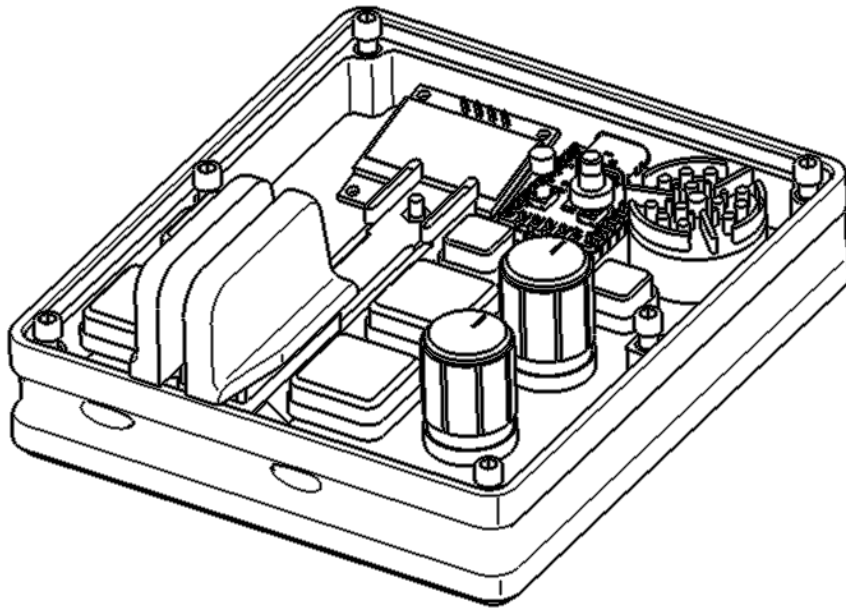


MORSEKEY-A01

(모스부호 키보드 입력장치)

조립 및 사용 매뉴얼
(한글판)



■ 제품 소개

MORSEKEY-A01은 RP2040-Zero 보드를 기반으로 한 모스부호 키보드 입력장치입니다.
한글과 영어를 모두 지원하며, 모스부호를 타이핑 신호로 변환하여 PC 입력장치로 동작합니다.

매크로 기능을 통해 자주 사용하는 문장을 자동 입력할 수 있으며, OLED 디스플레이를 통해 현재 언어, 속도(WPM), 간격(IWG), 사운드 설정 등 실시간으로 확인할 수 있습니다.

또한 누구나 쉽게 제작할 수 있도록 회로와 구조를 최대한 단순하게 설계하였으며,
조립 또한 기본적인 납땜만으로 완성할 수 있도록 구성되었습니다.

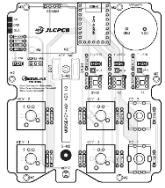
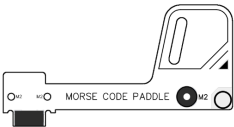
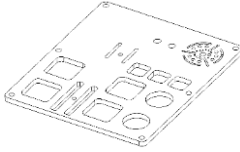
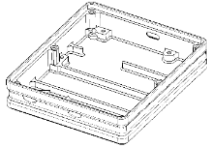
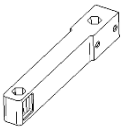





■ 가공 부품

MORSEKEY-A01 가공 부품들은 별도로 제작(가공)해야 하는 항목입니다.
PCB, 3D프린팅, 아크릴 커버는 각각 개별적으로 출력 또는 주문 제작이 필요합니다.

회로도 및 3D 도면 파일은 OSHWLab 공식 페이지에서 확인할 수 있습니다.

<https://oshwlab.com/kimgx05/morsekey-a01>

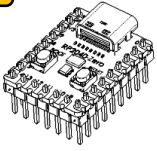
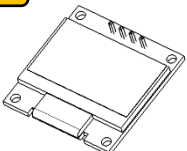
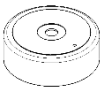
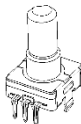








- P C B : 거버(Gerber) 데이터를 이용해 JLCPCB에서 직접 주문 가능합니다.
- 3D 프린팅 부품 : STL 파일을 이용해 출력합니다.
- 아크릴 부품 : 2D DX 데이터를 이용해 레이저 커팅합니다.

<div>A1</div>  <p>MORSEKEY MAIN PCB (1EA)</p> <p>JLCPCB PCB 주문 제작 파일: Gerber_MORSEKEY MAIN PCB.zip</p>	<div>A2</div>  <p>MORSE CODE PADDLE (1EA)</p> <p>JLCPCB PCB 주문 제작 파일: Gerber_MORSEKEY PADDLE PCB.zip</p>	<div>A3</div>  <p>MORSEKEY COVER (1EA)</p> <p>3T 아크릴 레이저 주문 제작 파일: MORSEKEY_COVER.dwg</p>	<div>A4</div>  <p>MORSEKEY CASE (1EA)</p> <p>3D프린팅 출력물 파일: MORSEKEY_CASE.stl</p>	
<div>A5</div>  <p>MORSEKEY BLOCK (1EA)</p> <p>파일: MORSEKEY_BLOCK.stl</p>	<div>A6</div>  <p>MORSEKEY KEY C1 (4EA)</p> <p>파일: MORSEKEY_KEY_C1.stl</p>	<div>A7</div>  <p>MORSEKEY KEY C2 (3EA)</p> <p>파일: MORSEKEY_KEY_C2.stl</p>	<div>A8</div>  <p>MORSEKEY PADDLE L (1EA)</p> <p>파일: MORSEKEY_PADDLE_L.stl</p>	<div>A9</div>  <p>MORSEKEY PADDLE R (1EA)</p> <p>파일: MORSEKEY_PADDLE_R.stl</p>
<div>A10</div>  <p>MORSEKEY RESET PIN (2EA)</p> <p>파일: MORSEKEY_RESET_PIN.stl</p>				

■ 전자 부품

MORSEKEY-A01 장치는 RP2040-Zero 를 기반으로 동작하며,
입력 신호 처리와 동작 제어에 필요한 핵심 전자 부품들로 구성되어 있습니다.

아래 표는 제작에 필요한 주요 전자 부품 목록입니다.

B1  RP2040-Zero (1EA)	B2  SSD1306 OLED (4Pin, 128x64) (1EA)	B3  스피커 (YZ2308) (1EA)	B4  로터리 엔코더 (Plum EC-11) (2EA)	B5  기계식 스위치 (Cherry MX) (4EA)	B6  푸시 스위치 (7x7 LOCKING) (2EA)
B7  푸시 스위치 (7x7 UNLOCK)) (1EA)	B8  LED-R (square LED) (7EA)	B9  LED-G (square LED) (1EA)	B11  LED-R (5mm LED) (1EA)	B12  저항 (220Ω) (9EA)	B13  저항 (10kΩ) (15EA)

■ 체결 부품

MORSEKEY-A01 조립 시 필요한 볼트, 너트, 자석, 고무발 등
외형 결합과 고정을 위해 사용되는 부품들을 정리한 목록입니다.

C1  알루미늄 노브 (14 x 16mm) (2EA)	C2  렌치 볼트 (M3 x 10mm) (6EA)	C3  접시 렌치 볼트 (M3 x 5mm) (4EA)	C4  접시 렌치 볼트 (M2 x 8mm) (8EA)	C5  접시 렌치 볼트 (M2 x 4mm) (6EA)	C6  무드 렌치 볼트 (M3 x 8mm) (2EA)
C7  지지대 볼트 (M2 x 10mm) (2EA)	C8  M2 육각 너트 (2EA)	C9  원형 사라자석 (6.5 x 4 x 4mm) (4EA)	C10  EVA 고무발 (8mm x 2T) (8EA)		

■ 필요 도구

조립 과정에서 필요한 기본 도구 목록입니다.
정확한 체결과 납땜을 위해 각 도구를 준비해 주세요.



