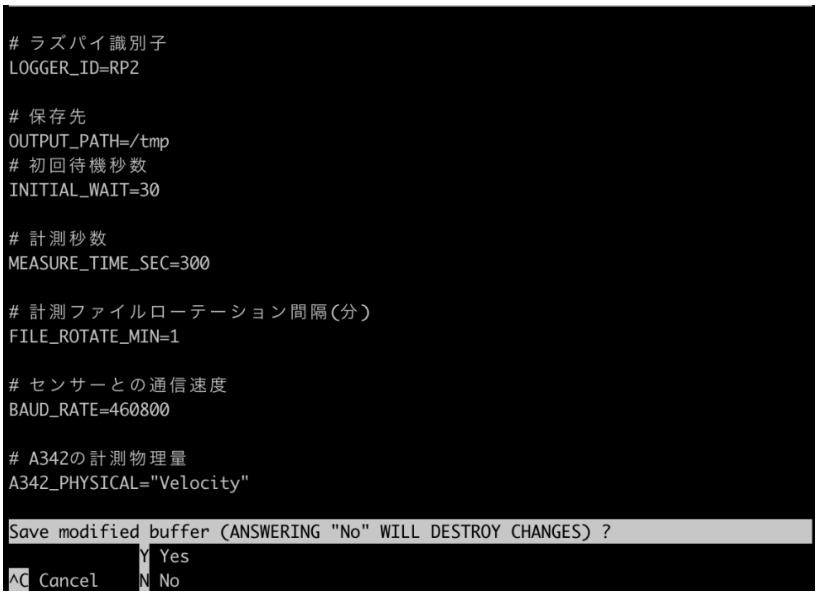

Figure 2-3

4. The file editing process is now complete.

3. Hardware Preparation

3.1. Common Preparation

Table 3-1

Items	Product Number	Manufacturer	Remarks
Raspberry Pi	Raspberry Pi 4B Raspberry Pi 5	Raspberry Pi Foundation	Memory size: 4GB or more
microSD Card	Extreme	SanDisk	(Verified product) 512GB
RTC-HAT	103030278	Seeed Studio	(Verified product)
Lithium Battery	CR1225	—	Used for RTC-HAT power retention
PC	—	—	Windows10 or later recommended
Wi-Fi Router	WMR-433W2-BK	Buffalo	(Verified product) When connecting the PC and Raspberry Pi via Wi-Fi

Note: To maintain time even then the power is off, it is strongly recommended to use an RTC backed up by a battery.

Additional items required for Raspberry Pi operation (please prepare items compatible with your Raspberry Pi model):

- Power supply (AC adapter) (must match your Raspberry Pi model)
- Equipment required only during the initial setup of the Raspberry Pi: Display, Keyboard
- Equipment required when connecting the Raspberry Pi and PC via wired LAN: Ethernet cable

In the standard procedure described in this setup manual, measurement data will be recorded on an SD card. Please ensure the SD card has sufficient capacity based on the following recommendations.

Table 3-2

Category	Details	Size
OS Area	Recommended in the procedure "4. Creating an SD Card for Raspberry Pi OS"	Approx. 3GB
Measurement Data	Vibration Sensor M-A342, Speed Measurement, 3000sps, 10 minutes	Approx. 130MB
Measurement Data	Acceleration Sensor M-A370, M-A352, Acceleration Measurement, 1000sps, 10 minutes	Approx. 50MB

If you save data to an external USB memory according to the procedure “8.3Saving Measurement Data to an External USB Memory," an additional external USB memory is required. (Verified product: SanDisk SDCZ430-512G-J57)

Additionally, to maintain measurements during commercial power interruptions or short power outages, it is recommended to use an uninterruptible power supply (UPS).

3.2. Measurement with M-A370AD, M-A352AD, M-A342VD

Table 3-3

Items	Product Number	Manufacturer	Remarks
Acceleration Sensor	M-A370AD M-A352AD	Seiko Epson	Use any of M-A370AD, M-A352AD, or M-A342VD
Vibration Sensor	M-A342VD	Seiko Epson	
Relay Board	M-G32EV51	Seiko Epson	Use M-G32EV51 and M-G32EV41 together
Evaluation Board	M-G32EV41	Seiko Epson	
USB Cable	—	—	Type A – Type C

3.3. Measurement with M-A552AR, M-A542VR

Table 3-4

Items	Product Number	Manufacturer	Remarks
Acceleration Sensor	M-A552AR	Seiko Epson	Either M-A552AR or M-A542VR
Vibration Sensor	M-A542VR	Seiko Epson	
RS422 Conversion Cable	TBD	TBD	Please contact us individually
AC Adaptor	ATS065T-P120	Adapter Technology	(Verified product)
RS422 – USB Converter	ICUSB422	StarTech	

3.4. Raspberry Pi 4B Case and Heat Dissipation

3.4.1. About the Case

- Due to interference with the RTC-HAT, do not use cases with thermal conduction protrusions that make direct contact with the CPU (e.g., Takachi Electric Industrial, model number RPH-4B-H). Use cases without thermal conduction protrusions inside (e.g., Takachi Electric Industrial, model number RPH-4B-N).
- When using a case, it is recommended to monitor the internal temperature during operation, as heat may accumulate inside the case, causing electronic devices to overheat and malfunction.

3.4.2. About Heat Dissipation

- It is recommended to attach a heatsink to the ICs such as the CPU (e.g., Seeed Studio, model number 110991329).
- Heat dissipation occurs not only from the surface of the Raspberry Pi board (component side) but also from the back (solder side), so it is recommended to provide space on the back of the board.
- If operating for extended periods in high-temperature environments, consider using a fan for additional cooling. Ensure it does not interfere with the RTC-HAT.
- If precise timekeeping is not required during measurements, you can remove the RTC-HAT and use a case with thermal conduction protrusions that make direct contact with the CPU for cooling (e.g., Takachi Electric Industrial, model number RPH-4B-H).

  When the RTC-HAT is removed, the time accuracy will depend on the internal clock of the Raspberry Pi.

3.5. About Raspberry Pi 5 Case

For Raspberry Pi 5, the case manufactured by DFROBOT, model number FIT0963 (Switch Science product code: DFROBOT-FIT0963) has been verified to work and is recommended.

4. Creating an SD Card Raspberry Pi OS

To create an SD card for booting Raspberry Pi OS using a PC, follow these steps:

1. Download the recommended OS image for operating a vibration measurement system using Raspberry Pi products from the following link:

   https://downloads.raspberrypi.com/raspios_lite_arm64/images/raspios_lite_arm64-2025-05-13/

   Click on the link ending with ".img.xz" to download.

