



KAmoRPI5 RTC (PL)



Rev. 20240712101809

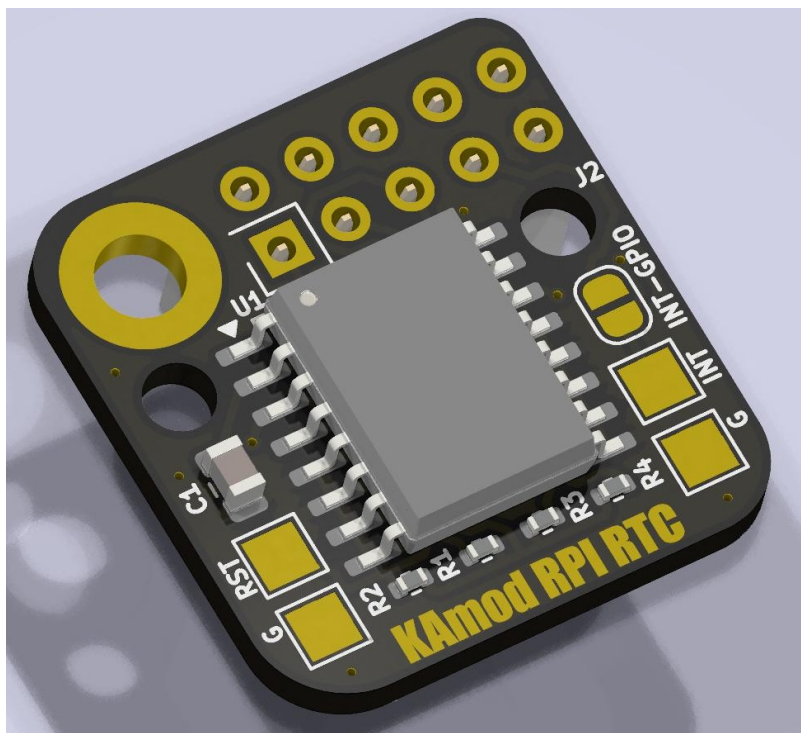
Źródło: [https://wiki.kamamilabs.com/index.php/KAmoRPI5_RTC_\(PL\)](https://wiki.kamamilabs.com/index.php/KAmoRPI5_RTC_(PL))

Spis treści

Opis	1
Podstawowe parametry	2
Wyposażenie standardowe	3
Schemat elektryczny	4
Złącze komunikacyjne zgodne z Raspberry Pi	5
Bateria podtrzymująca	6
Dodatkowe funkcje	7
Wymiary	8
Uruchomienie modułu z komputerkiem Raspberry Pi 5	9
Linki	15

Opis

Moduł precyzyjnego zegara czasu rzeczywistego – RTC, przeznaczony dla komputerków z serii Raspberry Pi. Bazuje na układzie DS3231, który został połączony z baterią podtrzymującą jego działanie przy zaniku zasilania głównego. Komunikacja z zegarem RTC odbywa się poprzez interfejs I2C.



Układ DS3231 jest kompletnym zegarem RTC, który wyróżnia się wysoką precyzją w odmierzaniu czasu. Doskonałe parametry uzyskano dzięki zintegrowaniu rezonatora kwarcowego w strukturze układu i zastosowaniu kompensacji temperaturowej. Na płytce znajduje się gniazdo na baterię typu CR1220, która umożliwia działanie zegara nawet przy braku zasilania głównego. Wymagane do poprawnego działania napięcie zasilania zawiera się w szerokim przedziale 2,3...5,5 V.