■ 펌웨어 및 스크립트

공식 GitHub 저장소 (MORSEKEY-A01 Project):
<https://github.com/Airmodeling/MORSEKEY-A01>

MORSEKEY-A01 은 CircuitPython 기반으로 동작하며
펌웨어와 매크로 편집 프로그램을 다운로드하여 사용하면 됩니다.

아래는 기본 프로그램 파일 구성입니다.

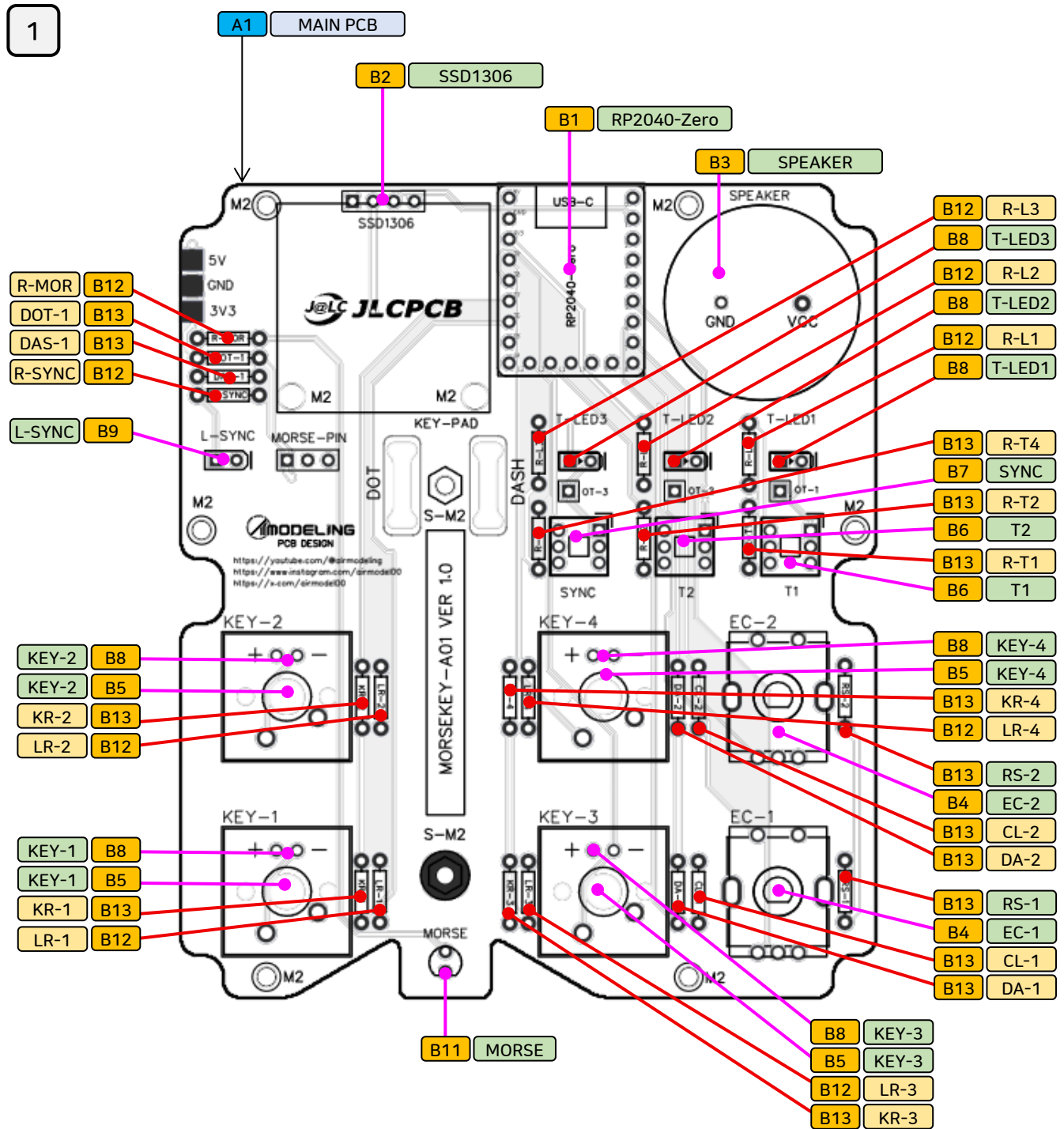
- MORSEKEY_CODE → 메인 프로그램 코드 (CircuitPython)
- MORSEKEY_Macro_Editor → 매크로 파일 편집용 PC 프로그램
- adafruit-circuitpython-raspberry_pi_pico-en_US-9.2.8 → RP2040-Zero 용 CircuitPython 펌웨어

■ PCB 납땜 작업

이 단계에서는 MORSEKEY-A01 의 전자 부품을 PCB 에 납땜하여 작업합니다.

작업 전, 부품의 방향과 극성을 반드시 확인하고

무연납 을 사용하여 깔끔하고 안정적인 납땜 품질을 유지하세요.

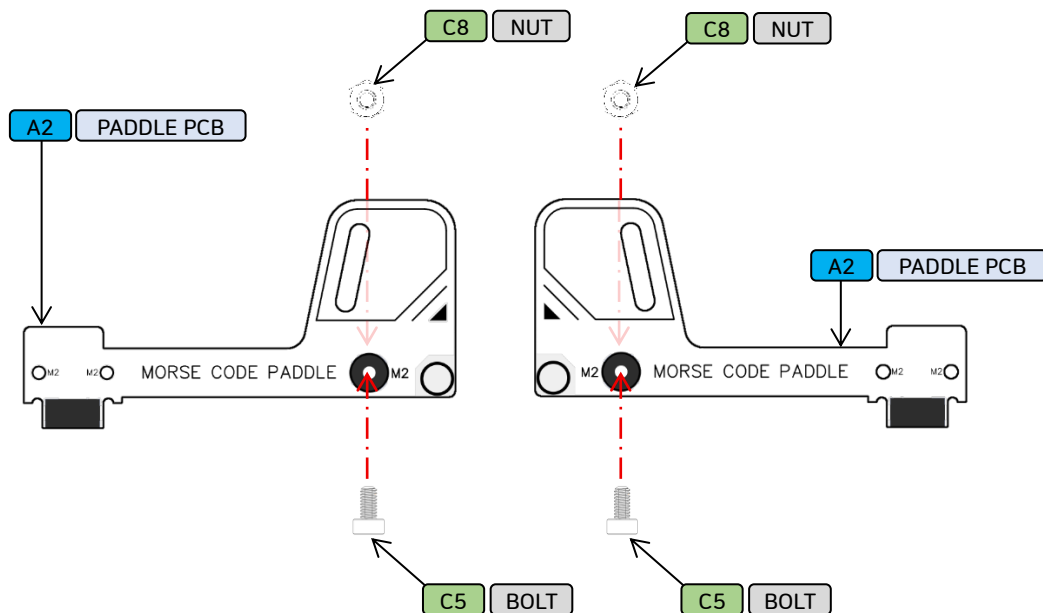


[PCB 부품 배치도]

