
Raspberry Pi Pico

Programirajte, izradite i savladajte više od 50 projekata
koristeći MicroPython i RP2040 mikrokontroler



Dr Dogan Ibrahim

Agencija Eho
www.infoelektronika.net

- Sva prava zadržana. Nijedan deo ove knjige ne sme biti reprodukovan u bilo kom materijalnom obliku, uključujući fotokopiranje ili slučajno ili nenamerno smeštanje na bilo koji elektronski medijum sa ili uz pomoć bilo kog elektronskog sredstva, bez pismenog odobrenja nosioca autorskih prava osim u skladu sa odredbama zakona o autorskim pravima, dizajnu i patentima iz 1988. godine ili pod uslovima izdatim od Copyright Licensing Agency Ltd, 90 Tottenham Court Road, London, England W1P 9HE. Prijave za pismene dozvole radi štampanja bilo kog dela ove publikacije upućuje se izdavaču ove knjige.
- Izjava: Autor i izdavač su uložili najveće napore da bi se obezbedila tačnost informacija sadržanih u ovoj knjizi. Autor i izdavač ne mogu da pretpostave neprijatnosti i ovom izjavom isključuju bilo kakvu odgovornost za bilo koju stranku koja bi imala gubitke ili štetu uzrokovanu greškama ili propustima u ovoj knjizi, bez obzira da li su greške ili propusti nastali usled nemara, nezgode ili bilo kog drugog razloga.

ISBN 978-86-80134-53-6

Raspberry Pi Pico

Naslov originala: Raspberry Pi Pico Essentials

Autor: Dogan Ibrahim

Prevod: Biljana Tešić

Recenzent: Aleksandar Jevtić

Izdaje i štampa: Agencija Eho, Niš

e-mail: redakcija@infoelektronika.net

Tiraž: 200

Godina izdanja: 2024

CIP - Каталогизација у публикацији
Народна библиотека Србије, Београд

004.383/.384

ИБРАХИМ, Доган, 1954-

Raspberry Pi Pico : programirajte, izradite i savladajte više od 50 projekata koristeći MicroPython i RP2040 mikrokontroler / Dogan Ibrahim ; [prevod Biljana Tešić]. - Niš : Agencija Eho, 2024 (Niš : Agencija Eho). - 247 str. : ilu-str. ; 24 cm

Prevod dela: Raspberry Pi Pico Essentials. - Tiraž 200.

ISBN 978-86-80134-53-6

а) Микроконтролери -- Примена

COBISS.SR-ID 140755465

Sadržaj	
Predgovor	9
Poglavlje 1 • Raspberry Pi Pico hardver	11
1.1 Pregled	11
1.2 Pico hardverski modul	11
1.3 Poređenje sa Arduino UNO	13
1.4 Uslovi rada i napajanje Pico-a	14
1.5 Raspored pinova mikrokontrolera RP2040 i Pico modula	14
1.6 Ostale ploče zasnovane na mikrokontroleru RP2040	16
Poglavlje 2 • Raspberry Pi Pico programiranje	23
2.1 Pregled	23
2.2 Instaliranje MicroPython-a na Pico-u	23
Poglavlje 3 • Jednostavni hardverski projekti za Raspberry Pi Pico	48
3.1 Pregled	48
3.2 Projekat 1: Trepćući LED - Upotreba ugrađenog LED-a	48
3.3 Projekat 2: Spoljni trepćući LED	51
3.4 Projekat 3: Trepćući SOS u Morzeovoj azbuci	53
3.5 Projekat 4: Trepćući LED – korišćenje tajmera	55
3.6 Projekat 5: Naizmenično trepćući LED-ovi	56
3.7 Projekat 6: Promena brzine treptanja LED-a – korišćenjem tastera za prekid	58
3.8 Projekat 7: Naizmenično treptanje crvenih, zelenih i plavih LED-ova - RGB	63
3.9 Projekat 8: Nasumično trepćuće crveni, zeleni i plavi LED-ovi - RGB	65
3.10 Projekat 9: Rotirajući LED-ovi	66
3.11 Projekat 10: LED-ovi koji broje u binarnom obliku	69
3.12 Projekat 11: Božićne lampice (nasumično treptanje 8 LED-ova)	72
3.13 Projekat 12: Elektronska kocka	74
3.14 Projekat 13: Srećan dan u nedelji	78
3.15 Projekat 14: Alarm za vrata sa LED-om koji svetli u 7 boja	80

3.16 Projekat 15: Dvocifreni, 7-segmentni displej	84
3.17 Projekat 16: 4-cifreni, 7-segmentni displej kao brojač sekundi.....	93
3.18 LCD-ovi	98
3.19 Projekat 17: LCD funkcije – prikazivanje teksta.....	100
3.20 Projekat 18: Brojač sekundi - LCD.....	104
3.21 Projekat 19: Reakcioni tajmer sa LCD-om	106
3.22 Projekat 20: Ultrazvučno merenje udaljenosti	108
3.23 Projekat 21: Visina osobe (stadiometar)	112
3.24 Projekat 22: Sistem za pomoć pri parkiranju unazad sa ultrazvučnim senzorom i zujalicom.....	114
Poglavlje 4 • Korišćenje analognih-digitalnih pretvarača (ADC)	117
4.1 Pregled	117
4.2 Projekat 1: Voltmetar	117
4.3 Projekat 2: Merenje temperature korišćenjem internog senzora temperature	119
4.4 Projekat 3: Merenje temperature korišćenjem spoljnog senzora temperature	120
4.5 Projekat 4: ON/OFF kontroler temperature	122
4.6 Projekat 5: ON/OFF kontroler temperature sa LCD-om	125
4.7 Projekat 6: Merenje intenziteta ambijentalnog svetla	128
4.8 Projekat 7: Ommetar	130
4.9 Projekat 8: Unutrašnja i spoljašnja temperatura	133
4.10 Projekat 9: Korišćenje termistora za merenje temperature	135
Poglavlje 5 • Evidencija podataka	140
5.1 Pregled	140
5.2 Projekat 1: Evidentiranje podataka o temperaturi	140
5.3 Projekat 2: Čitanje evidentiranih podataka	142
Poglavlje 6 • Modulisanje širine impulsa (PWM)	144
6.1 Pregled	144
6.2 Osnovna teorija modulisanja širine impulsa	144
6.3 PWM kanali Raspberry Pi Pico-a.....	146

6.4 Projekat 1: Generisanje PWM signala od 1000 Hz sa radnim ciklusom od 50%.....	147
6.5 Projekat 2: Promena svetline LED-a.....	148
6.6 Projekat 3: Promena brzine DC motora sa četkicama	149
6.7 Projekat 4: Generator fukcija sa LCD-om	150
6.8 Projekat 5: Merenje frekvencije i radnog ciklusa PWM signala.....	152
6.9 Projekat 6: Generisanje melodije (Melody Maker)	154
Poglavlje 7 • Serijska komunikacija (UART)	158
7.1 Pregled.....	158
7.2 Raspberry Pi Pico UART serijski portovi.....	160
7.3 Projekat 1: Slanje unutrašnje temperature Raspberry Pi Pico-a na Arduino Uno.....	160
7.4 Projekat 2: Prijem i prikazivanje brojeva sa Arduino Uno-a.....	165
7.5 Projekat 3: Komunikacija sa Raspberry Pi 4 preko serijske veze	166
Poglavlje 8 • Interfejs I2C magistrale.....	170
8.1 Pregled.....	170
8.2 I2C magistrala	170
8.3 I2C pinovi Raspberry Pi Pico-a	171
8.4 Projekat 1: I2C proširivač porta	172
8.5 Projekat 2: EEPROM memorija	177
Operacija čitanja iz memorije	180
8.6 Projekat 3: Senzor temperature TMP102	182
8.7 Projekat 4: Senzor temperature i atmosferskog pritiska BMP280	188
8.8 Projekat 5: Prikaz temperature i atmosferskog pritiska BMP280 na LCD-u.....	196
Poglavlje 9 • Interfejs SPI magistrale	198
9.1 Pregled.....	198
9.2 Raspberry Pi Pico dva SPI porta	199

Poglavlje 10 • Wi-Fi sa Raspberry Pi Picom	206
10.1 Pregled	206
10.2 Projekat 1: Upravljanje LED diodom sa pametnog telefona koristeći Wi-Fi	206
10.3 Projekat 2: Prikazivanje interne temperature na pametnom telefonu korišćenjem Wi-Fi mreže	212
Poglavlje 11 • Bluetooth sa Raspberry Pi Picom	217
11.1 Pregled	217
11.2 Bluetooth interfejs	217
11.3 Projekat 1: Upravljanje LED diodom sa pametnog telefona korišćenjem Bluetooth-a	217
11.4 Projekat 2: Slanje interne temperature Raspberry Pi Pica na pametni telefon	222
Poglavlje 12 • Korišćenje digitalno-analognih pretvarača (DAC)	225
12.1 Pregled	225
12.2 MCP4921 DAC	225
12.3 Projekat 1: Generisanje kvadratnog signala sa amplitudom ispod +3.3 V	226
12.4 Projekat 2: Generisanje fiksni napona	231
12.5 Projekat 3: Generisanje testerastog oblika signala	233
12.6 Projekat 4: Generisanje trougaonog signala	235
12.7 Projekat 5: Proizvoljni periodični signal	237
12.8 Projekat 6: Generisanje sinusnog signala	239
12.9 Projekat 7: Generisanje tačnog sinusnog signala koristeći tajmerske prekide	242
Poglavlje 13 • Automatsko izvršavanje programa nakon pokretanja Raspberry Pi Pico.....	245
Dodatak A • Spisak komponenti	247

Predgovor

Tradicionalno, računar je napravljen korišćenjem čipa mikroprocesora i mnogih eksternih čipova za podršku. Mikroprocesor uključuje centralnu procesorsku jedinicu (CPU – Central Processing Unit), aritmetičku i logičku jedinicu (ALU – Arithmetic and Logic Unit) i vremensko i kontrolno kolo — i kao takav nije posebno koristan samostalno. Da bi mikroprocesor bio koristan računar, mora da ga podržavaju mnogi eksterni čipovi kao što su memorija, ulaz/izlaz, tajmeri, prekidna kola itd. Nedostatak ovog tipa dizajna je bio veliki broj čipova, što je dovelo do složenog dizajna i ožičenja i visoke potrošnje energije.

