

BITRONICS

BITRONICS

BITRONICS

BITRONICS

BITRONICS

BITRONICS



Learning by building

SEED SIGNER

ENHANCED SECURITY



USER-FRIENDLY INTERFACE



AIR-GAPPED



OPEN SOURCE



THE SEEDSIGNER

SeedSigner is a DIY Bitcoin hardware wallet. It's a physical device leveraging generic and affordable hardware to sign Bitcoin transactions.

This guide introduces how to use it from scratch. For more details and insights about the project, you can check the description of the product.

KIT - DIY

Everything you need to assemble your SeedSigner from scratch



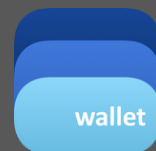
BUILD YOUR TRANSACTIONS

Sign your PSBT transactions easily



COMPATIBILITY WITH ANY WALLET

Sparrow Wallet, Blue Wallet, Nunchuck...



PREPARATION

Check all SeedSigner details downloading the full guide on the following QR code.



For any other question contact Bitronics team.

QUICK SETUP

Required time: 10 minutes



Assamble your hardware wallet:

1. Insert and secure the camera in the Raspberry Pi Zero, attach the camera to the case, and fit the Waveshare screen, ensuring all pins align.
2. Download the [SeedSigner image](#), verify, and flash it to your micro-SD card.
3. Insert the SD card, connect the power, and configure.

Create your seed

1. Insert the micro-SD card and connect the power cable.
2. Use the navigation buttons to enter the main menu.
3. Select "Create New Seed", choose the "Create New Seed" option and generate the Seed. Follow the on-screen instructions to generate entropy.
4. Write down the seed words in order and store them in a safe place.
5. Verify the Seed, enter the requested words to verify the seed.

Sign a transaction

1. Create a PSBT file on your Bitcoin wallet and save it to a micro-SD card.
2. Insert the micro SD Card into the SeedSigner and power it on.
3. Select "Sign the Transaction" from the main menu and select the PSBT file.
4. Review and confirm the transaction details.
5. Sign the transaction, saving the signed PSBT file back to the SD card.
6. To complete the transaction, insert the SD card back into your Bitcoin wallet and load the signed PSBT file and broadcast the transaction.

Additional resources:

For any problem check full guide or contact Bitronics team.

SEEDSIGNER DIY KIT

Required time: 10 minutes

Below, we present the steps to assemble your SeedSigner DIY kit.

For any other question contact Bitronics team.