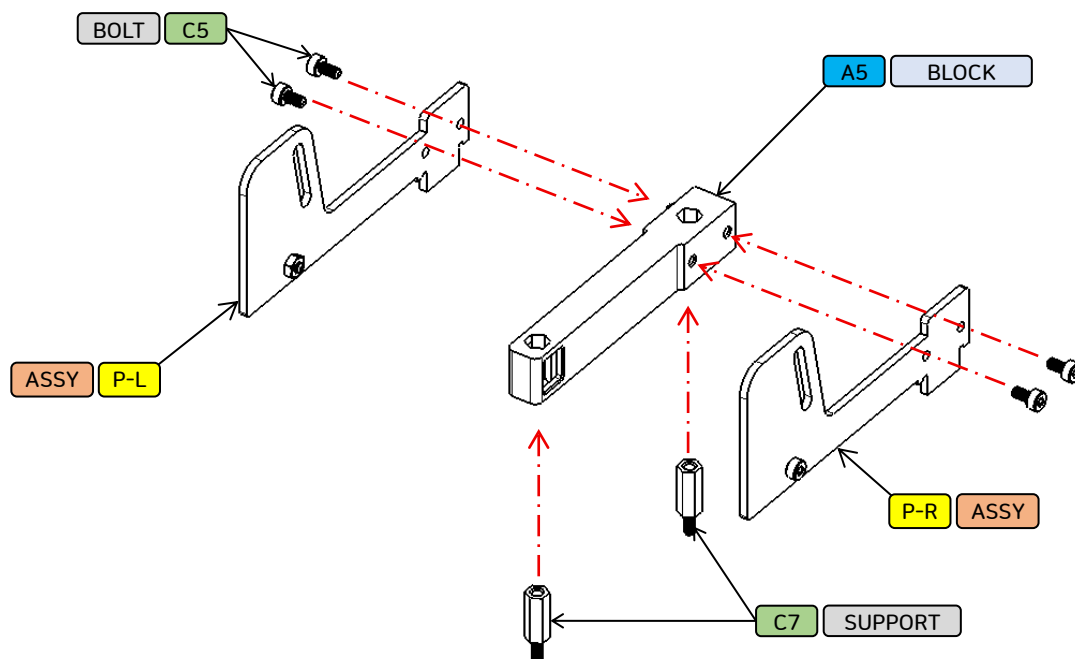
■ 패들 조립

PCB 납땜 작업이 끝나면, 다음은 모스부호 입력에 필요한 패들(Paddle)을 조립하는 단계입니다.
아래 순서를 참고하여 부품을 정확한 방향으로 조립해 주세요.

2



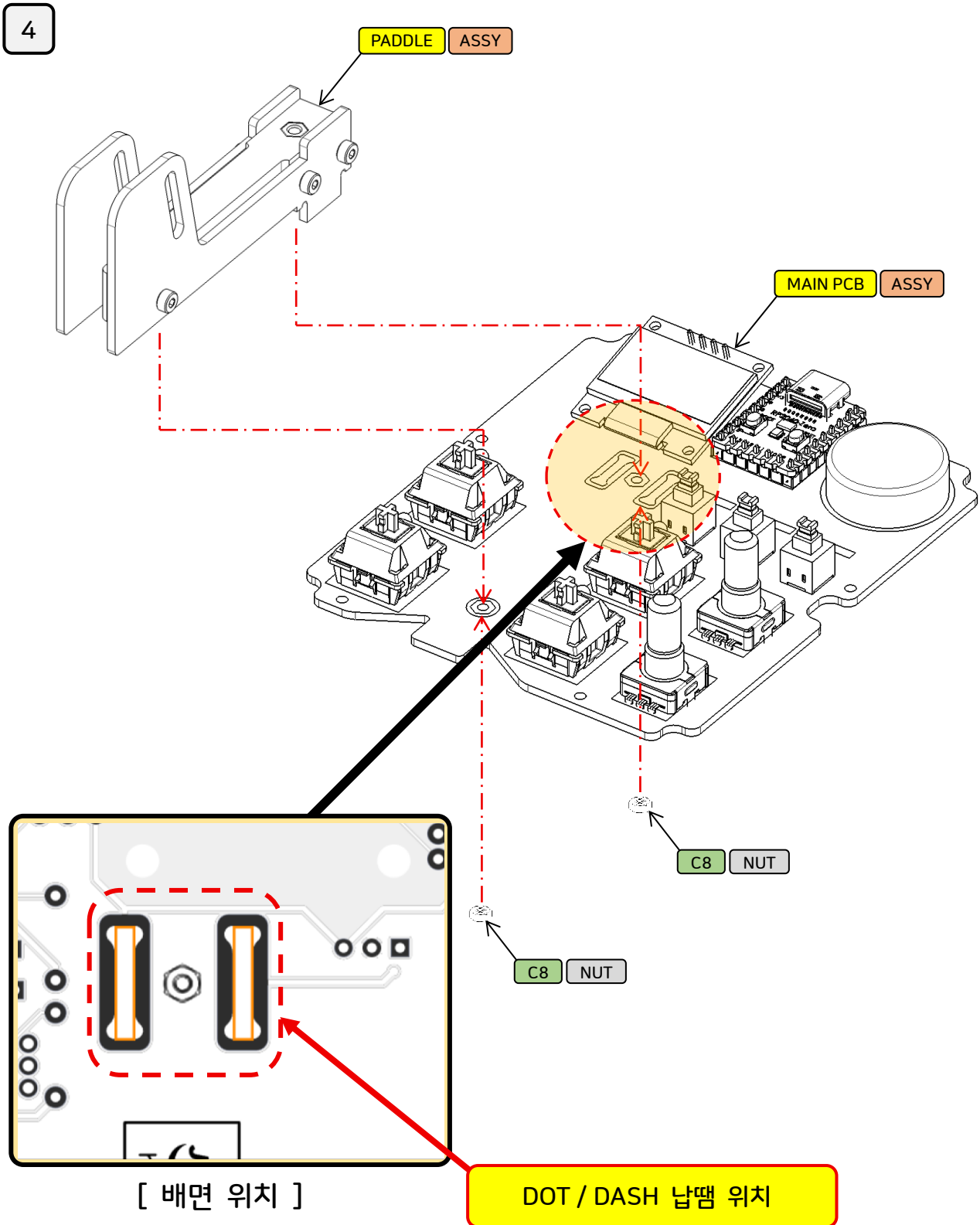
3



[패들 조립도]

