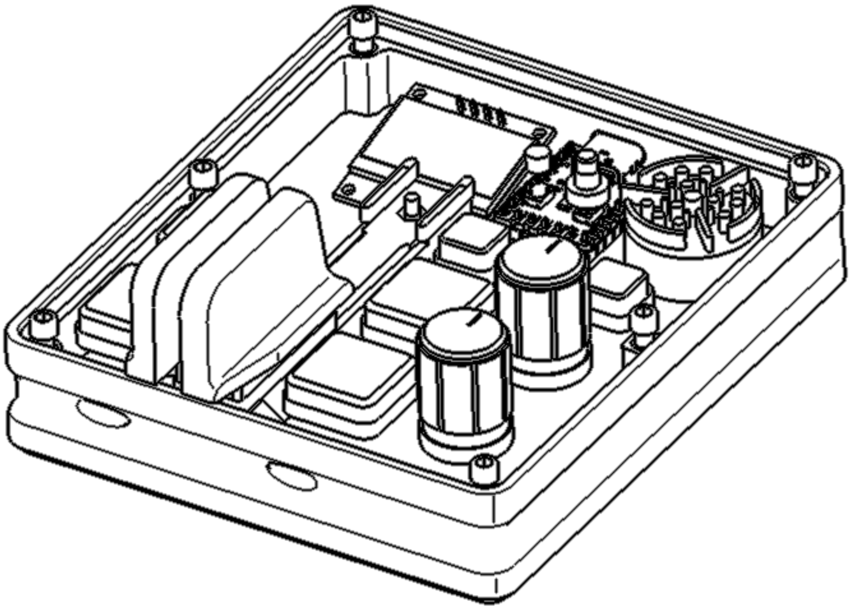
MORSEKEY-A01

(Morse Code Keyboard Device)

Assembly & User Manual
(English Edition)



Product Overview

The MORSEKEY-A01 is a Morse code keyboard input device based on the RP2040-Zero board.
It supports both Korean and English and converts Morse code into typing signals, functioning as a PC input device.

With the macro function, frequently used phrases can be entered automatically, and the OLED display provides real-time information such as language mode, transmission speed (WPM), interval (IWG), and sound configuration.
The circuit and structure are designed for simple assembly and easy customization, requiring only basic soldering to complete.

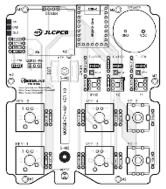
Manufactured Parts

The MORSEKEY-A01 includes components that must be individually fabricated.
The PCB, 3D-printed parts, and acrylic cover each require separate printing or custom production.

Circuit schematics and 3D design files can be found on the official OSHWLab project page.
https://oshwlab.com/kimgx05/morsekey-a01

- PCB : Can be directly ordered from JLCPCB using the Gerber data.
- 3D-Printed Parts : Produced using the provided STL files.
- Acrylic Parts : Cut with a laser cutter using the 2D DXF data.

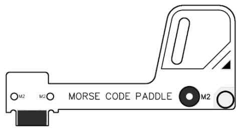
A1



MORSEKEY MAIN PCB
(1EA)

JLCPCB PCB Fabrication
File: Gerber_MORSEKEY_MAIN_PCB.zip

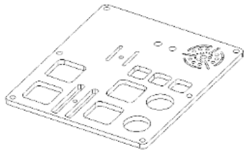
A2



MORSEKEY PADDLE PCB
(2EA)

JLCPCB PCB Fabrication
File: Gerber_MORSEKEY_PADDLE_PCB.zip

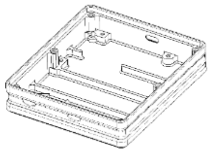
A3



MORSEKEY COVER
(1EA)

3T Acrylic Laser Fabrication
File: MORSEKEY_COVER.dwg

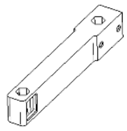
A4



MORSEKEY CASE
(1EA)

3D printed
File: MORSEKEY_CASE.stl

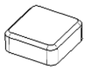
A5



MORSEKEY BLOCK.
(1EA)

File: MORSEKEY_BLOCK.stl

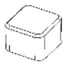
A6



MORSEKEY KEY C1
(4EA)

Files: MORSEKEY_KEY_C1.stl

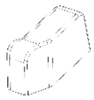
A7



MORSEKEY KEY C2
(3EA)

Files: MORSEKEY_KEY_C2.stl

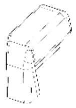
A8



MORSEKEY PADDLE L
(1EA)

Files: MORSEKEY_PADDLE_L.stl

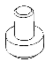
A9



MORSEKEY PADDLE R
(1EA)

Files: MORSEKEY_PADDLE_R.stl

A10



MORSEKEY RESET PIN
(2EA)

File: MORSEKEY_RESET_PIN.stl

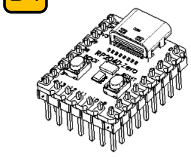
(JLCPCB: https://jlcpcb.com)

Electronic Components

The MORSEKEY-A01 operates based on the RP2040-Zero and is composed of key electronic components required for input signal processing and operation control.

The table below lists the main electronic components necessary for assembly.

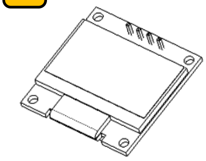
B1



RP2040-Zero

(1EA)

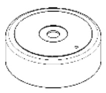
B2



SSD1306 OLED
(4Pin, 128×64)

(1EA)

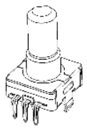
B3



스피커
(YZ2308)

(1EA)

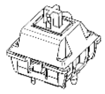
B4



Rotary Encoder
(Plum EC-11)

(2EA)

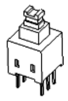
B5



Key Switch
(Cherry MX)

(4EA)

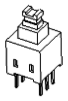
B6



Push Switch
(7x7 LOCKING)

(2EA)

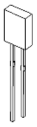
B7



Push Switch
(7x7 UNLOCK))

(1EA)

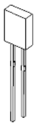
B8



LED-R
(square LED)

(7EA)

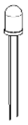
B9



LED-G
(square LED)

(1EA)

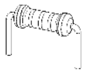
B11



LED-R
(5mm LED)

(1EA)

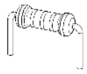
B12



Resistor
(120Ω)

(9EA)

B13



Resistor
(10kΩ)

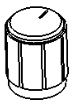
(15EA)

Fastening Components

The MORSEKEY-A01 assembly requires various fastening parts such as bolts, nuts, magnets, and rubber feet.

These components are used for external assembly and structural fixation, and are summarized in the list below.

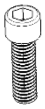
C1



Aluminum Knob
(14 x 16mm)

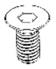
(2EA)

C2



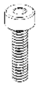
Hex Socket Head Cap Screw
(M3 × 10mm)
(6EA)

C3



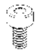
Flat Head Socket Screw
(M3 × 5mm)
(4EA)

C4



Hex Socket Head Cap Screw
(M2 × 8mm)
(8EA)

C5



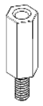
Hex Socket Head Cap Screw
(M2 × 4mm)
(6EA)

C6

Socket Set Screw
(M3 × 8mm)

(2EA)

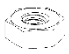
C7



Hex Standoff
(M2 x 10mm)

(2EA)

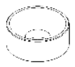
C8



M2 Hex Nut

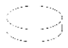
(2EA)

C9



Countersunk Round Neodymium Magnet
(6.5 x 4 x 4mm)
(4EA)

C10



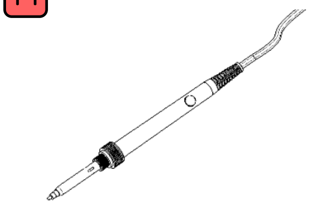
EVA Rubber Foot
(Adhesive Type)
(8mm x 2T)
(4EA)

Required Tools

This is a list of the **basic tools** needed during the assembly process.