KIT PARTS



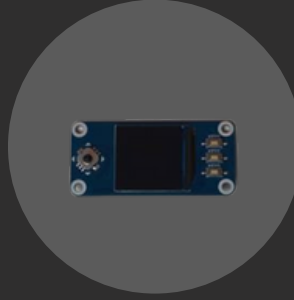
Case Parts



Raspberry Pi Zero 1.3



Waveshare 1.3" LCD Hat



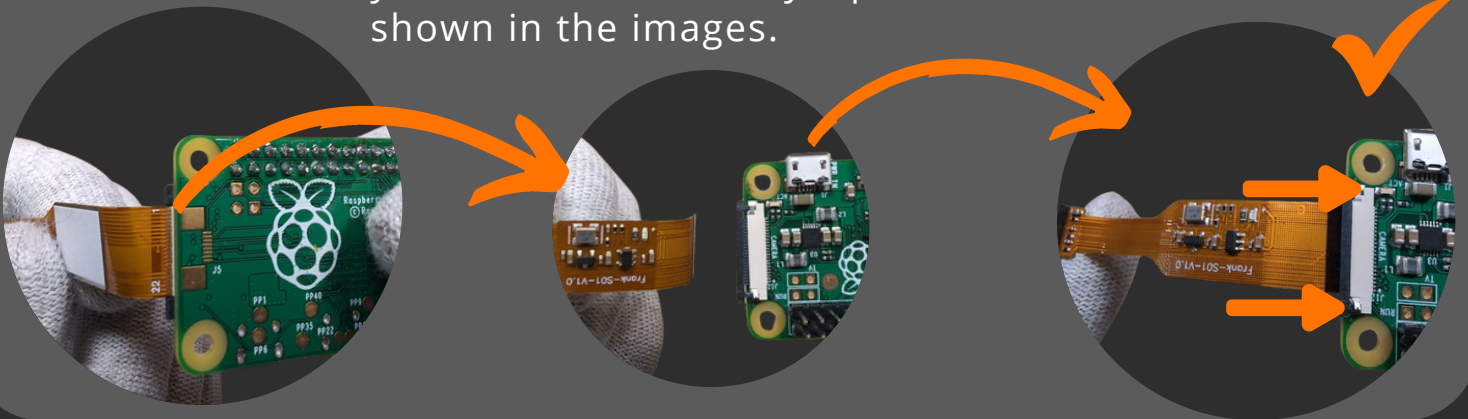
OV5670 Pi-cam & Micro SD



HOW TO BUILD

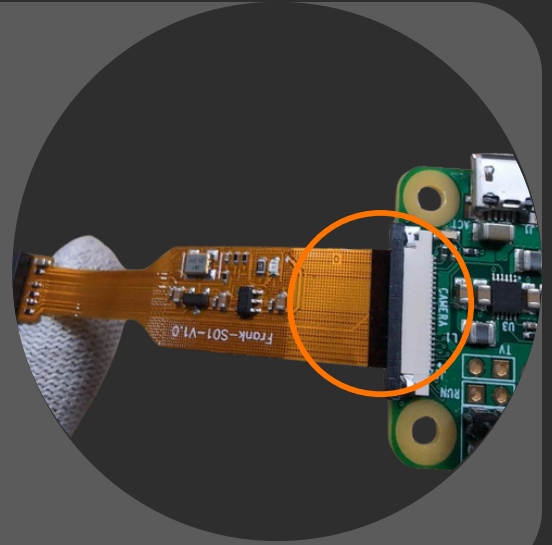
STEP 1

Start by positioning your Raspberry Pi Zero with the pins facing up. You'll notice a tab on its right side that you need to carefully open to insert the camera as shown in the images.

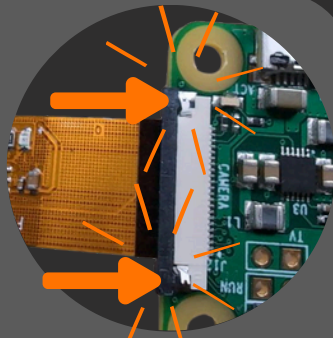


STEP 2

Ensure the camera aligns properly (the gold part should match the edge of the Raspberry Pi's slot). Make sure the shiny contacts face towards the green circuit board.

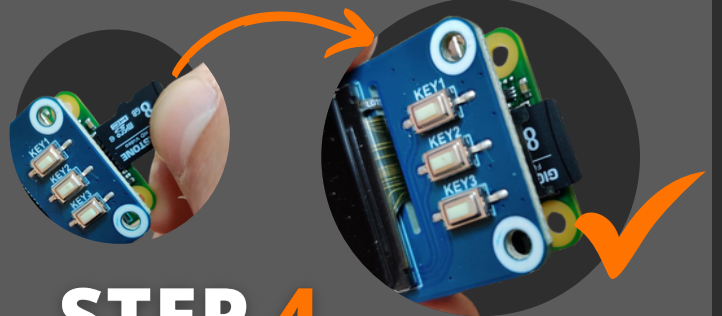


STEP 3



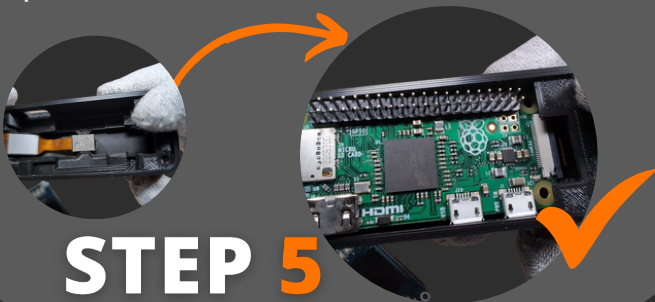
After inserting the camera, gently close the black tab to secure the camera in place, ensuring it matches the illustration.

