

Raspberry Pi

Skrócony opis produktu:

Nieduży zestaw uruchomieniowy z mikrokontrolerem RP2040 fundacji Raspberry Pi o rdzeniu Cortex-M0+ firmy ARM i maksymalnej częstotliwości taktowania 133 MHz wspierający dwa tryby pracy: nis-komocowy (ang. low-power mode) i uśpienia (ang. dormant mode). Obecność złącza GPIO 40 pinów zawierającego 26 wyprowadzeń cyfrowych, 3 wejścia analogowe i 16 kanałów PWM, a także dwóch pamięci: SRAM o pojemności 264 KB (w mikrokontrolerze) oraz Quad-SPI Flash o pojemności 2 MB (poza mikrokontrolerem). Ponadto dostępny przetwornik analogowo-cyfrowy o czterech kanałach i rozdzielczości 12 bitów, przy częstotliwości próbkowania na kanał równej 0,5 MSa/s, jak również kontroler DMA. Obsługiwane interfejsy: USB 1.1 (host i device), I²C, SPI i UART. Programowanie: przez kabel USB dołączony do gniazda micro-USB i przycisk BOOTSEL (w tym względzie zestaw jest traktowany jako pamięć masowa, do której należy wgrać stosowne pliki), złącze SWD i zewnętrzny programator, komputer Raspberry Pi 4 B i komputer Raspberry Pi 400. Zasilanie zewnętrzne: przez gniazdo micro-USB (5V) i złącze GPIO (zakres napięć: 1,8-5,5V). Współpraca z drugim zestawem Raspberry Pi zero, który może pełnić rolę debuggera w oparciu o złącze SWD i narzędzie Pico-probe. Produkt dedykowany językom programowania C, C++ oraz Python (wersja 3 w implementacji MicroPython). Jest on niskobudżetowy i dla wszystkich, w tym dla osób początkujących.

Więcej informacji:

Opis produktu:

Raspberry Pi Pico to mały zestaw uruchomieniowy zawierający mikrokontroler RP2040 fundacji Raspberry Pi o 32-bitowej magistrali danych i maksymalnej częstotliwości taktowania równej 133 MHz, który jest oparty o dwurdzeniową architekturę Cortex-M0+ firmy ARM i wspiera 2 tryby pracy: niskomocowy (ang. low-power mode) oraz uśpienia (ang. dormant mode). Zestaw ten obejmuje 40-pinowe złącze GPIO, na które składa się: 26 wyprowadzeń cyfrowych, 3 wejścia analogowe i 16 kanałów PWM, a także dwie pamięci: SRAM o pojemności 264 KB (obecna w mikrokontrolerze) i Quad-SPI Flash o pojemności 2 MB (poza mikrokontrolerem). Jego interfejsami są: USB 1.1 (host i device), I²C, SPI i UART. Natomiast peryferiami: gniazdo microUSB, przycisk BOOTSEL (wymagany przy programowaniu zestawu), dioda LED o kolorze zielonym i złącze SWD do programowania zestawu i debugowania programu w nim działającego z użyciem zewnętrznego programatora). Dodatkowo występuje w nim przetwornik analogowo-cyfrowy o rozdzielczości 12 bitów i częstotliwości próbkowania równej 0,5 MSa/s, którego liczba kanałów wynosi cztery (do jednego z nich jest dołączony wbudowany czujnik temperatury), jak również kontroler DMA - oba pod zaawansowane rozwiązania. Programowanie zestawu odbywa się czworako: przez kabel USB dołączony do gniazda microUSB i użycie przycisku BOOTSEL (w tym względzie Raspberry Pi Pico jest traktowane jako pamięć masowa, do której należy wgrać stosowne pliki), złącze SWD i zewnętrzny programator, przez komputer Raspberry Pi 4 B lub przez komputer Raspberry Pi 400. Jego zasilanie jest z kolei podwójne: przez gniazdo microUSB (5V) lub złącze GPIO (zakres napięć: 1,8-5,5V). Produkt niskobudżetowy dla wszystkich, w tym dla osób początkujących. Jest on dedykowany językom programowania: C, C++ oraz Python (wersja 3 w implementacji MicroPython), a także współpracuje z drugim zestawem Raspberry Pi Pico, komputerem Raspberry Pi 4 B i komputerem Raspberry Pi 400. Brak przy tym kabla USB i zasilacza zewnętrznego w zestawie. Niemniej jest możliwa współpraca z drugim zestawem Raspberry Pi Pico, który może pełnić rolę debuggera w oparciu o złącze SWD i narzędzie Picoprobe.

