



KAmoRPi5 PCIe-M.2 (PL)



Rev. 20240308115414

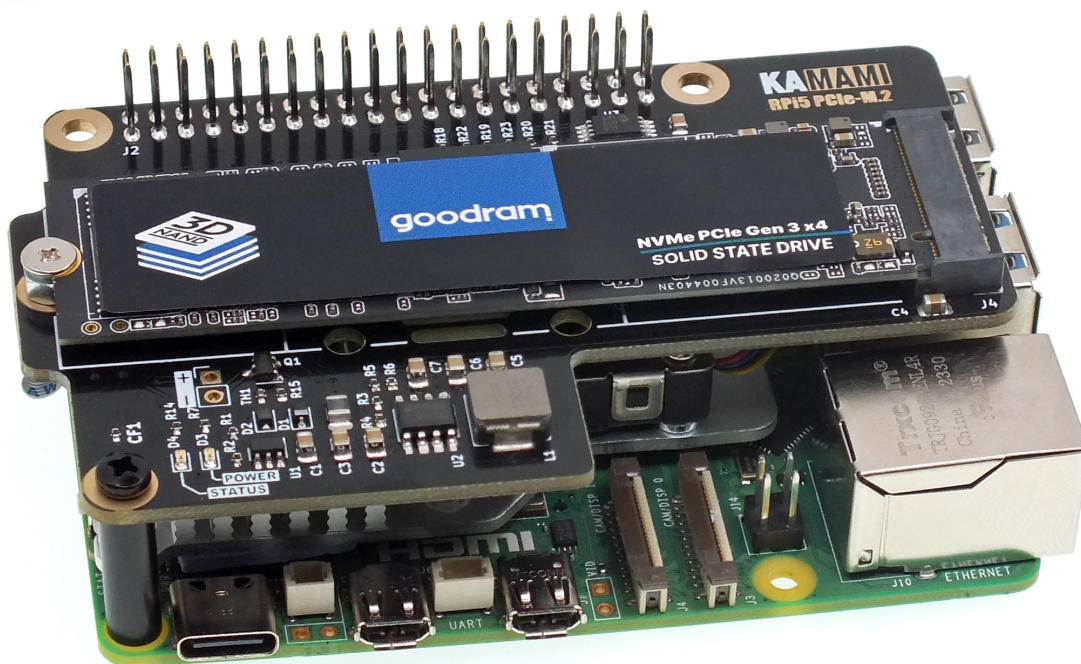
Źródło: [https://wiki.kamamilabs.com/index.php/KAmoRPi5_PCIe-M.2_\(PL\)](https://wiki.kamamilabs.com/index.php/KAmoRPi5_PCIe-M.2_(PL))

Spis treści

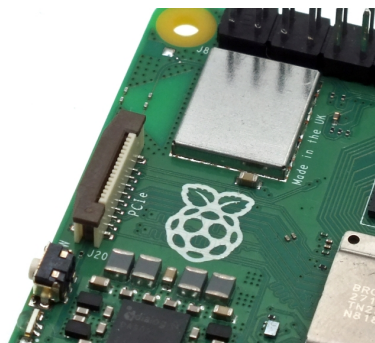
Opis	1
Podstawowe parametry	2
Wyposażenie standardowe	3
Opis najistotniejszych elementów	4
Wymiary	6
Montaż Dysku NVME M.2 w adapterze KAmoD RPi5 PCIe-M.2	7
Montaż adaptera KAmoD RPi5 PCIe-M.2 na płycie Raspberry Pi 5	8
Przebieg aktualizacji oprogramowania	9
Aktualizacja systemu operacyjnego	10
Aktualizacja zawartości pamięci EEPROM	11
Aktualizacja narzędzia raspi-config	12
Ustawianie sekwencji bootowania	13
Instalowanie systemu na dysku NVME M.2	14
Uruchomienie systemu z dysku NVME M.2	15
Informacje dodatkowe	17
Dodatkowe funkcje adaptera KAmoD RPi5 PCIe-M.2	29

Opis

KAmoDRPi5 PCIe-M.2 - Adapter dysku NVME M.2 dla Raspberry Pi 5



Komputer Raspberry Pi 5 udostępnia interfejs **PCI Express**, dzięki czemu jest kompatybilny z wieloma najnowszymi rozwiązaniami techniki komputerowej. Dla zachowania kompaktowych wymiarów serii Raspberry, interfejs został wyprowadzony na niestandardowe, miniaturowe złącze FFC o 16 stykach i rastrze 0,5 mm, oznaczone na płytce jako **PCIe**.



Nowe złącze pozwala na podłączenie nowoczesnych dysków SSD, które obsługują szybki i niezawodny protokół NVME. Takie rozwiązanie sprawia, że Raspberry Pi staje się idealnym centrum multimedialnym czy serwerem plików jednocześnie zachowując niewielkie rozmiary i efektywność energetyczną.

Zastosowany interfejs działa w wersji PCI Express Gen2 x1 i umożliwia podłączenie do komputera Raspberry Pi 5 dysku NVME ze złączem typu **M.2 M-KEY**. Wymaga to użycia adaptera **KAmoDRPi5 PCIe-M.2**.

Do poprawnego działania takiego zestawu wymagana jest najnowsza wersja oprogramowania dla komputerka Raspberry Pi 5.

Dopiero, gdy wszystkie aktualizacje będą wykonane można zainstalować obraz systemu operacyjnego na dysku NVME M.2 oraz ustawić sekwencję bootowania na rozruch z dysku NVME M.2. Wszystkie niezbędne czynności, w kolejności ich wykonywania, zostały opisane w instrukcji.

Podstawowe parametry

- Adapter jest przystosowany do łatwego montażu na płytce Raspberry Pi 5
- Połączenie z komputerkiem Raspberry Pi 5 odbywa się poprzez złącze 40-stykowe GPIO oraz taśmę FFC 16/0,5 (opcjonalnie możliwe jest połączenie tylko z użyciem taśmy FFC)
- Zajmuje typową przestrzeń zarezerwowaną dla standardowych modułów rozszerzeń HAT
- Umożliwia dołączenie kolejnych modułów typu HAT, do 40-stykowego złącza GPIO
- Umożliwia podłączenie dysku NVME ze złączem typu M.2 (nazywane również NGFF)
- Komunikacja poprzez interfejs PCI Express Gen2 x1 (opcjonalnie również Gen3 x1)
- Umożliwia zamontowanie dysku o rozmiarze 2230, 2242, 2260 lub 2280
- Dostarcza napięcie zasilania dysku równe 3,3V i maksymalny prąd 3A (wydajność prądowa zależy od mocy zasilacza dostarczającego energię do Raspberry Pi 5)
- Pobiera zasilanie z komputerka Raspberry Pi 5, dlatego należy zastosować zasilacz do Raspberry Pi 5 o odpowiedniej mocy (zalecany - co najmniej 5,1V, 3,0A, 15W)
- Wskaźniki w postaci diod LED sygnalizujące poprawne zasilanie oraz aktywność dysku
- Wymiary 56x85 mm (56x89 mm w środkowej części, gdzie znajduje się otwór dla śruby mocującej dysk w rozmiarze 2280)
- Konstrukcja adaptera nie blokuje możliwości zastosowania radiatora z wentylatorem dedykowanego dla Raspberry Pi 5
- **Nie każdy dysk NVME M.2 jest kompatybilny z komputerkiem Raspberry Pi 5**

Przetestowane i działające modele dysków to m.in.:

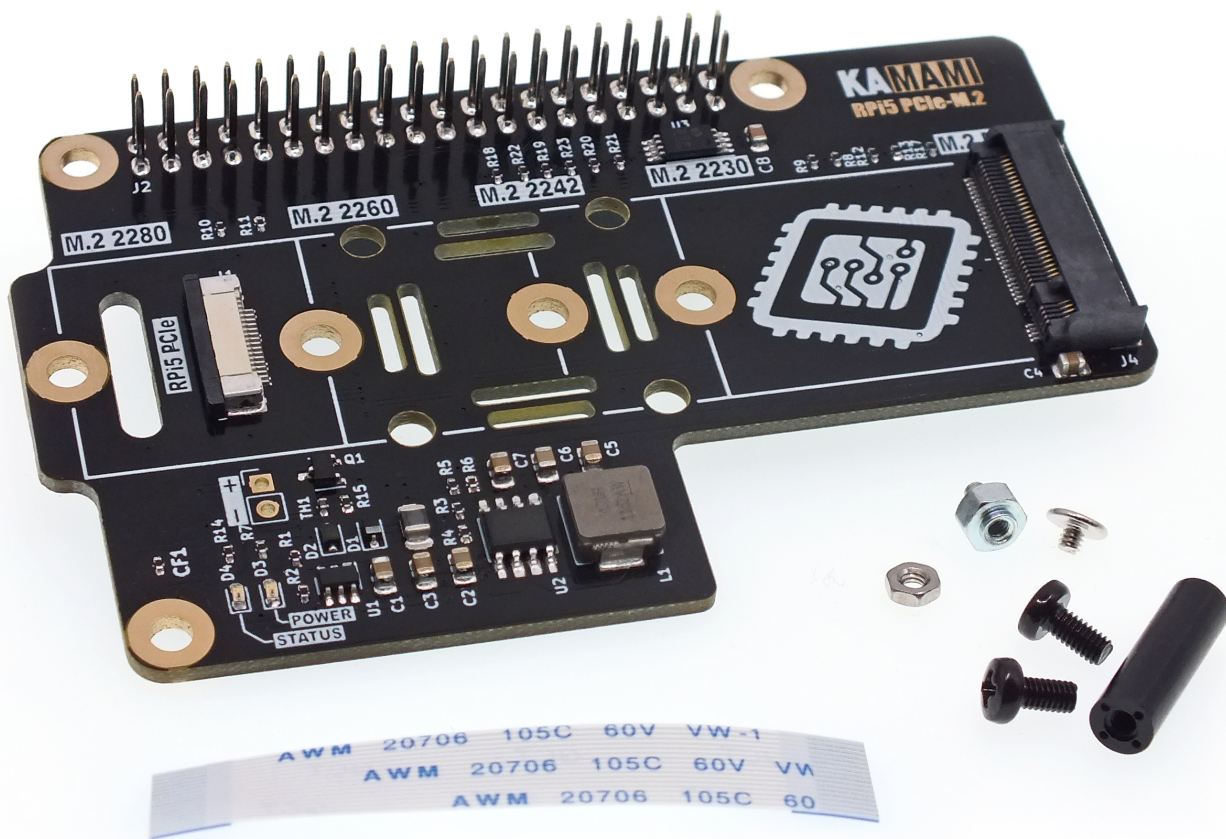
- Goodram PX500 (256 GB, 2280)
- Samsung 980 (500 GB, 2280)
- Samsung PM991a (256GB, 2242)
- Patriot P310 (240GB, 2280)

Dyski niekompatybilne (działają tylko jako pamięć dodatkowa, nie nadają się do bootowania systemu):

- Goodram PX600 (256GB, 2280)
- WD SN530 (256GB, 2242)

Wyposażenie standardowe

Kod	Opis
<ul style="list-style-type: none"> • Adapter KAmoDRPi5 PCIe-M.2 • 1 x Taśma FPC • 1 x Zestaw montażowy 	<ul style="list-style-type: none"> • Zmontowany i uruchomiony moduł • Taśma FFC 16/0,5 • Zestaw zapewniający stabilne umocowanie adaptera (2 śrubki M2, 1 dystans), śruba mocująca do dysku M.2
<ul style="list-style-type: none"> • Dysk NVME M.2 nie jest elementem zestawu • Raspberry Pi 5 nie jest elementem zestawu 	



Opis najistotniejszych elementów

Złącze J1 - RPi5 PCIe

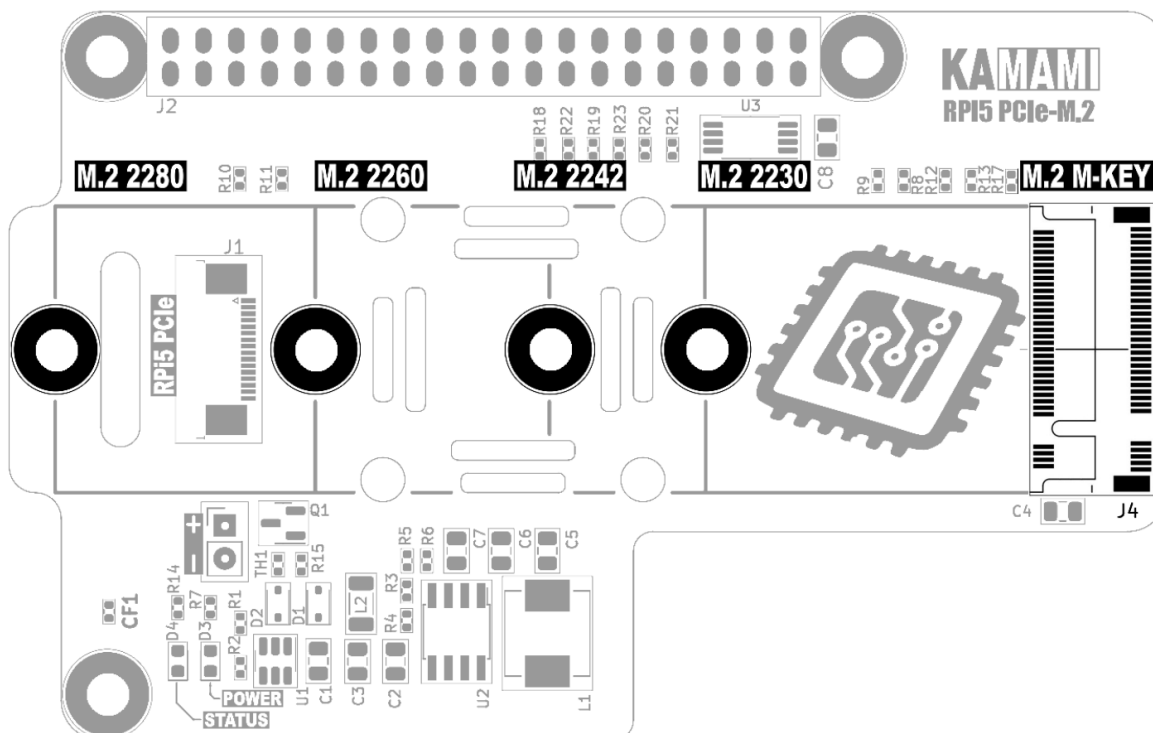
Typ złącza	Funkcja
FFC 16 styków, raster 0,5 mm	<ul style="list-style-type: none"> Łączy interfejs PCIe na płycie Raspberry Pi 5 z adapterem KAmoD RPi5 PCIe-M.2 Odpowiada za przesyłanie danych, ale także dostarcza zasilania do adaptera (adapter może działać, gdy jest połączony z Raspberry Pi 5 tylko samą taśmą, jeśli dysk nie pobiera prądu większego niż 1A) Przy złączu znajduje się otwór, przez który należy przeprowadzić taśmę połączeniową Łączy dysk NVME M.2 z interfejsem PCI Express Gen2 x1