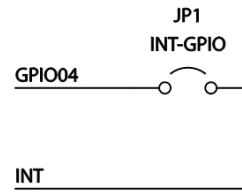
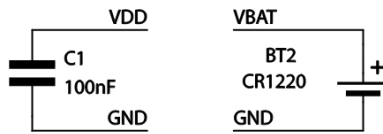
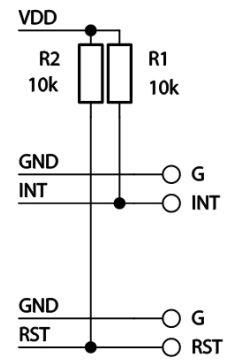
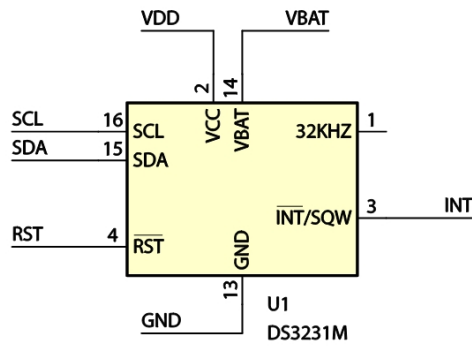
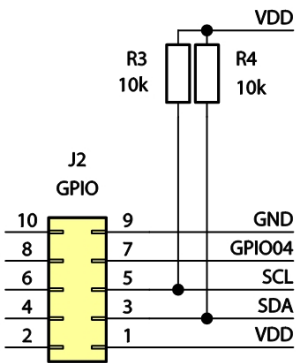
Podstawowe parametry

- Bazuje na układzie zegara RTC typu DS3231
- Zegar ze zintegrowanym rezonatorem kwarcowym i kompensacją temperaturową
- Deklarowany przez producenta układu DS3231 błąd odmierzenia czasu nie przekracza 2 minut na rok, przy pracy w temperaturze z zakresu -40...+85 °C
- Odlicza sekundy, minuty, godziny, dni tygodnia, miesiące oraz rok
- Komunikacja poprzez interfejs I2C (400 kHz)
- Zawiera gniazdo na baterię podtrzymującą typu CR1220
- Pobór prądu z baterii podtrzymującej nie przekracza 0,1 uA
- Opcjonalne wyjście przerwania (INT) / sygnału prostokątnego (SQW)
- Opcjonalne wyjście zerujące aktywowane niskim poziomem napięcia zasilania (RST)
- Wymaga zasilania o napięciu z zakresu 2,3...5,5 V
- Wymiary płytki 21x20 mm, wysokość ok. 8 mm (bez gniazda goldpin)

Wyposażenie standardowe

Kod	Opis
<ul style="list-style-type: none">• KAmoDRPI-RTC• 1 x Zestaw montażowy	<ul style="list-style-type: none">• Zmontowany i uruchomiony moduł• Zestaw zapewniający stabilne umocowanie (2 śrubki M2, 1 dystans)