Insert the micro SD card with the flashed SeedSigner image. (Details on how to do that below)



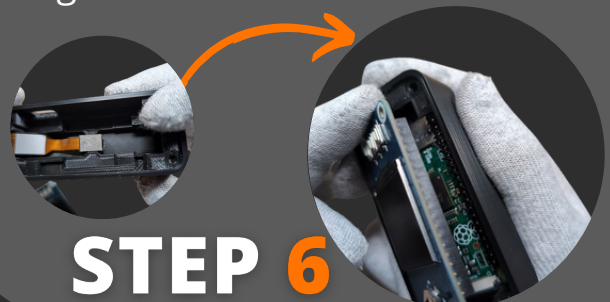
STEP 4

Next, place the camera on the top of the case, ensuring it fits into the camera hole. Apply gentle pressure to secure it in place. This is how it should look:



STEP 5

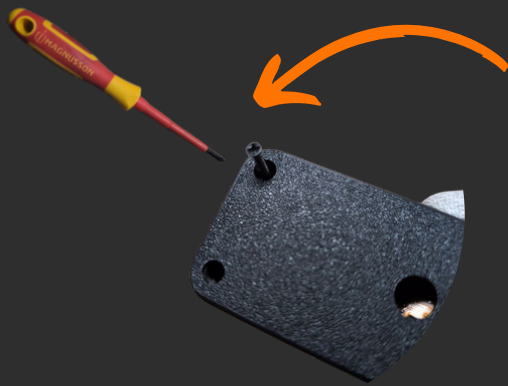
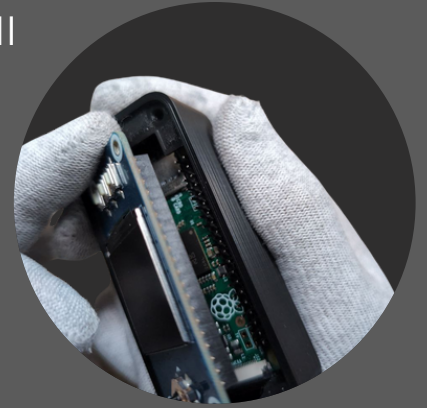
Once the camera is positioned in the casing, insert the Raspberry Pi Zero ensuring proper pin alignment.



STEP 6

STEP 7

Once the camera is correctly positioned in the casing, you only need to insert the Waveshare screen into the Raspberry Pi Zero, making sure all the pins align as shown in the following image:



Place the top cover on the screen and securely screw it in from the back.

And that's it! Now, you just need to use the SeedSigner software to start using it! (further explanation on how to do this follows)



Connect the power cable (Micro USB)

Once connected, the SeedSigner menu will appear. Click on Settings to set up.



With these buttons and joystick, you can configure the entire SeedSigner menu and navigate through all its options and functionalities.

SEEDSIGNER SCREENS

MAIN SCREEN

The following screen provides SeesSigner tools, including scan QR code to sign PSBT transaction, show your seeds, tools and settings.



TOOLS SCREEN

Here you will find various options on the "Tools" screen, such as creating a new seed, verifying addresses, and many more!



ATTRIBUTES



FEATURES

HIGH PERFORMANCE

Raspberry Pi Zero 1.3 without WIFI

PLUG PLAY

Easy to setup, fully assembled or DIY kit, configure and play

FOSS

MIT License

CLEAN INTERFACE

A beautiful and intuitive interface with everything you need to navigate the wallet

FULL SETUP

Required time: 15 minutes

Check all SeedSigner details downloading the full guide on the following QR code.



For any other question contact Bitronics team.



Assamble your hardware wallet

SeedSigner uses a modified version of Debian for Raspberry Pi. Debian is a Linux distribution that uses only free and open-source software (FOSS). You can download the latest version [HERE](#).

After downloading the .IMG file, we need to verify it. The file is signed by the project's maintainer, and we will use their credentials to verify the file's integrity.

For this, we use [GPG](#), a program that allows us to import authors' public keys and verify signatures. GPG is available on all Linux and Mac operating systems (which is also Linux-based). If not, you can download it [HERE](#).

Now, we need SeedSigner's public GPG key.