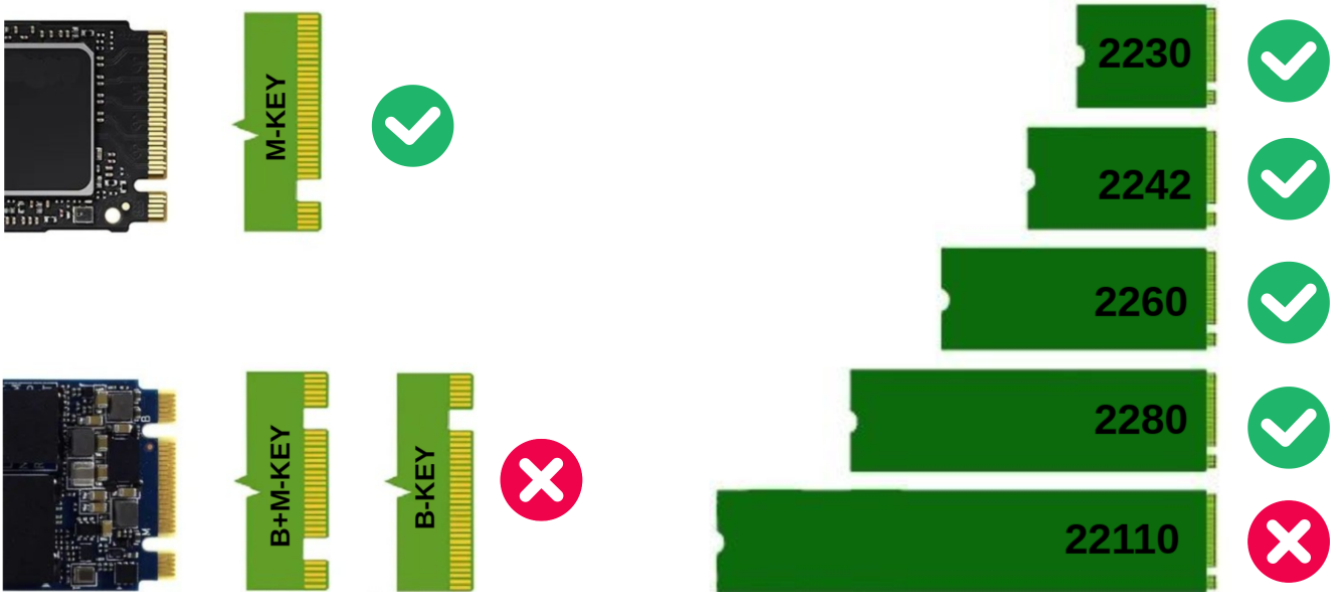
Złącze J2 - GPIO

Typ złącza	Funkcja
Złącze typu goldpin męsko-żeńskie, wysokie	<ul style="list-style-type: none"> Łączy adapter KAmoD RPi5 PCIe-M.2 z Raspberry Pi 5 - tworzy połączenia elektryczne oraz mocowanie mechaniczne Umożliwia dołączenie kolejnych modów rozszerzających możliwości Raspberry Pi 5 - nie blokuje funkcjonalności złącza GPIO Dostarcza zasilania do adaptera z komputera Raspberry Pi 5

Złącze J4 - M2 M-KEY

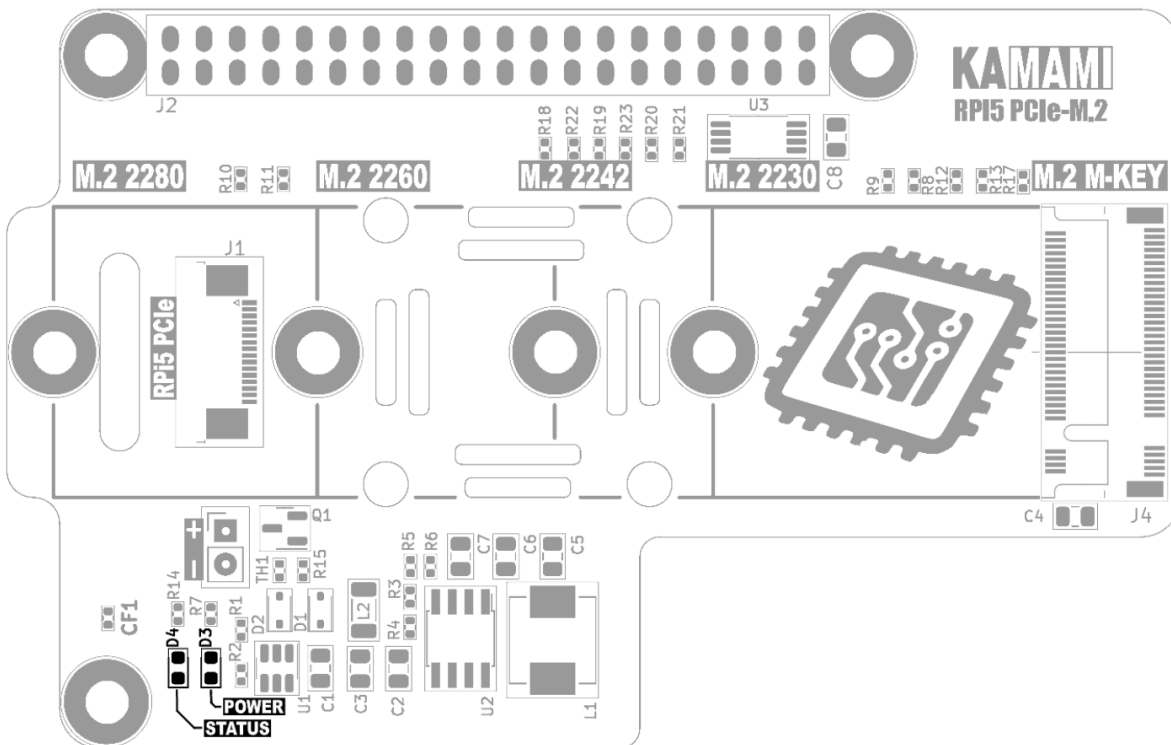
Typ złącza	Funkcja
M.2 (NGFF)	<ul style="list-style-type: none"> Umożliwia dołączenie dysku NVME ze złączem M.2, z tzw. kluczem typu „M” (M-KEY) Zestaw otworów pozwala na stabilne zamontowanie dysków w rozmiarach 2230, 2242, 2260 lub 2280 Dostarcza napięcie zasilania dysku równe 3,3 V i maksymalny prąd 3 A Łączy dysk NVME M.2 z interfejsem PCI Express Gen2 x1





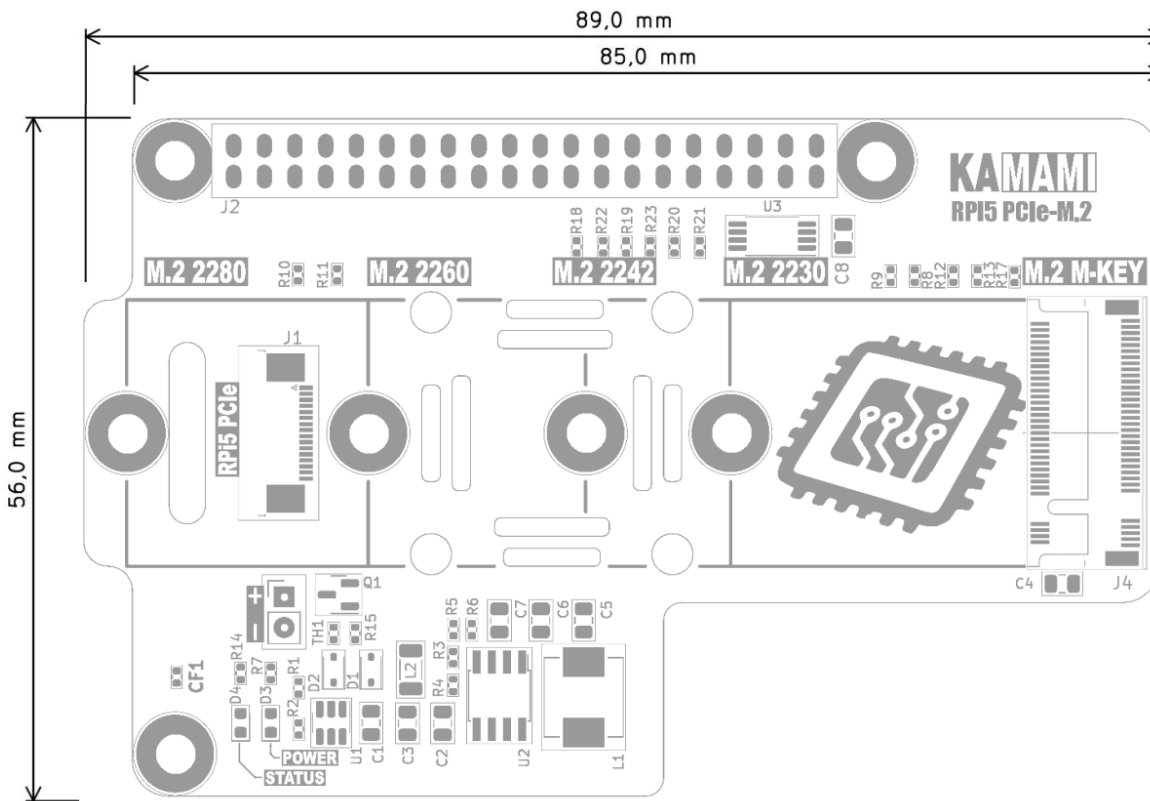
Kontrolki LED

Oznaczenie	Funkcja
D3 - POWER	• Wyraźne świecenie sygnalizuje prawidłowe parametry zasilania dysku NVME M.2
D4 - STATUS	• Zaświecenie kontrolki sygnalizuje wykonywanie operacji zapisu/odczytu/transferu danych na dysku NVME M.2



Wymiary

Wymiary 56x85 mm i pokrywają się w wymiarami komputerka Raspberry Pi 5. W środkowej części płytki znajduje się otwór dla śruby mocującej dysku w rozmiarze 2280, w tym miejscu wymiary wynoszą 56x89 mm.



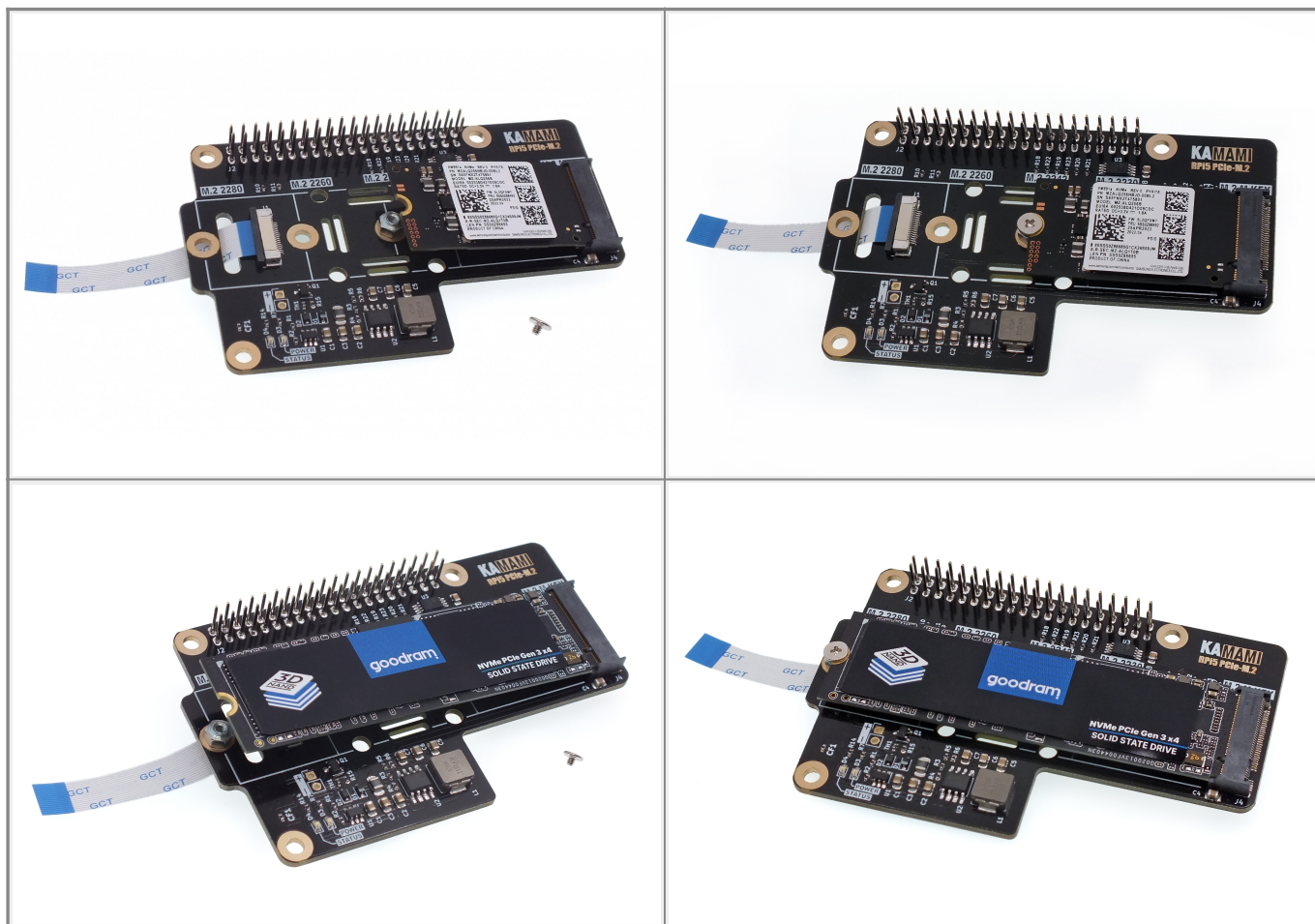
Montaż Dysku NVME M.2 w adapterze KAmoD RPi5 PCIe-M.2

Uwaga! Podłączanie i odłączanie dysku NVME M.2 należy wykonywać tylko wtedy, gdy komputerek Raspberry Pi 5 jest wyłączony i odłączony od zasilacza.