2. Download the PC software required to write the Raspberry Pi OS image to the SD card from the following link:

   https://www.raspberrypi.com/software/

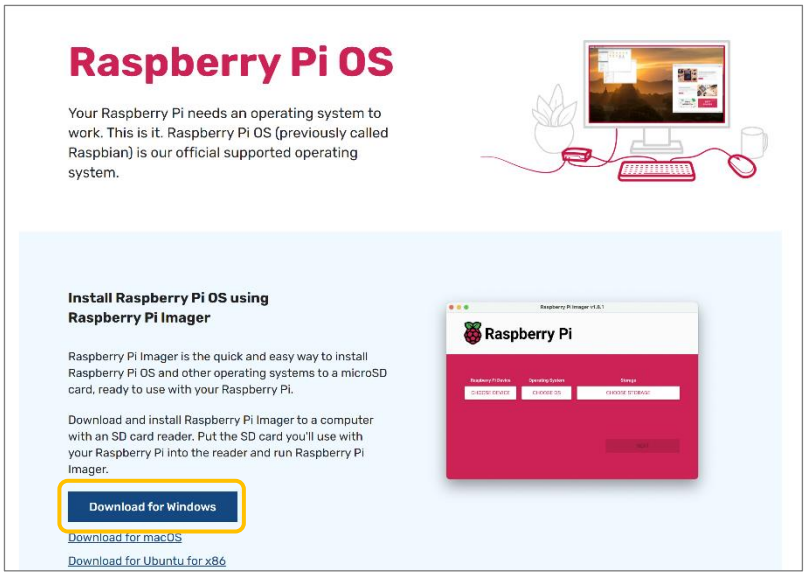

Figure 4-1

- The screenshots used in this manual are from Windows version 1.8.5. Please note that the screens may change without prior notice.

3. Double-click the downloaded "Imager_version.exe" to install Raspberry Pi Imager.
4. Launch Raspberry Pi Imager.
5. Insert the SD card into your PC.
6. Click "Choose Device" and selecte Raspberry Pi model you are setting up.

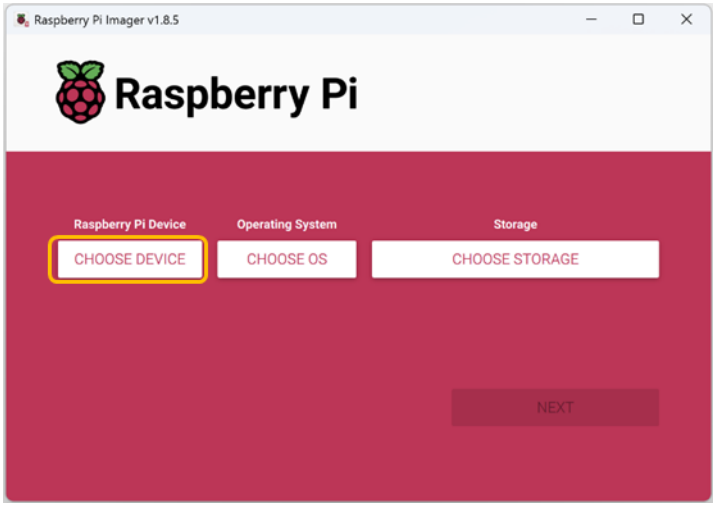

Figure 4-2

7. Click "Choose OS" and select "Use Custom" at the bottom. Then, choose the file downloaded in step 1.

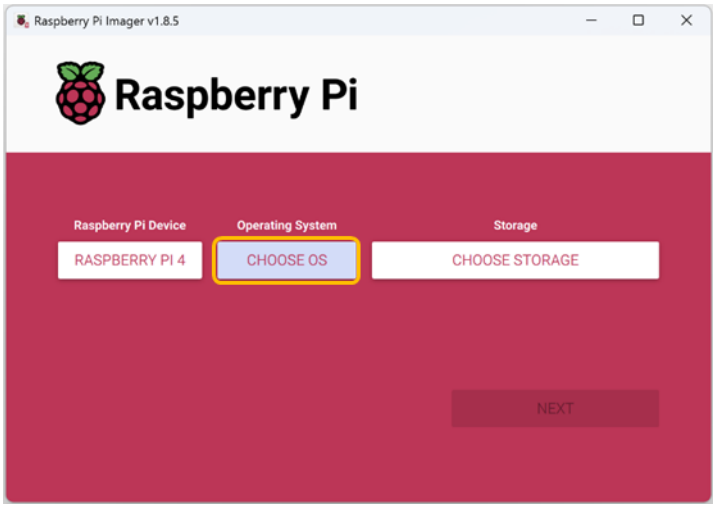

Figure 4-3

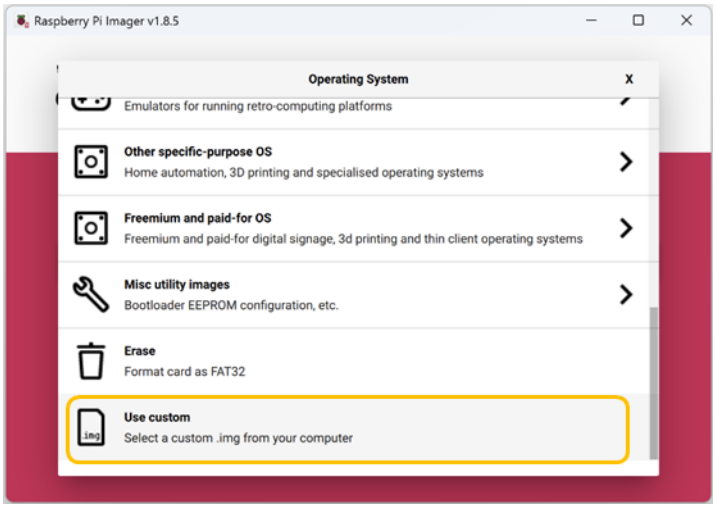

Figure 4-4

8. Click "Choose Storage" and select the SD card to write the OS image to.

Figure 4-5

9. Click "Next".

Figure 4-6

10. A screen will appear asking if you want to edit settings. Click "No". Another confirmation dialog will open; select "Yes" to proceed.