Please prepare the following tools to ensure **accurate fastening** and **proper soldering**.

T1

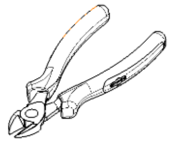


Soldering Iron

T2

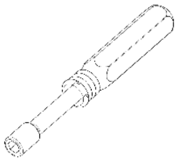
무연 납

T3



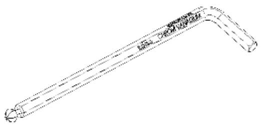
Nippers

T4



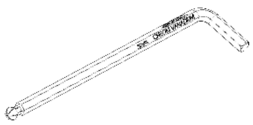
Hex Nut Driver
(4mm)

T5



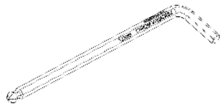
Hex Wrenches

T6



Hex Wrenches

T7



Hex Wrenches

Firmware and Scripts

Official GitHub Repository (MORSEKEY-A01 Project):
https://github.com/Airmodeling/MORSEKEY-A01

The MORSEKEY-A01 operates on CircuitPython.
Download and use the firmware and macro editor provided in the repository.

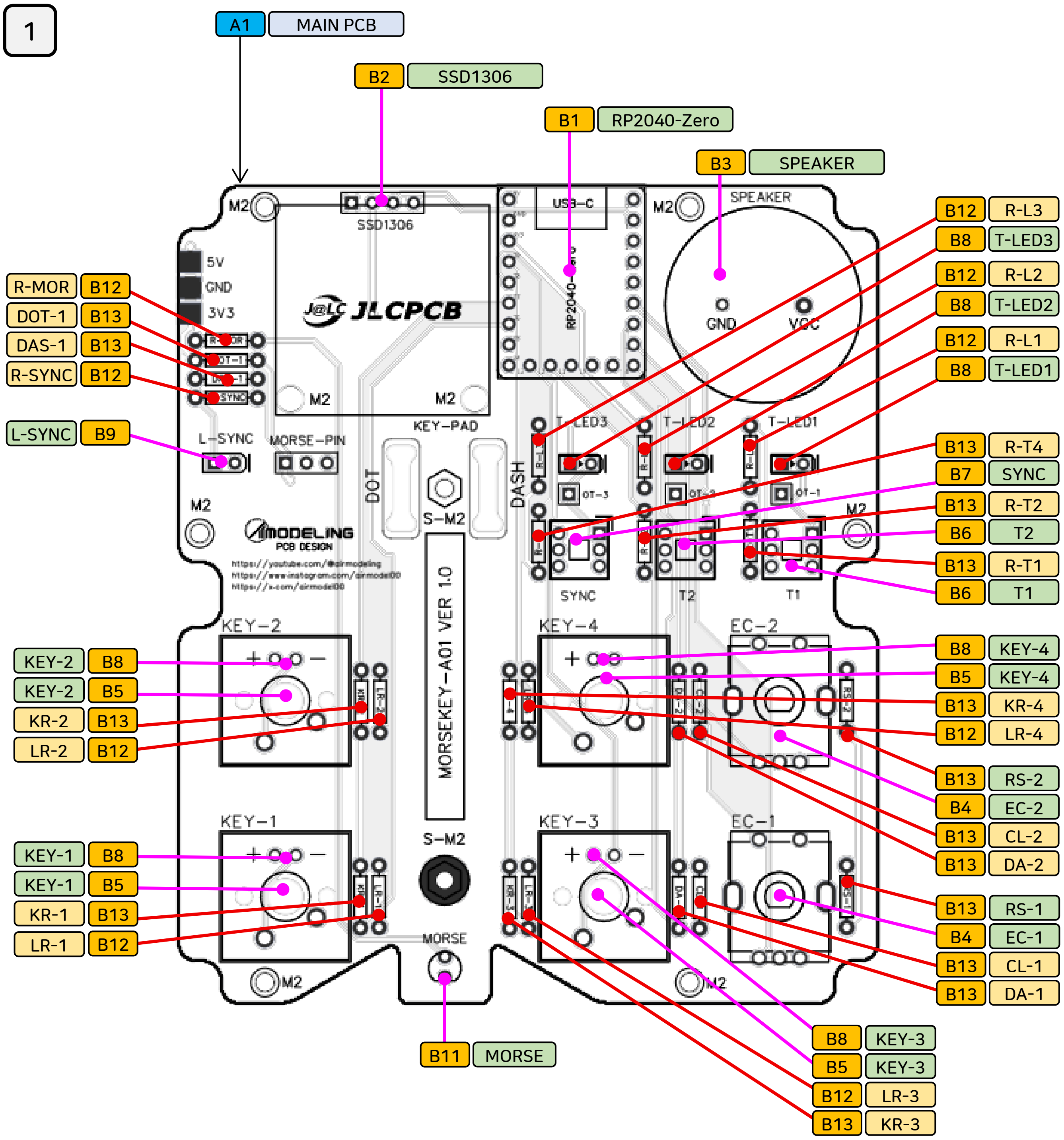
The following are the main program files included in the project:

- MORSEKEY_CODE → Main program code (CircuitPython)
- MORSEKEY_Macro_Editor → PC application for editing macro files
- adafruit-circuitpython-raspberry_pi_pico-en_US-9.2.8 → CircuitPython firmware for RP2040-Zero

PCB Soldering Work

At this stage, electronic components of the MORSEKEY-A01 are soldered onto the PCB.
Before soldering, always check the orientation and polarity of each component.

Use lead-free solder to ensure a clean and stable soldering quality.

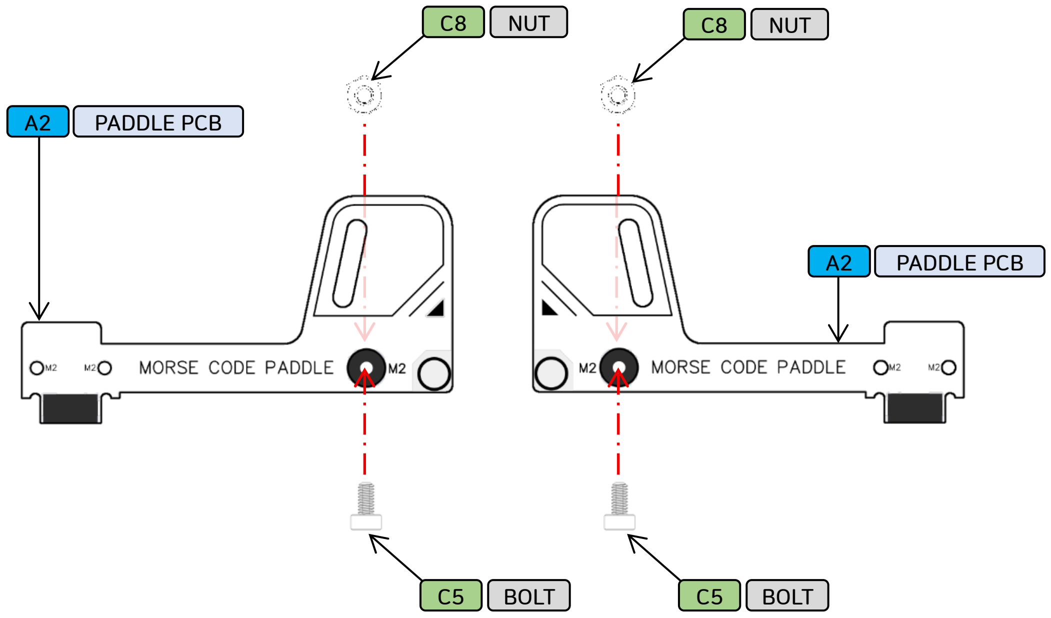


[PCB Component Layout]