S druge strane, mikrokontroler je u osnovi računar sa jednim čipom koji uključuje CPU, memoriju, ulazno/izlazno kolo, tajmere, kola za prekide, tajmer i nekoliko drugih kola i modula, koji smešteni u jednom silicijumskom čipu. Raniji mikrokontroleri su bili ograničeni u svojim kapacitetima, brzini, i trošili su znatno više energije. Većina ranijih mikrokontrolera su bili 8-bitni procesori sa frekvencijom od nekoliko MHz i nudili su samo stotine bajtova memorije programa i podataka. Ovi mikrokontroleri su tradicionalno programirani korišćenjem asemblerskih jezika ciljnih procesora. 8-bitni mikrokontroleri se i dalje koriste, posebno u malim projektima u kojima velike količine memorije ili velika brzina nisu glavni zahtevi. Napretkom tehnologije čipova sada imamo 32-bitne i 64-bitne mikrokontrolere sa brzinama od nekoliko GHz, koji nude nekoliko GB memorijskog prostora. Mikrokontroleri se danas programiraju korišćenjem jezika visokog nivoa kao što su C, C#, BASIC, PASCAL, JAVA itd.

Raspberry Pi Pico je mikrokontroler visokih performansi, dizajniran posebno za fizičko računarstvo. Čitaoci treba da shvate da se mikrokontroleri veoma razlikuju od računara sa jednom pločom kao što je Raspberry Pi 4 (i drugi članovi porodice Raspberry Pi-a). Ne postoji operativni sistem na Raspberry Pi Pico-u. Mikrokontroleri kao što su Raspberry Pi Pico mogu se programirati tako da izvršavaju jedan zadatak i mogu se koristiti u brzim aplikacijama za kontrolu i praćenje u stvarnom vremenu.

Raspberry Pi Pico je zasnovan na brzom i veoma efikasnom dvojezgarom ARM Cortex-M0+RP2040 čipu mikrokontrolera koji radi na najviše 133 MHz. Čip uključuje 264 KB SRAM memorije i 2 MB fleš memorije. Ono što Raspberry Pi Pico čini veoma atraktivnim je njegov veliki broj GPIO pinova i često korišćeni periferni interfejs moduli, kao što su SPI, I2C, UART, PWM, kao i brzi i precizni tajmer moduli.

Možda najveća prednost Raspberry Pi Pico-a u odnosu na mnoge druge mikrokontrolere na tržištu je vrlo niska cena, velika memorija i brzi i precizni tajmer moduli. U vreme pisanja ove knjige cena jedne jedinice bila je oko 6 dolara.

Raspberry Pi Pico se lako programira korišćenjem popularnih jezika visokog nivoa kao što su MicroPython ili C/C++. Na Internetu su dostupne mnoge beleške o primeni, uputstva i tabele sa podacima za Raspberry Pi Pico.

Ova knjiga je uvod u korišćenje mikrokontrolera Raspberry Pi Pico u kombinaciji sa programskim jezikom Micro-Python. Thonny razvojno okruženje (IDE) se koristi u svim opisanim projektima. U knjizi se nalazi više od 50 radnih i testiranih projekata kojima su obuhvaćeni gotovo svi aspekti Raspberry Pi Pico-a.

Sledeći podnaslovi su dati za svaki projekat kako bi ga bilo lakše pratiti:

- Naslov
- Kratak opis
- Cilj
- Blok dijagram
- Električna šema
- Listing programa sa punim opisom

Nadam se da ćete u vašim sledećim projektima zasnovanim na mikrokontrolerima koristiti Raspberry Pi Pico i da će ova knjiga biti korisna u razvoju vaših projekata.

Dr Dogan Ibrahim

Poglavlje 1 • Raspberry Pi Pico hardver

1.1 Pregled

Raspberry Pi Pico je modul mikrokontrolera sa jednom pločom koji je razvila Raspberry Pi Foundation. Ovaj modul se zasniva na čipu mikrokontrolera RP2040. U ovom poglavlju ćemo detaljno pogledati detalje o hardveru mikrokontrolera Raspberry Pi Pico. Od sada ćemo ovaj modul mikrokontrolera zvati skraćeno „Pico“ zbog uvažavanja i priznanja njegovog zvaničnog naziva Raspberry Pi Pico.

1.2 Pico hardverski modul

„Pico“ je veoma jeftin mikrokontrolerski modul od 4 dolara zasnovan na čipu mikrokontrolera RP2040 koji ima dualni Cortex-M0+ procesor. Na slici 1.1 je prikazan prednji deo Pico hardverskog modula koji je u osnovi mala ploča. U centru ploče je mali, 7×7 mm RP2040 čip mikrokontrolera smešten u QFN-56 paketu. Na dve ivice ploče nalazi se 40 metalnih GPIO (General-Input-Output) pinova sa otvorima u zlatnoj boji. Lemljenje pinova na ove otvore omogućava lako povezivanje spoljnih priključaka na ploču. Otvori su označeni počev od broja 1 u gornjem levom uglu ploče, a brojevi se povećavaju naniže do broja 40 koji se nalazi u gornjem desnom uglu ploče. Ploča je kompatibilna sa prototipskom pločom (tj. razmak između pinova od 2.54 mm), a nakon lemljenja pinova, ploča se može priključiti na prototipsku ploču radi lakšeg povezivanja sa GPIO pinovima pomoću kratkospojnika. Pored ovih otvora videćete neravne kružne izreze koji se mogu priključiti na druge module bez postavljanja fizičkih pinova.

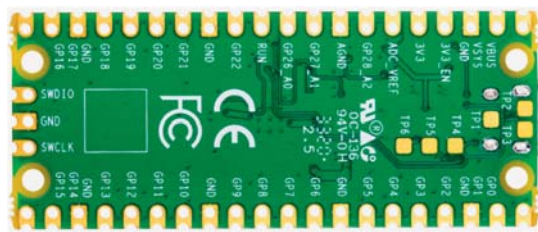


Slika 1.1: Prednja strana Pico hardverskog modula.

Na jednoj ivici ploče nalazi se mikro-USB B port za napajanje ploče, kao i za njeno programiranje. Pored USB porta nalazi se ugrađena korisnički LED koji se može koristiti tokom razvoja programa. Pored ovog LED-a nalazi se dugme BOOTSEL koje se koristi tokom programiranja mikrokontrolera, kao što ćemo videti u narednim poglavljima. Na drugoj ivici ploče, pored logotipa Raspberry Pi, nalaze se 3 konektora koji se mogu koristiti za debugovanje u vašim programima.

Na slici 1.2 je prikazan zadnji deo Pico hardverskog modula. Ovde su svi GPIO pinovi identifikovani slovima i brojevima. Primitićete sledeće vrste slova i brojeva:

GND	- Uzemljenje napajanja (digitalno uzemljenje)
AGND	- Uzemljenje napajanja (analogno uzemljenje)
3V3	- +3,3 V napajanje (izlaz)
GP0	- GP22 - Digitalni GPIO
GP26_A0	- GP28_A2 - analogni ulazi
ADC_VREF	- ADC referentni napon
TP1	- TP6 - testne tačke
SWDIO, GND, SWCLK	- interfejs za debugovanje
RUN	- Podrazumevani RUN pin. Povežite LOW da biste resetovali RP2040.
3V3_EN	- Ovaj pin podrazumevano omogućava +3,3V napajanje. +3,3 V se može onemogućiti povezivanjem ovog pina LOW.
VSYS	- Sistemski ulazni napon (od 1,8 V do 5,5 V) koji koristi ugrađeni sistem SMPS za generisanje +3,3 V napajanja za ploču.
VBUS	- mikro-USB ulazni napon (+5 V)



Slika 1.2: Zadnja strana Pico hardverskog modula.

GP29 (ulaz)	- koristi se u ADC režimu (ADC3) za merenje VSYS/3
GP25 (izlaz)	- povezan na ugrađeni korisnički LED
GP24 (ulaz)	- VBUS osećaj - HIGH ako je VBUS prisutan, inače LOW
GP23 (izlaz)	- kontroliše ugrađeni SMPS Power Save

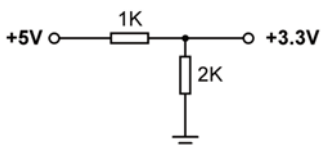
Specifikacije Pico hardverskog modula su sledeće:

- 32-bitni RP2040 Cortex-M0+ procesor sa dva jezgra koji radi na 133 MHz
- 2 megabajta Q-SPI fleš memorija
- 264 kilobajta SRAM memorije
- 26 GPIO (+3,3V kompatibilan)
- 3× 12-bitni ADC pinovi
- Serial Wire Debug (SWD) port
- Micro-USB port (USB 1.1) za napajanje (+5V) i podatke (programiranje)
- 2× UART, 2× I²C, 2× SPI interfejs magistrale
- 16× PWM kanala
- 1× tajmer (sa 4 alarma), 1× brojač u realnom vremenu
- Ugrađeni senzor temperature

- Ugrađeni LED (na portu GP25)
- MicroPython, C, C++ programiranje
- Programiranje masovne memorije preko USB-a prevlačenjem i otpuštanjem (Drag & drop)

Pico GPIO hardver je kompatibilan sa +3,3 V i stoga je važno da pazite da ne prekoracite ovaj napon kada povezujete spoljne uređaje sa GPIO pinovima. Kola logičkih pretvarača od +5 V do +3,3 V ili kola naponskog razdelnika moraju se koristiti ako je potrebno povezati uređaje sa +5 V izlazima na Pico GPIO pinove.

Na slici 1.3 prikazano je kolo naponskog razdelnika koji se može koristiti za smanjenje +5 V na +3,3 V.



Slika 1.3: Kolo naponskog razdelnika.