■ 패들-메인 PCB 조립

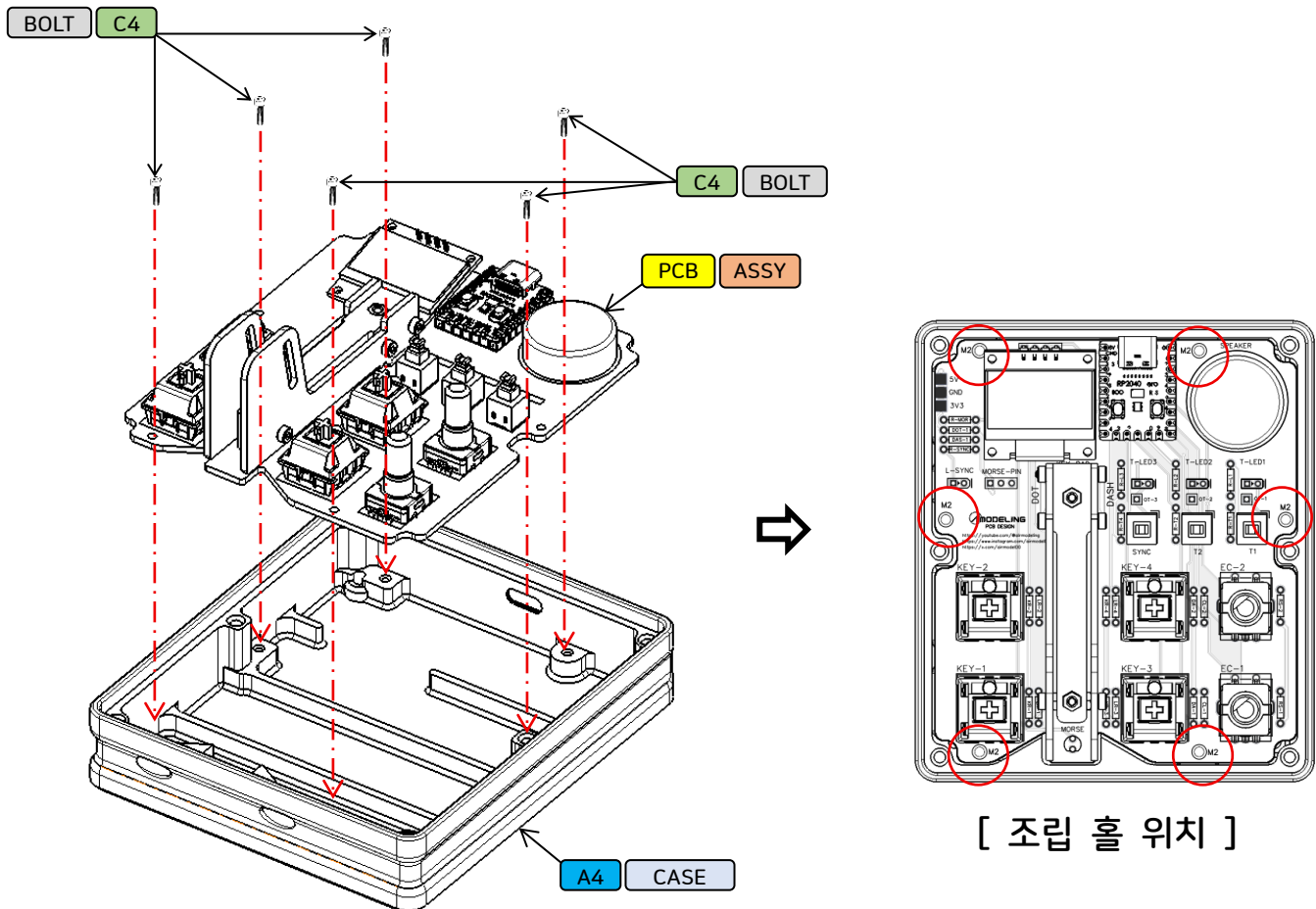
조립한 패들과 메인 PCB 를 구멍 위치에 맞춰 끼우고 너트로 체결합니다.
고정이 완료되면 점(DOT)과 선(DASH)를 PCB 에 납땜합니다.



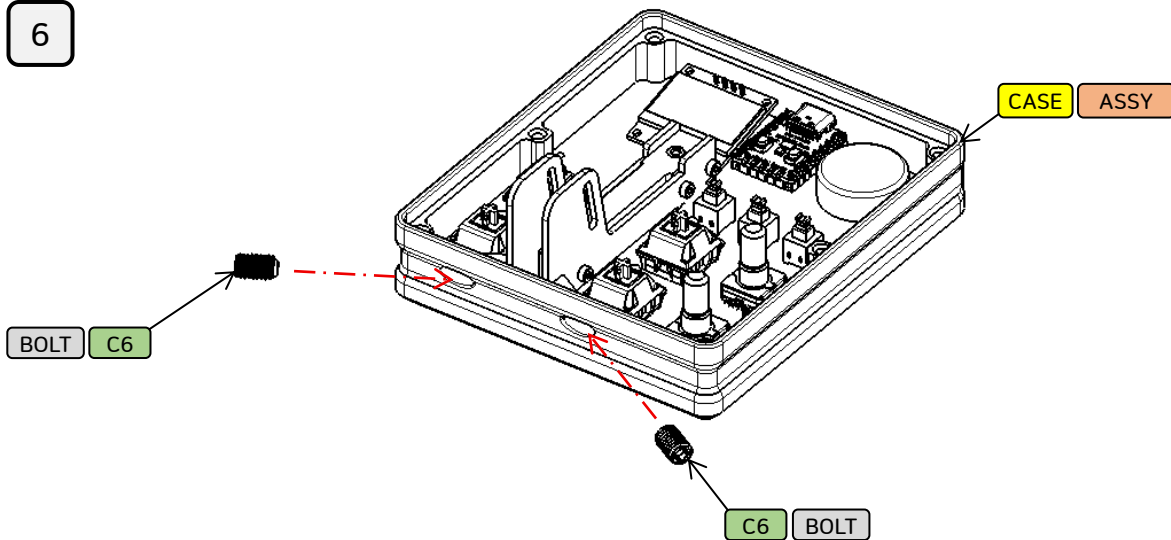
■ 케이스 결합

패들과 메인 PCB의 조립이 완료되면, 다음으로 PCB 보드를 케이스에 장착하고 나사로 고정합니다.
케이스의 방향과 나사 위치를 확인하여 정확히 결합하세요.

5



6



[패들 간극 조절 나사]