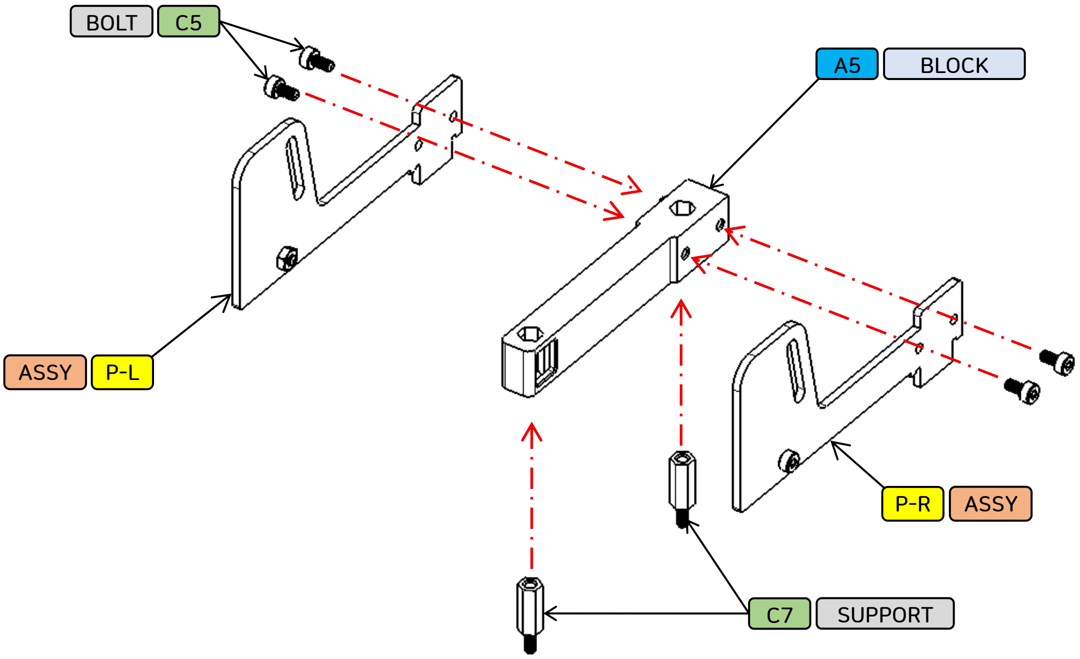
Paddle Assembly

After completing the PCB soldering, proceed to assemble the paddles, which are used for Morse code input. Follow the steps below to ensure each component is assembled in the correct orientation and proper order.

2



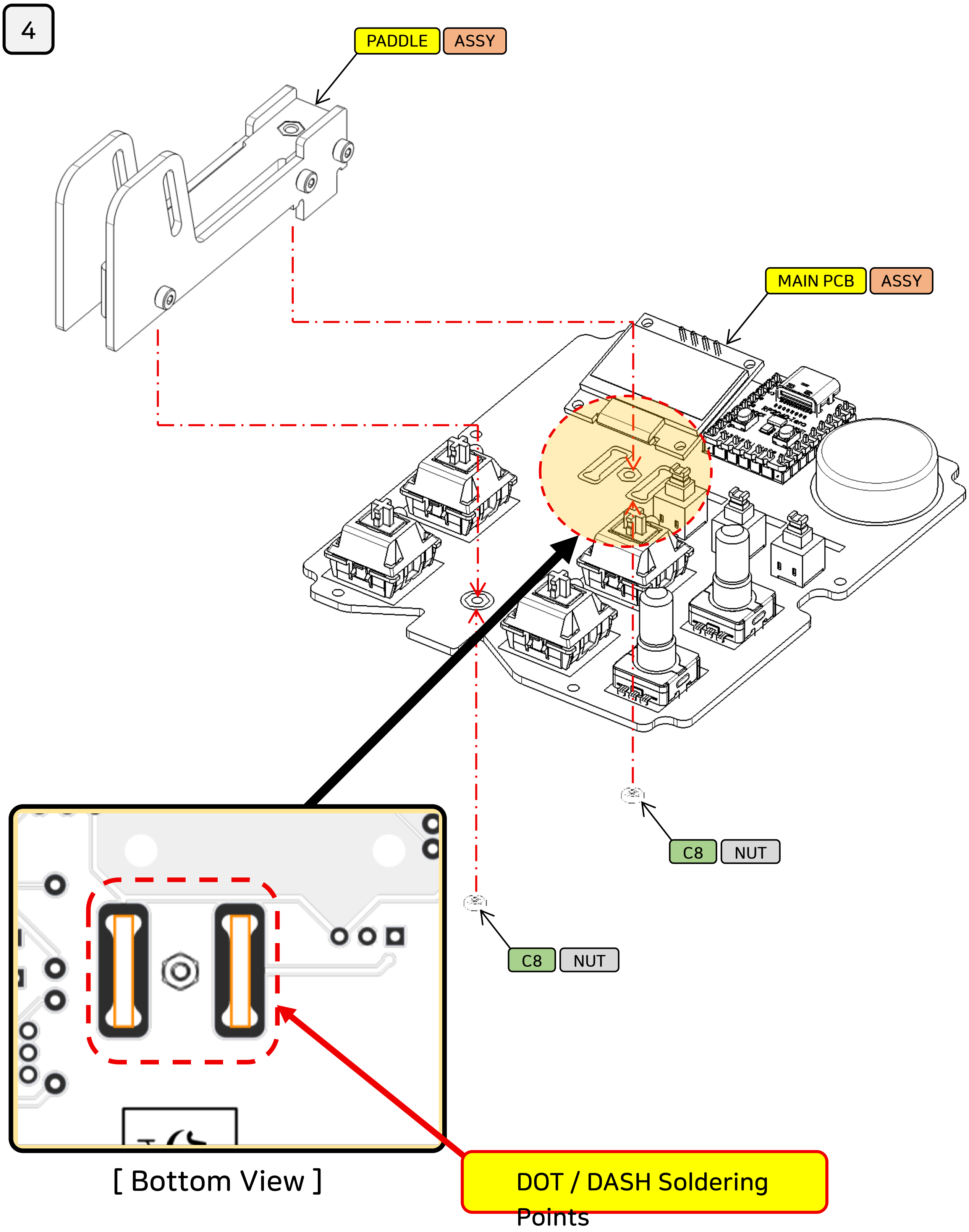
3



[Paddle Assembly]

Paddle-to-Main PCB Assembly

Align the assembled paddles with the mounting holes on the main PCB, then fasten them securely using nuts. Once fixed, solder the DOT and DASH connections to the PCB according to their designated pads.