1.3 Poređenje sa Arduino UNO

Arduino UNO je jedna od najpopularnijih razvojnih ploča mikrokontrolera koju koriste studenti, inženjeri i hobisti. U tabeli 1.1 je prikazano poređenje Raspberry Pi Pico-a sa Arduino UNO-m. Iz ove tabele je jasno da je Pico mnogo brži od Arduino UNO-a, ima veću fleš memoriju i memoriju podataka, obezbeđuje više digitalnih ulazno/izlaznih pinova i poseduje ugrađeni senzor temperature. Arduino UNO radi na +5 V i njegovi GPIO pinovi su kompatibilni sa +5 V. Možda neke prednosti Arduino UNO uključuju ugrađenu EEPROM memoriju i 6-kanalni ADC umesto 3-kanalnog.

Funkcija	Raspberry Pi Pico	Arduino UNO
Mikrokontroler	RP2040	Atmega328P
Jezgro i bitovi	dvojezgarni, 32-bitni, Cortex-M0+	jednojezgarni, 8-bitni
RAM	264 kilobajta	2 kilobajta
Fleš	2 megabajta	32 kilobajta
CPU brzina	od 48 MHz do 133 MHz	16 MHz
EEPROM	Nema	1 kilobajt
Ulaz napajanja	+5 V preko USB porta	+5 V preko USB porta
Alternativno napajanje	2–5 V preko VSYS pina	7–12 V
MCU radni napon	+3,3 V	+5 V
GPIO broj	26	20
ADC broj	3	6
Hardverski UART	2	1
Hardverski I2C	2	1
Hardverski SPI	2	1
Hardverski PWM	16	6
Programski jezici	MicroPython, C, C++	C (Arduino IDE)
Ugrađeni LED	1	1
Troškovi	4 dolara	20 dolara

Tabela 1.1: Poređenje Raspberry Pi-a sa Arduino UNO-m

1.4 Uslovi rada i napajanje Pico-a

Preporučeni uslovi rada za Pico su:

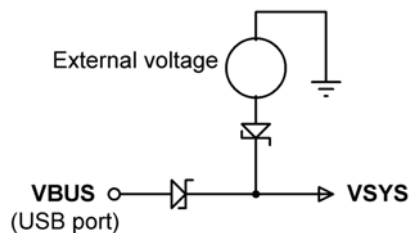
- Radna temperatura: $-20\text{ }^{\circ}\text{C}$ do $+85\text{ }^{\circ}\text{C}$
- VBUS napon: $+5\text{ V} \pm 10\%$
- VSYS napon: od $+1,8\text{ V}$ do $+5,5\text{ V}$

Ugrađeni SMPS se koristi za generisanje $+3,3\text{ V}$ za napajanje RP2040 iz opsega ulaznih napona od $1,8\text{ V}$ do $+5,5\text{ V}$. Na primer, 3 alkalne AA baterije se mogu koristiti za obezbeđenje $+4,5\text{ V}$ za napajanje Pico-a.

Pico se može napajati na nekoliko načina. Najjednostavniji metod je da priključite micro-USB port na izvor napajanja od $+5\text{ V}$, kao što je USB port računara ili adapter za napajanje od $+5\text{ V}$. To će obezbediti napajanje za VSYS ulaz (pogledajte sliku 1.4) preko Šotkijeve diode. Napon na VSIS ulazu je stoga VBUS napon minus pad napona Šotkijeve diode (oko $+0,7\text{ V}$). VBUS i VSYS pinovi mogu biti kratko spojeni ako se ploča napaja sa eksternog $+5\text{ V}$ USB porta. To će malo povećati ulazni napon i time smanjiti oscilacije na VSYS-u. VSYS napon se dovodi do SMPS-a preko RT6150 koji generiše fiksno $+3,3\text{ V}$ za MCU i druge delove ploče. VSYS je podeljen sa 3 i dostupan je na analognom ulaznom portu GPIO29 (ADC3) koji se lako može nadgledati. GPIO24 proverava postojanje VBUS napona i nalazi se na logički HIGH ako je VBUS prisutan.

Drugi metod za napajanje Pico-a je primena spoljnog napona (od $+1,8\text{ V}$ do $+5,5\text{ V}$) direktno na VSYS ulaz (npr. korišćenjem baterija ili spoljnog napajanja).

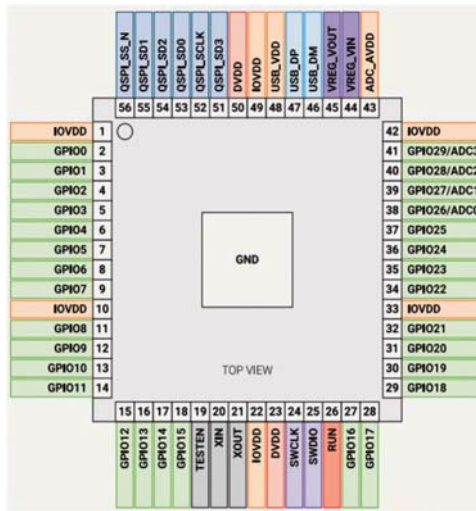
Takođe možemo da koristimo USB ulaz i VSYS ulaze zajedno za napajanje Pico-a, na primer za rad pomoću obe baterije i USB porta. Ako se koristi taj metod, onda bi trebalo koristiti Šotkijevu diodu na VSYS ulazu kako bi se sprečilo da se napajanja međusobno ometaju. Veći napon će napajati VSYS.



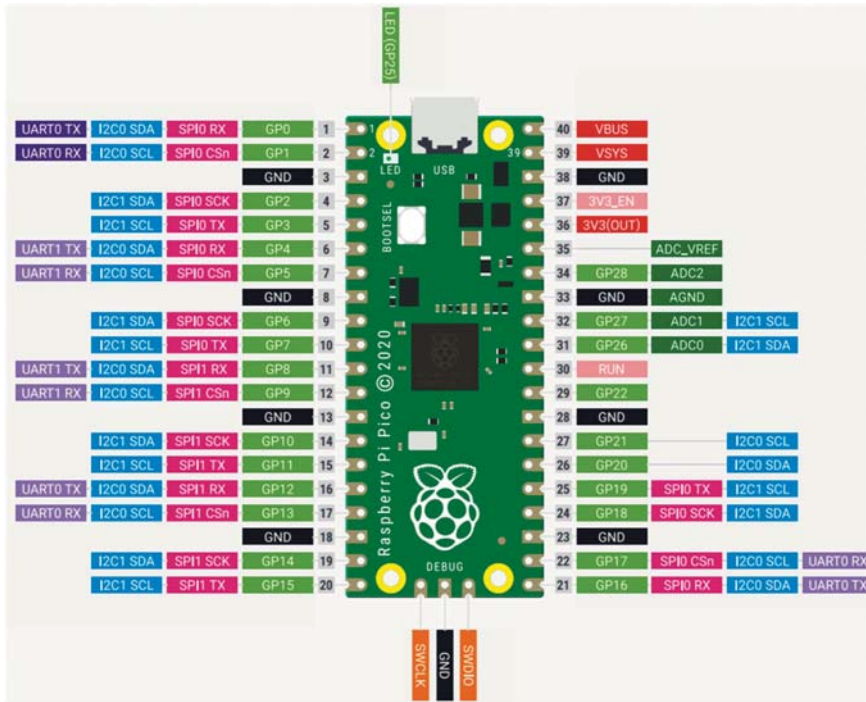
Slika 1.4: Napajanje Pico-a.

1.5 Raspored pinova mikrokontrolera RP2040 i Pico modula

Na slici 1.5 je prikazan raspored pinova mikrokontrolera RP2040, koji je smešten u kućište sa 56 pinova. Raspored pinova Pico modula je detaljno prikazan na slici 1.6. Kao što možete videti na slici, većina pinova ima više funkcija. Na primer, GPIO0 (pin 1) se udvostručuje kao UART0 TX, I2C0 SDA i SPI0 RX pin.



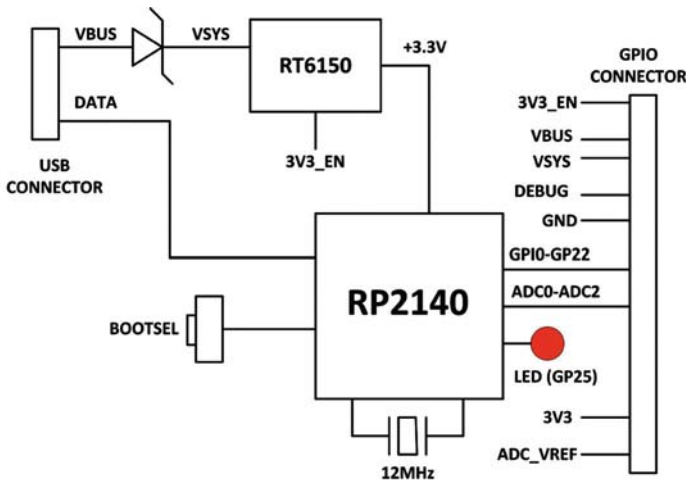
Slika 1.5: Raspored pinova mikrokontrolera RP2040.



Slika 1.6: Raspored pinova Raspberry Pi Pico-a.

Na slici 1.7 je prikazan pojednostavljeni blok dijagram Pico hardverskog modula. Napominjemo da su GPIO pinovi direktno povezani sa čipa mikrokontrolera na GPIO konektor. GPIO br. 26–28 se mogu koristiti ili kao digitalni GPIO ili kao ADC ulazi. ADC ulazi GPIO26-29 imaju diode na 3 V sa nepropusnom polarizacijom i stoga ulazni

napon ne sme biti veći od $3V3 + 300 \text{ mV}$. Još jedna stvar koju treba napomenuti je da ako se RP2040 ne napaja, primena napona na pinove GPIO26-29 može da procuri kroz diodu do izvora napajanja (sa drugim GPIO pinovima nema problema i napon se može primeniti kada se RP2040 ne napaja.)



Slika 1.7: Pojednostavljeni dijagram.