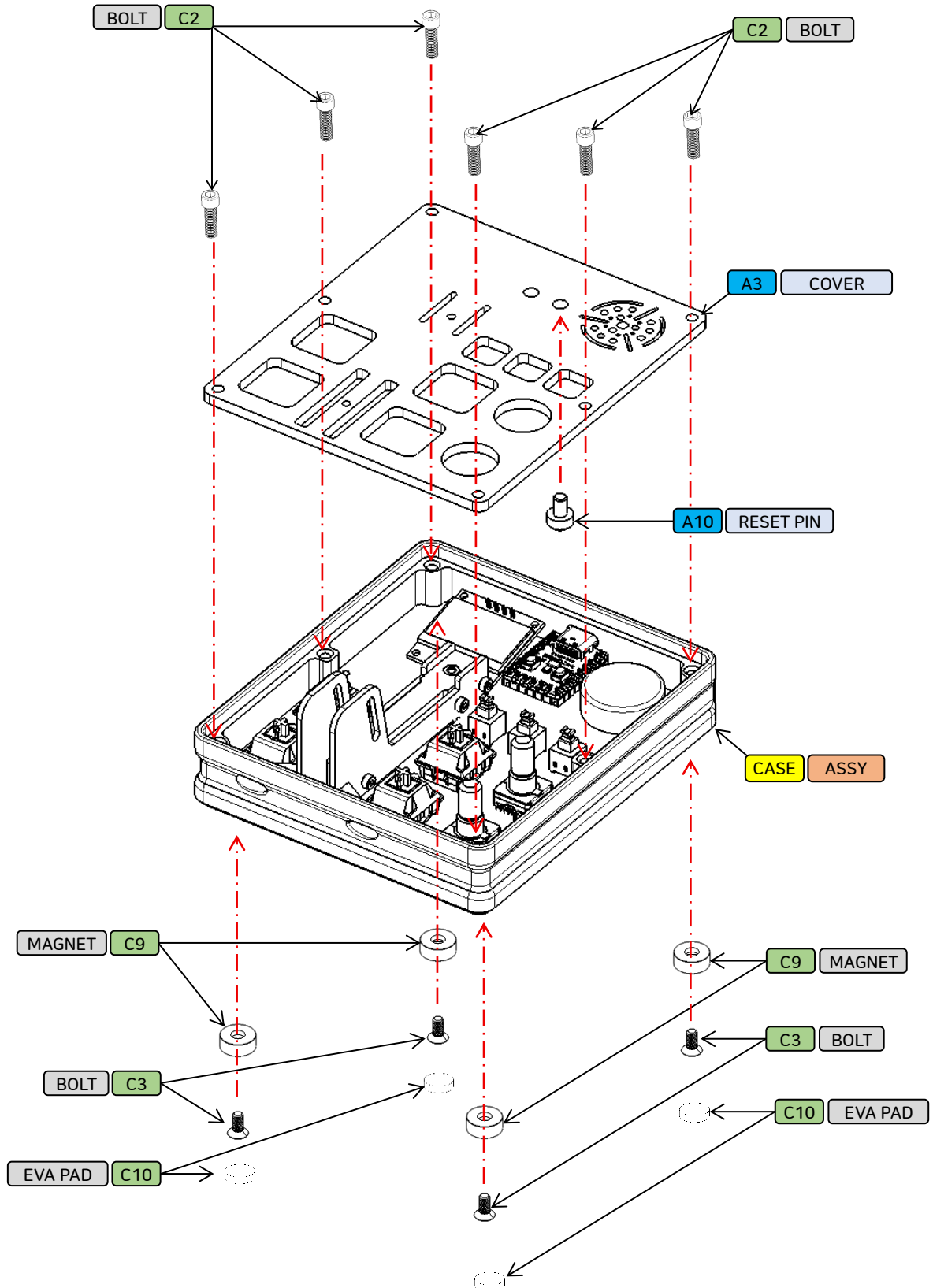
■ 커버 및 고정 부품 조립

케이스 조립이 완료되면, 아크릴 커버를 장착하기 전에 리셋 핀을 케이스와 커버 사이의 지정된 위치에 끼웁니다.

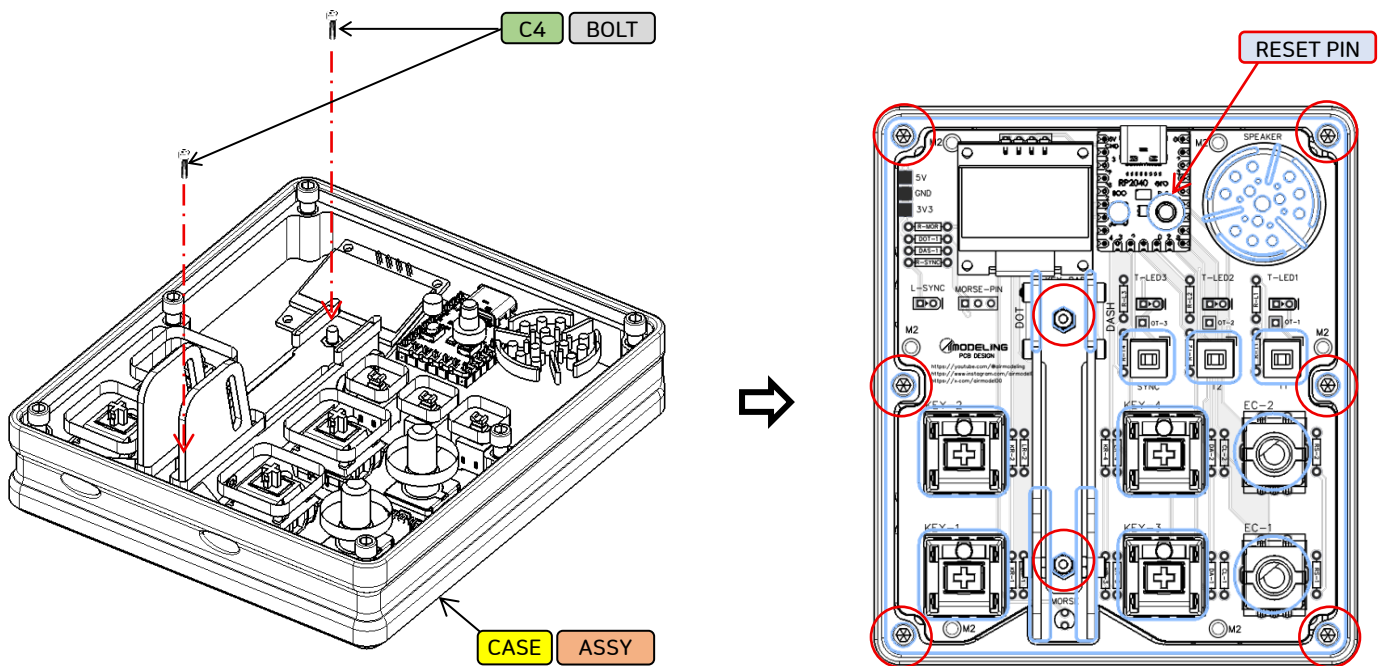
리셋 핀을 올바르게 고정한 후, 아크릴 커버를 나사로 체결합니다.

이어서 하단 자석과 EVA 패드를 나사로 먼저 고정 한 뒤, 패드를 부착합니다.

7



8



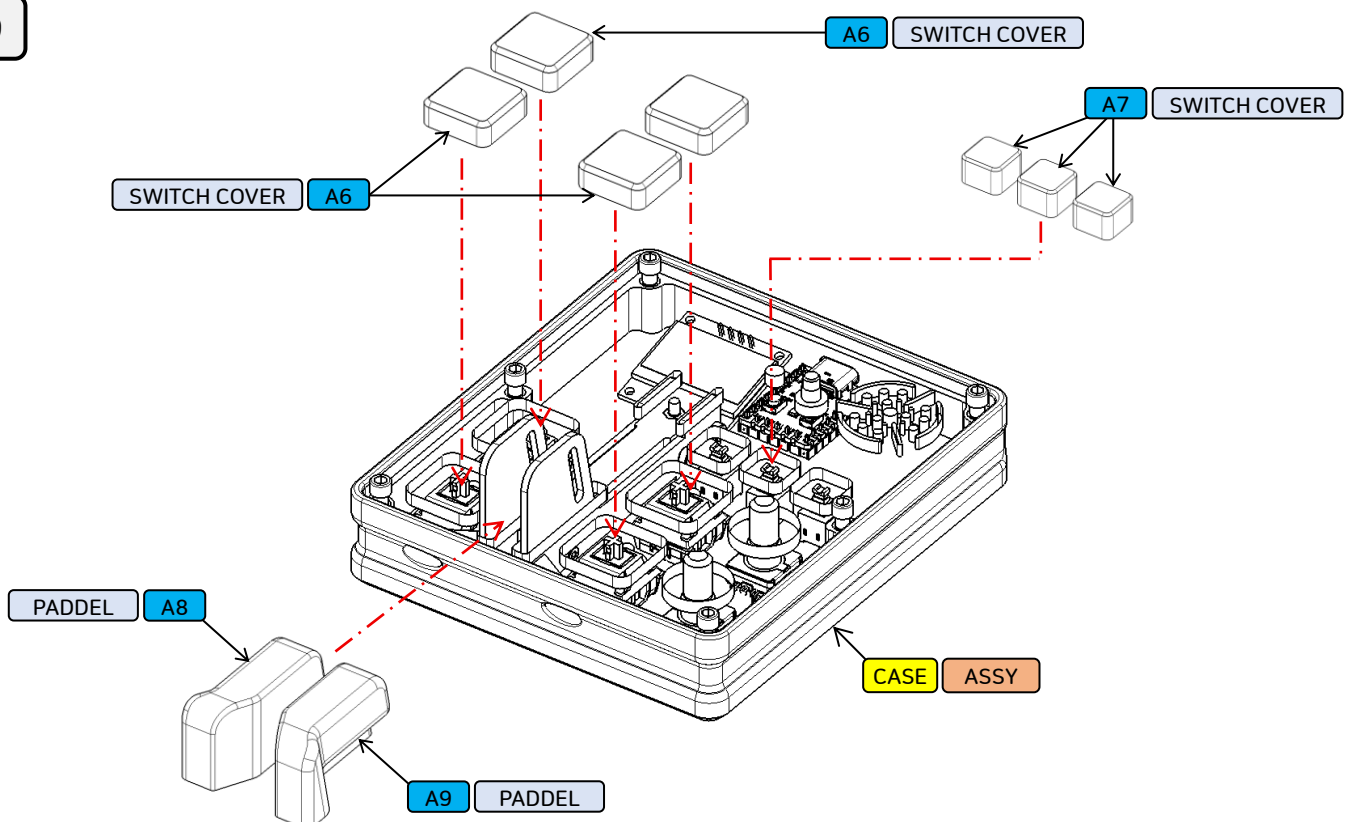
[커버 홀 위치]