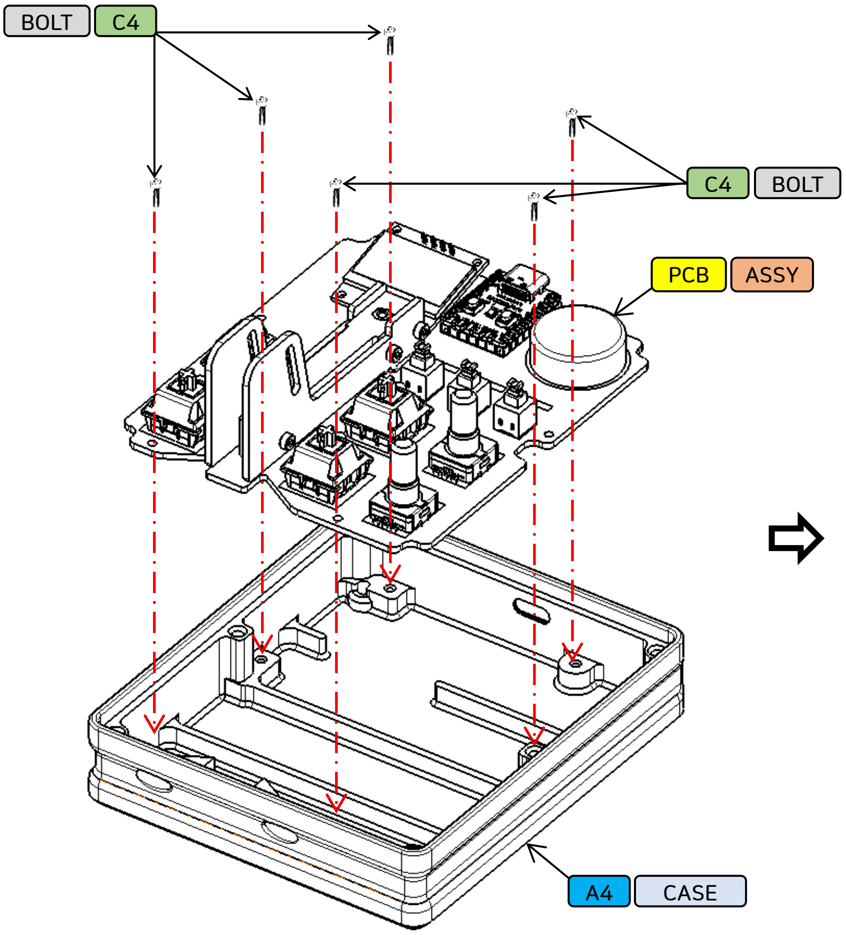


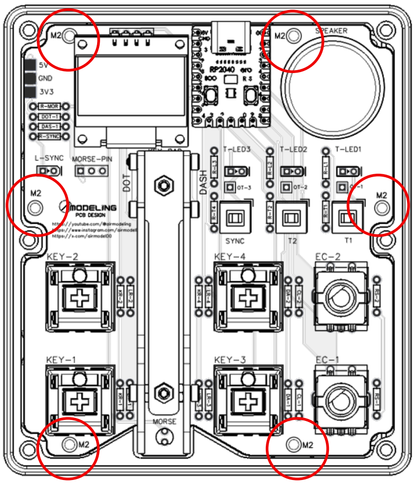
[Bottom View]

Case Assembly

After assembling the paddles and main PCB, mount the PCB board into the case and secure it using screws. Ensure the case orientation and screw hole positions are correct before tightening to achieve a precise fit.

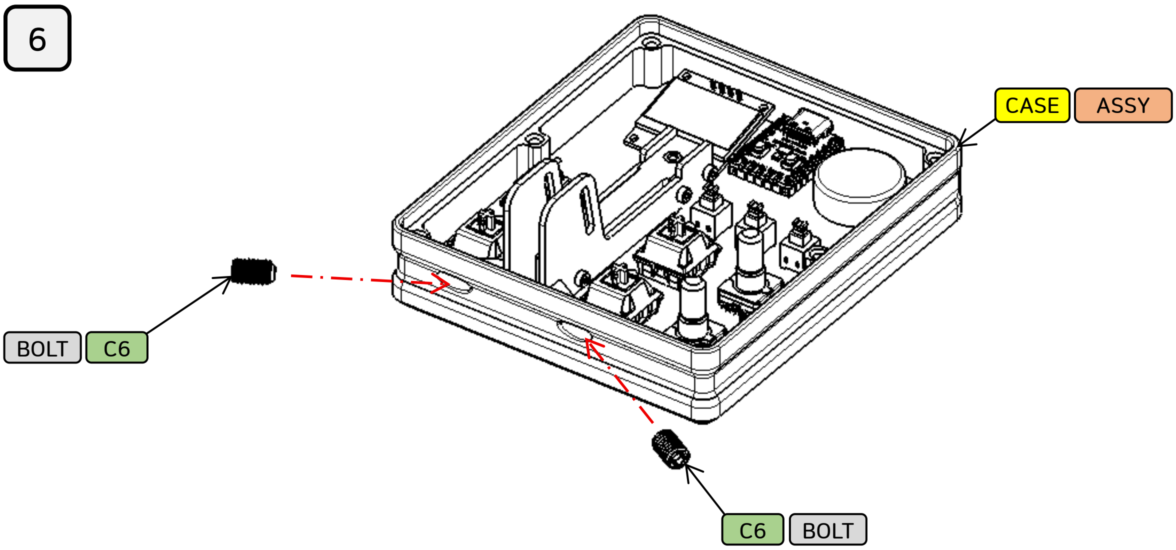
5





[Mounting Hole Positions]

6



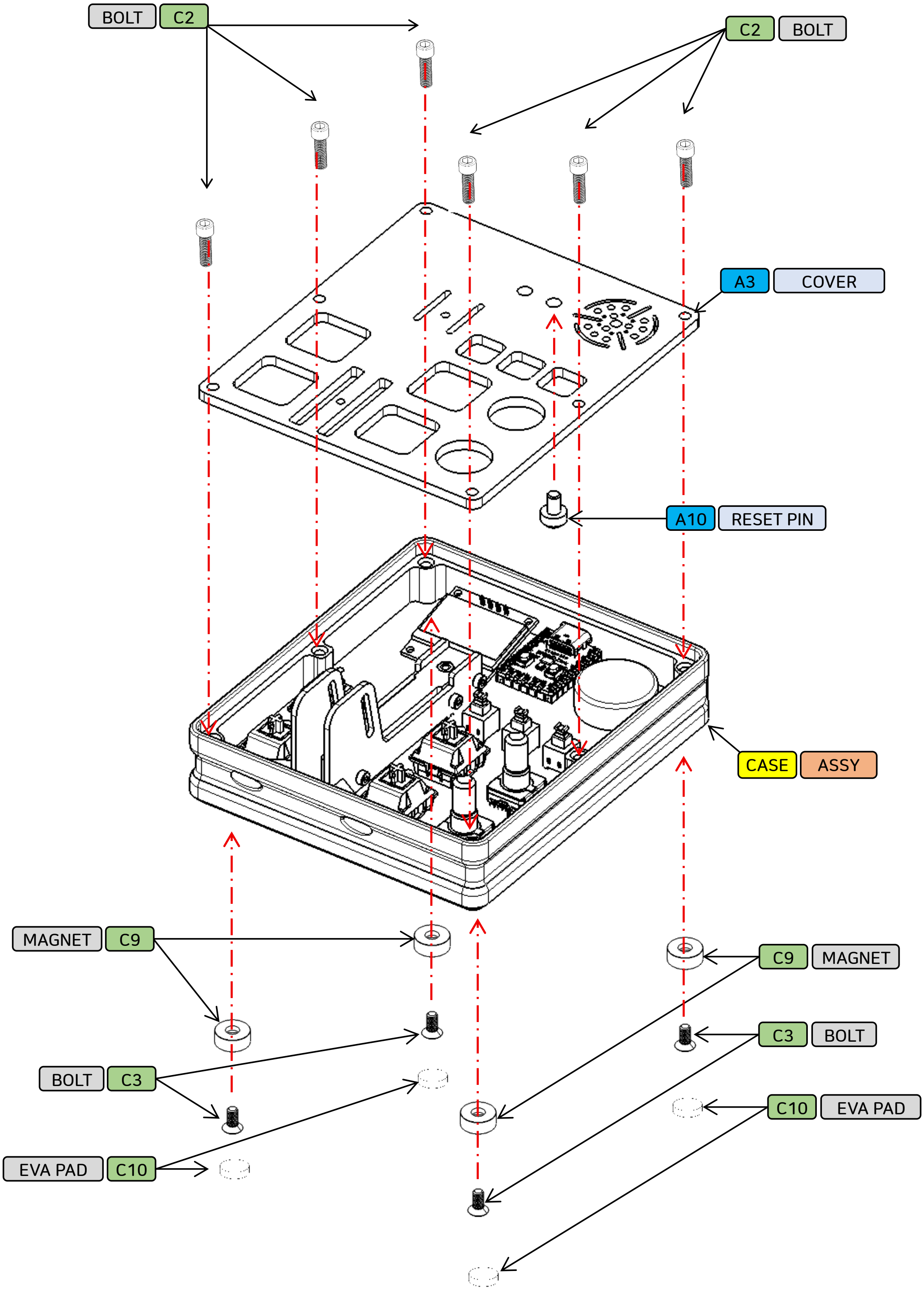
[Paddle Gap Adjustment Screw]