W pierwszej kolejności należy określić rozmiar dysku (nie pojemność) - akceptowane są dyski 2230, 2242, 2260 lub 2280. Znając rozmiar dysku należy zamontować śrubę męsko-żeńską mocującą dysk M.2 w otworze odpowiadającym danemu rozmiarowi dysku i dokręcić lekko nakrętką od spodu. Otwory w płytce są nieco większe, aby możliwe było ustawienie śruby w optymalnym miejscu dla danego dysku.

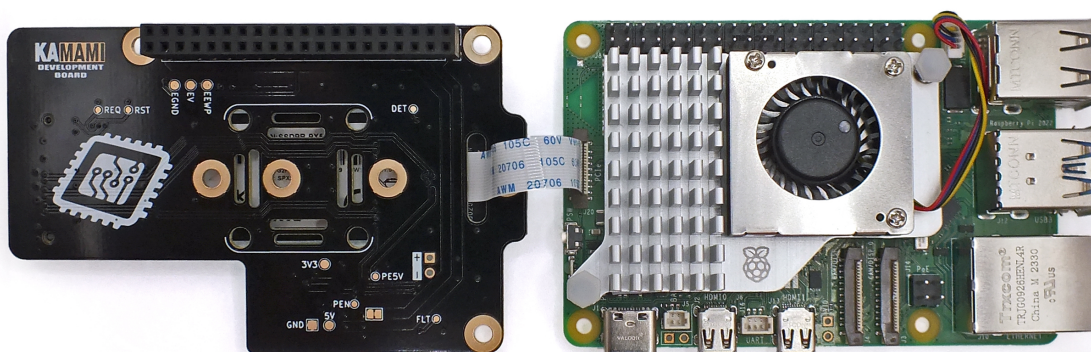
W drugim kroku należy taśmę FFC 16/0,5 podłączyć do złącza J1 (niebieskimi znacznikami do góry) i zacisnąć blokadę, a drugi koniec taśmy należy przełożyć przez otwór przy złączu J1 (RPi5 PCIe).

Teraz można wsunąć dysk NVME M.2 do złącza J4 (M.2 M-KEY), w taki sposób, aby przeciwna strona dysku lekko odstawała od płytki. Następnie należy delikatnie docisnąć dysk do śruby mocującej (i jeśli to konieczne) przesunąć lekko śrubę tak, aby precyzyjnie trafiła we wcięcie na dysku. Na koniec należy przykręcić śrubę górną i dokręcić nieco mocniej nakrętkę od spodu.



Montaż adaptera KAmoD RPi5 PCIe-M.2 na płytce Raspberry Pi 5

Adapter KAmoD RPi5 PCIe-M.2 z zamontowaną taśmą należy ułożyć obok Raspberry Pi 5. Teraz, w łatwy sposób można dołączyć taśmę FFC do złącza PCIe na płytce Raspberry Pi 5.



Po dołączeniu taśmy adapter należy umieścić nad komputerkiem Raspberry Pi 5, a następnie należy dolną stronę złącza J2 osadzić na 40-szpilekowym złączu GPIO komputerka. Na koniec warto zamontować tuleję o wysokości 16 mm przy złączu USB-C, która zapewni stabilne połączenie całości.



Przebieg aktualizacji oprogramowania

Wykonanie pełnej aktualizacji oprogramowania obejmuje 3 etapy:

- aktualizację **systemu operacyjnego**,
- aktualizację zawartości **pamięci EEPROM**,
- aktualizację narzędzia **raspi-config**.

Przeprowadzenie wszystkich niezbędnych czynności wymaga posiadania działającego komputerka Raspberry Pi 5 z systemem operacyjnym uruchomionym, np. z karty pamięci microSD i dostępem do internetu. Dokładna instrukcja przygotowania systemu operacyjnego na karcie pamięci znajduje się w rozdziale „*Przygotowanie karty pamięci z systemem operacyjnym dla Raspberry Pi 5*”.

Aktualizacja systemu operacyjnego

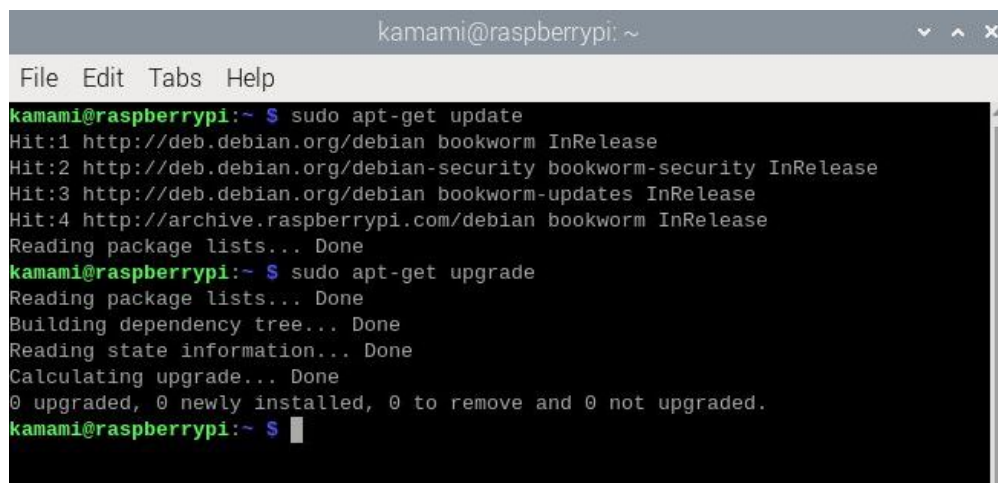
Uruchamiamy Raspberry Pi 5 z systemem operacyjnym przygotowanym wcześniej na karcie pamięci (na tym etapie adapter KAmoD RPi5 PCIe-M.2 powinien być zdemonstrowany). Pierwsze uruchomienie systemu może trwać nieco dłużej niż zwykle. Po wyświetleniu pulpitu systemu otwieramy okno konsoli (Terminal) np. za pomocą kombinacji klawiszy *Ctrl+Alt+T* i wpisujemy:

```
sudo apt-get update
```

a następnie:

```
sudo apt-get upgrade
```

Ukończenie wszystkich działań może potrwać kilka minut, w zależności od liczby komponentów wymagających uaktualnienia (w oknie konsoli może wyświetlić się znacznie więcej komunikatów niż na poniższym przykładzie). Wszelkie pytania potwierdzamy klawiszem Y(Yes).



```
kamami@raspberrypi: ~  
File Edit Tabs Help  
kamami@raspberrypi:~ $ sudo apt-get update  
Hit:1 http://deb.debian.org/debian bookworm InRelease  
Hit:2 http://deb.debian.org/debian-security bookworm-security InRelease  
Hit:3 http://deb.debian.org/debian bookworm-updates InRelease  
Hit:4 http://archive.raspberrypi.com/debian bookworm InRelease  
Reading package lists... Done  
kamami@raspberrypi:~ $ sudo apt-get upgrade  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
Calculating upgrade... Done  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
kamami@raspberrypi:~ $
```

Na koniec restartujemy system, np. wpisując polecenie:

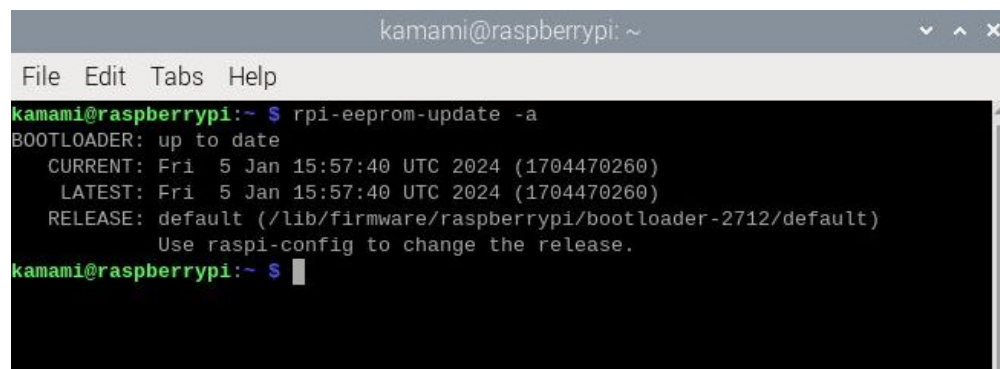
```
sudo reboot
```

Aktualizacja zawartości pamięci EEPROM

Otwieramy okno konsoli (Terminal) np. za pomocą kombinacji klawiszy *Ctrl+Alt+T* i wpisujemy:

```
sudo rpi-eeprom-update -a
```

W oknie konsoli może wyświetlić się więcej komunikatów niż na poniższym przykładzie, jeżeli zawartość będzie wymagała aktualizacji. Wszelkie pytania potwierdzamy klawiszem Y (Yes).



```
kamami@raspberrypi: ~  
File Edit Tabs Help  
kamami@raspberrypi:~$ rpi-eeprom-update -a  
BOOTLOADER: up to date  
CURRENT: Fri 5 Jan 15:57:40 UTC 2024 (1704470260)  
LATEST: Fri 5 Jan 15:57:40 UTC 2024 (1704470260)  
RELEASE: default (/lib/firmware/raspberrypi/bootloader-2712/default)  
Use raspi-config to change the release.  
kamami@raspberrypi:~$
```

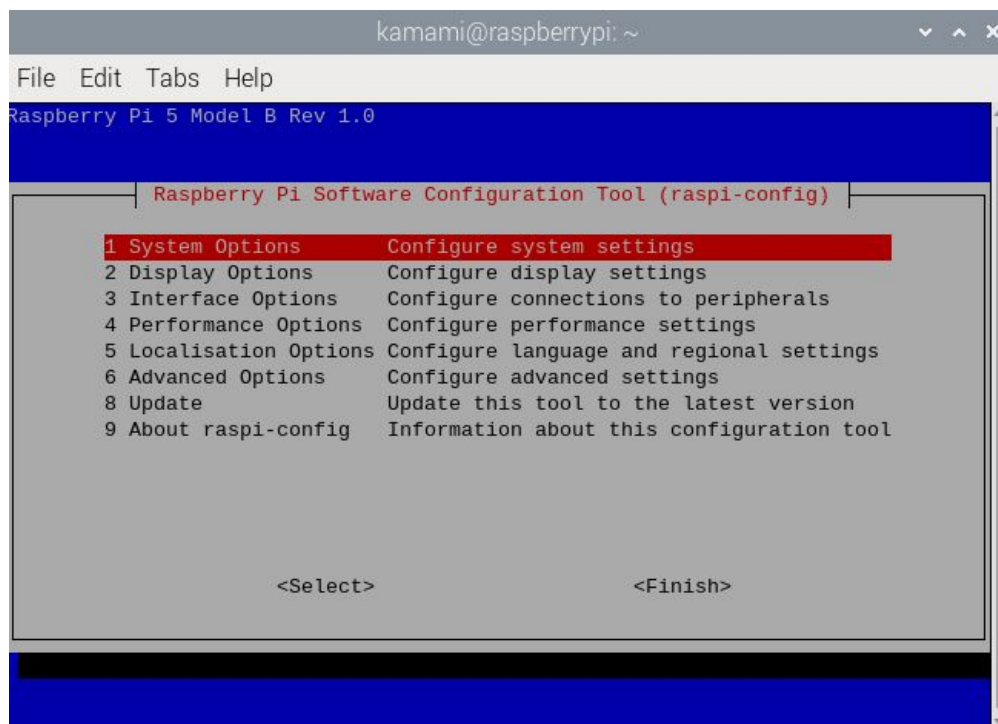
Po zakończeniu działań konieczny jest restart systemu, który wywołamy np. wpisując polecenie:

```
sudo reboot
```

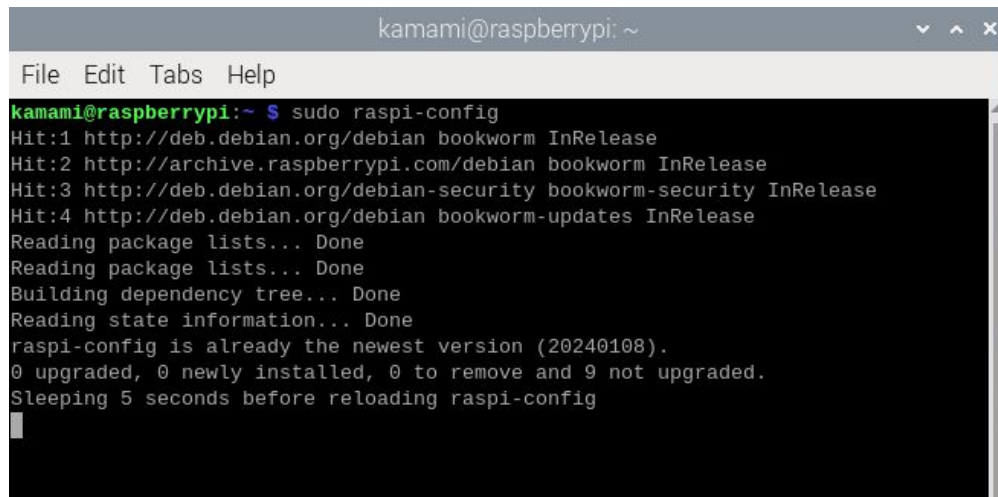
Aktualizacja narzędzia raspi-config

Otwieramy okno konsoli (Terminal) np. za pomocą kombinacji klawiszy *Ctrl+Alt+T* i wpisujemy:

```
sudo raspi-config
```



Przy pomocy klawiszy strzałek wskazujemy wiersz: **8 Update**. Czekamy na zakończenie wszystkich działań. Po chwili okno otworzy się ponownie.