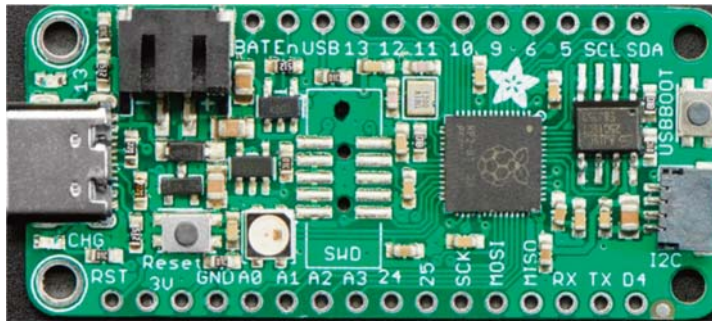
1.6 Ostale ploče zasnovane na mikrokontroleru RP2040

Tokom pisanja ove knjige, neki nezavisni proizvođači su razvijali module zasnovane na RP2040 čipu. Neki primeri su dati u ovom odeljku.

1.6.1 Adafruit Feather RP2040

Ova ploča mikrokontrolera (slika 1.8) ima sledeće osnovne specifikacije:

- RP2040 32-bitni Cortex-M0+ radi na 125 MHz
- 4 MB fleš memorije
- 264 KB RAM-a
- 4× 12-bitni ADC
- 2× I2C, 2× SPI, 2× UART
- 16× PWM
- 200 mA LiPo punjač
- dugmad Reset i Bootloader
- 24 MHz kristal
- +3,3 V regulator sa strujnim izlazom od 500 mA
- USB priključak tipa C
- ugrađeni crveni LED
- RGB NeoPixel
- ugrađeni STEMMA QT konektor sa opcionim SWD portom za debugovanje



Slika 1.8: Adafruit Feather Rp2040.

1.6.2 Adafruit ItsyBitsy RP2040

ItsyBitsy RP2040 (slika 1.9) je još jedna ploča mikrokontrolera zasnovana na RP2040 kompanije Adafruit. Njegove osnovne funkcije su veoma slične funkcijama za Feather RP2040. Ima USB-micro B konektor i obezbeđuje +5 V izlaz.



Slika 1.9: Adafruit ItsyBitsy RP2040.

1.6.3 Pimoroni PicoSystem

Ovo je minijaturna ploča za igre (slika 1.10) razvijena oko mikrokontrolera RP2040.

Njene osnovne funkcije su:

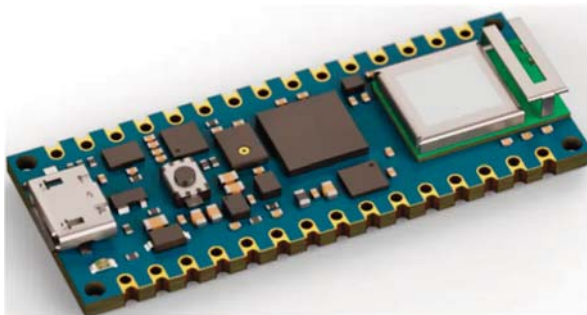
- 133 MHz radni takt
- 264 KB SRAM
- LCD ekran
- džojped
- dugmad
- LiPo baterija
- USB-C konektor za napajanje



Slika 1.10: Pimoroni PicoSystem.

1.6.4 Arduino Nano RP2040 Connect

Ova ploča (slika 1.11) nudi 16 MB fleš, 9-osni IMU i mikrofoni. Ima veoma efikasan deo za napajanje opremljen Wi-Fi/Bluetooth-om.

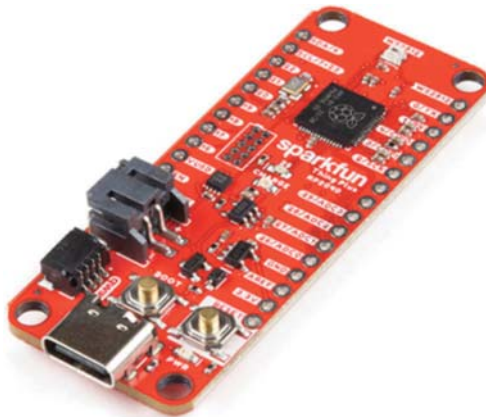


Slika 1.11: Arduino Nano RP2040 Connect.

1.6.5 SparkFun Thing Plus RP2040

Ova razvojna platforma (slika 1.12) obezbeđuje slot za SD karticu, 16 MB fleš memoriju, JST konektor za jednočelijsku bateriju, WS2812 RGB LED, JTAG pinove i Qwiic konektor. Njegove osnovne funkcije su:

- 133 MHz brzina
- 264 KB SRAM
- 4× 12-bitni ADC
- 2× UART, 2× I²C, 2× SPI
- 16× PWM
- 1× tajmer sa četiri alarma

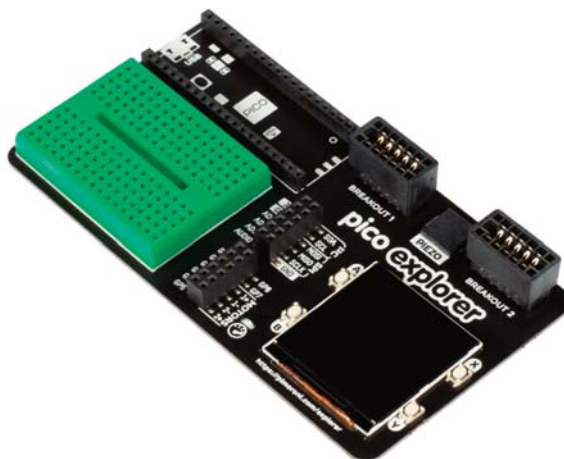


Slika 1.12: SparkFun Thing Plus RP2040.

1.6.6 Pimoroni Pico Explorer Base

Ova razvojna ploča (slika 1.13) uključuje malu matičnu ploču i 240 × 240 IPS LC displej sa 4 taktilna dugmeta. Na ploči se nalazi utičnica za priključivanje Raspberry Pi Pico ploče. Osnovne funkcije ove razvojne ploče su:

- piezo zvučnik
- 1,54-inčni IPS LCD
- 4× dugmad
- 2× polumostni motorni pogon
- dve breakout I²C utičnice
- lak pristup GPIO i ADC pinovima
- mini prototipska ploča
- nije potrebno lemljenje
- Raspberry Pi Pico nije isporučen



Slika 1.13: Pimoroni Pico Explorer Base.

1.6.7 SparkFun MicroMod RP2040 Processor

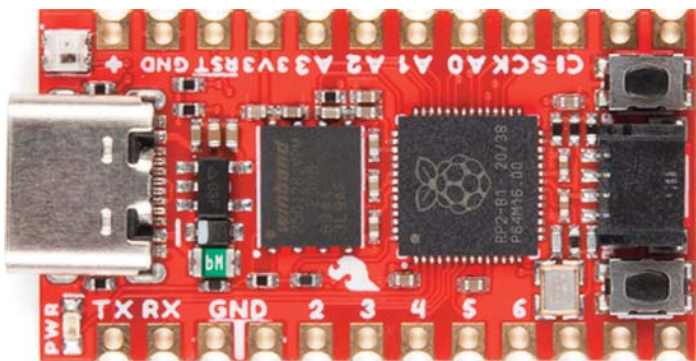
Ova ploča (slika 1.14) uključuje MicroMod M.2 konektor za pristup GPIO pinovima.



Slika 1.14: SparkFun MicroMod RP 2040 Processor.

1.6.8 SparkFun Pro Micro RP2040

Ova ploča (slika 1.15) uključuje ES2812B adresabilni LED, dugme za pokretanje, dugme za resetovanje, Qwiic konektor, USB-C interfejs za napajanje, PTC osigurač i aluminijske GPIO pločice.

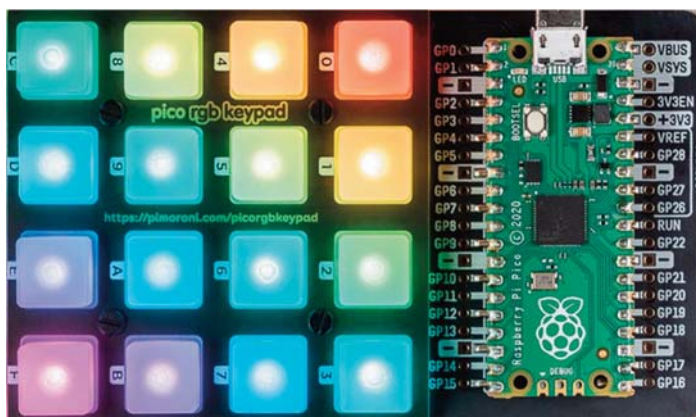


Slika 1.15: SparkFun Pro Micro RP2040.

1.6.9 Pico RGB Keypad Base

Ova ploča je opremljena tastaturom 4×4 osvetljenom u duginim bojama (slika 1.16) sa APA102 LED-ovima. Osnovne funkcije su:

- 4×4 tastatura
- 16× APA102 RGB LED-a
- tastatura povezana preko I²C I/O proširivača
- označeni GPIO pinovi

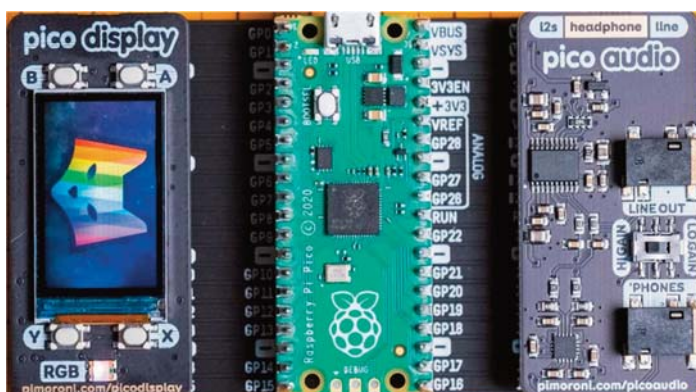


Slika 1.16 Pico RGB Keypad Base.