Cover and Fastening Parts Assembly

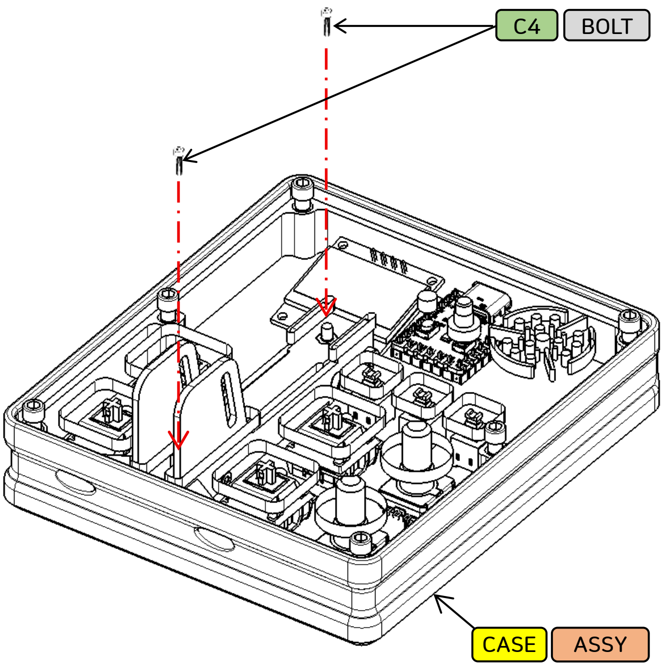
After completing the case assembly, insert the reset pin into its designated position between the case and acrylic cover.

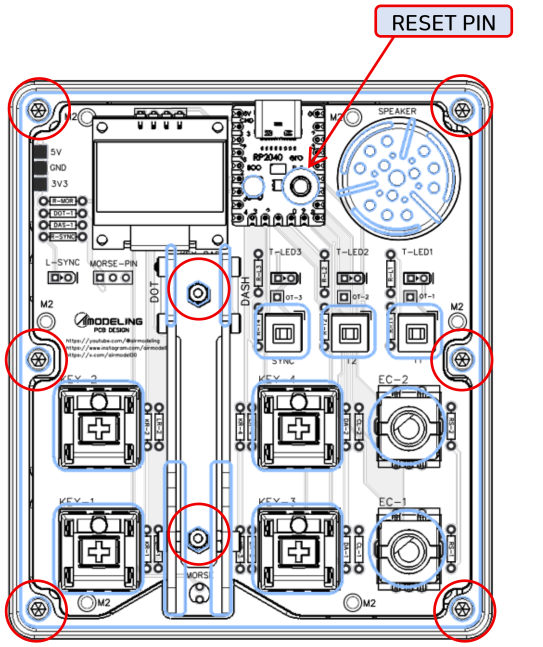
Once the reset pin is securely in place, fasten the acrylic cover using screws.
Next, attach the bottom magnets and EVA pads—first secure them with screws, then apply the pads firmly in place.