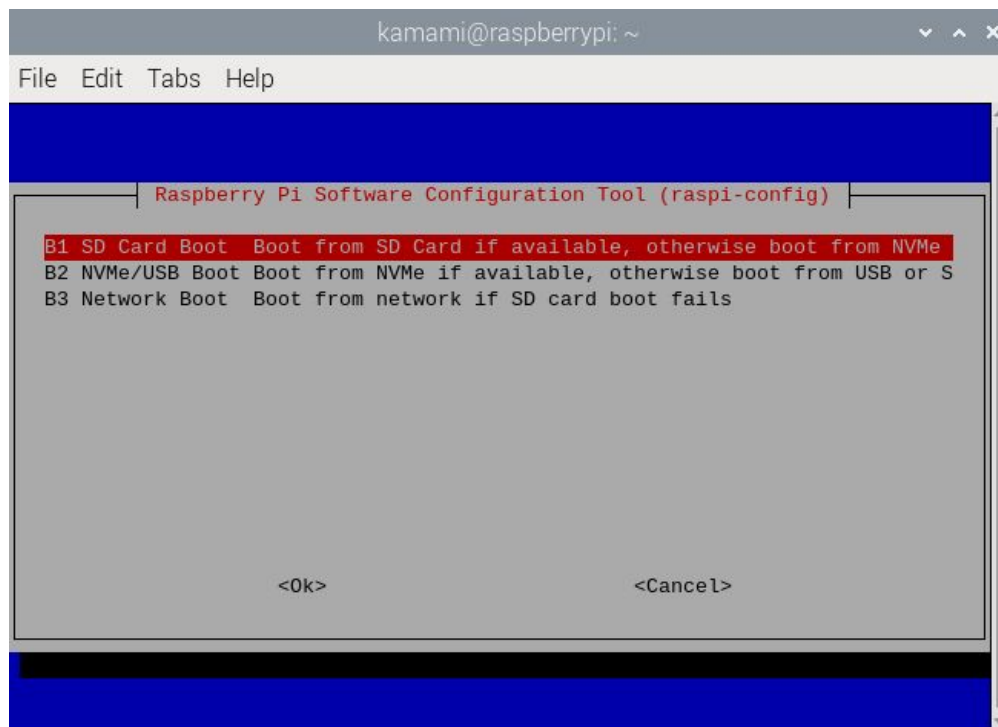
Ustawianie sekwencji bootowania

Sekwencja bootowania określa z jakich nośników (karta pamięci, pamięć USB, dysk NVME) i w jakiej kolejności Raspberry będzie uruchamiał system operacyjny.

Konfigurację sekwencji umożliwia narzędzie `raspi-config`, które uruchamiamy wpisując w konsoli:

```
sudo raspi-config
```

Teraz wybieramy wiersz: **6 Advanced Options**, a następnie **A4 Boot Order**. Do wyboru są trzy opcje, które widać na poniższym zrzucie ekranu:



Opcja pierwsza: **B1 SD Card Boot** - powoduje, że system będzie w pierwszej kolejności wczytywany z karty pamięci, a jeśli karta nie zostanie wykryta, to system będzie wczytywany z dysku NVME. Jest to bezpieczne ustawienie i należy je wybrać.

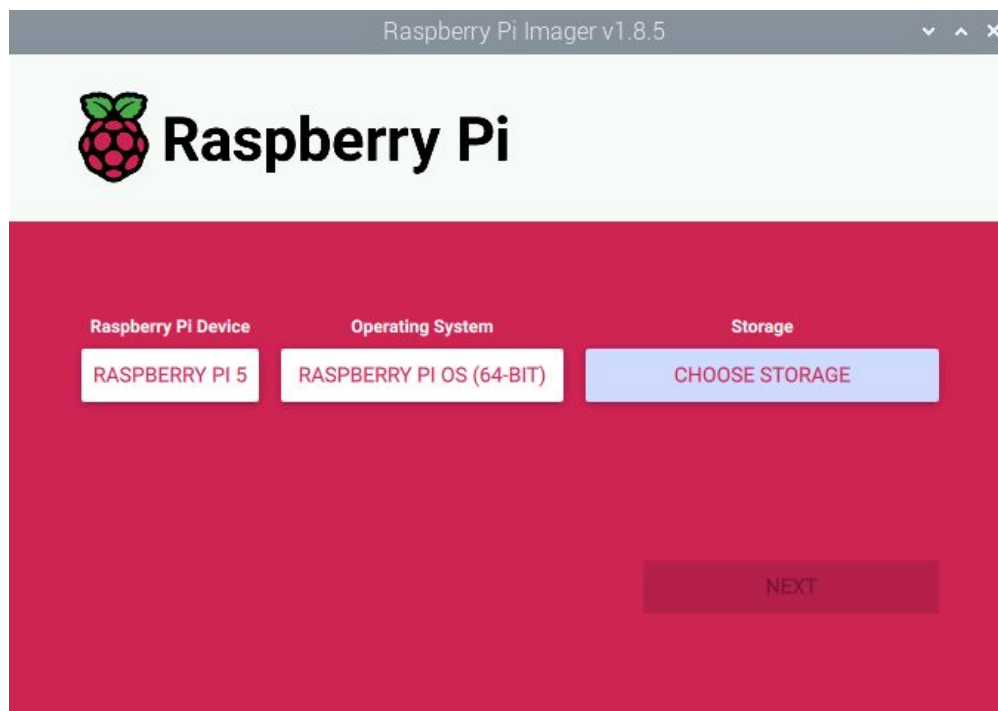
Potwierdzamy zmiany, a w oknie początkowym wybieramy polecenie **<Finish>** (w dolnej części okna). Na pytanie o restart systemu wybieramy **<Yes>**.

Instalowanie systemu na dysku NVME M.2

Montujemy adapter KAmoD RPi5 PCIe-M.2 wraz z zamontowanym dyskiem NVME M.2. Dokładne instrukcje tych czynności znajdziemy w rozdziale „Montaż adaptera KAmoD RPi5 PCIe-M.2 na płycie Raspberry Pi 5” oraz „Montaż Dysku NVME M.2 w adapterze KAmoD RPi5 PCIe-M.2”.

Uruchamiamy Raspberry Pi 5, otwieramy okno konsoli (Terminal) np. za pomocą kombinacji klawiszy *Ctrl+Alt+T* i wpisujemy:

```
sudo rpi-imager
```



W oknie które się pojawi wybieramy:

- model komputerka (Raspberry Pi Device): **RASPBerry PI 5**,
- system operacyjny (Operating System): **RASPBerry PI OS (64-BIT)**,
- dysk (Storage): tu wskazujemy dysk NVME M.2, który został zamontowany w adapterze KAmoD RPi5 PCIe-M.2.

Dalszy przebieg instalowania obrazu systemu operacyjnego wygląda podobnie jak w przypadku karty pamięci i został dokładnie opisany w rozdziale „Przygotowanie karty pamięci z systemem operacyjnym dla Raspberry Pi 5”.

Uruchomienie systemu z dysku NVME M.2

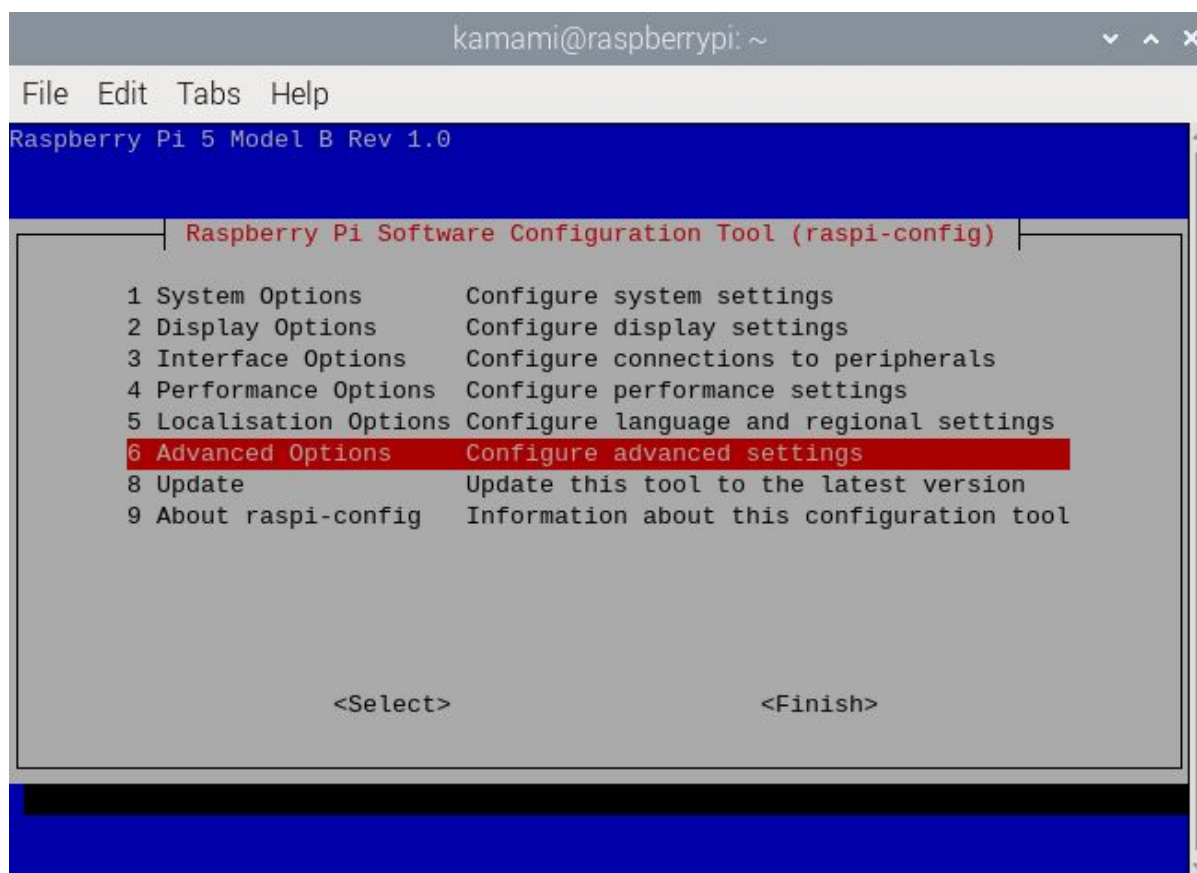
Gdy mamy zamontowany dysk NVME M.2 z zainstalowanym obrazem systemu operacyjnego, to przed uruchomieniem komputerka Raspberry Pi 5 usuwamy kartę pamięci z dedykowanego gniazda – nie będzie już potrzebna. Kartę pamięci możemy usunąć/zamontować, tylko wtedy, gdy komputer jest wyłączony i odłączony od zasilania.

Pierwsze uruchomienie nowego systemu operacyjnego potrwa nieco dłużej niż kolejne, ale po chwili pokaże się pulpit gotowego do pracy Rasbiana zainstalowanego na dysku NVME M.2. Jest to nowy system operacyjny i nie zawiera zmian, które wprowadziliśmy w systemie znajdującym się na karcie pamięci. Zatem musimy przeprowadzić aktualizację systemu operacyjnego oraz narzędzia raspi-config, tak jak to zostało opisane wcześniej. Nie musimy aktualizować zawartości pamięci EEPROM – nie zmieniła się po zmianie systemu operacyjnego.