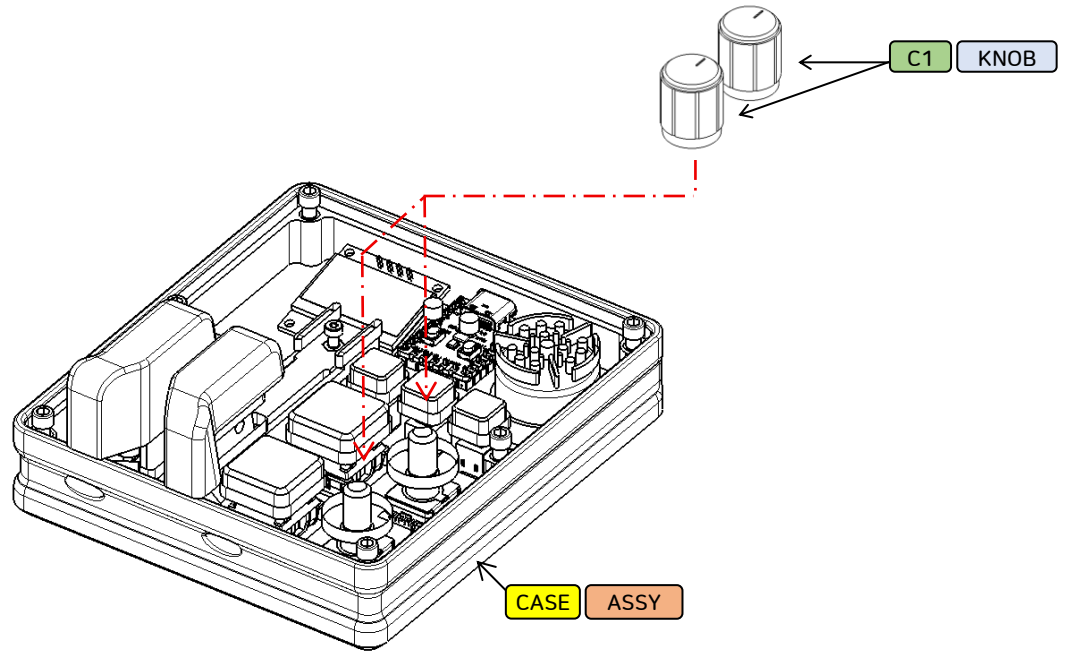
스위치 및 패들 커버 조립

스위치 커버, 패들 커버, 그리고 노브를 장착합니다.

각 부품을 지정된 위치에 맞춰 끼우고 끝까지 밀어 넣어 정확히 고정되었는지 확인하세요.

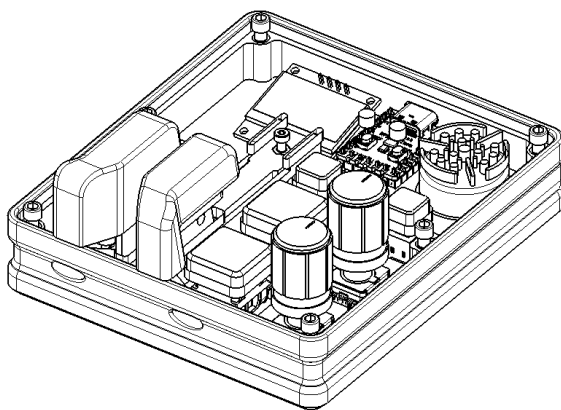
9



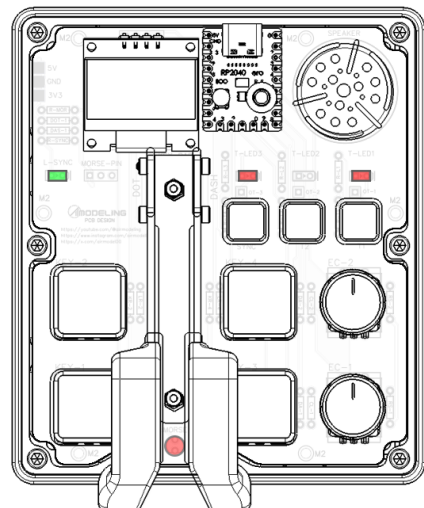


■ 조립 완료

조립이 완료되면, 모든 부품이 올바르게 장착되었는지 확인합니다.
체결 상태와 움직임이 부드러운지 점검하세요.



[ISO VIEW]



[TOP VIEW]

■ 펌웨어 설치 및 코드 업로드

장치에 이상이 없으면, USB 케이블을 이용해 RP2040-Zero 를 PC 에 연결합니다.

그 후, 준비해둔 CircuitPython 펌웨어(UF2 파일) 과 프로그램 코드를 순서에 맞춰 업로드합니다.

■ 설치 방법

1. CircuitPython 펌웨어(UF2 파일) 을 RP2040-Zero 가 인식된 드라이브에 복사하여 업로드합니다.
2. 업로드 후 자동으로 생성되는 CIRCUITPY 드라이브에 MORSEKEY_CODE 폴더와 관련 파일을 복사합니다.
3. 업로드가 완료되면 장치의 리셋 버튼을 눌러 재시작합니다.
4. OLED 화면에 장치 화면이 표시되면 업로드가 정상적으로 완료된 것입니다.

1 RP2040-Zero를 PC에 USB 케이블로 연결합니다.
연결하면 장치가 자동으로 인식되어 드라이브로 표시됩니다.

2 인식된 드라이브에 CircuitPython UF2 파일을 복사합니다.
복사후 자동으로 재시작되고, 새로운 드라이브(CIRCUITPY)가 생성됩니다.