7



8



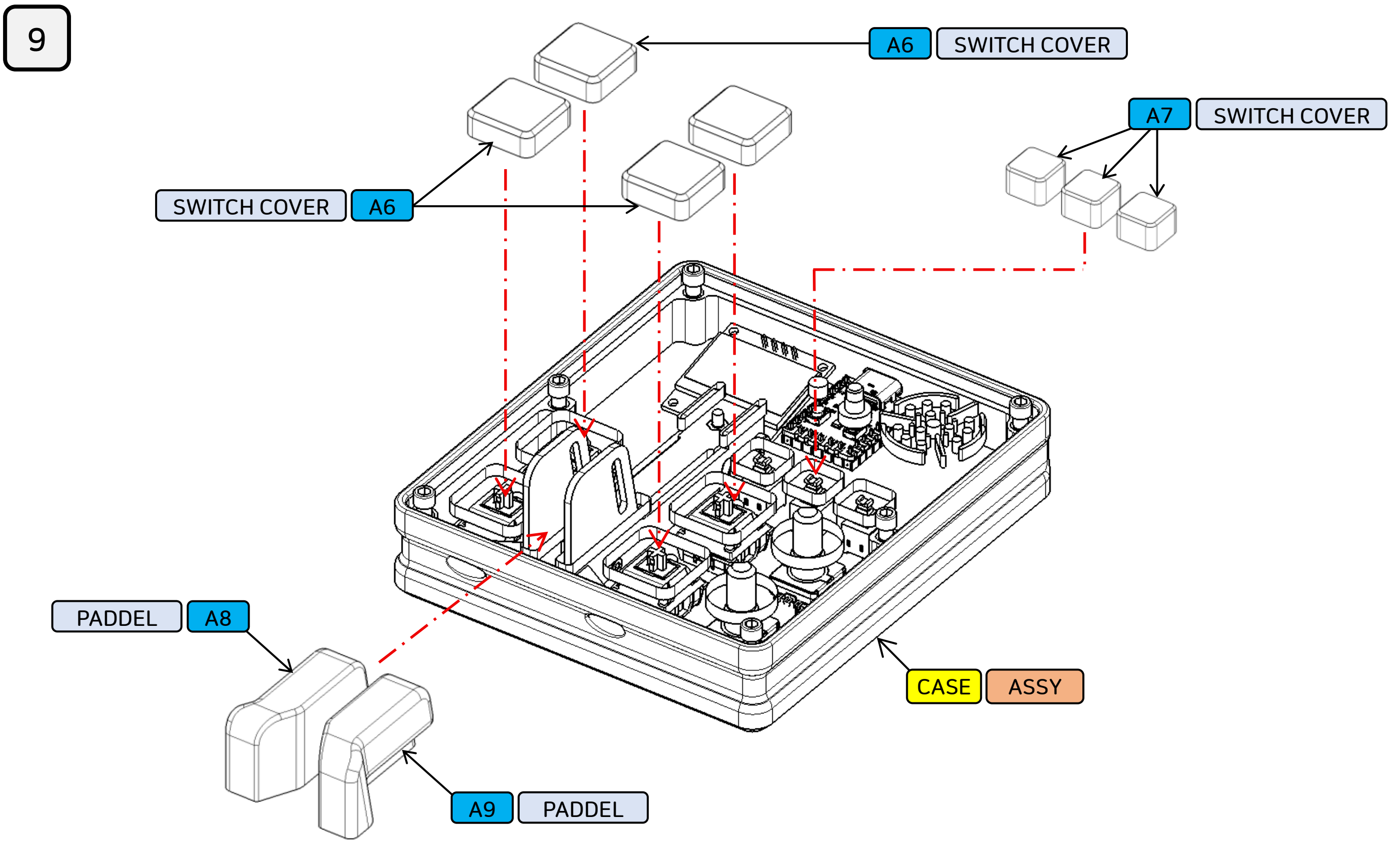


[Cover Hole Positions]

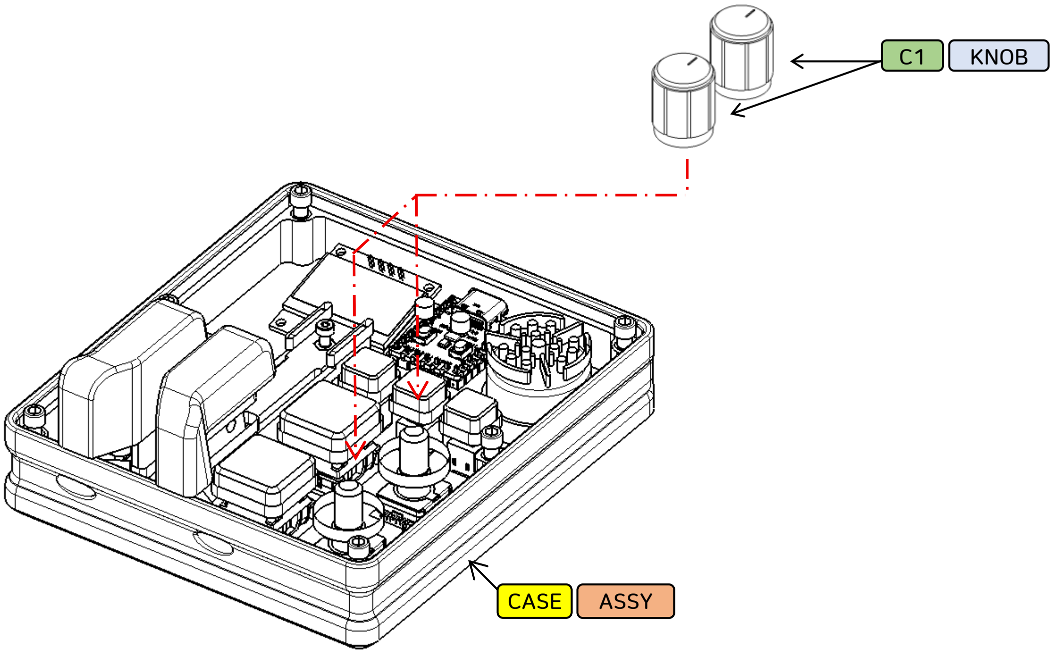
Switch and Paddle Cover Assembly

Install the switch cover, paddle cover, and knob in this step.
Align each part with its designated position and press it firmly until it is fully seated and securely fixed in place.

9

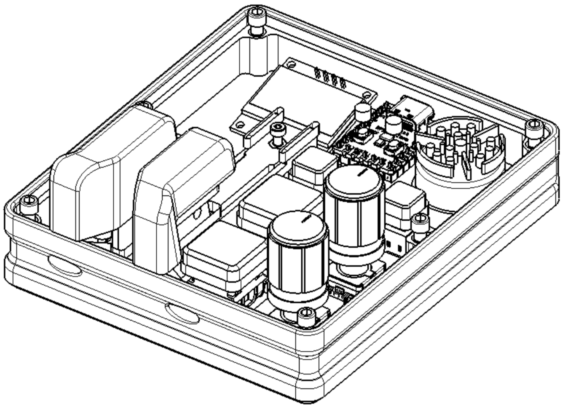


10

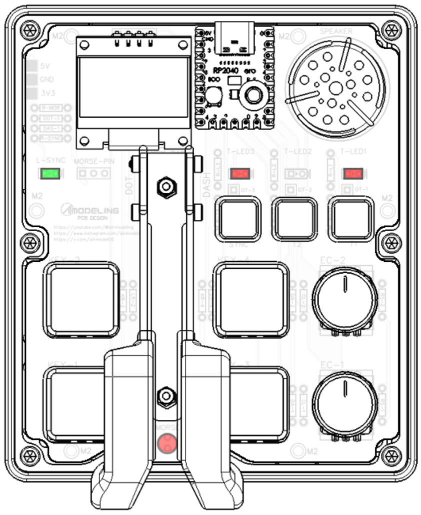


Assembly Completion

After completing the assembly, verify that all components are correctly installed.
Check that all fastenings are secure and the paddle and switch movements operate smoothly without resistance.



[ISO VIEW]



[TOP VIEW]

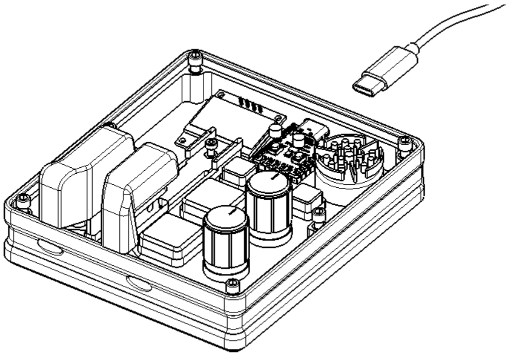
Firmware Installation and Code Upload

If the device passes inspection, connect the RP2040-Zero to your PC via USB cable.
Then, upload the prepared CircuitPython firmware (UF2 file) followed by the main program code, in the correct order.

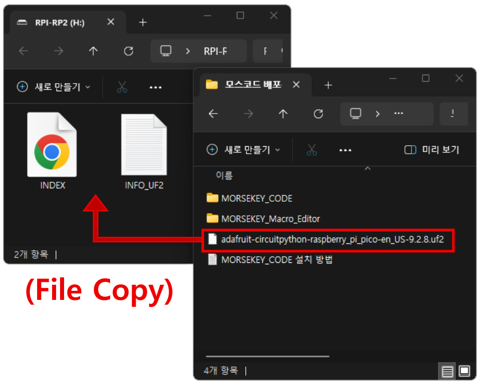
Installation Guide

1. Copy the CircuitPython firmware (UF2 file) to the drive recognized by the RP2040-Zero to begin the upload.
2. After the upload, a new drive named CIRCUITPY will be created automatically — copy the MORSEKEY_CODE folder and related files into this drive.
3. Once the upload is complete, press the reset button on the device to restart it.
4. If the OLED display shows the device interface, the upload has been completed successfully.

1 Connect the RP2040-Zero to your PC using a USB cable. Once connected, the device will be automatically recognized and appear as a removable drive on your computer.