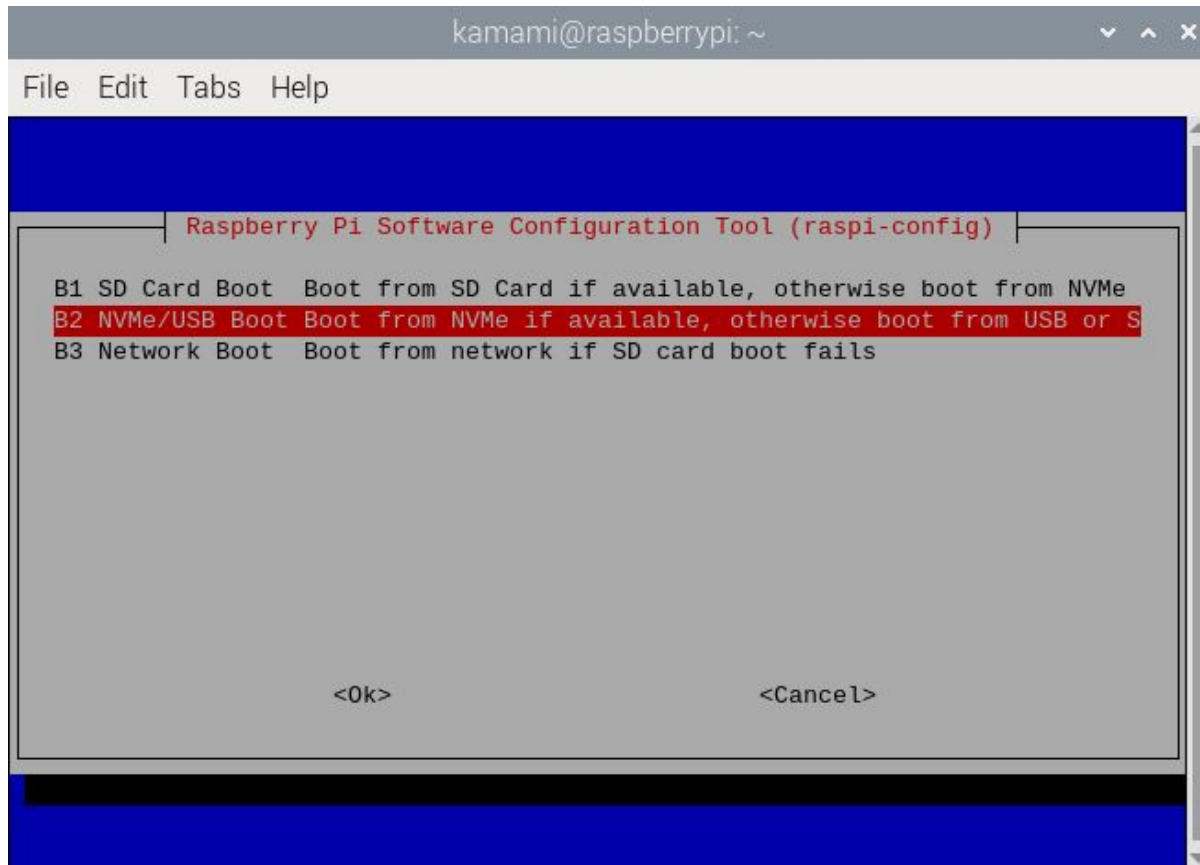
Na koniec, przy pomocy narzędzia raspi-config, należy ustawić sekwencję bootowania na rozruch z dysku NVME, co nieco przyspieszy uruchamianie systemu, ponieważ zostanie pominięta próba odczytu z karty pamięci.

Otwieramy okno konsoli (Terminal) np. za pomocą kombinacji klawiszy *Ctrl+Alt+T* i wpisujemy:

```
sudo raspi-config
```



Teraz wybieramy wiersz: **6 Advanced Options**, a następnie **A4 Boot Order**. Wybieramy opcję: **B2 NVMe/USB Boot**, która powoduje, że system będzie w pierwszej kolejności wczytywany z dysku NVME.



Potwierdzamy zmiany, a w oknie początkowym wybieramy polecenie **<Finish>** (w dolnej części okna). Na pytanie o restart systemu wybieramy **<Yes>**.

Informacje dodatkowe

Dysk NVME M.2 nie jest widoczny w systemie

W sytuacji, gdy dysk nie jest widoczny w systemie, np. program rpi-imager nie ma wśród dostępnych nośników dołączonego dysku, należy w pierwszej kolejności sprawdzić poprawność działania interfejsu PCIe. Można tego dokonać analizując komunikaty startowe ładowania systemu. Należy otworzyć okno konsoli (Terminal) np. za pomocą kombinacji klawiszy **Ctrl+Alt+T** i wpisać:

```
dmesg | grep pcie
```

Wyświetli się treść podobna do tej, która znajduje się na poniższym zrzucie ekranu:

```
File Edit Tabs Help
kamami@raspberrypi:~ $ dmesg | grep pcie
[ 0.000000] Kernel command line: reboot=w coherent_pool=1M 8250.nr_uarts=1 pci=pcie_bus_safe snd_bcm28
000000 console=ttyAMA10,115200 console=tty1 root=PARTUUID=c0b784bf-02 rootfstype=ext4 fsck.repair=yes rd
[ 0.393740] brcm-pcie 1000110000.pcie: host bridge /axi/pcie@110000 ranges:
[ 0.393748] brcm-pcie 1000110000.pcie: No bus range found for /axi/pcie@110000, using [bus 00-ff]
[ 0.393759] brcm-pcie 1000110000.pcie: MEM 0x1b00000000..0x1bfffffffb -> 0x0000000000
[ 0.393765] brcm-pcie 1000110000.pcie: MEM 0x1800000000..0x1affffffff -> 0x0400000000
[ 0.393771] brcm-pcie 1000110000.pcie: IB MEM 0x0000000000..0x0fffffffb -> 0x1000000000
[ 0.394949] brcm-pcie 1000110000.pcie: setting SCB_ACCESS_EN, READ_UR_MODE, MAX_BURST_SIZE
[ 0.394955] brcm-pcie 1000110000.pcie: Forcing gen 2
[ 0.394992] brcm-pcie 1000110000.pcie: PCI host bridge to bus 0000:00
[ 0.503755] brcm-pcie 1000110000.pcie: link up, 5.0 GT/s PCIe x1 (!SSC)
[ 0.515874] pcieport 0000:00:00.0: enabling device (0000 -> 0002)
[ 0.515910] pcieport 0000:00:00.0: PME: Signaling with IRQ 39
[ 0.515974] pcieport 0000:00:00.0: AER: enabled with IRQ 39
[ 0.516133] brcm-pcie 1000110000.pcie: clkreq control enabled
[ 0.516199] brcm-pcie 1000120000.pcie: host bridge /axi/pcie@120000 ranges:
[ 0.516203] brcm-pcie 1000120000.pcie: No bus range found for /axi/pcie@120000, using [bus 00-ff]
[ 0.516212] brcm-pcie 1000120000.pcie: MEM 0x1f00000000..0x1fffffffb -> 0x0000000000
[ 0.516217] brcm-pcie 1000120000.pcie: MEM 0x1c00000000..0x1effffffff -> 0x0400000000
[ 0.516225] brcm-pcie 1000120000.pcie: IB MEM 0x1f00000000..0x1f003fffff -> 0x0000000000
[ 0.516230] brcm-pcie 1000120000.pcie: IB MEM 0x0000000000..0x0fffffffb -> 0x1000000000
[ 0.517294] brcm-pcie 1000120000.pcie: setting SCB_ACCESS_EN, READ_UR_MODE, MAX_BURST_SIZE
[ 0.517302] brcm-pcie 1000120000.pcie: Forcing gen 2
[ 0.517336] brcm-pcie 1000120000.pcie: PCI host bridge to bus 0001:00
[ 0.623754] brcm-pcie 1000120000.pcie: link up, 5.0 GT/s PCIe x4 (!SSC)
[ 0.635866] pcieport 0001:00:00.0: enabling device (0000 -> 0002)
[ 0.635904] pcieport 0001:00:00.0: PME: Signaling with IRQ 45
[ 0.635961] pcieport 0001:00:00.0: AER: enabled with IRQ 45
[ 1.152576] input: Logitech USB Optical Mouse as /devices/platform/axi/1000120000.pcie/1f00200000.usb/l
[ 1.676526] input: Logitech USB Keyboard as /devices/platform/axi/1000120000.pcie/1f00300000.usb/xhci-
[ 1.804428] input: Logitech USB Keyboard Consumer Control as /devices/platform/axi/1000120000.pcie/1f0
[ 1.863807] input: Logitech USB Keyboard System Control as /devices/platform/axi/1000120000.pcie/1f00
kamami@raspberrypi:~ $
```

Widać tam informację o interfejsie PCIe x4, który jest wewnętrznym interfejsem komputerka Raspberry Pi 5 i nie jest dostępny dla użytkownika. Powinna również znajdować się informacja o interfejsie PCIe x1, który odpowiada za połączenie z dyskiem NVME M.2:

```
File Edit Tabs Help
kamami@raspberrypi:~ $ dmesg | grep pcie
[ 0.000000] Kernel command line: reboot=w coherent_pool=1M 8250.nr_ua
rts=1 pci=pcie_bus_safe snd_bcm2835.enable_via_firmware=1 rootfs=
000000 console=ttyAMA10,115200 console=tty1 root=PARTUUID=c0b784bf-02
rootfstype=ext4 fsck.repair=yes
[ 0.393740] brcm-pcie 1000110000.pcie: host bridge /axi/pcie@110000
ranges:
[ 0.393748] brcm-pcie 1000110000.pcie: No bus range found for /axi/pcie@110000, using [bus 00-ff]
[ 0.393759] brcm-pcie 1000110000.pcie: MEM 0x1b00000000..0x1bfffffffb -> 0x0000000000
[ 0.393765] brcm-pcie 1000110000.pcie: MEM 0x1800000000..0x1affffffff -> 0x0400000000
[ 0.393771] brcm-pcie 1000110000.pcie: IB MEM 0x0000000000..0x0fffffffb -> 0x1000000000
[ 0.394949] brcm-pcie 1000110000.pcie: setting SCB_ACCESS_EN, READ_UR_MODE, MAX_BURST_SIZE
[ 0.394955] brcm-pcie 1000110000.pcie: Forcing gen 2
[ 0.394992] brcm-pcie 1000110000.pcie: PCI host bridge to bus 0000:00
[ 0.503755] brcm-pcie 1000110000.pcie: link up, 5.0 GT/s PCIe x1 (!SSC)
[ 0.515874] pcieport 0000:00:00.0: enabling device (0000 -> 0002)
[ 0.515910] pcieport 0000:00:00.0: PME: Signaling with IRQ 39
[ 0.515974] pcieport 0000:00:00.0: AER: enabled with IRQ 39
```

W sytuacji, gdy interfejs PCIe x1 nie zostanie poprawnie uruchomiony, znajdziemy wpis podobny do takiego:

```
File Edit Tabs Help
kamami@raspberrypi:~ $ dmesg | grep pcie
[ 0.000000] Kernel command line: reboot=w coherent_pool=1M 8250.nr_ua
rts=1 pci=pcie_bus_safe snd_bcm2835.enable_via_firmware=1 rootfs=
000000 console=ttyAMA10,115200 console=tty1 root=PARTUUID=c0b784bf-02
rootfstype=ext4 fsck.repair=yes
[ 0.393958] brcm-pcie 1000110000.pcie: host bridge /axi/pcie@110000
ranges:
[ 0.393966] brcm-pcie 1000110000.pcie: No bus range found for /axi/pcie@110000, using [bus 00-ff]
[ 0.393977] brcm-pcie 1000110000.pcie: MEM 0x1b00000000..0x1bfffffffb -> 0x0000000000
[ 0.393983] brcm-pcie 1000110000.pcie: MEM 0x1800000000..0x1affffffff -> 0x0400000000
[ 0.393989] brcm-pcie 1000110000.pcie: IB MEM 0x0000000000..0x0fffffffb -> 0x1000000000
[ 0.395165] brcm-pcie 1000110000.pcie: setting SCB_ACCESS_EN, READ_UR_MODE, MAX_BURST_SIZE
[ 0.395172] brcm-pcie 1000110000.pcie: Forcing gen 2
[ 0.395207] brcm-pcie 1000110000.pcie: PCI host bridge to bus 0000:00
[ 0.823755] brcm-pcie 1000110000.pcie: link down
[ 0.828559] pcieport 0000:00:00.0: PME: Signaling with IRQ 39
[ 0.828622] pcieport 0000:00:00.0: AER: enabled with IRQ 39
[ 0.828909] brcm-pcie 1000120000.pcie: host bridge /axi/pcie@120000
ranges:
```

Komunikat „Link down” oznacza, że podłączenie dysku nie powiodło się.

W tej sytuacji warto jeszcze zmodyfikować zawartość pliku konfiguracyjnego *config.txt*. W konsoli wpisujemy:

```
sudo nano /boot/firmware/config.txt
```

(we wcześniejszych wersjach systemu operacyjnego plik *config.txt* był umieszczony bezpośrednio w katalogu /boot)

```
File Edit Tabs Help
kamami@raspberrypi:~ $ sudo nano /boot/firmware/config.txt
```

Na końcu pliku (przewijamy strzałkami do samego dołu) należy dopisać linię:

```
dtparam=nvme
```

Następnie należy zapisać zmiany za pomocą klawiszy *Ctrl+O*, zamknąć edytor za pomocą klawiszy *Ctrl+X* i uruchomić system ponownie.

```
File Edit Tabs Help
GNU nano 7.2 /boot/firmware/config.txt *

# Run in 64-bit mode
arm_64bit=1

# Disable compensation for displays with overscan
disable_overscan=1

# Run as fast as firmware / board allows
arm_boost=1

[cm4]
# Enable host mode on the 2711 built-in XHCI USB controller.
# This line should be removed if the legacy DWC2 controller is required
# (e.g. for USB device mode) or if USB support is not required.
otg_mode=1

[all]

dtparam=nvme

File Name to Write: /boot/firmware/config.txt
^G Help      M-D DOS Format  M-A Append     M-B Backup File
^C Cancel    M-M Mac Format  M-P Prepend    ^T Browse
```

Jeżeli te działania nie przyniosą efektu, to znaczy, że podłączony dysk nie jest kompatybilny z komputerkiem Raspberry Pi 5.

Zwiększenie szybkości interfejsu PCIe

Interfejs PCIe komputerka Raspberry Pi 5 domyślnie uruchamia się w trybie **gen 2**, który pozwala na komunikację z maksymalną przepustowością 5 GT/s (Gigatransferów na sekundę). Istnieje sposób na uruchomienie trybu **gen 3**, który oferuje przepustowość do 8 GT/s. W tym celu należy zmodyfikować zawartość pliku konfiguracyjnego *config.txt*.