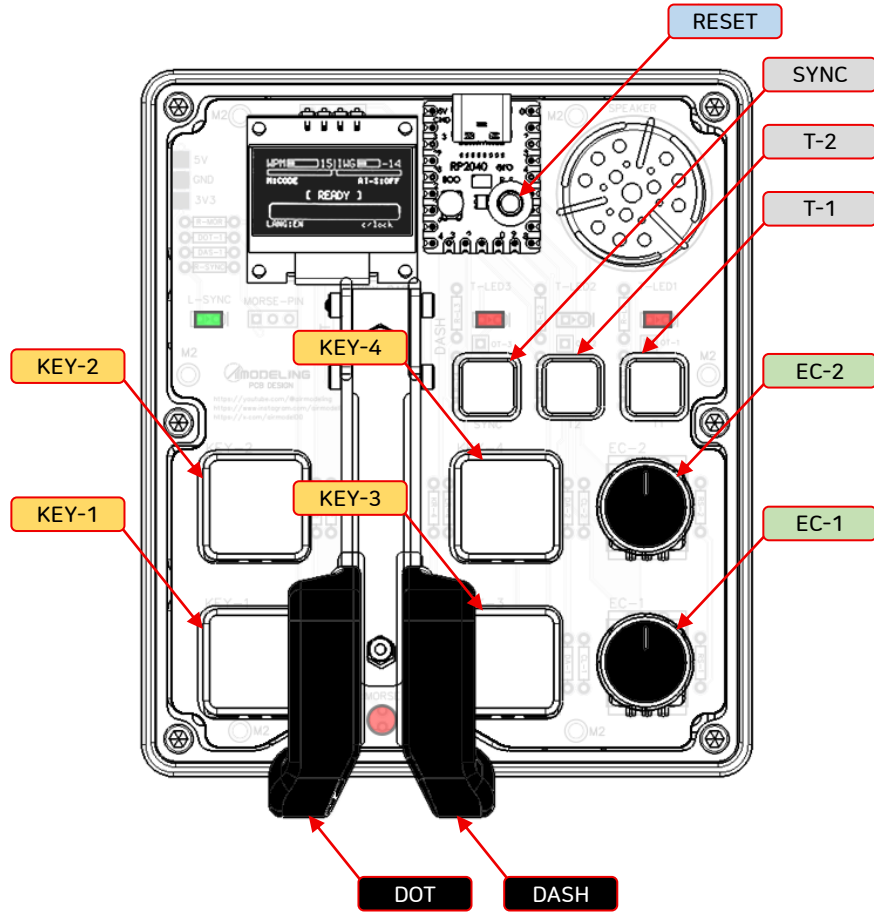
3 CIRCUITPY 드라이브에 MORSEKEY_CODE 폴더에 관련 파일을 복사합니다.

4 업로드 완료 후, 장치의 리셋 버튼을 눌러 재시작합니다.
OLED 화면에 장치 화면이 표시되면 업로드가 정상적으로 완료된 것입니다.

■ 조작 방법

펌웨어 업로드가 완료되면, OLED 화면에 메인 화면이 표시됩니다.

이 화면에서는 모스부호 입력, 매크로 실행, 설정 조정 등의 기본 기능을 조작할 수 있습니다.



[버튼 구성도]

■ 모스부호 입력

- 점(·)과 선(-)을 조합해 모스부호 패턴을 만들면, 키보드 문자로 입력됩니다.
- WPM(속도)과 IWG(입력 완료 대기 시간)를 조절해 점·선 속도와 확정 딜레이를 설정합니다.
- IWG 가 끝나기 전에 스페이스(KEY-4)를 누르면 즉시 문자를 확정해 보낼 수 있습니다.
(※ 매크로 모드가 켜져 있는 경우, 매크로보다 문자 입력이 우선 적용됩니다.)

■ 언어 설정 (SYNC)

- SYNC 버튼을 눌러 장치의 모스부호 언어 모드를 전환할 수 있습니다..
- 모드는 KR / EN / MO 세 가지이며, 버튼을 짧게 누르면 KR ⇄ EN, 3 초 이상 길게 누르면 MO 모드로 변경됩니다.
(※ MO 모드는 모스부호 점과 선을 그대로 입력하는 직접 입력 모드입니다.)
- 다시 누르면 이전 언어 모드로 돌아갑니다.

■ 자동 스페이스 (AT-S / T-1 버튼)

- T-1 버튼을 누르면 자동 스페이스(AT-S) 기능이 활성화됩니다.
- 이 기능은 IWG가 종료된 후, 설정된 시간(ms)이 지나면 자동으로 스페이스가 입력되도록 동작합니다.
(※ AT-S 수치는 옵션에서 설정 가능합니다.)
- AT-S 기능은 EN 모드에서만 작동하며, 영문자와 숫자 입력 시에만 적용됩니다.

■ 매크로 모드 (T-2 버튼)

- T-2 버튼을 누르면 매크로 모드(MACRO)가 활성화됩니다.
- 매크로 데이터는 macro.json 파일에 저장되어 있으며, PC에서 편집하여 자유롭게 수정할 수 있습니다.
- 매크로 모드에서는 미리 등록된 단어나 문장을 자동으로 입력할 수 있습니다.
- 매크로 모드가 활성화된 상태에서도 일반 문자 입력은 가능합니다.
(※ 입력한 모스 패턴이 문자와 동일할 경우 매크로가 우선 실행됩니다.)

■ 기본 키 기능 (KEY-1 ~ KEY-4)

- KEY-1 : CAPS LOCK 기능으로, 대문자와 소문자를 전환합니다.
(※ 모스부호 입력 시 영문자의 대소문자를 구분할 수 있습니다.)
- KEY-2 : 한/영 전환 기능으로, 장치와 PC의 입력 언어를 동시에 변경합니다.
(※ KR ⇄ EN 전환 시 OLED 화면에 현재 언어 모드가 표시됩니다.)
- KEY-3 : 문자 확정 / 스페이스 / 엔터 키
 - 모스 입력 중 IWG가 끝나기 전 누르면 즉시 문자가 확정됩니다.
 - 입력된 패턴이 없을 때는 스페이스로 작동합니다.
 - 길게 누른 상태에서 점(.)이나 선(-)을 누르면 Enter로 작동합니다.
 - 매크로 모드가 켜져 있어도 문자 확정이 우선 적용됩니다.
(※ 스페이스 기능은 AT-S 자동 스페이스 기능과 중복되지 않습니다.)
- KEY-4 : 백스페이스(BACKSPACE) 기능으로, 마지막으로 입력된 문자를 삭제합니다.

■ 로터리 엔코더 노브 (EC-1 / EC-2)

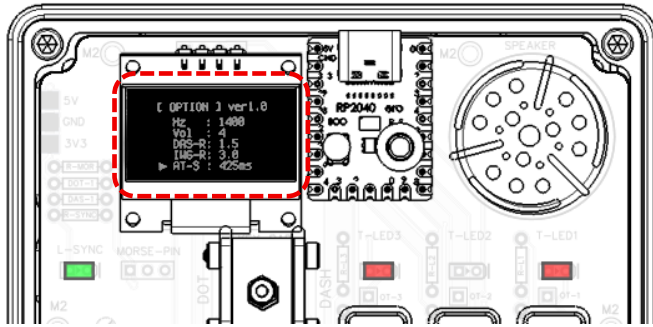
- EC-1 노브 : WPM(속도) 조절 및 저장 기능
 - 노브를 돌려 5~30 범위에서 조절합니다.
 - 클릭 시 현재 모든 설정값이 함께 저장되며, 'SAVED', 또는 'FAILED'가 화면에 표시됩니다.
- EC-2 노브 : IWG(입력 완료 대기 시간) 및 옵션 설정
 - 노브를 돌려 -40 ~ +40 범위에서 조절하여 IWG를 보정합니다.
 - 클릭 시 [OPTION] 항목으로 전환되며, 여러 가지 설정을 변경할 수 있습니다.
(※ 자세한 내용은 별도로 설명합니다.)