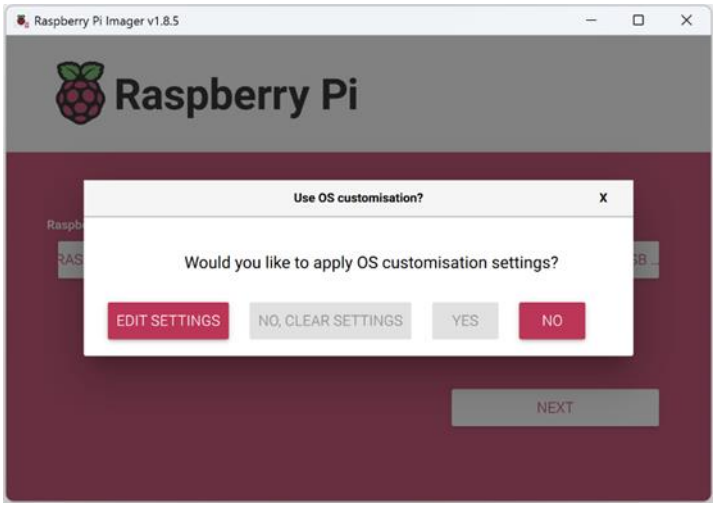

Figure 4-7

11. In the “General” tab, enter the following:

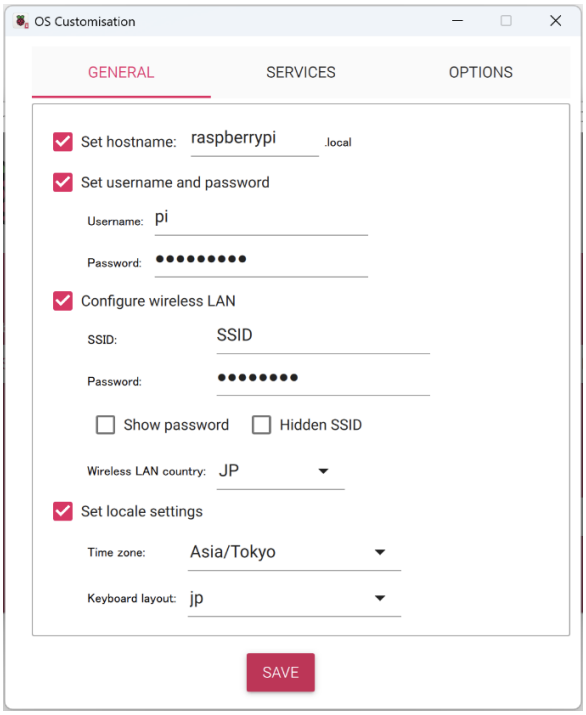

Figure 4-8

- Hostname: Set the hostname for the Raspberry Pi.
- Username and Password: Set the credentials for OS login.
- Wi-Fi: Enter the SSID and password for your Wi-Fi router. Also, set the country (e.g., JP for Japan).
- Locale: Set the time zone and keyboard layout.

12. In the “Services” tab, enable SSH and select password authentication.

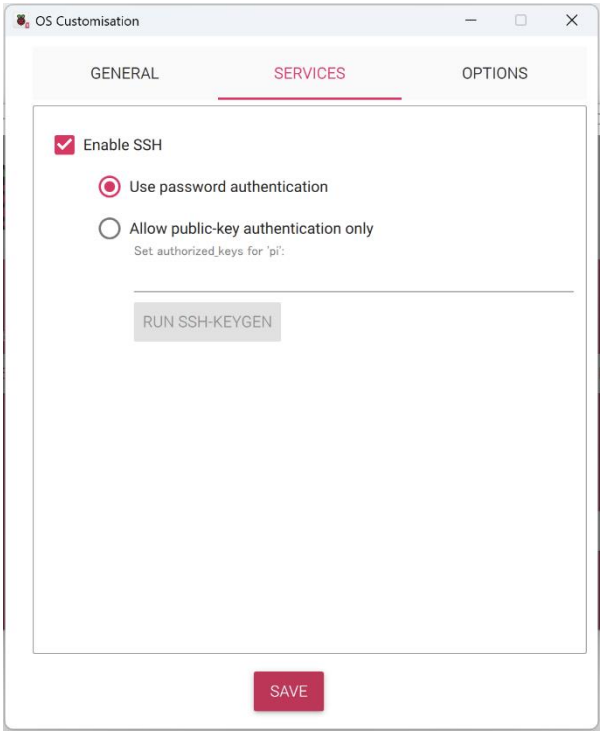

Figure 4-9

13. In the “Options” tab, no changes are needed. Click “Save.”
14. When prompted again to confirm customization, click “Yes.”
15. A warning will appear about erasing the media. Click “Yes” to begin writing.

16. Wait for the writing process to complete.

The OS image writing process is now complete.

Insert the SD card into the Raspberry Pi and proceed with "5. Initial Setup".

5. Initial Setup

This chapter describes the initial setup steps for the Raspberry Pi system.

The following instructions assume:

- Username created on Raspberry Pi: `pi`
- Static IP address assigned to Raspberry Pi: `192.168.1.52`

If you use different values, please substitute them accordingly.

Note: Until completing step "5.2.2 Connecting Raspberry Pi and Windows PC," it is not possible to operate the Raspberry Pi from the PC. Please prepare a display and keyboard.

5.1. Log in to Raspberry Pi OS

Log into Raspberry Pi OS by following steps.

1. Insert the SD card created in Chapter 4 into the Raspberry Pi's SD card slot.
2. Connect a display and keyboard to the Raspberry Pi.
3. Connect the AC adapter to power on the Raspberry Pi.
4. After a few reboots, a login prompt will appear. Log in using the username and password you set during OS customization.

Confirm OS Customization.

1. Run `date` command to check the system time. Ensure the time zone is correctly set (e.g., JST for Asia/Tokyo).

   Note: If the Raspberry Pi is not connected to the internet, the displayed time may not be accurate.
2. Run `nmcli` command to check network settings. You'll see something like:

```
If Wi-Fi is configured and connected, you'll see something like:
wlan0: connected to preconfigured
        "Broadcom Wi-Fi"
        wifi (brcmfmac), _MAC_ADDRESS_, hw, mtu 1500
        ip4 default, ip6 default
         (Network settings will be displayed)

If not connected:
wlan0: disconnected
        "Broadcom Wi-Fi"
```

Figure 5-1

5.2. PC Remote Connection Setup Using a Wired Lan Cable

This section explains how to configure the Raspberry Pi for remote access from a PC using a wired LAN connection.