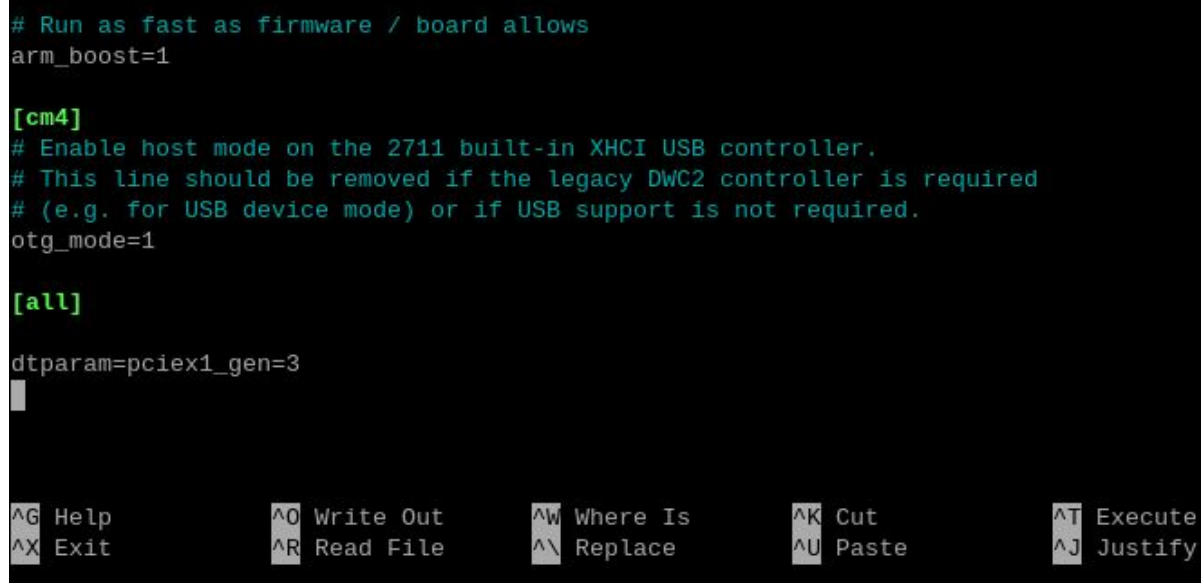
W konsoli wpisujemy:

```
sudo nano /boot/firmware/config.txt
```

(we wcześniejszych wersjach systemu operacyjnego plik *config.txt* był umieszczony bezpośrednio w katalogu */boot*).

Na końcu pliku (przewijamy strzałkami do samego dołu) należy dopisać linię:

```
dtparam=pciex1_gen=3
```



```
# Run as fast as firmware / board allows
arm_boost=1

[cm4]
# Enable host mode on the 2711 built-in XHCI USB controller.
# This line should be removed if the legacy DWC2 controller is required
# (e.g. for USB device mode) or if USB support is not required.
otg_mode=1

[all]

dtparam=pciex1_gen=3
```

Następnie należy zapisać zmiany za pomocą klawiszy *Ctrl+O*, zamknąć edytor za pomocą klawiszy *Ctrl+X* i uruchomić system ponownie.

Aby sprawdzić, czy modyfikacja zadziałała można przeanalizować komunikaty startowe ładowania systemu. Należy otworzyć okno konsoli (Terminal) np. za pomocą kombinacji klawiszy *Ctrl+Alt+T* i wpisać:

```
dmesg | grep pcie
```

Wyświetli się treść podobna do tej, która znajduje się na poniższym zrzucie ekranu:

File Edit Tabs Help

```
kamami@raspberrypi:~$ dmesg | grep pcie
[ 0.000000] Kernel command line: reboot=w coherent_pool=1M 8250.nr_uarts=1 pci=pcie_bus_safe snd_bcm
000000 console=ttyAMA10,115200 console=tty1 root=PARTUUID=c0b784bf-02 rootfstype=ext4 fsck.repair=yes
[ 0.393754] brcm-pcie 1000110000.pcie: host bridge /axi/pcie@110000 ranges:
[ 0.393762] brcm-pcie 1000110000.pcie: No bus range found for /axi/pcie@110000, using [bus 00-ff]
[ 0.393773] brcm-pcie 1000110000.pcie: MEM 0x1b00000000..0x1bfffffffb -> 0x0000000000
[ 0.393779] brcm-pcie 1000110000.pcie: MEM 0x1800000000..0x1affffffff -> 0x0400000000
[ 0.393785] brcm-pcie 1000110000.pcie: IB MEM 0x0000000000..0x0fffffffb -> 0x1000000000
[ 0.394961] brcm-pcie 1000110000.pcie: setting SCB_ACCESS_EN, READ_UR_MODE, MAX_BURST_SIZE
[ 0.394967] brcm-pcie 1000110000.pcie: Forcing gen 3
[ 0.395004] brcm-pcie 1000110000.pcie: PCI host bridge to bus 0000:00
[ 0.503759] brcm-pcie 1000110000.pcie: link up, 8.0 GT/s PCIe x1 (!SSC)
[ 0.515879] pcieport 0000:00:00.0: enabling device (0000 -> 0002)
[ 0.515917] pcieport 0000:00:00.0: PME: Signaling with IRQ 39
[ 0.515980] pcieport 0000:00:00.0: AER: enabled with IRQ 39
```

Widać tam wpisy: „Forcing gen 3” oraz „Link up, 8.0 GT/s PCIe x1”, który oznacza, że modyfikacja się powiodła. Jednak **nie gwarantuje to pełnej stabilności systemu w każdych warunkach.**

Ustawienie sekwencji bootowania poprzez edycję zawartości pamięci EEPROM

Jeżeli z jakiegoś powodu nie mamy możliwości zaktualizowania oprogramowania komputerka Raspberry Pi 5 to niektóre ustawienia dotyczące rozruchu systemu z dysku NVME mogą być niedostępne. W takiej sytuacji można ustawić odpowiednią sekwencję edytując bezpośrednio zawartość pamięci EEPROM komputerka Raspberry Pi 5.

Otwieramy okno konsoli (Terminal) np. za pomocą kombinacji klawiszy *Ctrl+Alt+T* i wpisujemy:

```
sudo rpi-eeeprom-config --edit
```

Linie rozpoczynającą się od „*BOOT_ORDER=*” zamieniamy na „*BOOT_ORDER=0xf461*”.

```
File Edit Tabs Help
GNU nano 7.2 /tmp/tmpial7icvm/boot.conf
[all]
BOOT_UART=1
BOOT_ORDER=0xf461
POWER_OFF_ON_HALT=0
```

Ten wpis ustawia bezpieczną sekwencję, która powoduje załadowanie systemu z dysku NVME M.2 jeśli nie zostanie wykryta kara pamięci. Następnie należy zapisać zmiany za pomocą klawiszy *Ctrl+O*, zamknąć edytor za pomocą klawiszy *Ctrl+X* i uruchomić system ponownie.

```
File Edit Tabs Help
procesorowiec@raspberrypi:~ $ sudo rpi-eeeprom-config --edit
Updating bootloader EEPROM
image: /lib/firmware/raspberrypi/bootloader-2712/latest/pieeprom-2024-01-22.bin
config_src: /boot/firmware/pieeprom.upd
config: /tmp/tmp6k9a2nwb/boot.conf
#####
#
[all]
BOOT_UART=1
BOOT_ORDER=0xf461
POWER_OFF_ON_HALT=0
#####
#
*** To cancel this update run 'sudo rpi-eeeprom-update -r' ***
*** CREATED UPDATE /tmp/tmp6k9a2nwb/pieeprom.upd ***
CURRENT: Fri 5 Jan 15:57:40 UTC 2024 (1704470260)
UPDATE: Mon 22 Jan 14:44:36 UTC 2024 (1705934676)
BOOTFS: /boot/firmware
'/tmp/tmp.7JSuY9vY1G' -> '/boot/firmware/pieeprom.upd'
Copying recovery.bin to /boot/firmware for EEPROM update
EEPROM updates pending. Please reboot to apply the update.
To cancel a pending update run "sudo rpi-eeeprom-update -r".
procesorowiec@raspberrypi:~ $
```


Tworzenie partycji na nowym dysku NVME

Dysk NVME M.2 nie musi być dyskiem głównym, na którym zostanie zainstalowany system operacyjny - może działać jako dysk dodatkowy do przechowywania dużych ilości danych. Gdy fabrycznie nowy dysk NVME M.2 dołączymy do Raspberry Pi, to konieczne będzie utworzenie partycji, aby stał się widoczny w systemie plików.

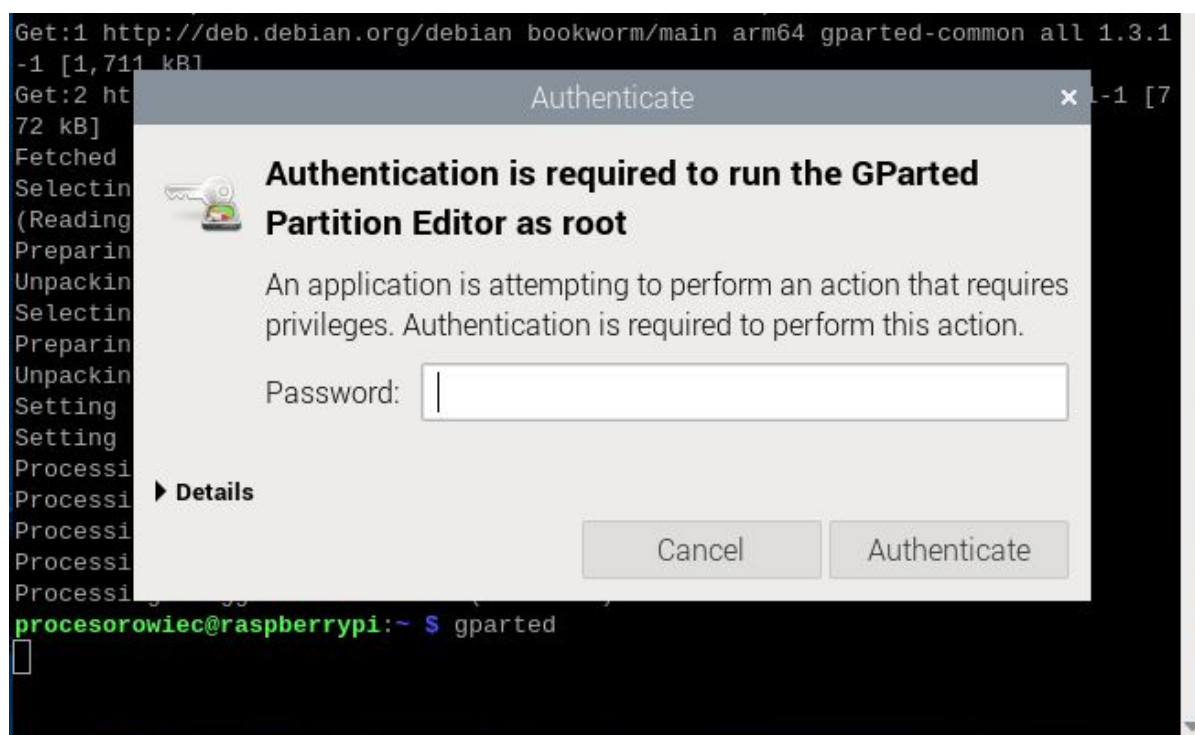
Raspberry Pi 5 pozwala na zainstalowanie narzędzia programowego GParted, które służy do tworzenia partycji i innych powiązanych operacji, a obsługiwane jest poprzez intuicyjny interfejs graficzny. Aby zainstalować GParted należy otworzyć okno konsoli (Terminal) np. za pomocą kombinacji klawiszy *Ctrl+Alt+T* i wpisać poniższą komendę uruchamiającą instalację:

```
sudo apt install gparted -y
```

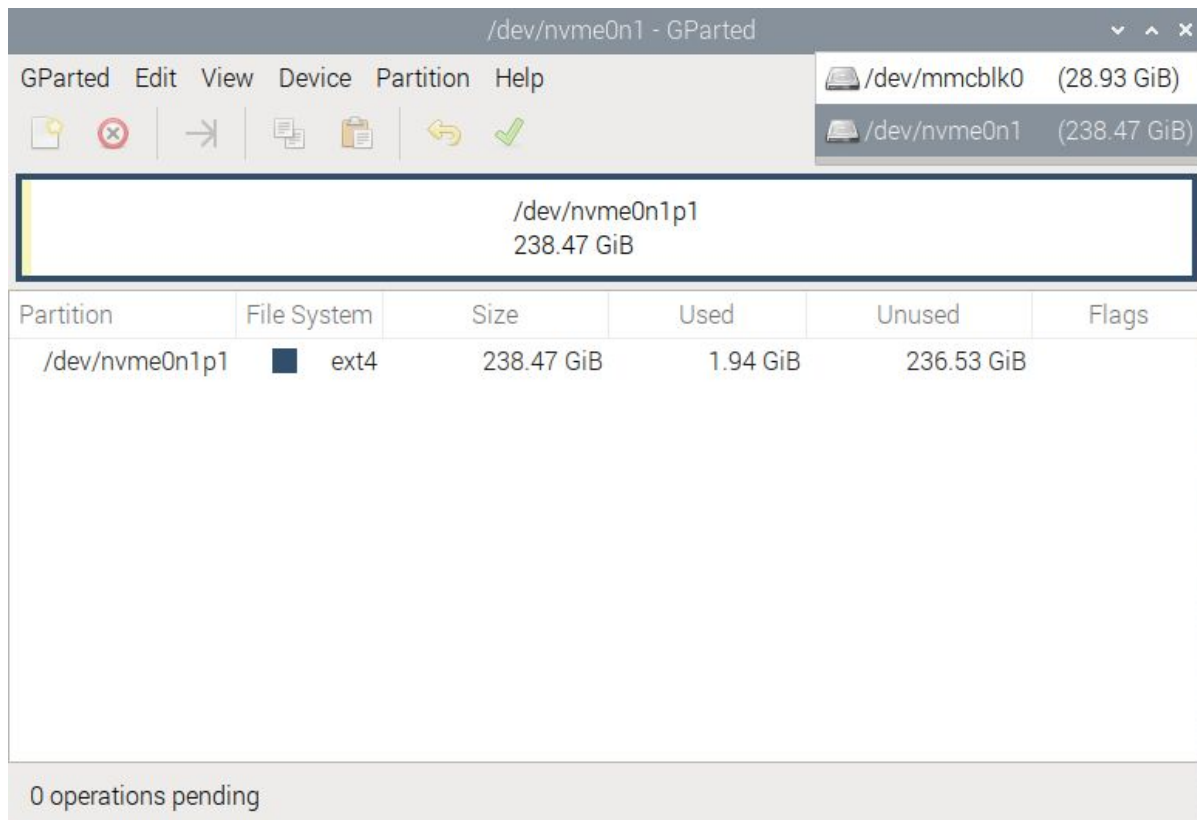
Po zakończeniu instalacji należy wpisać polecenie:

```
gparted
```

a w oknie, które się pojawi należy wpisać hasło dostępu do systemu:

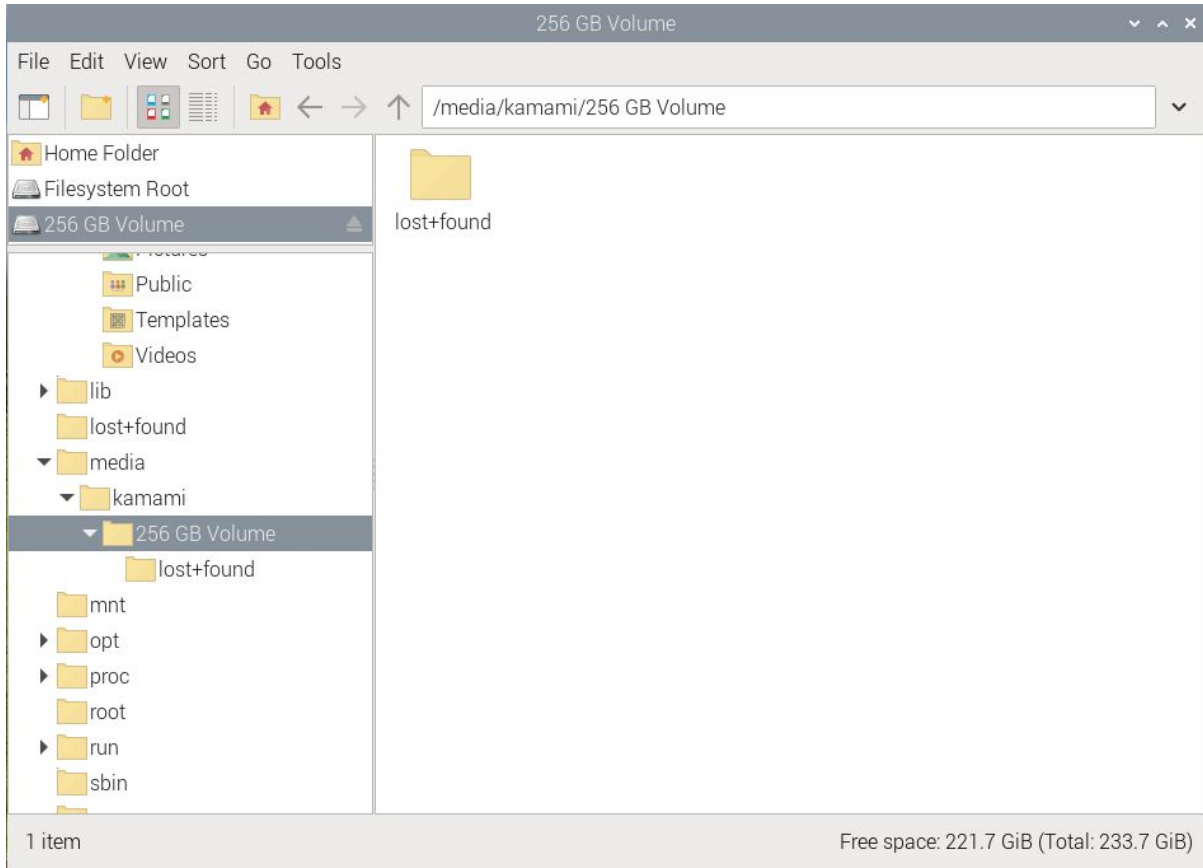


Główne okno programu zostało pokazane na poniższym zrzucie ekranu. Interfejs jest przejrzysty i intuicyjny:



W prawym górnym rogu znajduje się pole wyboru dysku, na którym wykonywane będą działania, w zakładce *Partition* znajdują się najważniejsze polecenia, takie jak utworzenie nowej partycji - *New*. Wszystkie operacje należy zatwierdzić zielonym przyciskiem.

Po utworzeniu partycji stanie się ona widoczna w menedżerze plików, podobnie jak na poniższym zrzucie ekranu:



Przygotowanie karty pamięci z systemem operacyjnym dla Raspberry Pi 5

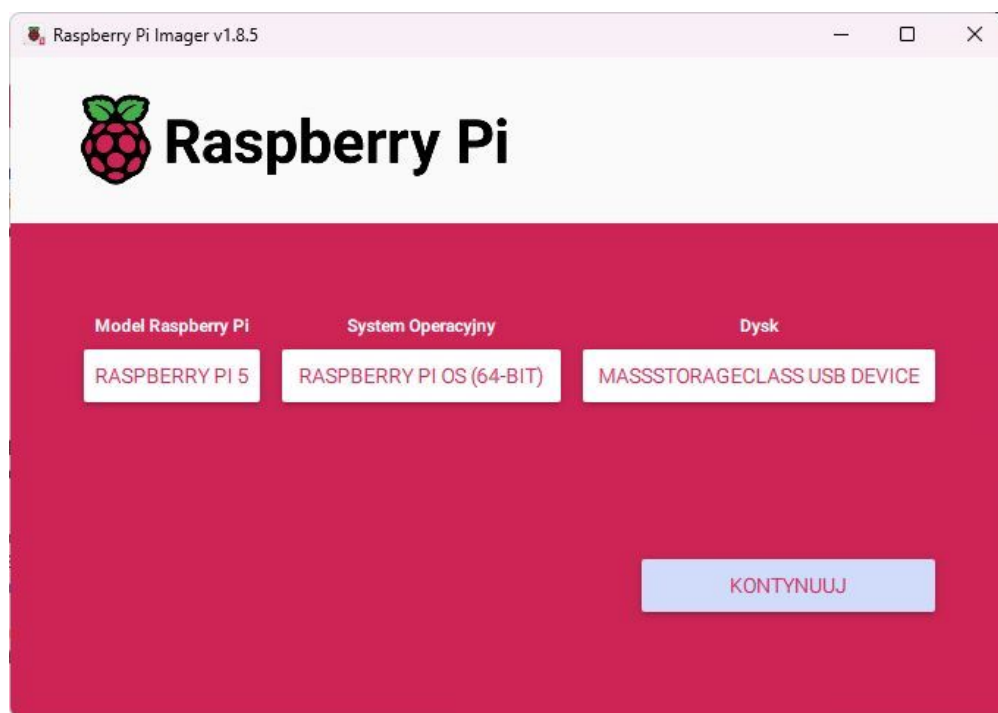
Oprogramowanie, które pozwala przygotować kartę pamięci można pobrać ze strony <https://www.raspberrypi.com/software/>.

Po jego zainstalowaniu i uruchomieniu wybieramy:

model komputerka: *RASPBERRY PI 5*,

system operacyjny: *RASPBERRY PI OS (64-BIT)*,

dysk: tu wskazujemy kartę pamięci (lub inną pamięć np. pendrive USB).



Klikamy *KONTYNUUJ*, a następnie wybieramy *EDYTUJ USTAWIENIA*.

Teraz warto ustawić login i hasło dostępu do instalowanego systemu, co jest konieczne dla zachowania jego bezpieczeństwa. Następnie należy podać parametry dostępu do Wi-Fi (jeżeli będziemy korzystali z takiej formy dostępu do internetu) oraz należy uzupełnić informacje dotyczące lokalizacji.

Konfiguracja systemu

OGÓLNE USŁUGI OPCJE

ustaw hostname: .local

Ustaw login i hasło

 Login:

 Hasło:

Skonfiguruj sieć Wi-Fi

 SSID:

 Hasło:

Pokaż hasło Ukryte SSID

 Kraj Wi-Fi: ▼

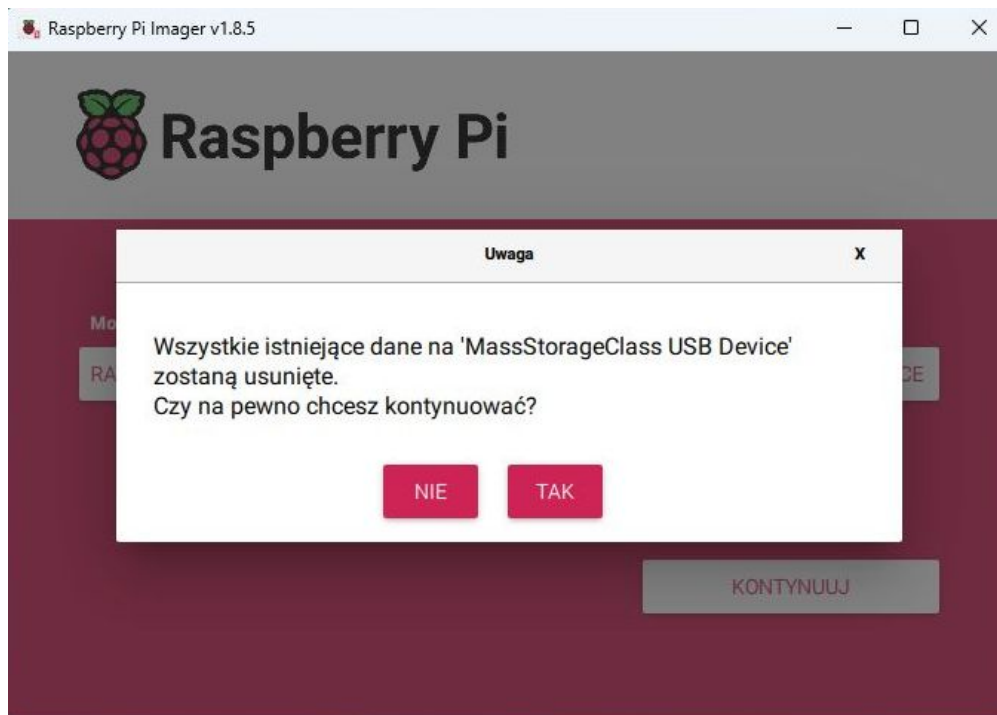
Ustawienia lokalizacji

 Strefa czasowa: ▼

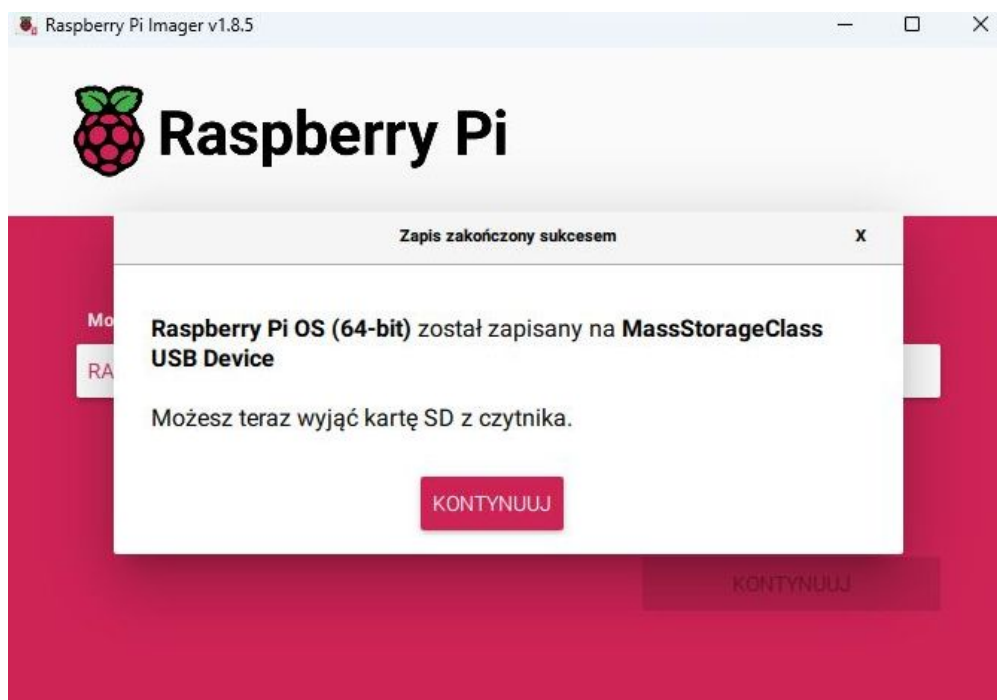
 Układ klawiatury: ▼

ZAPISZ

Aby zastosować ustawienia klikamy **ZAPISZ**, a w kolejnym oknie potwierdzamy **TAK**. Wyświetli się ostrzeżenie o tym, że wszystkie dane znajdujące się teraz na karcie pamięci zostaną usunięte. Klikamy **TAK**, jeśli zgadzamy się na to i chcemy kontynuować.



Instalowanie systemu na karcie pamięci potrwa od kilku do kilkunastu minut. Po zakończeniu działań, jeśli wszystko przebiegło pomyślnie, zostanie wyświetlony odpowiedni komunikat, a karta pamięci z systemem operacyjnym będzie gotowa.