The maintainer has the key listed in several places. Why, you might ask?

We aim to ensure that an attacker cannot replace the file we intend to write on the SD card with a malicious one. However, if an attacker compromises the source of our files, they would likely also replace the public key with their own, making it impossible for us to detect the tampering.

That's why you should always verify the key in multiple locations.

You can find the key on:

- [Telegram](#)
- [GitHub](#)
- [Twitter](#)

So, open a terminal and type:

```
curl -sS https://raw.githubusercontent.com/SeedSigner/seedsigner/main/seedsigner_pubkey.gpg | gpg --import
```

Having imported the key into our keyring, let's obtain the files from:

- Seedsigner_0_4_4.img.zip: the archived image
- Seedsigner_0_4_4.img.zip.txt.gpg: the signature produced by the author

The following command will confirm the file's integrity:

```
gpg --verify seedigner_0_4_4.img.zip.txt.gpg
```

This guide and review of the SeedSigner Bitcoin hardware wallet helps users protect their private keys while preserving entropy. We're looking for a "good signature," a confirmation that everything is in order.

Now that we know the .IMG file is authentic, we can write it to the SD card. For this, I use balenaEtcher, a software that lets you burn the .IMG file to the SD card. It's quite simple and supports drag-and-drop functionality.

This is a modified guide from two guides and information available online. You can read the full guides here:

- [[How to Use SeedSigner for Secure Bitcoin](#)] (Bitcoin Magazine)
- [[SeedSigner: What It Is and Assembly Guide](#)] (Estudio Bitcoin)

Power on the SeedSigner

Insert the micro-SD card and connect the power cable to turn on the device.

Navigate to the Seed Generation Menu, use the navigation buttons to access the main menu, then select the option to create a new seed.

Generate the Seed

Follow the on-screen instructions to generate entropy, typically by performing specific actions such as moving the device around or pressing buttons.

Write Down the Seed Words

Once the seed is generated, it will be displayed as a series of words (usually 12 or 24). Write these words down in the exact order they appear. This is your backup.

Verify the Seed Re-entering the Seed Words, the SeedSigner may prompt you to re-enter certain words from your seed to verify accuracy. Carefully follow the prompts and enter the requested words correctly.

Complete the Verification

Once verification is successful, the device will confirm that your seed is correctly recorded.

Store the Backup Securely

Physical Storage: Keep the written seed words in a secure location, such as a fireproof safe, to protect against physical threats like fire or theft.

Digital Backup: Consider creating a secure digital backup if you have a reliable method, ensuring it is encrypted and stored in a safe manner.

Prepare the PSBT (Partially Signed Bitcoin Transaction):

Create the PSBT using your Bitcoin wallet software (e.g., Sparrow Wallet) to create a PSBT. This involves setting up the transaction details such as recipient address, amount and fees. Save the PSBT File and export the PSBT file and save it onto a micro-SD card.

Insert the SD Card

Load the PSBT, insert the micro-SD card containing the PSBT file into the SeedSigner device. Power On the SeedSigner, connect the power cable to power on the SeedSigner.

Access the Main Menu

Navigate to "Sign Transaction", use the navigation buttons to enter the main menu and select the "Sign Transaction" option.

Load the PSBT File

Select the PSBT File, browse through the files on the micro-SD card and select the PSBT file you saved earlier.

Verify Transaction Details

Review Details and carefully review the transaction details displayed on the SeedSigner screen. This includes verifying the recipient address, amount, and transaction fee. If everything looks correct, confirm the transaction details to proceed.

Sign the Transaction

Follow the on-screen prompts to sign the transaction. The SeedSigner will use your private key to sign the PSBT, creating a signed transaction. Once the transaction is signed, save the signed PSBT file back to the micro-SD card.

Complete the Transaction

Insert the SD Card Back into Your Wallet Software, remove the micro-SD card from the SeedSigner and insert it back into your computer.

Load the Signed PSBT

Open your Bitcoin wallet software and load the signed PSBT file from the micro-SD card.

Broadcast the Transaction

Finalize the transaction by broadcasting it to the Bitcoin network. Your transaction is now signed and will be processed by the network.