5.2.1. Setting a Static IP Address

1. Enter the following command to set the IP address for the wired LAN connection:

```
sudo nmcli connection modify “Wired connection 1” \
   ipv4.addresses 192.168.1.52/24 \
   ipv4.route-metric 1000 \
   ipv4.method manual \
   ipv6.method disabled
```

Note：

- Use a backslash (\) at the end of each line to continue the command across multiple lines.
- If your Wi-Fi network uses the 192.168.1.x subnet, change to a different subnet such as 192.168.2.x.
- If you are setting up multiple Raspberry Pi devices, change the last number in the IP address (e.g., 52) to a unique value between 2 and 255.

2. Reboot the Raspberry Pi. (Refer to 2.1.3Rebooting the Raspberry Pi)

5.2.2. Connecting Raspberry Pi and Windows PC

To set up a remote (SSH) connection between a Raspberry Pi and a PC, follow these steps:

1. Connect the Raspberry Pi and the PC using an Ethernet cable.
2. On the PC, open “Settings” → “Network & internet,” then click “Ethernet”.

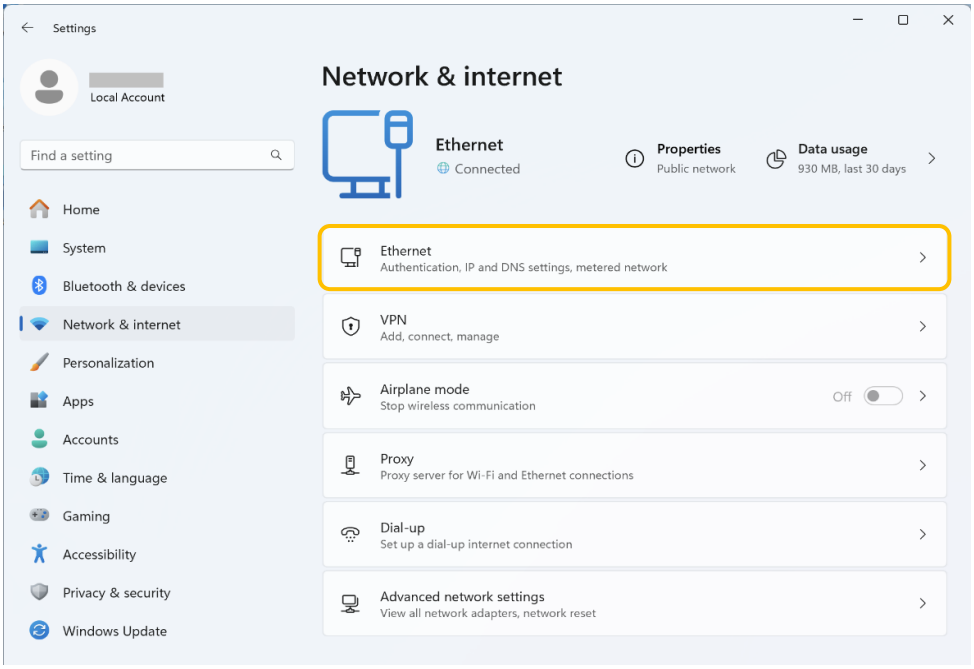

Figure 5-2

3. In the “IP assignment”, click “Edit.”

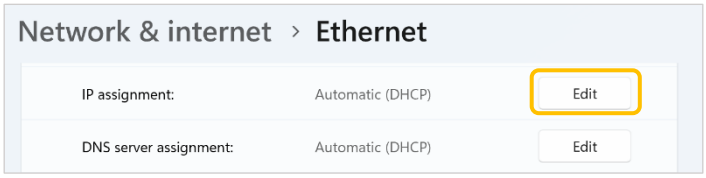

Figure 5-3

In the “Edit IP settings” dialog:

- Change from Automatic (DHCP) to “Manual”
- Enable “IPv4”
- Enter an IP address and subnet mask as follows:
  - Use the same subnet as the Raspberry Pi (e.g., 192.168.1.x)
  - Ensure the last number is different from the Raspberry Pi’s IP

4. From the PC's PowerShell, execute the following command to verify the LAN connection.
   - `ping 192.168.1.52`
     - If you receive replies with no packet loss, the connection is successful.
5. From the PC's PowerShell, execute the following command to log into the Raspberry Pi. When prompted, enter the password set on the Raspberry Pi to log in.
   - `ssh pi@192.168.1.52`
     - On first access, confirm the SSH prompt by typing `yes`
     - Enter the password you set for the Raspberry Pi

After this setup, you can access the Raspberry Pi remotely via SSH without needing a display or keyboard.

5.3. Internet Connection Setup Using a Broadband Wi-Fi Router

To connect to the internet, configure the Wi-Fi settings. If you already configured Wi-Fi during the OS customization step in [Chapter 4: Creating SD Card for Raspberry Pi OS], you may skip this section.

1. Ensure your broadband Wi-Fi router is powered on and ready for connection.
2. Open the Raspberry Pi terminal and execute `sudo raspi-config` to open the configuration tools
3. From the "System Options" menu, select "Wireless LAN."

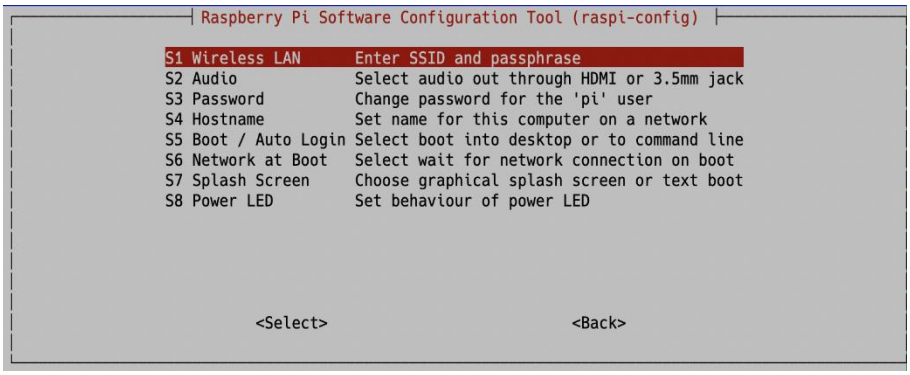

Figure 5-4

4. If prompted to select the country where the Raspberry Pi will be used, choose "JP" and select "<Ok>."

   (This screen may not appear in some cases.)