■ 리셋 스위치

- 장치를 재시작할 때 사용합니다.

■ 옵션 항목

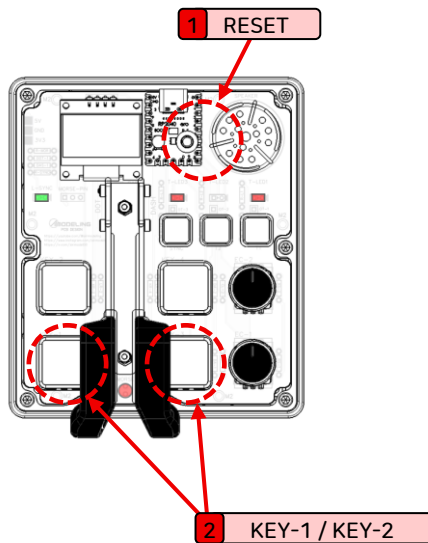
옵션은 EC-2 노브를 누르면 [OPTION] 항목이 표시되며, 다음과 같은 설정 항목을 변경할 수 있습니다.



- H Z : 모스부호 점(·)과 선(-)의 사운드 주파수를 100~3000Hz 범위에서 조절합니다.
- V O L : 모스부호 사운드의 볼륨을 0~5 단계로 조절합니다.
- D A S - R : 모스부호 선(-)의 길이를 1.5~4.0 범위에서 0.5 단위로 조절합니다.
- I W G - R : 문자 간 간격을 3.0~10.0 범위에서 1.0 단위로 조절합니다.
- A T - S : 동 스페이스 타이밍을 25~3000ms 범위에서 25ms 단위로 조절합니다.
- D E F A U L T : 모든 설정값을 초기값으로 복원합니다.
- S A V E : 현재 설정값을 저장합니다. (※ 성공하면 'DONE', 실패하면 'FAIL' 이 화면에 표시됩니다.)
- B A C K : 옵션 메뉴를 종료하고 메인 화면으로 돌아갑니다.

■ 부트 모드 (편집 모드) 진입 방법

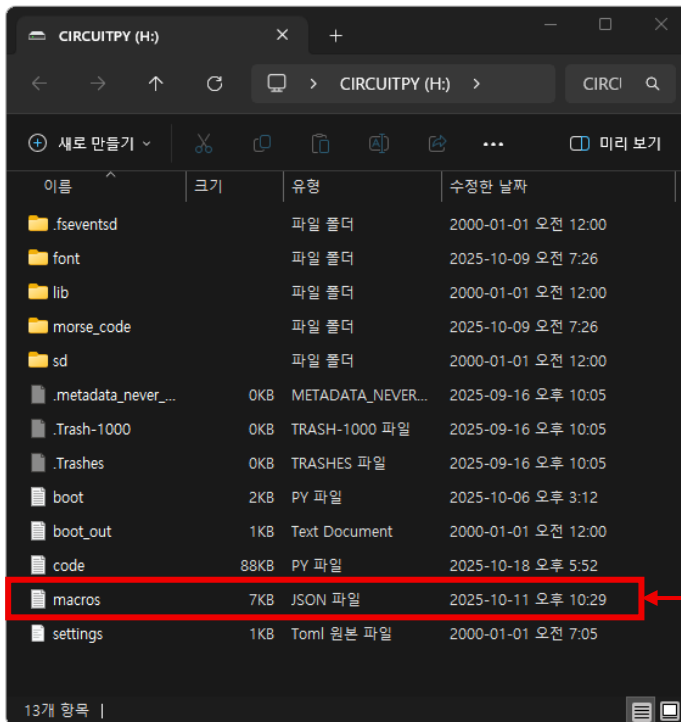
내부 파일을 변경하거나 매크로 파일을 업로드할 때 장치를 부트 모드(BOOT MODE) 로 전환해야 합니다. 이 모드는 PC 에서 장치를 저장장치로 인식하여 파일을 직접 교체하거나 수정할 수 있도록 합니다.



■ 진입방법

1. 리셋 스위치를 누른 후, 바로 KEY-1 과 KEY-2 를 동시에 계속 누릅니다.
2. 잠시 후 PC 화면에 드라이브 창이 표시되면 부트 모드 진입이 완료됩니다.

■ 매크로 파일 교체 방법



1. 장치를 부트 모드로 진입합니다.
2. PC에서 표시된 CIRCUITPY 드라이브 폴더가 자동으로 열립니다.
3. 새로운 macro.json 파일을 기존 파일을 삭제하거나 덮어씁니다.
4. 파일 복사가 완료되면 리셋 스위치를 눌러 장치를 재시작합니다.

(※ 매크로 파일은 별도의 매크로 편집 프로그램(MORSEKEY_Macro_Editor)을 사용하여 수정할 수 있습니다.)

■ 마무리를 하며

MORSEKEY-A01은 '누구나 만들 수 있는 모스부호 키보드'를 목표로 설계했습니다. 단순한 입력 장치를 넘어, 직접 조립하고 설정하며 전자 제작의 과정을 배우고 즐길 수 있도록 구성해 봤습니다.

이 작품이 여러분의 제작과 학습, 그리고 새로운 아이디어에 도움이 되길 바랍니다. 마지막으로, 제 작품을 다운로드하고 직접 제작해 주신 모든 분들께 진심으로 감사의 말씀을 드립니다.

감사합니다.

■ 면책 및 저작권 안내

작품명 : MORSEKEY-A01 ver1.0

제작자 : Airmodeling

이 프로젝트는 Adafruit CircuitPython (MIT License) 을 기반으로 제작된 오픈소스 하드웨어입니다.

MORSEKEY-A01 은 학습과 취미 제작, 개인 연구를 목적으로 자유롭게 활용할 수 있습니다.

프로젝트에 포함된 회로, 3D 데이터, 펌웨어, 매뉴얼, 디자인 등 모든 자료는 공개된 형태로 제공됩니다.

■ 사용 및 배포 조건

1. 모든 자료는 비상업적·비영리적 목적에 한해 자유롭게 이용할 수 있습니다.
2. 상업적 판매, 수정 후 재배포, 2 차 제작물 판매 등 영리 목적의 사용은 금지됩니다.
3. 개인 학습을 위한 수정은 가능하나, 재배포 시에는 반드시 원 저작자(Airmodeling) 를 명시해야 합니다.
4. 프로젝트 사용 중 발생하는 문제·오류·손상·기기 고장 등에 대한 책임은 사용자 본인에게 있습니다.

■ 참고 자료

GitHub (소스코드 및 펌웨어)

<https://github.com/Airmodeling/MORSEKEY-A01>

OSHWLab (회로 및 3D 도면)

<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>

※ 본 프로젝트는 오픈소스 정신에 따라, 누구나 배우고 공유하며 함께 발전시켜 나가는 것을 목표로 합니다.

문의: airmodel00@gmail.com

© 2025 Airmodeling. All Rights Reserved.

본 문서는 CC BY-NC-SA 4.0 라이선스를 따릅니다.

이 프로젝트는 Adafruit CircuitPython (MIT License)을 기반으로 제작되었습니다.