Schemat elektryczny

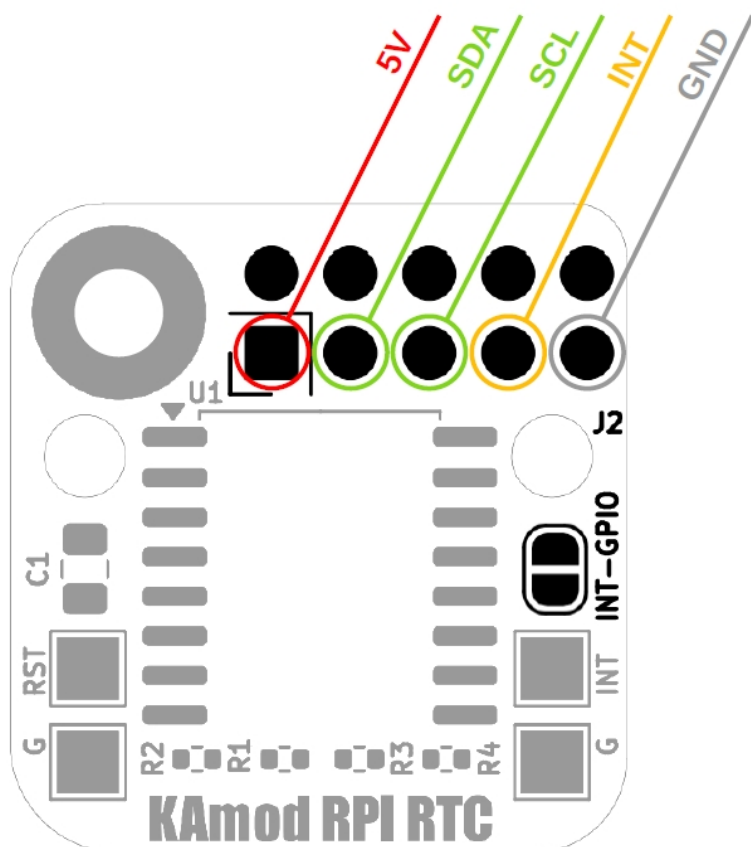


Złącze komunikacyjne zgodne z Raspberry Pi

Oznaczenie	Funkcja
J2 Gniazdo goldpin 2x5, 2,54 mm	<ul style="list-style-type: none"> • Wyprowadzone sygnały interfejsu I2C • Wyprowadzony dodatkowy sygnał przerwania INT • Wejście zasilania

Sygnały na złączu komunikacyjnym zostały tak rozmieszczone, aby odpowiadały sygnałom na złączu GPIO stosowanym w płytkach Raspberry Pi. Moduł KamodRPI-RTC zajmuje tylko pierwsze 10 szpilek, 40-stykowego złącza GPIO. Funkcje sygnałów na złączu **J2** są następujące:

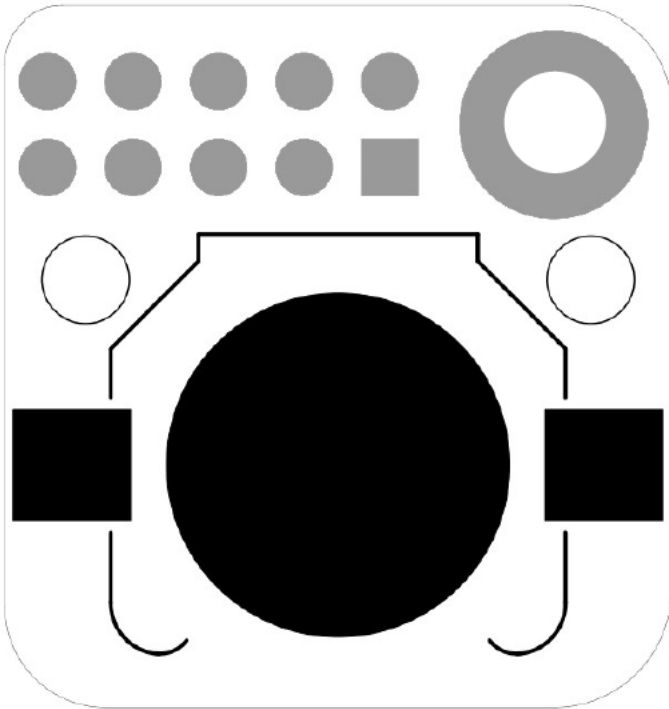
- styk nr 1: **+5V** - wejście zasilania o napięciu 2,3...5,5 V;
- styk nr 3: **SDA** - linia danych interfejsu I2C;
- styk nr 5: **SCL** - linia zegarowa interfejsu I2C;
- styk nr 7: **INT** - opcjonalne wyjście przerwania INT lub sygnału prostokątnego SQW. Sygnał jest dostępny dopiero po wykonaniu zworki z kropli cyny na padach oznaczonych INT-GPIO i trafia do portu GPIO04 na płytce Raspberry Pi;
- styk nr 9: **GND** - masa zasilania;
- styk nr 2, 4, 6, 8, 10: wyprowadzenia nie są podłączone.