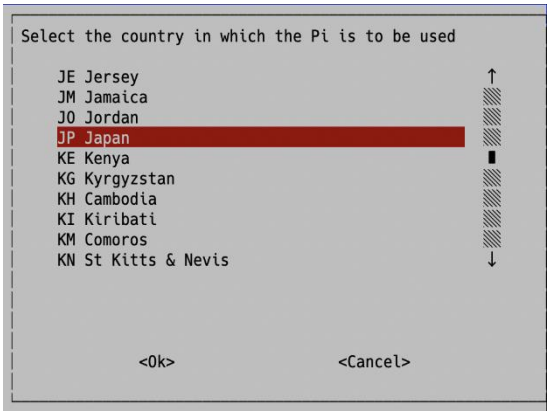

Figure 5-5

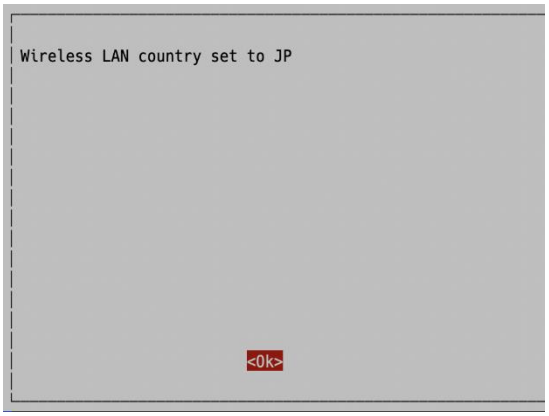

Figure 5-6

5. Next, enter the SSID and password for the Wi-Fi router you want to connect to.

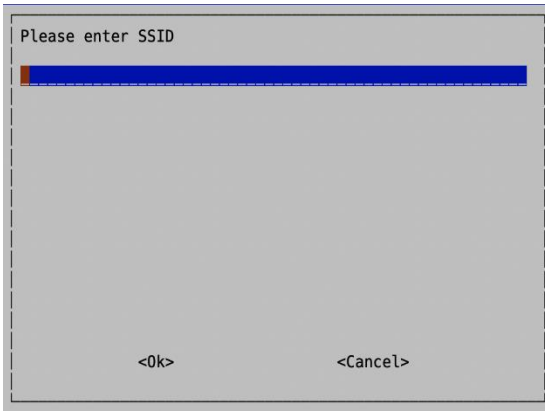

Figure 5-7

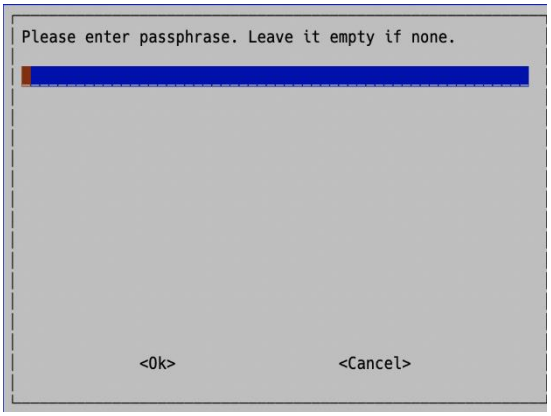

Figure 5-8

6. Select to <Finish> to exit the configuration tool.
7. To verify the internet connection, run the following command in terminal:
   - `ping google.co.jp`
     - If you receive replies and “packet loss” is 0%, the connection is successful.

5.4. Time Setting Using RTC-HAT

Due to the significant inaccuracy of the internal clock of the Raspberry Pi 4B, the following steps describe how to connect an expansion module and set the correct time.

Note: This setup requires the Raspberry Pi to be connected to the internet.

For Raspberry Pi 5 Uses

- Raspberry Pi 5 includes a built-in RTC.
- To maintain time while powered off, connect a battery module (e.g., Raspberry Pi Foundation SC1163) to the RTC battery connector.
- If using the built-in RTC, refer to publicly available setup guides online.

The logger software does not detect connection issues with the RTC-HAT or improper installation of the lithium battery. If the RTC-HAT malfunctions due to these issues, the timekeeping accuracy will default to the internal clock accuracy of the Raspberry Pi. It is recommended to verify the correct time before measurements using the `date` command or similar tools.

Setup Steps for RTC-HAT (Raspberry Pi 4B)

1. To download the RTC-HAT setup script, execute the following command.
   - `sudo apt update`
   - `sudo apt install git -y`
2. Execute `git clone https://github.com/Seeed-Studio/pi-hats.git` to download the setup script.

   Note: This repository was archived in October 2024 and is no longer maintained.
3. Execute `sudo ~/pi-hats/tools/install.sh -u rtc_ds3231`

   Note: This script may not support newer OS versions, so the following modifications are required.
   Please note that these modifications are **not officially recommended by the manufacturer**.

   Edit the configuration file:

   - Run `cat /boot/config.txt` and view current config:

```
DO NOT EDIT THIS FILE

The file you are looking for has moved to /boot/firmware/config.txt
dtparam=i2c_arm=on
dtoverlay=i2c-rtc,ds3231
```

Figure 5-9

   - Run `sudo nano /boot/firmware/config.txt` to open the actual config file. Insert the following two lines near the top.

```
# For more options and information see
# http://rptl.io/configtxt
# Some settings may impact device functionality. See link above for details

# Uncomment some or all of these to enable the optional hardware interfaces
#dtparam=i2c_arm=on
#dtparam=i2s=on
#dtparam=spi=on
* Insert the last two lines from "/boot/config,.txt" file.
```

Figure 5-10

4. Execute: `sync`
5. Turn off the Raspberry Pi (refer to section 2.1.2 Shutting Down the Raspberry Pi). Attach the RTC-HAT to the leftmost GPIO pins as shown in the diagram, then install the lithium battery into the RTC-HAT.