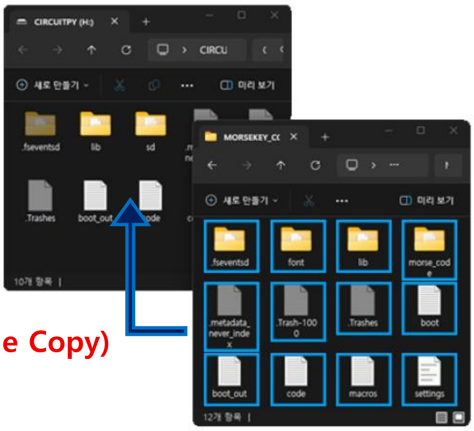


2 Copy the CircuitPython UF2 file to the recognized drive. After the copy is complete, the device will automatically restart, and a new drive named CIRCUITPY will be created.



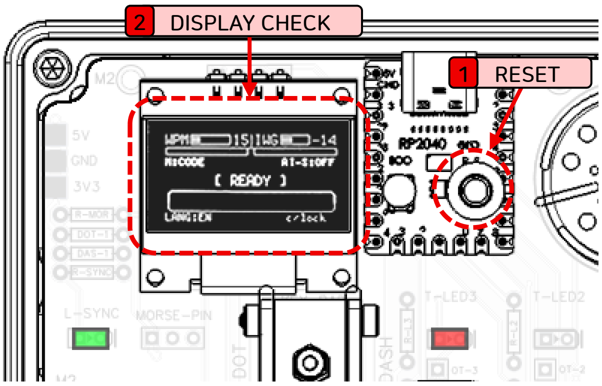
(File Copy)

3 Copy the MORSEKEY_CODE folder and its related files into the CIRCUITPY drive.



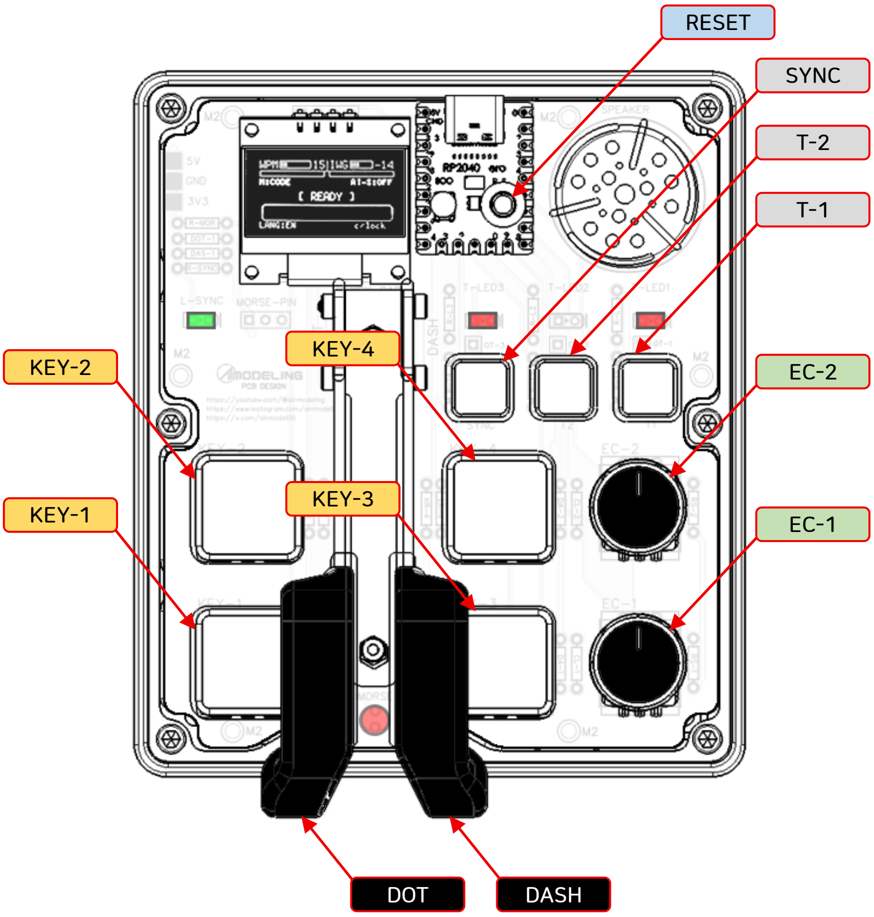
(File Copy)

4 After the upload is complete, press the reset button on the device to restart it.
If the OLED display shows the device interface, the upload has been completed successfully.



Operation Method

This section explains how to operate and use the MORSEKEY-A01 after installation.
Follow the instructions below to understand the basic controls, input functions, and device behavior during operation.



[Button Layout]

Morse Code Input

- Combine dots (·) and dashes (–) to create Morse code patterns, which are then converted into keyboard characters.
- Adjust WPM (Words Per Minute) and IWG (Inter-Word Gap) to control the input speed and confirmation delay.
- Press the SPACE key (KEY-4) before the IWG timer ends to immediately confirm and send the current character.

(※ When the macro mode is enabled, character input takes priority over macro execution.)

Language Setting (SYNC)

- Press the SYNC button to switch the Morse code language mode of the device.
- There are three modes: KR / EN / MO.
  - short press toggles between KR ↔ EN
  - long press (over 3 seconds) switches to MO mode.

  (※ MO mode is a direct input mode, where dots and dashes are entered exactly as Morse signals.)
- Press the button again to return to the previous language mode.

Auto Space (AT-S / T-1 Button)

● the T-1 button to activate the Auto Space (AT-S) function.

● After the IWG period ends, a space is automatically entered following the configured delay time (in milliseconds). (※ The AT-S delay can be adjusted in the options menu.)

● The AT-S function operates only in EN mode and applies to letters and numbers.

Macro Mode (T-2 Button)

● Press the T-2 button to activate Macro Mode (MACRO).

● Macro data is stored in the macro.json file, which can be freely edited on a PC.

● In this mode, pre-registered words or phrases can be entered automatically.

● Normal character input remains available even when Macro Mode is active.

(※ *If the entered Morse pattern matches a macro, the macro will take priority.*)

Basic Key Functions (KEY-1 ~ KEY-4)

● KEY-1 : Acts as CAPS LOCK, toggling between uppercase and lowercase letters.

(※ Allows Morse code input to distinguish letter cases.)

● KEY-2 : Switches language mode between Korean and English for both the device and PC.

(※ The current language mode (KR / EN) is displayed on the OLED screen.)

● KEY-3 : Functions as Confirm / Space / Enter.

- Press before IWG ends → immediately confirms the current character.
- When no pattern is entered → works as a Space key.
- Hold down and press DOT (·) or DASH (–) → acts as Enter.
- Character confirmation takes priority even when Macro Mode is active.

(※ The Space key does not conflict with the Auto Space (AT-S) function.)

● KEY-4 : Acts as Backspace, deleting the last entered character.

Rotary Encoder Knobs (EC-1 / EC-2)

● EC-1 Knob: Adjusts and saves WPM (Speed).

- Rotate to adjust between 5–30.
- Press to save all current settings — the OLED displays “SAVED” or “FAILED”.

● EC-2 Knob: Adjusts IWG (Inter-Word Gap) and opens the Options Menu.

- Rotate to fine-tune IWG within a –40 to +40 range.
- Press to enter [OPTION] mode, where additional settings can be adjusted.