1.6.10 Pico Omnibus

Ovo je ploča za proširenje (slika 1.17) za Pico. Osnovne funkcije ove ploče uključuju:

- označeni GPIO pinovi
- dve ciljne oblasti sa označenim (preslikanim) muškim zaglavljima za pričvršćivanje dodataka
- 4x gumene nožice
- kompatibilna sa Raspberry Pi Pico-om
- potpuno sastavljena
- dimenzija cca. 94 × 52 × 12 mm



Slika 1.17: Pico Omnibus.

1.6.11 Pimoroni Pico VGA Demo Base

Ova ploča (slika 1.18) ima VGA izlaz i slot za SD karticu. Osnovne funkcije su:

- pokreće je Raspberry Pi Pico
- 15-pinski VGA konektor
- I²S DAC za linijski audio izlaz

- PWM audio izlaz
- Slot za SD karticu
- dugme Reset
- zaglavlja za instaliranje Raspberry Pi Pico-a
- tri korisnička prekidača
- nije potrebno lemljenje



Slika 1.18: Pimoroni Pico VGA Demo Base.

Poglavlje 2 • Raspberry Pi Pico programiranje

2.1 Pregled

U vreme pisanja ove knjige, Raspberry Pi Pico prihvata programiranje na sledećim programskim jezicima:

- C/C++
- MicroPython
- asemblerski jezik

Iako je Pico podrazumevano podešen za upotrebu sa moćnim i popularnim jezikom C/C++, mnogim početnicima je lakše da koriste MicroPython, koji je verzija programskog jezika Python razvijena posebno za mikrokontrolere.

U ovom poglavlju ćemo naučiti kako da instaliramo i koristimo programski jezik MicroPython. Koristićemo Thonny uređivač teksta koji je posebno razvijen za Python programe.

Mnogi radni i potpuno testirani projekti biće prikazani u narednim poglavljima korišćenjem MicroPython-a sa našim Pico-om. Upotreba C jezika će takođe biti razmatrana u kasnijim poglavljima sa nekim projektima.

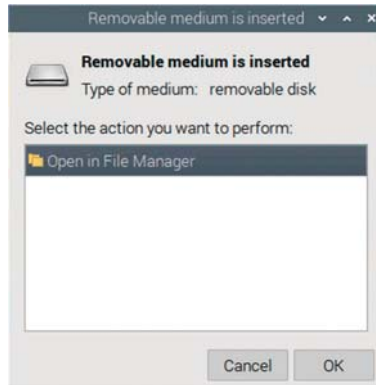
2.2 Instaliranje MicroPython-a na Pico-u

MicroPython mora biti instaliran na Pico-u da bi se ploča mogla koristiti. Nakon instalacije, MicroPython ostaje na Pico-u, osim ako nije zamenjen nečim drugim. Instaliranje MicroPython-a zahteva Internet vezu, a to je potrebno samo jednom. Pošto Pico nema Wi-Fi vezu, moraćemo da koristimo računar sa pristupom Internetu. To se može uraditi ili korišćenjem Raspberry Pi-a (npr. Raspberry Pi-a 4) ili korišćenjem računara. U ovom odeljku ćete videti kako da instalirate MicroPython pomoću oba metoda.

2.2.1 Korišćenje Raspberry Pi-a 4 za pomoć pri instaliranju MicroPython-a na Pico-u

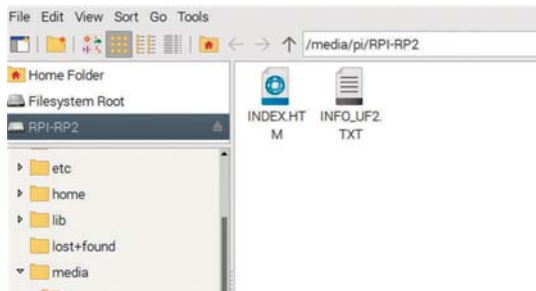
Koraci su sledeći.

- Pokrenite Raspberry Pi 4 i prijavite se na Desktopu.
- Proverite da li je vaš Raspberry Pi povezan na Internet.
- Držite pritisnuto dugme **BOOTSEL** na Pico-u.
- Povežite Pico sa jednim od USB portova Raspberry Pi-a 4 pomoću micro-USB kabla dok držite pritisnuto dugme.
- Sačekajte nekoliko sekundi i otpustite dugme **BOOTSEL**.
- Trebalo bi da vidite da se Pico pojavljuje kao prenosivi disk. Kliknite na **OK** u prozoru **Removable medium is inserted** (slika 2.1).



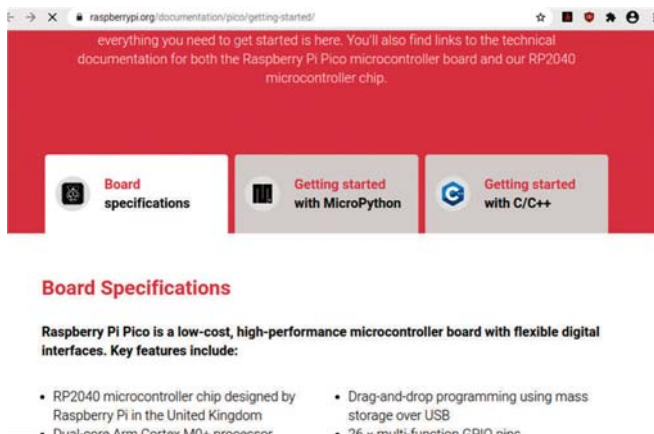
Slika 2.1: Kliknite na OK.

- U prozoru **File Manager** videćete dve datoteke koje se zovu **INDEX.HTM** i **INFO_UF2.TXT** (slika 2.2).



Slika 2.2: Primitićete dve datoteke.

- Dva puta kliknite na datoteku **INDEX.HTM** i skrolujte nadole.
- Trebalo bi da vidite prikazanu poruku **Welcome to your Raspberry Pi Pico** prikazanoj na veb stranici.



Slika 2.3: Prikazana poruka.

Poglavlje 3 • Jednostavni hardverski projekti za Raspberry Pi Pico

3.1 Pregled

U ovom poglavlju ćemo razvijati jednostavne hardverske projekte pomoću Raspberry Pi Pico-a tako što ćemo koristiti Thonny tekstualni uređivač. Za svaki projekat će biti dati sledeći podnaslovi kad god je to primenljivo:

- Naslov
- Opis
- Cilj
- Blok dijagram
- Električna šema
- Listing programa
- Predlozi za budući rad

Svi programi u ovom poglavlju su razvijeni pomoću Thonny-a na računaru, pri čemu je Raspberry Pi Pico povezan na USB port računara.

3.2 Projekat 1: Trepćući LED - Upotreba ugrađenog LED-a

Opis: U ovom projektu ugrađeni LED trepće svaku sekundu.

Cilj: Cilj ovog projekta je da upozna čitaoca sa osnovnim kontrolnim iskazima za GPIO.

Listing programa: Na slici 3.1 je prikazan listing programa (Program: **LEDINT**). Na početku programa se uvoze moduli **machine** i **utime**. Zatim se LED dodeljuje port pinu GP25 i konfiguriše kao izlaz. Ostatak programa se izvršava u beskonačnoj petlji, sve dok je korisnik ne zaustavi. Unutar ove petlje, LED se uključuje iskazom **LED.value(1)**. Nakon pauze od jedne sekunde, LED se isključuje iskazom **LED.value(0)**. Funkcija **utime.sleep(n)** stvara **n** sekundi kašnjenja u programu.

```
#-----  
#           FLASHING THE ON-BOARD LED  
#           =====  
#  
# In this program the on-board LED (at GP25) is flashed  
# every second  
#  
# Author: Dogan Ibrahim  
# File  : LEDINT.py  
# Date  : February, 2021  
#-----  
  
import machine  
import utime
```

```

LED = machine.Pin(25, machine.Pin.OUT)           # LED at GP25

while True:                                     # DO FOREVER
    LED.value(1)                                # LED ON
    utime.sleep(1)                              # Wait 1 second
    LED.value(0)                                # LED OFF
    utime.sleep(1)                              # Wait 1 second

```

Slika 3.1: Program LEDINT.

Na slici 3.1, modul **import machine** uvozi i druge funkcije, osim funkcije **Pin**. Možemo pojednostaviti program tako da uvozimo samo funkcije **Pin**, kao što je prikazano na slici 3.2 (Program: **LEDINT2**). Moguće je da morate zaustaviti program tako što ćete izabrati **Run**, a zatim **Stop/Restart backend** kako biste mogli sačuvati novi program dok je već pokrenut program na Pico-u.

```

#-----
#           FLASHING THE ON-BOARD LED
#           =====
#
# In this program the on-board LED (at GP25) is flashed
# every second. In this version only module Pin is imported
#
# Author: Dogan Ibrahim
# File  : LEDINT2.py
# Date  : February, 2021
#-----

from machine import Pin
import utime

LED = Pin(25, Pin.OUT)           # LED at GP25

while True:                       # DO FOREVER
    LED.value(1)                   # LED ON
    utime.sleep(1)                 # Wait 1 second
    LED.value(0)                   # LED OFF
    utime.sleep(1)                 # Wait 1 second

```

Slika 3.2: Program LEDINT2.

Modul **machine** podržava sledeće funkcije (u narednim poglavljima ćemo videti kako se koriste te funkcije):

- Pin
- Tajmer
- ADC
- I²C i Soft I²C

Poglavlje 4 • Korišćenje analognih-digitalnih pretvarača (ADC)

4.1 Pregled

Većina senzora u stvarnom životu su analogni, generišući analogni izlazni napon ili struju koja je proporcionalna izmerenoj promenljivoj. Bez upotrebe ADC-a, takvi senzori ne mogu biti direktno povezani sa digitalnim računarima. U ovom poglavlju ćemo naučiti kako da koristimo ADC kanale Raspberry Pi Pico-a.

Većina ADC-ova za opštu upotrebu su široki 8 ili 10 bitova, iako su neki profesionalni višeg stepena široki 16 ili čak 32 bita. Vreme konverzije ADC-a je jedna od njegovih važnih specifikacija. To je vreme potrebno ADC-u da pretvori analogni ulaz u digitalni. Što je vreme konverzije manje, to je bolje. Neki jeftiniji ADC-ovi generišu konvertovane digitalne podatke u serijskom formatu, a neki skuplji, profesionalni generišu paralelne digitalne izlaze.