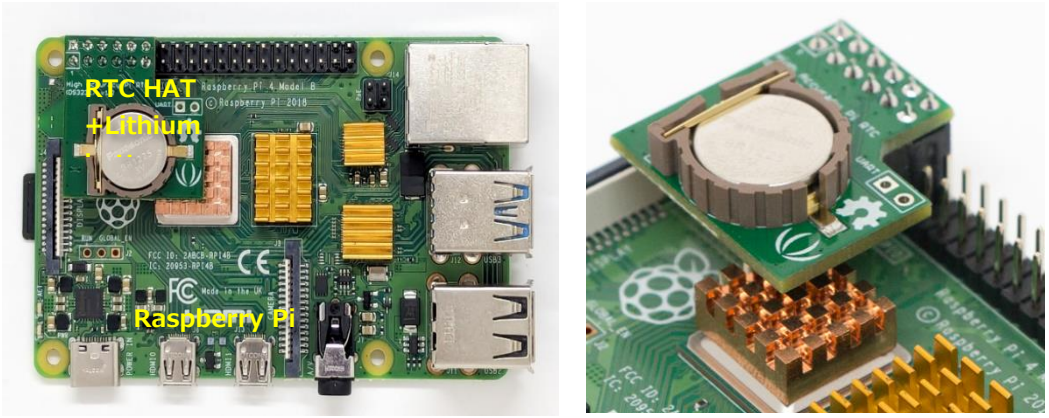

Figure 5-11

Note：Be cautious when attaching or detaching the RTC-HAT while the Raspberry Pi is powered on or the lithium battery is installed. This can reset the RTC, so handle it carefully.

- Please also refer to the RTC-HAT manual at https://wiki.seeedstudio.com/High_Accuracy_Pi_RTC-DS3231/.

6. Turn the Raspberry Pi power back on.
7. Execute `sudo hwclock -s` to synchronize the Raspberry Pi time with the RTC-HAT time.
8. Execute `date`, verify that the correct current time is displayed.
9. To periodically correct the internal clock of the Raspberry Pi, execute `crontab -e` to open the scheduled command execution settings screen, and add the following content. (Editing is possible with the same operations as nano)

```
0 * * * * sudo hwclock -s
```

Figure 5-12

5.5. Creating a Data Storage Folder

This program will create a folder to store measurement data. While this program can set any folder as the storage location for measurement data, here it will create the folder in the home directory of the configured Raspberry Pi user.

1. Enter `cd` to move to the home directory.
2. Execute `mkdir measure  create` a folder to store measurement data.
   - This program does not automatically create the storage folder, so it must be created manually in advance.
   - In the configuration file of this program, set /home/pi/measure.

6. Application Configuration

Note：This configuration requires the Raspberry Pi to be connected to the internet.

6.1. Extracting the Source Code

To run the program on the Raspberry Pi, copy the downloaded source code to the Raspberry Pi

1. Unzip the downloaded source code on your PC.
2. Connect the Raspberry Pi and PC using the method described in section 5.2 PC Remote Connection Setup Using a Wired Lan Cable.
3. To run the program on the Raspberry Pi, copy the downloaded source code to the Raspberry Pi.
   - `sudo mkdir /app`
   - `sudo chown pi:pi /app`
4. On your PC, open the unzipped folder (e.g., MSG002-001a_version) in PowerShell.
   - While holding the Shift key, right-click in Explorer to display the "Open PowerShell window here" option.
5. Execute the following command in PowerShell to install the program on the Raspberry Pi：
   - `scp -r MSG002-001a pi@192.168.1.52:/app`
6. On the Raspberry Pi, confirm that the source code in "/app/MSG002-001a" folder has been copied.
   - Run ls `/app/MSG002-001a` and confirm the contents

6.2. Installing Packages

To run the program, create a Python virtual environment and install the packages used by the program on the Raspberry Pi.

1. On the Raspberry Pi terminal, execute the following command to move to “/app/MSG002-001a” folder and make the virtual environment:
   - `cd /app/MSG002-001a`
   - `python -m venv venv`
   - `source venv/bin/activate`
2. Update the tool for installing packages and install the required packages:
   - `pip install -U pip`
   - `pip install .`

Note: If you use a different name for the virtual environment, you must update the path in the next step (6.3 Service Registration).

6.3. Service Registration

This section explains how to register the system as a service so that it can be automatically started by the OS.

1. Open the service definition file for editing: `nano /app/MSG002-001a/bin/logger@.service`
   - If the username you created on Raspberry Pi is not "pi", change line 13:
     - User=pi

     to match your actual username.
   - If you named the virtual environment something other than venv, update line 10 accordingly:
     - ExecStart=.../venv/...
2. Install the service definition file:
   - `sudo cp /app/MSG002-001a/bin/logger@.service /etc/systemd/system/`
3. Reload the system configuration:
   - `sudo systemctl daemon-reload`
4. Check the service status:
   - `systemctl status logger@auto.service`
     - If you see "Loaded: loaded", the setup was successful.

Note: When measuring time by synchronizing with **internet time synchronization (NTP) without using an RTC**, it is necessary to delay the system startup until NTP synchronization is complete. Execute the following command to enable an additional service:

- `sudo systemctl enable systemd-time-wait-sync.service`

6.4. Installing Programs Required for Measurement

Some measurement methods use Linux programs.

1. Execute following commands to install "at" program:
   - `sudo apt update`
   - `sudo apt install at -y`

6.5. Creating Configuration Files

Create the configuration file that the program will load:

- `cp /app/MSG002-001a/.env.default /app/MSG002-001a/.env`

The program operates using the contents of the "/app/MSG002-001a/.env" file as configuration values. For configurable values, please refer to the 'Operation Manual for Vibration Measurement System Using Raspberry Pi Products.

7. Connecting Sensors

7.1. Connecting M-A370AD, M-A352AD , M-A342VD to Raspberry Pi 4B

1. Connect the sensor with the M-G32EV51 & M-G32EV41.

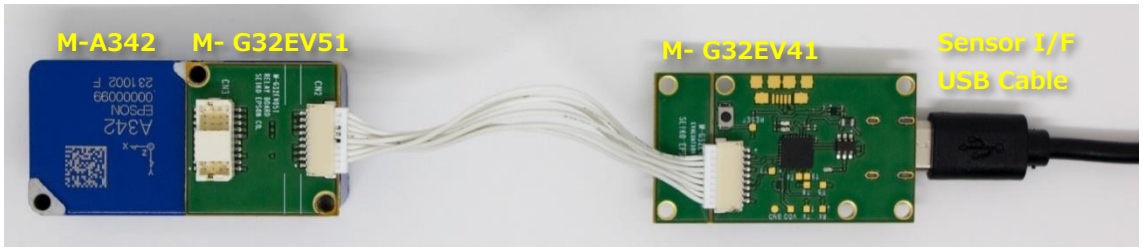

Figure 7-1

2. Shut down the OS and turn off the power of the Raspberry Pi. Connect the M-G32EV41 and Raspberry Pi 4B using a USB Type A – Type C cable. Please connect to USB 2.0 of the Raspberry Pi.