Dodatkowe funkcje adaptera KAmoD RPi5 PCIe-M.2

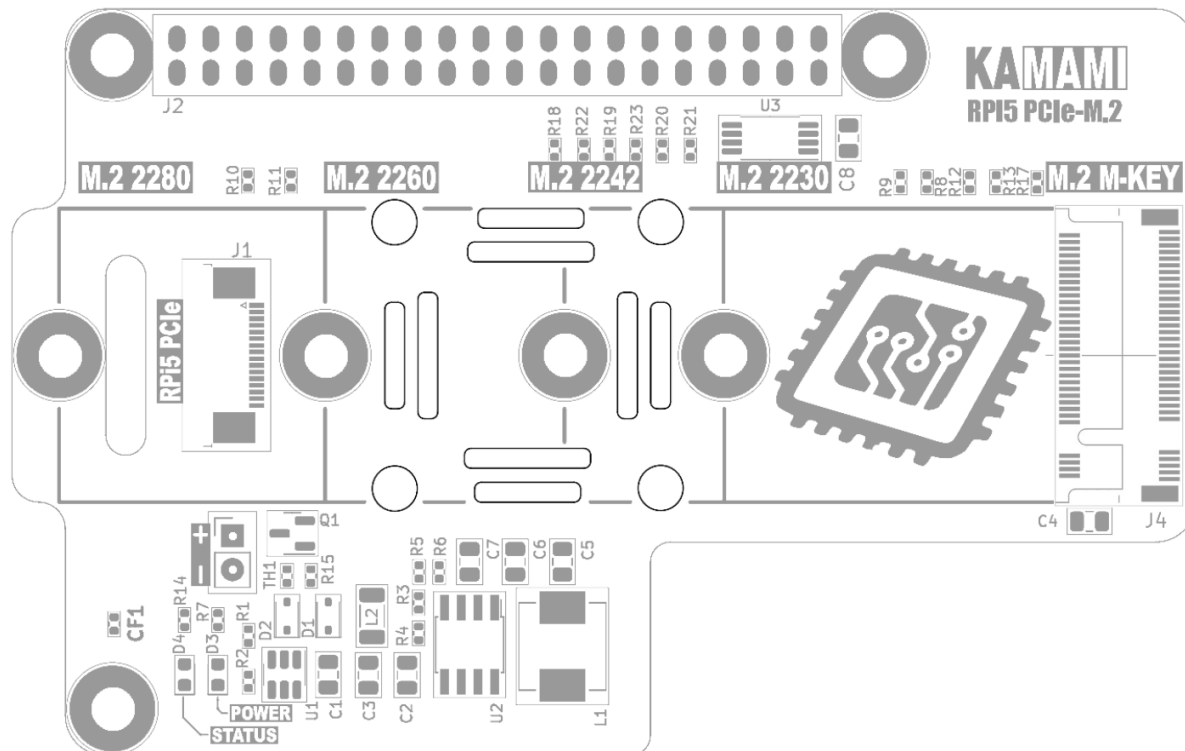
Podłączanie za pomocą samej taśmy

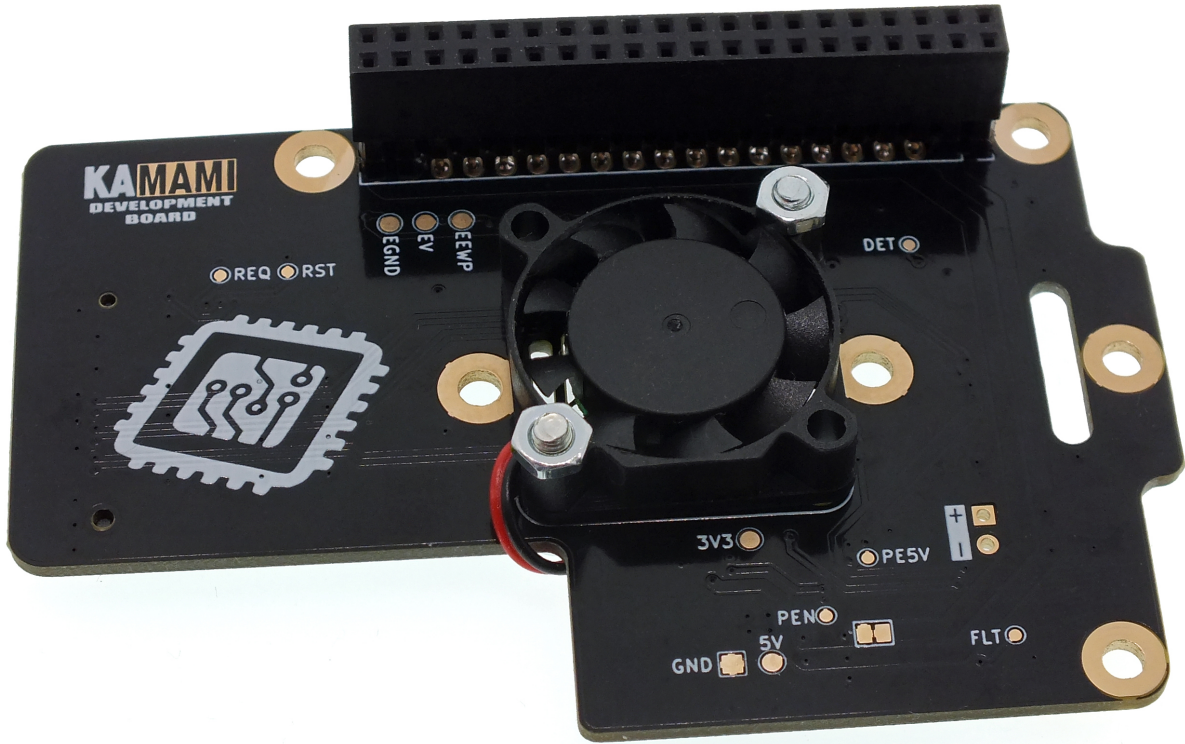
Adapter KAmoD RPi5 PCIe-M.2 może działać prawidłowo, gdy będzie podłączy z Raspberry Pi 5 tylko samą taśmą FFC 16/0,5. Jedyne warunki, który musi być wtedy spełniony jest taki, że pobór prądu przez dysk NVME M.2 nie może przekraczać 1 A.

Standard M.2 wymaga zapewnienia zasilania o napięciu 3,3V i prądzie o wartości do 3 A. Znamionowy pobór prądu, podawany przez producentów dysków, dotyczy pracy z pełnym interfejsem PCI Express GenX **x4**, przy maksymalnej wydajności dysku, natomiast podczas pracy z interfejsem **x1** i mniejszą wydajnością pobór prądu jest mniejszy. W praktyce wiele dysków może działać poprawnie przy ograniczeniu prądu do 1 A z interfejsem PCI Express Gen2 x1, zastosowanym w Raspberry Pi 5.

Zamontowanie wentylatora

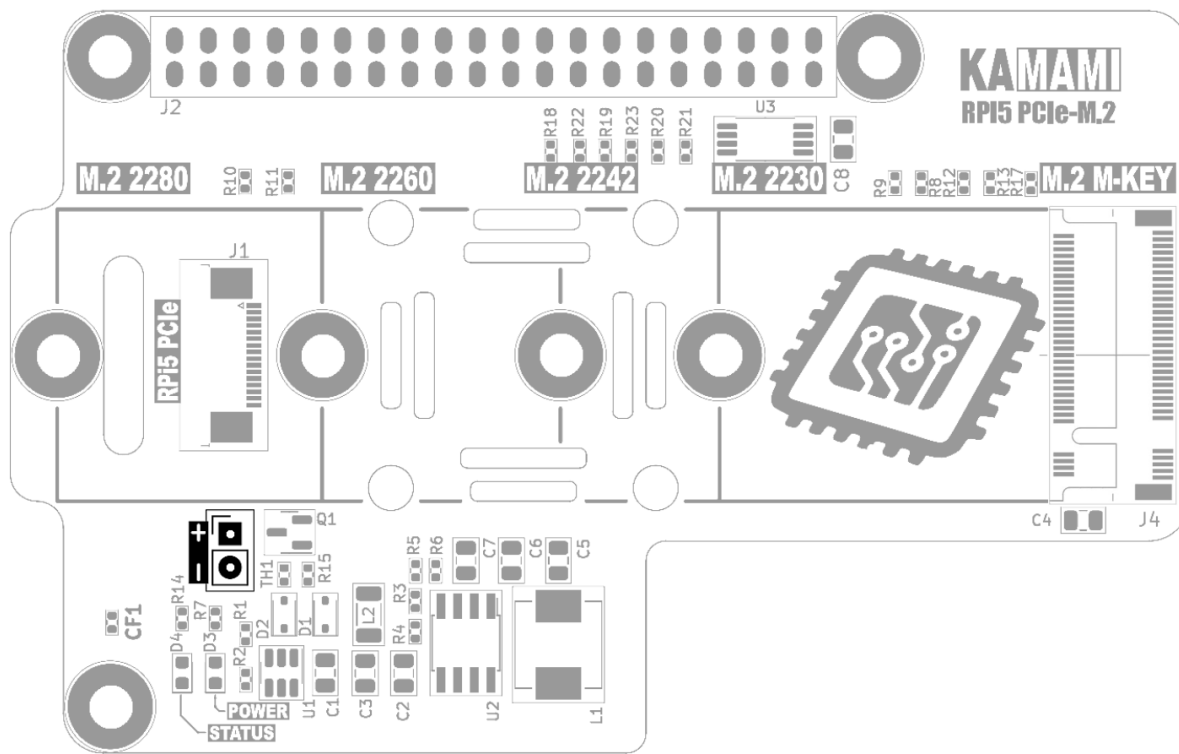
Na płytce adaptera KAmoD RPi5 PCIe-M.2 znajdują się podłużne otwory, które usprawniają cyrkulację powietrza chłodzącego komputerów Raspberry Pi 5. Konstrukcja adaptera nie blokuje możliwości zastosowania radiatora z wentylatorem dedykowanego dla Raspberry Pi 5. W przypadku, gdy dodatkowe chłodzenie nie jest zamontowane, istnieje możliwość zamontowania wentylatorka o rozmiarze 25x25 mm na dolnej stronie płytki adaptera. Na płytce znajdują się otwory umożliwiające umieszczenie odpowiednich śrub.





Zasilanie wentylatora

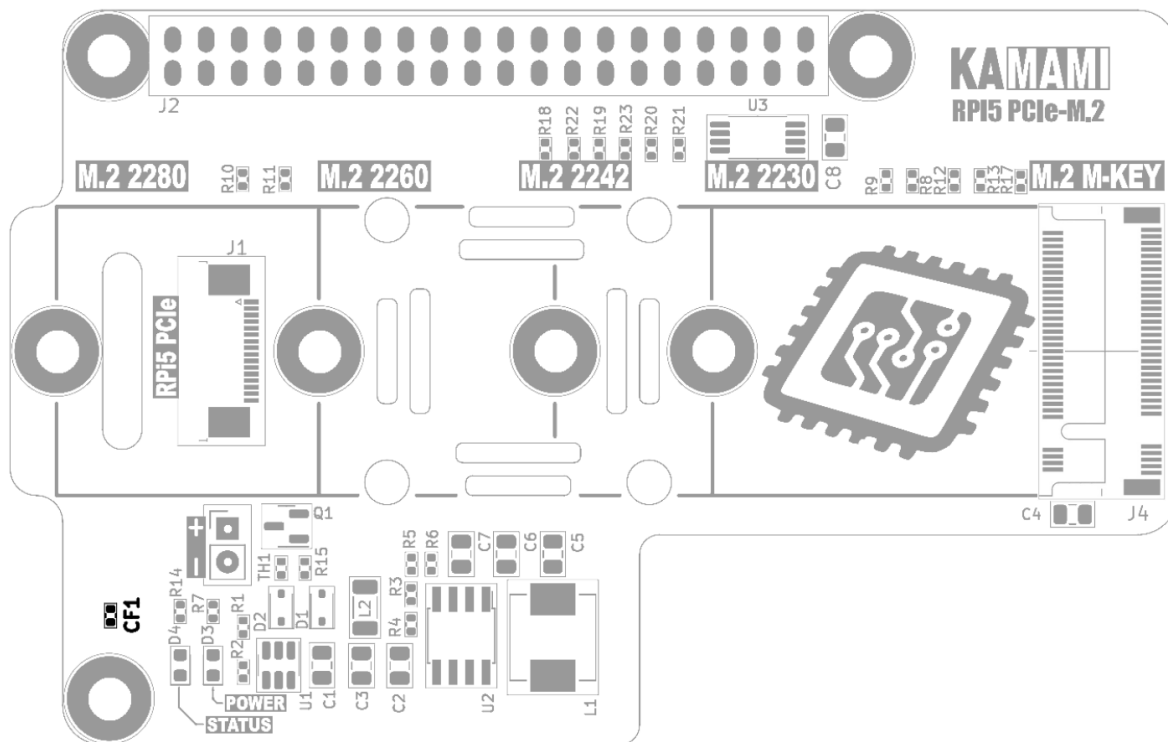
Na płytce znajduje się złącze, które może zasilac dodatkowy wentylator. Dostarcza napięcia, które jest regulowane wartością temperatury - im temperatura adaptera będzie wyższa, tym napięcie będzie bliższe wartości 5 V. Dodatkowo, wentylator będzie całkowicie wyłączany po wyłączeniu komputera (w stanie czuwania). Maksymalny prąd wentylatora nie może przekroczyć wartości 0,3 A.



Stałe zasilanie dysku NVME

Adapter doprowadza zasilanie do dysku NVME M.2 w sposób stały – dysk jest zasilany także wtedy, gdy komputer Raspberry Pi 5 jest wyłączony (w stanie czuwania). Niektóre dyski pozwalają na prawidłowe uruchomienie systemu operacyjnego tylko w takiej konfiguracji.

Adapter KAmoD RPi5 PCIe-M.2 oferuje, także inną konfigurację – zasilanie dysku będzie wyłączane po wyłączeniu Raspberry Pi 5 (w stanie czuwania). Takie rozwiązanie pozwala ograniczyć pobór energii całego zestawu w stanie czuwania, ale może zakłócić prawidłowe uruchamianie systemu. Aby włączyć tryb oszczędny należy usunąć z płytki element oznaczony CF1.





Zastrzegamy prawo do wprowadzania zmian bez uprzedzenia.

Oferowane przez nas płytki drukowane mogą się różnić od prezentowanej w dokumentacji, przy czym zmianom nie ulegają jej właściwości użytkowe.

BTC Korporacja gwarantuje zgodność produktu ze specyfikacją.

BTC Korporacja nie ponosi odpowiedzialności za jakiegokolwiek szkody powstałe bezpośrednio lub pośrednio w wyniku użycia lub nieprawidłowego działania produktu.

BTC Korporacja zastrzega sobie prawo do modyfikacji niniejszej dokumentacji bez uprzedzenia.