Bateria podtrzymująca

Oznaczenie	Funkcja
BT1	<ul style="list-style-type: none">• Gniazdo na baterię typu CR1220• Umożliwia działanie zegara RTC po zaniku zasilania głównego

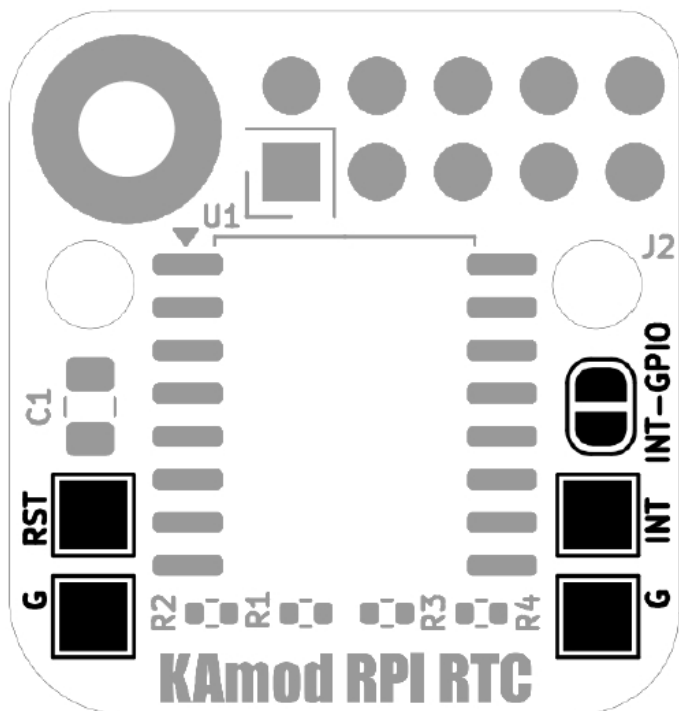
Gniazdo baterii znajduje się na dolnej stronie płytki modułu. Pobór prądu z baterii nie przekracza 0,1 uA, co umożliwia poprawną pracę zegara przez wiele lat.



Dodatkowe funkcje

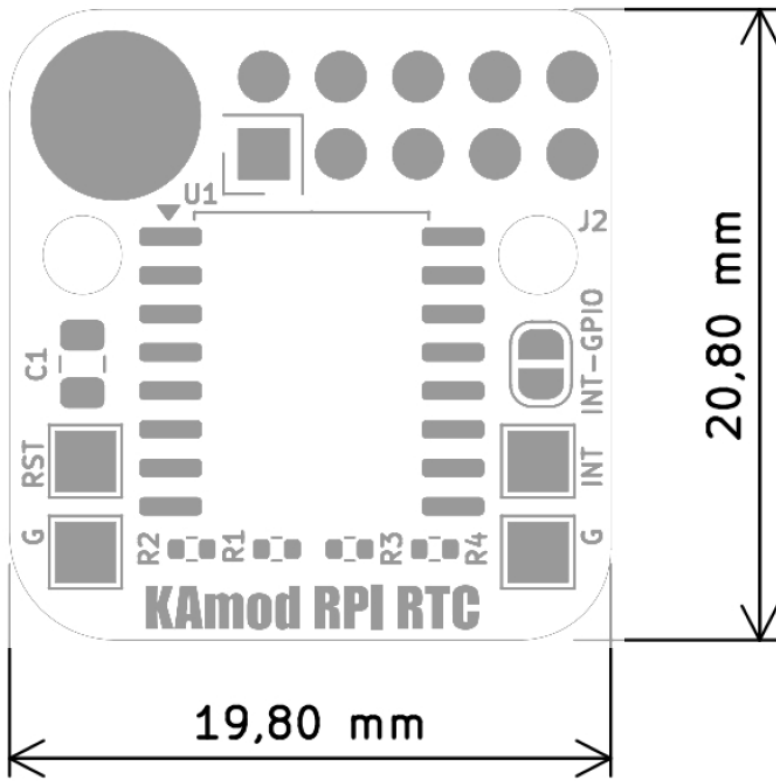
Oznaczenie	Funkcja
JP1 INT-GPIO	• Zwarcie padów za pomocą kropli cyny powoduje dołączenie wyjścia INT/SQW układu DS3231 do styku przypisanego do portu GPIO04 płytek Raspberry Pi
INT, G	• Pad oznaczony INT to wyjście INT/SQW układu DS3231, które może być aktywowane alarmem lub może być wyjściem przebiegu prostokątnego. Pad oznaczony G to masa.
RST, G	• Pad oznaczony RST to wyjście zerujące aktywowane niskim poziomem napięcia zasilania. Próg zadziałania jest ustawiony na ok. 2,57 V. Pad oznaczony G to masa.