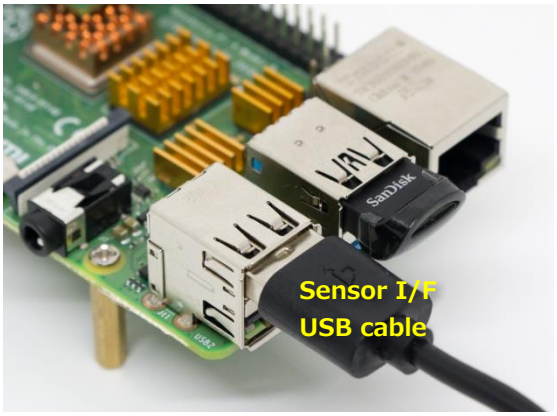

Figure 7-2

7.2. Connecting M-A552AR, M-A542VR to Raspberry Pi 4B

1. Connect the M-A552AR or M-A542VR with the ICUSB422 using an RS422 conversion cable.

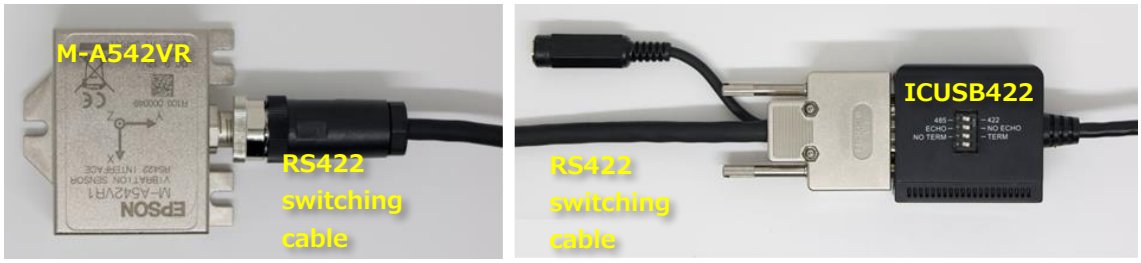

Figure 7-3

2. Configure the dip switches of the ICUSB422.

Table 7-1

Dip SW no	Setting
1	422
2	NO ECHO
3	TERM
4	- (Either ON/OFF acceptable)

Figure 7-4

3. Shut down the OS and turn off the power of the Raspberry Pi. Connect the ICUSB422 to the USB 2.0 connector of the Raspberry Pi.
4. Connect the ICUSB422 and the RS422 conversion cable, then connect the AC adapter.

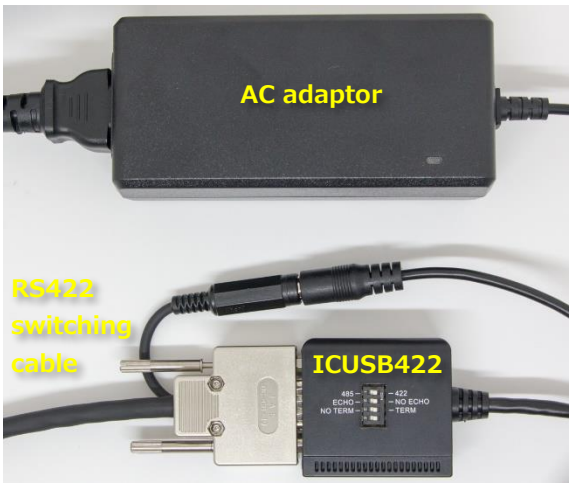

Figure 7-5

This completes the setup. Please refer to the 'Operation Manual for Vibration Measurement System Using Raspberry Pi Products' to conduct vibration measurements.

8. Appendix

8.1. Setting Up the MQTT Broker for System Status Messages

This system includes functionality to send messages via the MQTT protocol, such as measurement status, sensor conditions, and error notifications. To handle these messages, an MQTT broker is required. This section explains how to set up the MQTT broker **Mosquitto** on the Raspberry Pi.

1. Install Mosquitto on the Raspberry Pi by executing the following commands:
   - `sudo apt update`
   - `sudo apt install mosquitto`
2. Once installed, Mosquitto will start automatically. Confirm its status with:
   - `systemctl status mosquitto`
     - If you see “Active: active (running)”, the broker is running successfully.
3. Apply a minimal configuration file by executing:
   - `sudo cp /app/MSG002-001a/bin/mosquitto-logger.conf /etc/mosquitto/conf.d`
4. Restart the Mosquitto service to apply the configuration:
   - `sudo systemctl restart mosquitto`

Additional Notes

- By default, the system’s message transmission feature is disabled.
- For details on configuration options and message types, refer to the “Operation Manual for Vibration Measurement System Using Raspberry Pi Products.”
- Messages sent by the system can be viewed via a web browser using Epson’s “Monitoring Application for Vibration Measurement System.” For details, refer to the “Monitoring Application User Guide.”

8.2. Remote Connection Using a Broadband Wi-Fi Router

This manual describes PC remote connection via wired LAN and internet connection via wireless LAN to a broadband Wi-Fi router. However, remote connection to a PC using a Wi-Fi router is also possible. The setup for remote connection using the recommended WMR-433W2-BK is described.

Setup Steps

1. Configure wireless LAN either during OS customization in [Section 4: Creating the Raspberry Pi OS SD Card] or via [Section 5.3: Internet Connection Using Broadband Wi-Fi Router].
2. Set a static IP address for the wireless LAN connection by entering the following command:

```
sudo nmcli connection modify “preconfigured”  \
  ipv4.addresses  192.168.13.135/24  \
  ipv4.gateway  192.168.13.1  \
  ipv4.dns  192.168.13.1  \
  ipv4.method  manual
```

Note: The above assumes the wireless LAN connection name is “preconfigured” (as set during OS customization). If you configured Wi-Fi using the method in Section 5.3, use `nmcli connection show` to find the actual connection name and replace it accordingly.

Note: The third segment of the IP address "13" is based on the time of this manual's creation. Please refer to the router's IP address listed in the WMR-433W2-BK manual and configure accordingly.

Note: To connect and operate multiple Raspberry Pi devices on a PC, change the number "135" in the "static ip_address" line to any number between "66" and "255" for each Raspberry Pi.

3. Restart the Raspberry Pi.
4. Connect the PC to the Wi-Fi router's network.
5. Use PowerShell to remotely connect by entering:
   - `ssh pi@configured-IP-address`

8.3. Saving Measurement Data to an External USB Memory

This section explains how to use an external USB memory as the storage location for measurement data. Using an external USB memory makes it easier to access measurement data from a PC.