Specyfikacja produktu:

- Mikrokontroler RP2040 fundacji Raspberry Pi:
 - ❖ Architektura: ARM Cortex-M0+
 - ❖ Typ architektury: RISC
 - ❖ Długość magistrali danych: 32 bity
 - ❖ Standardowa (domyślna) częstotliwość taktowania: 48 MHz
 - ❖ Maksymalna częstotliwość taktowania: 133 MHz
 - ❖ Liczba rdzeni: 2
 - ❖ Liczba wyprowadzeń I/O: 30
 - ❖ Wbudowana pamięć SRAM o pojemności 264 KB
 - ❖ Interfejsy:
 - UART - w liczbie sztuk 2* (w ramach złącza GPIO)

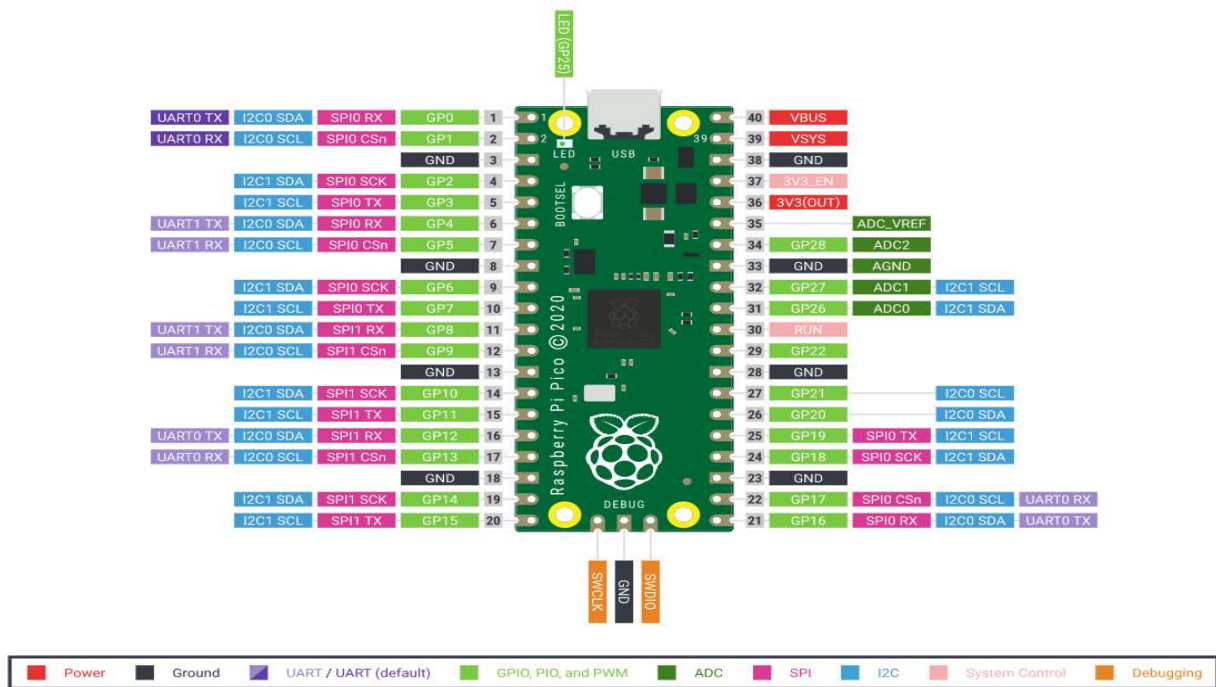
- SPI - w liczbie sztuk 2* (w ramach złącza GPIO)
 - I2C - w liczbie sztuk 2* (w ramach złącza GPIO)
 - USB 1.1 - w liczbie sztuk 1 (host i device)
- ❖ Kontrolery:
 - DMA
 - PHY (warstwy fizycznej modelu sieciowego OSI)
- ❖ Przetwornik analogowo-cyfrowy:
 - Liczba kanałów: 4 (jeden zarezerwowany pod wbudowany czujnik temperatury)
 - Rozdzielczość: 12 bitów
 - Częstotliwość próbkowania na kanał: 0,5 MSa/s
- ❖ Wbudowany czujnik temperatury (w przetworniku analogowo-cyfrowym)
- ❖ 2 pętle PLL (generacja zegarów dla mikrokontrolera oraz interfejsu USB)
- ❖ Programowalne źródło napięcia (dla rdzeni mikrokontrolera)
- ❖ 8 maszyn stanów PIO (odrębnych procesorów programowanych z użyciem asemblera)
- ❖ Zewnętrzne przerwania
- ❖ Tryby pracy:
 - Niskomocowy (ang. low-power mode)
 - Uśpienia (ang. dormant mode)
- Pamięć Quad-SPI Flash o pojemności 2 MB
- Wyprowadzenia* (w ramach złącza GPIO):
 - ❖ Cyfrowe:
 - Dwukierunkowe (wejścia i wyjścia)
 - W liczbie sztuk: 26
 - ❖ PWM:
 - Jednokierunkowe (wyjścia)
 - W liczbie sztuk: 16
 - ❖ Analogowe:
 - Jednokierunkowe (wejścia)
 - W liczbie sztuk: 3
- Peryferia:
 - ❖ Złącze GPIO 40 pinów - rozstaw 0,1 cala
 - ❖ Gniazdo microUSB
 - ❖ Przycisk BOOTSEL (wymagane do zaprogramowania zestawu)
 - ❖ Dioda LED o kolorze zielonym (dla użytkownika)
 - ❖ Złącze SWD (do programowania zestawu i debugowania programu w nim działającego - w oparciu o zewnętrzny programator)
- 4 otwory montażowe
- Programowanie:
 - ❖ przez kabel USB (gniazdo microUSB) i przycisk BOOTSEL:
 - Zestaw jest traktowany jako pamięć masowa, do której należy wgrać stosowne pliki (wolumin RPI-RP2)
 - ❖ przez złącze SWD i zewnętrzny programator
 - ❖ przez komputer Raspberry Pi 4B
 - ❖ przez komputer Raspberry Pi 400
- Zasilanie:
 - ❖ Zewnętrzne:
 - Przez kabel USB - gniazdo micro-USB (5V)
 - Przez złącze GPIO (zakres napięć: 1,8-5,5V)
- Dedykowane języki programowania:
 - ❖ C
 - ❖ C++
 - ❖ Python (wersja 3 w implementacji MicroPython)
- Dedykowane środowiska programistyczne - m.in.:
 - ❖ Eclipse
 - ❖ Thonny (tylko język Python)
- Współpraca z drugim zestawem Raspberry Pi Pico (drugi może pełnić rolę debuggera dla pierwszego w oparciu o złącze SWD i narzędzie Picoprobe)

- Opcja konwertera USB-SWD i USB-UART w oparciu o narzędzie Picoprobe
- Wymiary: 51 x 21 x 1 mm (pierwsze to długość, drugie to szerokość, a trzecie to wysokość)
- Temperatura pracy: od -20 do 85 stopni Celsjusza

* Niektóre z nich należy uprzednio skonfigurować.