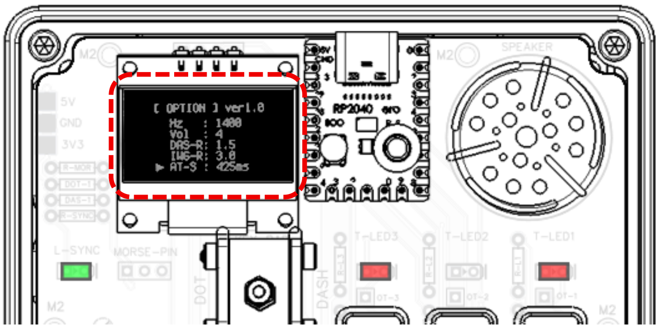
(※ Details are explained in a separate section.)

Reset Switch

● Used to restart the device.

Option Menu

Press the EC-2 knob to enter the [OPTION] menu, where you can adjust the following settings:

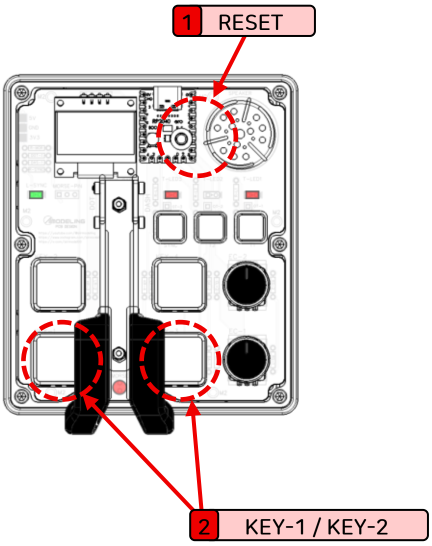


```
[ OPTION ] ver1.0
   Hz    : 1400
   Vol   : 4
   DAS-R : 1.5
   IWG-R : 3.0
 ▶ AT-S  : 425ms
```

- H Z : Adjusts the Morse tone frequency between 100–3000 Hz.
- V O L : Sets the Morse tone volume level from 0 to 5.
- DAS-R : Adjusts the dash (–) length from 1.5 to 4.0 in 0.5 increments.
- IWG-R : Adjusts the inter-character gap from 3.0 to 10.0 in 1.0 increments.
- AT-S : Sets the auto-space timing between 25 ms and 3000 ms, adjustable in 25 ms steps.
- DEFAULT : estores all settings to default values.
- SAVE : Saves the current settings. (※ Displays “DONE” if successful, or “FAIL” if saving fails.)
- BACK : Exits the option menu and returns to the main screen.

Entering Boot Mode (Edit Mode)

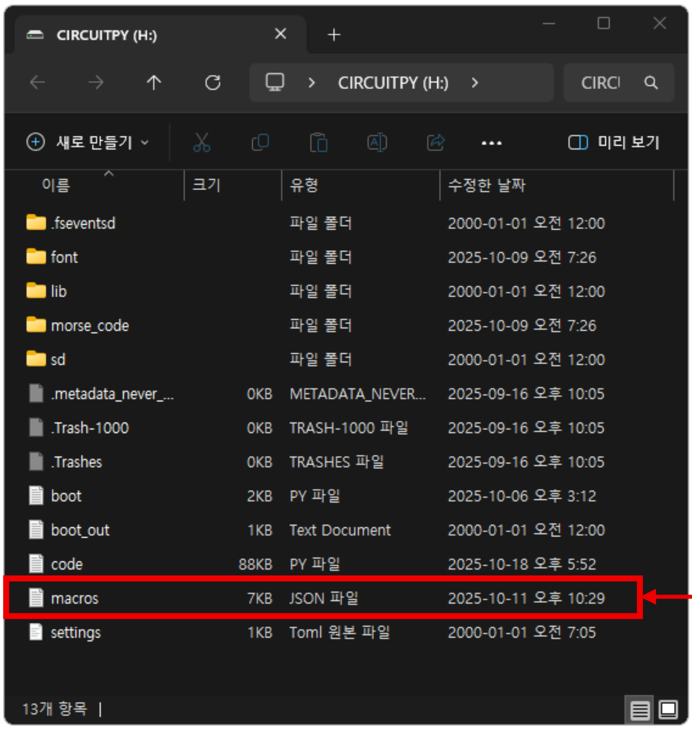
To replace or upload internal files such as the macro file, switch the device to Boot Mode (BOOT MODE). This mode allows the PC to recognize the device as a storage drive, enabling direct file modification or replacement.



How to Enter Boot Mode

1. Press the reset switch, then immediately hold down KEY-1 and KEY-2 together.
2. After a few seconds, a drive window will appear on the PC screen, indicating that Boot Mode has been successfully activated.

Macro File Replacement



이름	크기	유형	수정한 날짜
.fseventsd		파일 폴더	2000-01-01 오전 12:00
font		파일 폴더	2025-10-09 오전 7:26
lib		파일 폴더	2000-01-01 오전 12:00
morse_code		파일 폴더	2025-10-09 오전 7:26
sd		파일 폴더	2000-01-01 오전 12:00
.metadata_never_...	0KB	METADATA_NEVER...	2025-09-16 오후 10:05
.Trash-1000	0KB	TRASH-1000 파일	2025-09-16 오후 10:05
.Trashes	0KB	TRASHES 파일	2025-09-16 오후 10:05
boot	2KB	PY 파일	2025-10-06 오후 3:12
boot_out	1KB	Text Document	2000-01-01 오전 12:00
code	88KB	PY 파일	2025-10-18 오후 5:52
macros	7KB	JSON 파일	2025-10-11 오후 10:29
settings	1KB	Toml 원본 파일	2000-01-01 오전 7:05

Replace the file

1. Enter Boot Mode on the device.
2. The CIRCUITPY drive folder will automatically open on your PC.
3. Replace the existing macro.json file by either deleting or overwriting it with the new version.
4. the file transfer is complete, press the reset switch to restart the device.

(※ The macro file can be edited using the dedicated MORSEKEY_Macro_Editor program.)

Closing Remarks

The MORSEKEY-A01 was designed with the goal of creating a Morse code keyboard that anyone can build.
It is more than just an input device — it is an opportunity to learn, assemble, and enjoy the process of electronic creation.

I hope this project inspires your own making, learning, and new ideas.
Finally, I would like to express my sincere gratitude to everyone who has downloaded and built this project.

Thank you.

■ Disclaimer and Copyright Notice

Project Name : MORSEKEY-A01 ver1.0
Creator : Airmodeling

This project is an open-source hardware developed based on Adafruit CircuitPython (MIT License).

The MORSEKEY-A01 may be freely used for educational, hobby, or personal research purposes.
All related resources — including circuit schematics, 3D data, firmware, manuals, and design files — are provided in an open format.

■ Terms of Use and Distribution

1. All materials may be freely used only for non-commercial and non-profit purposes.
2. Commercial sales, redistribution of modified versions, or resale of derivative works are strictly prohibited.
3. Personal modifications for learning are allowed; however, redistribution must clearly credit the original creator (Airmodeling).
4. The creator assumes no responsibility for any problems, errors, damage, or hardware failures that may occur during use of the project.

■ Reference Resources

GitHub (Source Code & Firmware)
https://github.com/Airmodeling/MORSEKEY-A01

OSHWLab (Schematics & 3D Models)
https://oshwlab.com/kimgx05/morsekey-a01

YouTube (AIRMODELING)
https://www.youtube.com/@airmodeling

Instagram
https://www.instagram.com/airmodel00/

X (Twitter)
https://x.com/airmodel00

This project follows the open-source philosophy, encouraging everyone to learn, share, and grow together.

Contact: airmodel00@gmail.com