In the following steps, connect the USB memory to "/media/usb0" and save the measurement data to "/media/usb0/measure".

1. Execute `sudo mkdir /media/usb0` to create a folder to recognize the USB memory.
2. Execute `sudo chmod 777 -R /media/usb0` .
3. Connect the USB memory formatted with exFAT to the USB 3.0 connector.

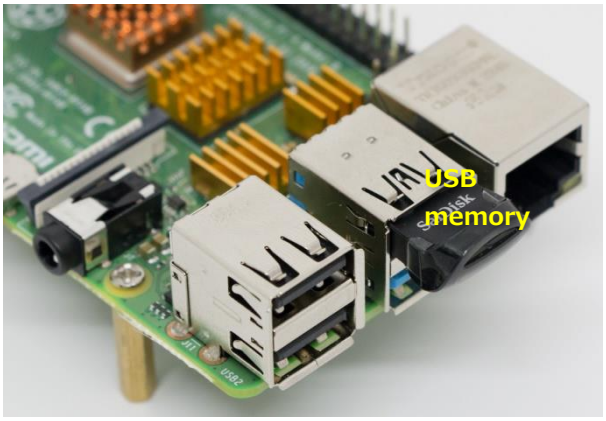

Figure 8-1

4. With the USB memory connected, execute `lsblk -o UUID,NAME,FSTYPE,SIZE,MOUNTPOINT,LABEL,MODEL` to obtain the UUID of the USB memory.
   - The execution results will display items other than the USB memory (such as SD cards). Confirm the USB memory by checking the MODEL and capacity, then obtain the UUID.
5. Open the fstab file using the command `sudo nano /etc/fstab` and append the following content.

```
UUID= the obtained UUID /media/usb0  exfat  defaults,nofail,uid=1000,gid=1000  0  2
```

Figure 8-2

   - By adding this entry to fstab, the USB memory will be automatically recognized on subsequent OS boots.。
6. Execute `sudo mount -a` to recognize the USB memory on the Raspberry Pi.
7. Execute `mount -v` to ensure that the output starts with the following.
   （Other mounted devices will also be displayed.）

```
/dev/sda1 on /media/usb0 type exfat ⋯
```

Figure 8-3

8. Execute `mkdir /media/usb0/measure` to create a folder for saving measurement data.
   - This program does not automatically create the save folder, so it must be created manually in advance
   - Set "/media/usb0/measure" in the configuration file of this program.

Note: If you use external USB memory for data storage, you must modify the service definition file installed in [Section 6.3: Service Registration] to wait for the USB memory to mount before starting the system.

- Open the service file:
  - `sudo nano /etc/systemd/system/logger@.service`
- Add the following to the “After=…” line (around line 5):
  - After=time-sync.target media-usb0.mount
- Save the file and reload the service definitions:
  - `sudo systemctl daemon-reload`

Note: If the USB memory is connected to a location other than "/media/usb0", check the results of the following command and check the correct mount unit name and adjust the “After=…” line accordingly:

- `systemctl list-units –-type=mount`

8.4. Power Supply via PoE (Power over Ethernet)

For the Raspberry Pi 4B, power supply via PoE has been confirmed to work using the equipment listed below.

Note: PoE functionality has not been verified for other Raspberry Pi models.

Table 8-1

Items	Product Number	Manufacturer	Remarks
PoE injector	500-SWH011	Sanwa Supply	
PoE HAT	SC1022	Raspberry Pi	
LAN cable	-	-	Use Cat.5e or higher-grade cable

8.5. Measurement Using Raspberry Pi Zero 2 W

The software of this system can also be run on the Raspberry Pi Zero 2 W. Please refer to the following for setting up the Raspberry Pi Zero 2 W (hereafter referred to as Zero).

Note：Please note that compared to the 4B, the Zero has lower hardware performance, which may result in data loss during long-term measurements or when connecting multiple sensors.

8.5.1. Hardware Preparation

To set up the Zero as a substitute for the Raspberry Pi 4B, the following hardware is required.

Table 8-2

Items	Description
Raspberry Pi	Raspberry Pi Zero 2 W
microSD Card	A card with sufficient capacity to record measurement data is required
AC Adapter	Prepare one suitable for the Zero. (Tested product: SanDisk Extreme)
Display	A display that can connect to the Zero's mini HDMI (use an adapter if necessary) is required.
Keyboard	A keyboard that can connect to a USB-A port is required.
USB Hub	USB Hub A USB hub that can connect to the Zero's micro USB port (use an adapter if necessary) and has multiple USB-A ports is required. (USB 2.0 ports are recommended for sensor connections)

	If connecting to a PC with a wired LAN cable as described in this manual, a hub with an Ethernet port is also required. (Setup is also possible with wireless LAN only)
PC	Windows 10 or later is recommended.
Wireless LAN	A wireless LAN environment, such as a Wi-Fi router, that can connect the PC and Raspberry Pi to the internet is required.

8.5.2. Setup Procedure

The following steps describe the setup procedure for the Zero.

1. Follow the steps in "3. Creating an SD Card for Raspberry Pi OS" to create the SD card.
   - Select "Raspberry Pi Zero 2 W" as the target device.
2. Connect the hardware components.
   - Insert the SD card into the Zero.
   - Connect the display to the Zero
   - Connect the USB hub to the Zero, and connect the keyboard to the USB hub.
   - Finally, connect the AC adapter to the Zero and power it on.
3. Follow the steps in "4. Initial Setup" to set up the Zero.
   - If using the Zero with a connected keyboard and display, you can skip the steps in "4.2. PC Remote Connection Setup Using a Wired LAN Cable".
   - If you skip the steps in "4.2", you will access the Zero via wireless LAN.
   - After following the steps in "4.3. Internet Connection Setup Using a Broadband Wi-Fi Router", to find the IP address assigned to the Zero, execute the `ifconfig` command from the Zero's terminal and refer to the "inet" entry in the block starting with "wlan0".
4. Continue with the steps from "6. Application Settings" onwards.

```
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.50.XXX  netmask 255.255.255.0  broadcast 192.168.50.255
```

Figure 8-4 if config execution result

9. Contact Information

Seiko Epson Corporation

Sales Headquarters MD Sales Department

Contact Information via the Internet

https://global.epson.com/products_and_drivers/sensing_system/privacy/area_select_inquiry_contact.html