Raspberry Pi Pico ima 5 ADC kanala. Četiri od njih se nalaze na pinovima GP26, GP27, GP28 i GP29 – poznati kao analogni kanali 0, 1, 2 i 3. Prva 3 kanala su dostupna na GPIO pinovima, a četvrti se može koristiti za merenje napona VSYS ploče. Takođe postoji ugrađeni ADC kanal 4 koji je interno povezan sa senzorom temperature.

ADC Pico-a ima rezoluciju od 12 bita, pretvarajući analogni ulazni napon u 4096 (od 0 do 4095) nivoa. Međutim, MicroPython transformiše izlaz u 16-bitni broj u opsegu od 0 do 65535.

Referentni napon ADC-a koji se koristi na Pico-u je +3,3 V. Kada se koristi takav ADC, rezolucija je $3300 \text{ mV} / 65535 = 0,050 \text{ mV}$ po bitu, ili $50 \mu\text{V/bit}$. Stoga, analogni ulazni napon od $0,050 \text{ mV}$ daje digitalni izlaz od 00000000 00000001, $0,1 \text{ mV}$ daje 00000000 00000010 i tako dalje.

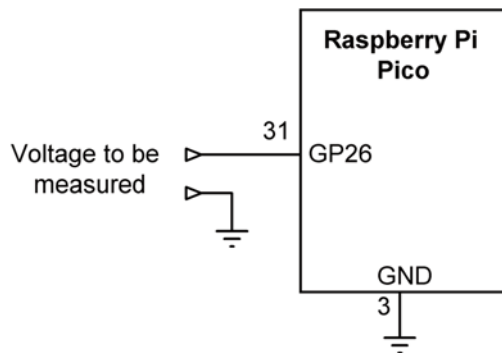
U ovom poglavlju ćemo razviti nekoliko projekata pomoću ADC-a koji Pico nudi.

4.2 Projekat 1: Voltmetar

Opis: Ovo je jednostavan projekat voltmetra u kojem se meri napon spoljnog izvora i prikazuje na ekranu u milivoltima.

Cilj: Cilj ovog projekta je da pokaže kako se Pico ADC kanali mogu koristiti za čitanje analognog ulaznog napona.

Električna šema: Na slici 4.1 je šema projekta. U ovom projektu, napon koji treba izmeriti se primenjuje na analogni ulaz GP26 (pin 31, kanal 0). Morate se postarati da ulazni napon ne premaši +3,3 V. Ako je potrebno meriti više napone, tada možete koristiti kola potencijalnog razdelnika napona na ulazu ADC-a.



Slika 4.1: Električna šema projekta.

Listing programa: Na slici 4.2 je prikazan listing programa i primer ispisa programa (Program: **Voltmeter**). Na početku programa modul **ADC** 'mašine' se uvozi u program i promenljivoj **AnalogIn** dodeljuje se analogni ulazni kanal 0. Zatim se definiše faktor konverzije kao $3300 / 65535$. Vrednost pročitana sa analognog kanala mora biti pomnožena ovim brojem kako bi bila izračunata stvarna vrednost izmerenog napona. Ostatak programa se izvršava u petlji u kojoj se očitava ulazni napon i konvertuje u milivolte, a zatim prikazuje na ekranu računara.

```

1 #-----
2 #           VOLTMETER
3 #           =====
4 #
5 # This is a voltmeter project. The voltage to be measured
6 # is applied to GP26 (pin 31) of the Pico
7 #
8 # Author: Dogan Ibrahim
9 # File  : Voltmeter.py
10 # Date  : February 2011
11 #-----
12 from machine import ADC
13 import utime
14
15 AnalogIn = ADC(0)                # ADC channel 0
16 Conv = 3300 / 65535              # Conversion factor
17
18 while True:                      # Do forever
19     mV = AnalogIn.read_u16()     # Read input
20     mV = mV * Conv                # Input in mV
21     print("Voltage = %5.2f" %mV)  # Display
22     utime.sleep(1)               # Wait 1 second
23
Shell
Voltage = 53.98
Voltage = 53.98
Voltage = 53.17
Voltage = 53.17

```

Slika 4.2: Program Voltmeter.

Prikazivanje napona na LCD-u

Jednostavno je modifikovati program tako da prikazuje izmereni napon na LCD-u. Pre svega, izradite kolo koje je prikazano na slici 3.71 i primenite napon koji treba meriti na GP26 Pico-a (pin 31), pri čemu vodite računa da napon ne premaši +3,3 V. Program za prikaz izmerenog napona na LCD-u (Program: **VoltLCD**) je prikazan na slici 4.3.

Poglavlje 5 • Evidencija podataka

5.1 Pregled

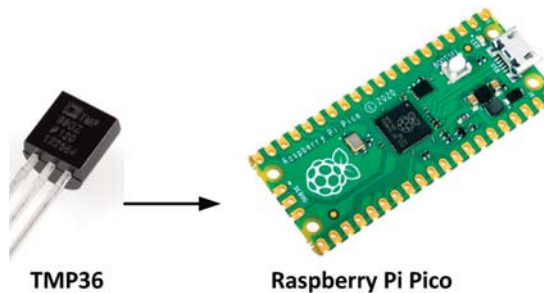
Sistem za upravljanje datotekama na Raspberry Pi Pico-u omogućava manipulaciju datotekama u memoriji, što uključuje kreiranje novih datoteka, čitanje iz datoteka i pisanje u datoteke. U ovom poglavlju ćemo naučiti kako da evidentiramo podatke o temperaturi u datoteke.

5.2 Projekat 1: Evidentiranje podataka o temperaturi

Opis: U ovom projektu ćemo kreirati datoteku koja je nazvana **Temp.txt** i sačuvati podatke o ambijentalnoj temperaturi svake sekunde, sa vremenom označenim u relativnim sekundama. Podaci će biti sačuvani 30 sekundi, tj. biće sačuvano 30 zapisa.

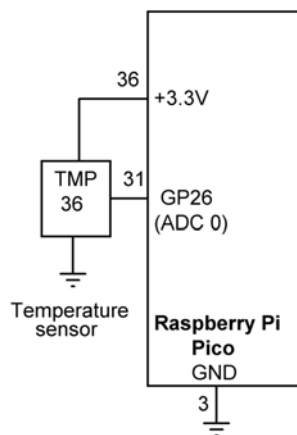
Cilj: Cilj ovog projekta je da pokaže kako se podaci o temperaturi (ili bilo koji drugi podaci) mogu sačuvati u datoteci.

Blok dijagram: Na slici 5.1 je prikazan blok dijagram projekta.



Slika 5.1: Blok dijagram projekta.

Električna šema: Električna šema je prikazan na slici 5.2. Čip senzora temperature TMP36 je povezan na analogni kanal 0 (GP26, pin 31).



Slika 5.2: Električna šema projekta.

Listing programa: Na slici 5.3 je prikazan listing programa (Program: **LogTemp**). Na početku programa, moduli **ADC** mašine i **utime** su uvezeni u program. Kreira se nova datoteka (tj. otvara se u režimu pisanja ‘w’) koja se zove **Temp.txt** i zaglavljuje **Ambiental Temperature** nakon čega se u datoteku upisuje znak za novi red. Ostatak programa se izvršava u petlji **for** koja se ponavlja 30 puta. Unutar ove petlje, temperatura se čita sa TMP36, konvertuje u stepene Celzijusa, a zatim smešta u datoteci sa relativnim sekundama. Na kraju petlje, datoteka se zatvara, a na ekranu se ispisuje poruka **Data has been written to file...**

```
#-----
#           LOGGING THE TEMPERATURE DATA
#           =====
#
# This program measures the temperature using an external
# temperature sensor chip and logs the data every second
# for 30 seconds (i.e. 30 records)
#
# Author: Dogan Ibrahim
# File  : LogTemp.py
# Date  : February 2021
#-----
from machine import ADC
import utime

AnalogIn = ADC(0)                # ADC channel 0
Conv = 3300 / 65535              # Conversion factor

file = open("Temp.txt", "w")     # Create a new file
file.write("Ambient Temperature\n") # Write heading

for secs in range(30):          # Do forever
    V = AnalogIn.read_u16()      # Read temp
    mV = V * Conv                # Convert to Volts
    Temp = (mV - 500.0) / 10.0   # Convert to temp
    Tempstr = str(Temp)[:5]      # Convert to string
    data = str(secs+1) + "    " + Tempstr + "\n"
    file.write(data)             # Write to file
    utime.sleep(1)              # Wait 1 second

file.close()                    # Close file
print("Data has been written to file...")
```

Slika 5.3: Program LogTemp.

Sadržaj datoteke može se prikazati na Thonny ekranu klikom na **File**, a zatim na **Open**. Izaberite Raspberry Pi Pico, pa kliknite na datoteku **Temp.txt** da biste je prikazali kako

Poglavlje 6 • Modulisanje širine impulsa (PWM)

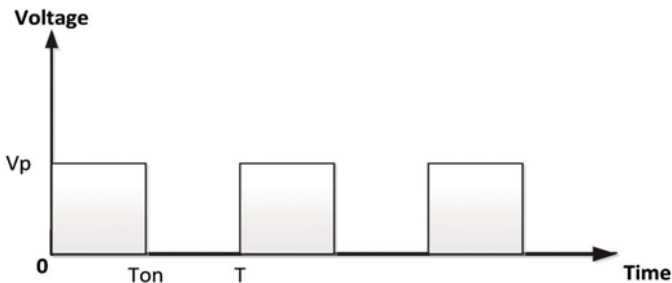
6.1 Pregled

Digitalni oblik modulisanja širine impulsa (PWM) se često koristi za pokretanje velikih opterećenja kao što su motori, aktuatori, grejači i slično. Kao što ćemo videti u ovom poglavlju, PWM je u osnovi pozitivni kvadratni talas čija se širina impulsa može menjati. Menjanjem širine impulsa, efikasno možemo promeniti prosečnu vrednost napona isporučenog opterećenju.