Dokładne informacje dotyczące funkcjonalności i konfigurowania dodatkowych wyprowadzeń można znaleźć w dokumentacji układu DS3231.



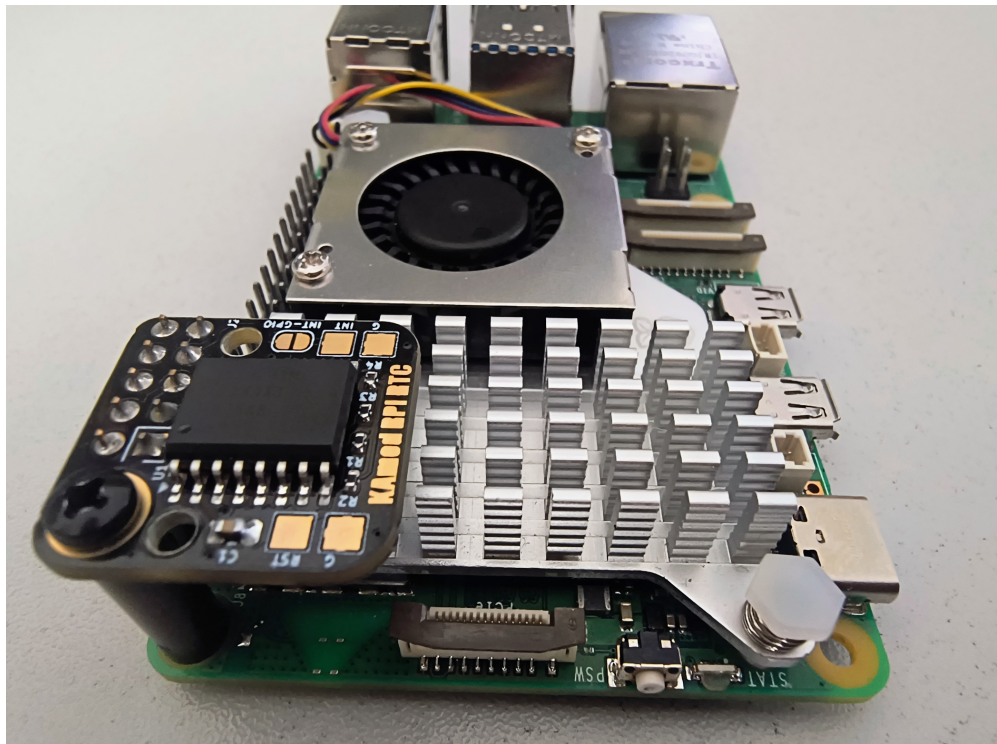
Wymiary

Wymiary modułu **KAmoRPI-RTC** to ok. 21x20 mm, a wysokość bez złącza to ok. 8 mm. Moduł jest kompatybilny z płytkami Raspberry Pi 5 oraz wcześniejszymi wersjami Raspberry Pi. Na płytce zostały umieszczone dodatkowe otwory, które mogą być przydatne w przypadku niestandardowych aplikacji modułu RTC.



Uruchomienie modułu z komputerkiem Raspberry Pi 5

Moduł KamodRPI-RTC montujemy na pierwszych 10 szpilkach złącza GPIO komputerka Raspberry Pi tak, jak pokazano na poniższym zdjęciu:



Konfigurowanie magistrali I2C

Pierwszym krokiem zawsze będzie odświeżenie repozytoriów i pakietów oprogramowania. Otwieramy konsolę np. poprzez kombinację klawiszy *Ctrl+Alt+T*, a następnie wpisujemy polecenia:

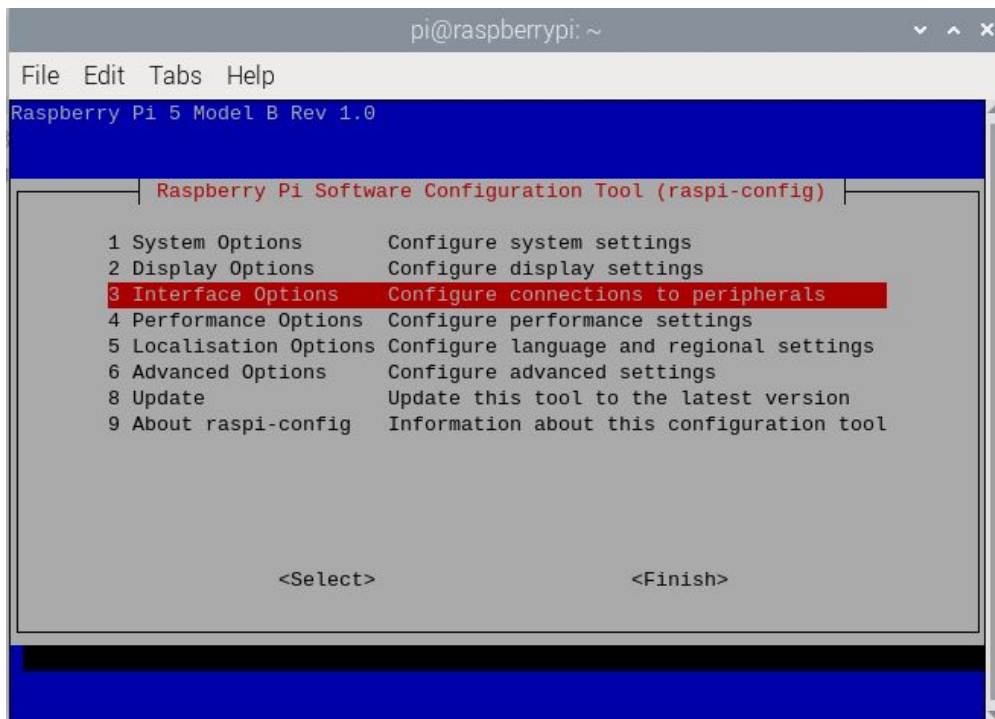
```
sudo apt-get update  
sudo apt-get upgrade
```

Teraz zainstalujemy pakiet oprogramowania ułatwiający pracę z interfejsem I2C. W tym celu wpisujemy polecenie:

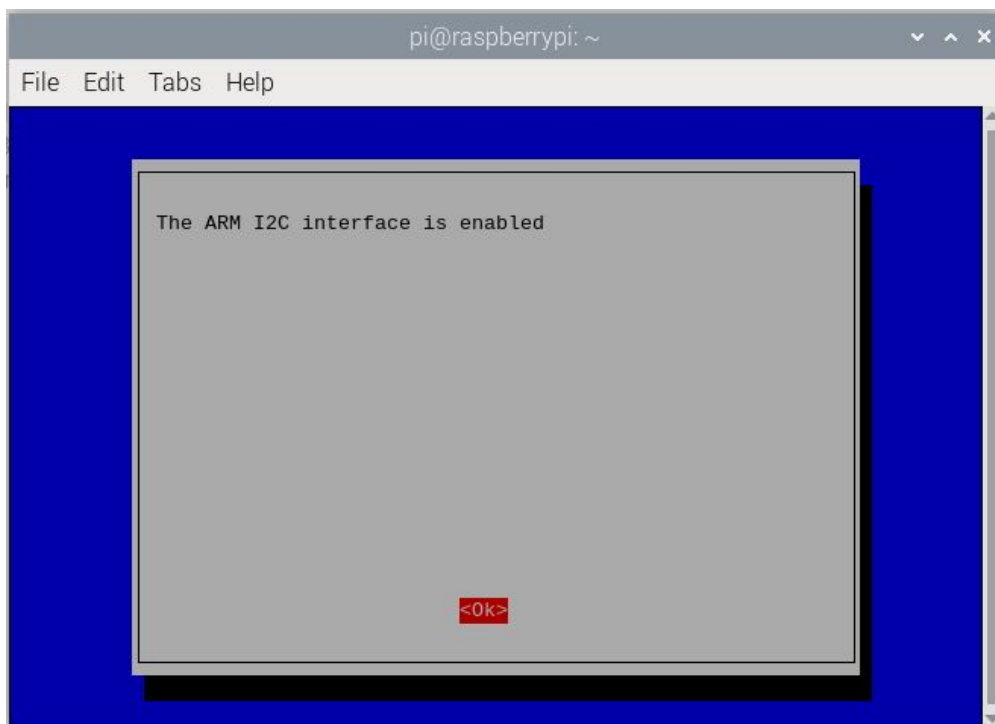
```
sudo apt-get install i2c-tools
```

Kolejnym zadaniem, które musimy wykonać jest włączenie kontrolera magistrali I2C. Uruchamiamy konfigurator Raspberry Pi poleceniem:

```
sudo raspi-config
```



Wybieramy opcję 3 (*Interface options*), a następnie wybieramy opcję I5 (*I2C*). Po zatwierdzeniu wszystkich zmian zobaczymy komunikat, że interfejs I2C jest gotowy.



W tym momencie, za pomocą programu *i2c-detect* możemy sprawdzić jakie urządzenia zostały podłączone do magistrali I2C. Wpisujemy:

```
i2cdetect -y 1
```

Efekt działania będzie podobny do tego pokazanego poniżej:

```
kamami@raspberrypi:~ $ i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  68  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
kamami@raspberrypi:~ $
```

Oznacza to, że na magistrali I2C znajduje się urządzenie z adresem 68 (hex) - to właśnie adres układu DS3231.

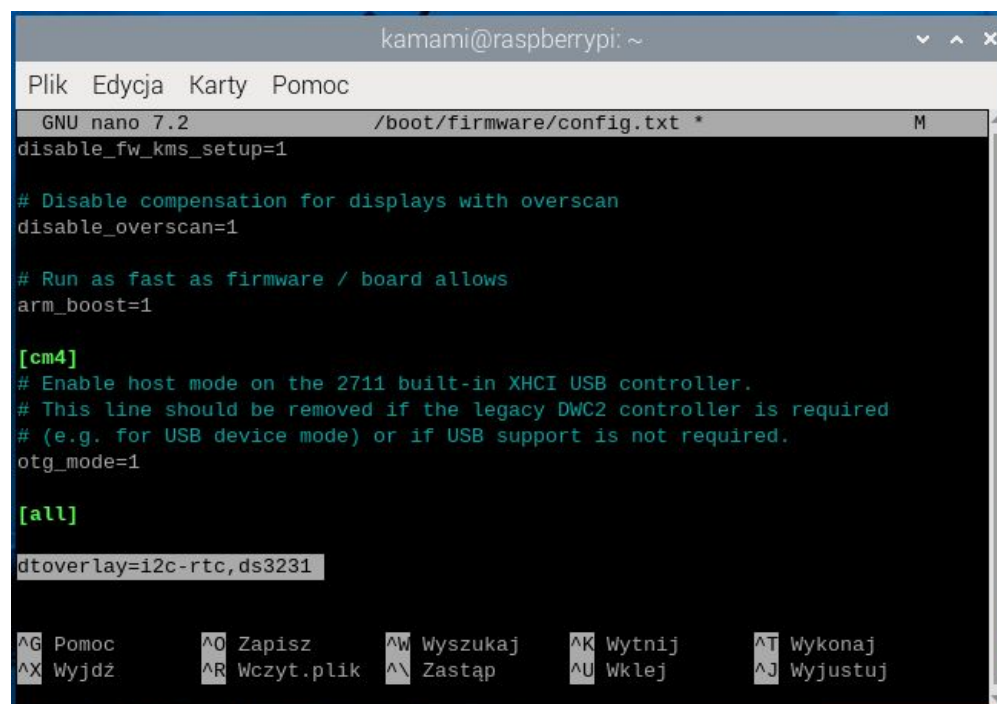
Przygotowanie systemu Raspberry Pi

Przygotowanie systemu rozpoczynamy od edycji pliku `/boot/firmware/config.txt` (w starszych wersjach systemu operacyjnego Raspberry Pi jest to plik `/boot/config.txt`). W tym celu wpisujemy polecenie:

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

Na końcu istniejącej treści dopisujemy linię:

```
dtoverlay=i2c-rtc,ds3231
```



```
kamami@raspberrypi: ~
Plik Edycja Karty Pomoc
GNU nano 7.2 /boot/firmware/config.txt * M
disable_fw_kms_setup=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]
dtoverlay=i2c-rtc,ds3231
^G Pomoc      ^O Zapisz      ^W Wyszukaj   ^K Wytnij     ^T Wykonaj
^X Wyjdź      ^R Wczyt.plik ^\ Zastąp    ^U Wklej      ^J Wyjustuj
```

Następnie zapisujemy zmiany poprzez `Ctrl+O` oraz zamykamy edytor poprzez `Ctrl+X`. Uruchamiamy system ponownie, np. wpisując:

```
reboot
```

Po ponownym uruchomieniu systemu sprawdzamy skuteczność wprowadzonych zmian, wpisując polecenie:

```
i2cdetect -y 1
```

Tym razem efekt powinien być następujący:

```
kamami@raspberrypi: ~  
Plik Edycja Karty Pomoc  
kamami@raspberrypi:~$ i2cdetect -y 1  
 0 1 2 3 4 5 6 7 8 9 a b c d e f  
00: -- -- -- -- -- -- -- -- -- -- -- -- -- --  
10: -- -- -- -- -- -- -- -- -- -- -- -- -- --  
20: -- -- -- -- -- -- -- -- -- -- -- -- -- --  
30: -- -- -- -- -- -- -- -- -- -- -- -- -- --  
40: -- -- -- -- -- -- -- -- -- -- -- -- -- --  
50: -- -- -- -- -- -- -- -- -- -- -- -- -- --  
60: -- -- -- -- -- -- -- UU -- -- -- -- -- --  
70: -- -- -- -- -- -- -- -- -- -- -- -- -- --  
kamami@raspberrypi:~$
```

Oznaczenie UU, które zastąpiło wcześniejszą wartość 68 oznacza, że system poprawnie skomunikował się z zegarem i zastrzegł jego adres.

Ustawianie i odczyt z RTC

Aktualny systemowy czas możemy sprawdzić za pomocą polecenia:

```
date
```

Aby zapisać aktualny czas do modułu RTC należy wpisać polecenie:

```
sudo hwclock -w
```

natomiast, aby odczytać czas z modułu RTC wpisujemy:

```
sudo hwclock -r
```

```
kamami@raspberrypi:~ $ date
śro, 19 cze 2024, 12:46:15 CEST
kamami@raspberrypi:~ $ sudo hwclock -w
kamami@raspberrypi:~ $ sudo hwclock -r
2024-06-19 12:46:34.459887+02:00
```

Zegar RTC możemy ustawić dowolną wartością czasu, służy do tego polecenie:

```
hwclock --set --date="2024-06-19 12:22:22"
```

Oczywiście data i czas mogą być inne ale muszą być zapisane analogicznie jak w powyższym przykładzie. Teraz możemy ustawić w systemie czas pobrany z zegara RTC, służy do tego polecenie:

```
sudo hwclock -s
```

```
kamami@raspberrypi:~ $ sudo hwclock --set --date="2024-06-19 12:22:22"
kamami@raspberrypi:~ $ sudo hwclock -r
2024-06-19 12:22:41.939911+02:00
kamami@raspberrypi:~ $ date
śro, 19 cze 2024, 12:50:07 CEST
kamami@raspberrypi:~ $ sudo hwclock -s
kamami@raspberrypi:~ $ date
śro, 19 cze 2024, 12:23:20 CEST
kamami@raspberrypi:~ $
```

Czas systemowy w Raspberry Pi 5 będzie automatycznie pobierany z zegara RTC, przy uruchomieniu komputerka. Czas zostanie automatycznie skorygowany, gdy Raspberry Pi uzyska dostęp do internetu. Wtedy również wartość czasu RTC zostanie uaktualniona.

Linki

- [Karta katalogowa układu DS3231](#)



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.