Opis złączy produktu:

Źródło: <https://www.raspberrypi.org/documentation/pico/getting-started/static/15243f1ffd3b8ee646a1708bf4c0e866/Pico-R3-Pinout.svg>



Legenda (od lewej strony):

1. Wyprowadzenia napięciowe
2. Masa
3. Wyprowadzenia interfejsu UART
4. Wyprowadzenia GPIO, PIO i PWM
5. Wyprowadzenia przetwornika analogowo-cyfrowego
6. Wyprowadzenia interfejsu SPI
7. Wyprowadzenia interfejsu I2C
8. Wyprowadzenia sterujące (kontrolne)
9. Wyprowadzenia debuggera (SWD)

Narzędzia:

- Środowisko programistyczne Eclipse: <https://www.eclipse.org/downloads/>
- Środowisko programistyczne Thonny: <https://thonny.org>
- Narzędzie Picoprobe: <https://github.com/raspberrypi/picoprobe>

Do pobrania:

- Skrócony opis zestawu Raspberry Pi Pico (w postaci pliku PDF): <https://datasheets.raspberrypi.org/pico/pico-product-brief.pdf>
- Podręcznik zestawu Raspberry Pi Pico (w postaci pliku PDF): <https://datasheets.raspberrypi.org/pico/pico-datasheet.pdf>
- Pliki projektowe zestawu Raspberry Pi Pico (schematy elektryczne, plik BRD środowiska Eagle oraz arkusz Excel zawierający podsumowanie podzespołów zestawu - BOM (ang. Bill of Materials): <https://datasheets.raspberrypi.org/pico/RPi-Pico-R3-PUBLIC-20200119.zip>

- Model trójwymiarowy zestawu Raspberry Pi Pico (w postaci pliku STEP):
<https://datasheets.raspberrypi.org/pico/Pico-R3-step.zip>
- Plik zestawu Raspberry Pi Pico do użycia w środowisku Fritzing (format FZPZ):
<https://datasheets.raspberrypi.org/pico/Pico-R3-Fritzing.fzpz>
- Samouczek zestawu Raspberry Pi Pico (w postaci pliku PDF):
<https://datasheets.raspberrypi.org/pico/getting-started-with-pico.pdf>
- Opis bibliotek i narzędzi umożliwiających programowanie w języku C i C++ w przypadku zestawu Raspberry Pi Pico (w postaci pliku PDF):
<https://datasheets.raspberrypi.org/pico/raspberry-pi-pico-c-sdk.pdf>
- Poradnik po bibliotekach i narzędziach umożliwiających programowanie w języku C i C++ w przypadku zestawu Raspberry Pi Pico (w postaci pliku PDF):
<https://datasheets.raspberrypi.org/pico/getting-started-with-pico.pdf>
- Opis generatora dokumentacji Doxygen (język C i C++) dla zestawu Raspberry Pi Pico:
<http://rptl.io/pico-doxygen>
- Opis bibliotek i narzędzi umożliwiających programowanie w języku Python (wersja 3 w implementacji MicroPython) w przypadku zestawu Raspberry Pi Pico (w postaci pliku PDF):
<https://datasheets.raspberrypi.org/pico/raspberry-pi-pico-python-sdk.pdf>
- Dokumentacja mikrokontrolera RP2040 (w postaci pliku PDF)
<https://datasheets.raspberrypi.org/rp2040/rp2040-datasheet.pdf>
- Poradnik mikrokontrolera RP2040 (w postaci pliku PDF):
<https://datasheets.raspberrypi.org/rp2040/hardware-design-with-rp2040.pdf>
- Pozostała dokumentacja i pliki zestawu Raspberry Pi Pico, a także skrócone samuczki i skrócony opis zestawu:
<https://www.raspberrypi.org/documentation/pico/getting-started/>
- Repozytorium Pico SDK (GitHub):
<https://github.com/raspberrypi/pico-sdk>
- Repozytorium zawierające przykładowe kody dla zestawu Raspberry Pi Pico (GitHub):
<https://github.com/raspberrypi/pico-examples>