6.2 Osnovna teorija modulisanja širine impulsa

PWM je često korišćena tehnika za kontrolu snage isporučene analognim opterećenjima korišćenjem digitalnih signala. Iako se analogni naponi (i struja) mogu koristiti za kontrolu isporučene snage, oni imaju nekoliko nedostataka. Kontrolisanje velikih analognih opterećenja zahteva velike napone i struju koji se ne mogu lako dobiti korišćenjem standardnih analognih kola i DAC-ova. Precizna analogna kola mogu biti teška, velika i skupa, a i osetljiva su na šum. Korišćenjem PWM tehnike, prosečna vrednost napona (i struje) koji se dovodi do opterećenja kontroliše se uključivanjem i isključivanjem napona velikom brzinom. Što je duže vreme snaga uključena, to je veći efektivni napon isporučen opterećenju.

Na slici 6.1 je prikazan tipičan PWM signala, u kojem je signal u osnovi ponavljajući pozitivni impuls, koji ima period T , vreme uključjenja T_{ON} i vreme isključenja $T - T_{ON}$ sekundi. Minimalne i maksimalne vrednosti napona kojim se snabdeva uređaj su 0 i V_p . Prekidačka frekvencija PWM-a obično je postavljena da bude vrlo visoka (obično reda veličine nekoliko kHz) tako da ne utiče na opterećenje koje koristi snagu. Glavna prednost PWM-a je da se opterećenje efikasno koristi, jer je gubitak snage u prekidaču veoma nizak. Kada je prekidač uključen, praktično nema pada napona na prekidaču, a kada je prekidač isključen, uređaj se ne snabdeva strujom.



Slika 6.1: Osnovni PWM signal.

Radni ciklus D PWM signala je definisan kao odnos vremena uključjenja i njegovog perioda. Matematički izraženo,

$$D = T_{ON} / T$$

Radni ciklus se obično izražava kao procenat i stoga je,

$$D = (T_{ON} / T_{OFF}) \times 100\%$$

Promenom radnog ciklusa između 0% i 100% možemo efikasno kontrolisati prosečni napon koji se dovodi do opterećenja između 0 i V_p .

Prosečna vrednost napona primenjenog na opterećenje se može izračunati uzimajući u obzir opšti PWM signal prikazan na slici 1. Prosečna vrednost A signala $f(t)$ sa periodom T , maksimalnom vrednošću y_{\max} i minimalnom vrednošću y_{\min} izračunava se kao:

$$A = \frac{1}{T} \int_0^T f(t) dt$$

ili

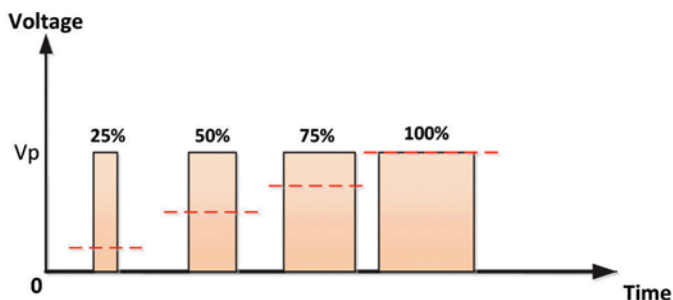
$$A = \frac{1}{T} \left(\int_0^{T_{ON}} y_{\max} dt + \int_{T_{ON}}^T y_{\min} dt \right)$$

U PWM signalu je $y_{\min} = 0$, a gornja jednačina postaje

$$A = \frac{1}{T} (T_{ON} y_{\max})$$

ili $A = D y_{\max}$

Kao što se može videti iz prethodne jednačine, prosečna vrednost napona koji se isporučuje opterećenju direktno je proporcionalna radnom ciklusu PWM signala, a promenom radnog ciklusa kontrolišemo prosečan napon opterećenja. Na slici 6.2 je prikazan prosečan napon za različite vrednosti radnog ciklusa.



Slika 6.2: Prosečan napon (prikazan kao isprekidana linija) isporučen opterećenju.

Zanimljivo je primetiti da, uz pravilnu primenu niskopropusnog filtera, PWM može biti korišćen kao DAC ukoliko MCU ne poseduje DAC kanal. Promenom radnog ciklusa možemo efikasno menjati prosečan analogni napon koji se isporučuje opterećenju.

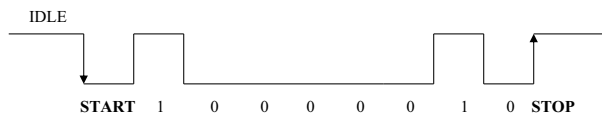
Poglavlje 7 • Serijska komunikacija (UART)

7.1 Pregled

Serijska komunikacija je jednostavan način slanja podataka na velike udaljenosti brzo i pouzdano. Najčešći metod serijske komunikacije zasniva se na RS-232 standardu. Prema ovom standardu, podaci se šalju preko jedne linije od predajnog uređaja do prijemnog uređaja u bit-serijskom formatu po unapred određenoj brzini, poznatoj i kao bodna brzina, ili broj bitova koji se šalju svake sekunde. Tipične bodne brzine su 4800, 9600, 19200, 38400 itd.

RS-232 serijska komunikacija je oblik asinhronne transmisije podataka u kojoj se podaci šalju karakter po karakter. Svakom karakteru prethodi početni bit, nakon čega slede sedam ili osam bitova podataka, opcioni bit parnosti i jedan ili više zaustavnih bitova. Najčešće korišćeni format je osam bitova podataka, bez bita parnosti i jednim zaustavnim bitom (8-N-1). Stoga se okvir podataka sastoji od 10 bitova. Uz bodnu brzinu od 9600, možemo prenositi i primati 960 karaktera svake sekunde. Najmanje značajan bit podataka se prenosi prvi, a najznačajniji bit se prenosi poslednji.

U standardnoj RS-232 komunikaciji, logička jedinica je definisana na -12 V , a logička 0 na $+12\text{ V}$. Na slici 7.1 je prikazano kako se karakter "A" (ASCII binarni obrazac 0010 0001) prenosi preko serijske linije. Signalna linija je obično u mirovanju na -12 V . Početni bit se prvo šalje tako što linija prelazi sa high na low. Zatim se šalje osam bitova podataka, počevši od najmanje značajnog bita. Na kraju, zaustavni bit se šalje podizanjem linije sa Low na High.



Slika 7.1: Slanje karaktera "A" u serijskom formatu.

U serijskoj komunikaciji koriste se najmanje tri linije za komunikaciju: predajnik (TX), prijemnik (RX) i uzemljenje (GND). Neki sistemi serijske komunikacije visoke brzine koriste dodatne kontrolne signale za sinhronizaciju, kao što su CTS, DTR i slično. Neki sistemi koriste tehnike softverske sinhronizacije u kojima se specijalan karakter (XOFF) koristi da se pošiljaocu ukaže da prestane slanje, a drugi karakter (XON) se koristi da se pošiljaocu ukaže da nastavi prenos. RS-232 uređaji su međusobno povezani pomoću dva tipa konektora: 9-polni konektor i 25-polni konektor. U tabeli 7.1 su prikazani TX, RX i GND pinovi svakog tipa konektora. Konektori koji se koriste u RS232 serijskoj komunikaciji prikazani su na slici 7.2.

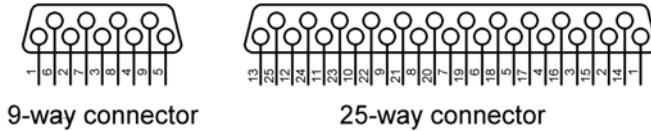
9-pinski konektor

Pin	Function
2	Transmit (TX)
3	Receive (RX)
5	Ground (GND)

25-pinski konektor

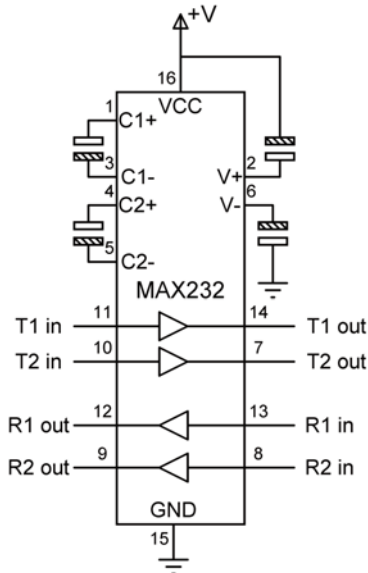
Pin	Function
2	Transmit (TX)
3	Receive (RX)
7	Ground (GND)

Tabela 7.1: Minimalni broj pinova potrebnih za RS232 serijsku komunikaciju.



Slika 7.2: RS-232 konektori.

Kao što je prethodno opisano, nivoi napona RS-232 su na ± 12 V. S druge strane, ulazno/izlazni portovi mikrokontrolera obično rade na nivoima napona od 0 do +5 V. Zbog toga je neophodno prevesti nivoe napona pre povezivanja mikrokontrolera sa uređajem kompatibilnim sa RS-232. Dakle, izlazni signal iz mikrokontrolera mora biti pretvoren u ± 12 V, a ulaz iz uređaja RS-232 mora biti pretvoren u 0 do +5 V pre nego što se može povezati sa mikrokontrolerom. Ovaj prevod napona se obično izvodi pomoću specijalnih čipova za pretvaranje napona RS232. Jedan takav popularan čip je MAX232. To je dvostruki pretvarački čip koji ima konfiguraciju pinova kao što je prikazano na slici 7.3. Ovaj poseban uređaj zahteva četiri eksterna kondenzatora od $1\mu\text{F}$ za svoj rad.



Slika 7.3: Konfiguracija pinova za MAX232.

Poglavlje 8 • Interfejs I²C magistrale

8.1 Pregled

I²C magistrala se često koristi u projektima zasnovanim na mikrokontrolerima. U ovom poglavlju ćemo razmatrati upotrebu ove magistrale na Raspberry Pi Pico-u. Cilj je da čitaoca upoznamo sa funkcijama biblioteke I²C magistrale i pokažemo kako se mogu koristiti u stvarnom projektu. Pre nego pogledamo detalje projekta, korisno je upoznati osnovne principe I²C magistrale.

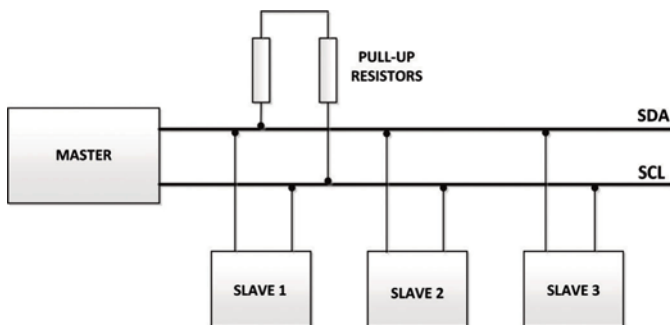
8.2 I²C magistrala

I²C je jedan od najčešće korišćenih protokola za komunikaciju mikrokontrolera sa spoljnim uređajima kao što su senzori i aktuatori. Magistrala je mrežna struktura sa jednim master uređajem i više slave uređaja, i može da radi u standardnom režimu na 100 Kbit/s, pri maksimalnoj brzini od 400 Kbit/s; u brzom režimu od 1 Mbit/s i brzom prenošenju podataka od 3,2 Mbit/s. Magistrala se sastoji od dve open-drain žice, SDA i SCL, koje se povlače nagore pomoću otpornika:

SDA: linija podataka

SCL: linija takta

Na slici 8.1 je prikazana I²C struktura magistrale sa jednim master uređajem i tri slave uređaja.



Slika 8.1: I²C magistrala sa jednim master uređajem i tri slave uređaja.

Budući da se I²C magistrala zasniva na samo dve žice, treba postojati način za adresiranje pojedinačnog slave uređaja na istoj magistrali. Iz tog razloga, protokol definiše da svaki slave uređaj ima jedinstvenu adresu za datu magistralu. Ova adresa je obično široka 7 bita. Kada je magistrala slobodna, obe linije su u stanju HIGH. Svu komunikaciju na magistrali pokreće i završava master, koji prvo šalje početni (START) bit, a transakciju završava slanjem zaustavnog (STOP) bita. To obaveštava sve slave uređaje da neki podaci pristižu na magistralu, pa svi slave uređaji oslušuju magistralu. Nakon početnog bita, šalje se 7 bita jedinstvene adrese slave uređaja. Svaki slave uređaj na magistrali ima svoju adresu, što omogućava da u bilo kom trenutku na magistrali komu-

nicira samo adresirani slave kako bi se izbegli sudari. Poslednji poslani bit je bit za čitanje/pisanje, pa ako je ovaj bit 0, to znači da master želi da piše na magistrali (npr. u registar slave uređaja). Ako je ovaj bit 1, to znači da master želi da čita sa magistrale (npr. iz registra uređaja). Podaci se šalju na magistralu sa MSB bitom prvo. Nakon svakog bajta dodaje se bit za potvrdu (ACK), pri čemu primalac može da signalizira pošiljaocu da je bajt uspešno primljen i da se može poslati još jedan bajt. ACK bit se šalje devetom taktom impulsu.

Komunikacija preko I²C magistrale se obavlja sledećim redosledom:

- Master šalje na magistralu adresu slave uređaja sa kojim želi da komunicira.
- LSB je R/W bit koji određuje smer prenosa podataka, tj. od master do slave uređaja (R/W = 0) ili od slave do master uređaja (R/W = 1).
- Traženi bajtovi se šalju, od kojih je svaki isprepleten sa ACK bitom, dok se ne pojavi uslov zaustavljanja.

U zavisnosti od tipa korišćenog slave uređaja, neke transakcije mogu zahtevati zasebnu transakciju. Na primer, koraci za čitanje podataka iz uređaja za memoriju kompatibilnog sa I²C su dati ispod.

- Master započinje transakciju u režimu pisanja (R/W = 0) slanjem adrese slave uređaja na magistralu.
- Lokacija memorije koja treba da se preuzme zatim se šalje kao dva bajta (pretpostavljajući memoriju od 64 kilobajta).
- Master šalje uslov zaustavljanja (STOP) da završi transakciju.
- Master započinje novu transakciju u režimu čitanja (R/W = 1) slanjem adrese slave uređaja na magistralu.
- Master čita podatke iz memorije. Ako se čita memorija u sekvencijalnom formatu, tada će se čitati više od jednog bajta.
- Master postavlja uslov zaustavljanja na magistralu.

8.3 I²C pinovi Raspberry Pi Pico-a

Raspberry Pi Pico ima dve I²C magistrale, koji se zovu I2C0 i I2C1 (pogledajte sliku 8.2). Kao što je prikazano na slici, I²C pinovi su duplirani i dele se sa drugim pinovima. Na primer, GP0 (pin 1) je I2C0 SDA pin i GP1 (pin 1) je I2C0 SCL pin. Takođe, GP16 (pin 21) je I2C0 SDA pin i GP17 (pin 22) je I²C I2C0 SCL pin.

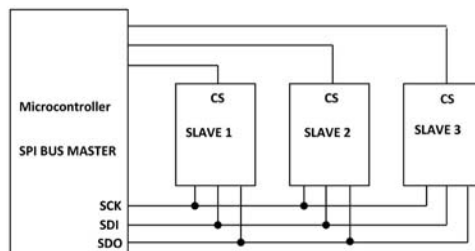
Poglavlje 9 • Interfejs SPI magistrale

9.1 Pregled

U ovom poglavlju ćemo razvijati projekte koristeći SPI magistralu (serijski periferni interfejs) sa Raspberry Pi Pico. SPI magistrala je obično široko korišćen protokol za povezivanje senzora i mnogih drugih uređaja sa mikrokontrolerima. SPI magistrala je magistrala tipa master-slave protokola. U ovom protokolu, jedan uređaj (mikrokontroler) je označen kao glavni, a jedan ili više drugih uređaja (obično senzori) su označeni kao slave. U minimalnoj konfiguraciji magistrale postoji jedan master i samo jedan slave. Master uspostavlja komunikaciju sa slave-ovima i kontroliše sve aktivnosti na magistrali.

Slika 9.1 prikazuje primer SPI magistrale sa jednim master uređajem i 3 slave uređaja. SPI magistrala koristi 3 signala: Radni takt (SCK), ulaz podataka (SDI ili RX) i izlaz podataka (SDO ili TX). SDO master uređaja je povezan sa SDI-ovima slave uređaja, a SDO-ovi slave uređaja su povezani sa SDI-ovima master uređaja. Master generiše SCK signale kako bi omogućio prenos podataka na magistralu. U svakom impulsu takta jedan bit podataka se premešta od master uređaja do slave uređaja, ili od slave uređaja do master uređaja. Komunikacija je samo između master i slave uređaja, a slave uređaji ne mogu komunicirati jedni sa drugima. Važno je napomenuti da samo jedan slave može biti aktivan u bilo kom trenutku jer ne postoji mehanizam za identifikaciju slave-a. Dakle, slave uređaji imaju linije za omogućavanje (npr. CS ili CE) koje obično kontroliše master. Tipična sekvenca komunikacije između master uređaja i nekoliko slave uređaja je data u nastavku.

- Master omogućava SLAVE 1.
- Master šalje SCK signale za čitanje ili pisanje podataka slave uređaju 1.
- Master deaktivira CS=SLAVE 1 i omogućava SLAVE 2.
- Master šalje SCK signale za čitanje ili pisanje podataka SLAVE uređaju 2.
- Gore navedeni proces se nastavlja po potrebi.



Slika 9.1: SPI magistrala sa jednim master uređajem i 3 slave uređaja.

Master i slave moraju se dogovoriti o polaritetu radnog takta i faze takta na liniji, koji su poznati kao režimi SPI magistrale (**SPI bus modes**). Ova dva podešavanja se zovu Clock Polarity (CPOL) i „Clock Phase“ (CPHA). CPOL i CPHA mogu imati sledeće vrednosti:

CPOL	Clock-Active State
1	Clock active High
1	Clock active Low

CPHA	Clock Phase
1	Clock out of phase with data
2	Clock in phase with data

Četiri SPI režima su:

Mode	CPOL	CPHA
0	0	0
1	0	1
2	1	0
3	1	1

Kada je CPOL = 0, aktivno stanje takta je 1, a neaktivno stanje je 0. prijem podataka se odvija na rastućoj ivici takta, a slanje podataka se odvija na opadajućoj ivici takta. Kada je CPHA = 1, prijem podataka se odvija na opadajućoj ivici takta, a slanje podataka na rastućoj ivici takta.

Kada je CPOL = 1, aktivno stanje takta je 0, a neaktivno stanje je 1. Kada je CPHA = 0, prijem podataka se odvija na opadajućoj ivici takta, a slanje podataka na rastućoj ivici. Kada je CPHA = 1, prijem podataka se odvija na rastućoj ivici takta, a slanje podataka na opadajućoj ivici.

9.2 Raspberry Pi Pico dva SPI porta

Na Raspberry Pi Pico-u postoje 2 SPI porta, nazvana SPI0 i SPI1. Slika 9.2 prikazuje konfiguraciju SPI pinova.

Poglavlje 10 • Wi-Fi sa Raspberry Pi Picom

10.1 Pregled

U ovom poglavlju razvijaćemo projekte koristeći Wi-Fi vezu za uspostavljanje komunikacije između Raspberry Pi Pico i pametnog telefona.

10.2 Projekat 1: Upravljanje LED diodom sa pametnog telefona koristeći Wi-Fi

Opis: U ovom projektu ćemo slati komande preko Wi-Fi veze sa mobilnog telefona kako bismo kontrolisali LED diodu (LED se može zameniti relejom, na primer, za kontrolu opreme) koja je povezana sa Raspberry Pi Picom. Komande moraju biti završene sa povratnim karakterom (CR/LF ili 'nova linija').

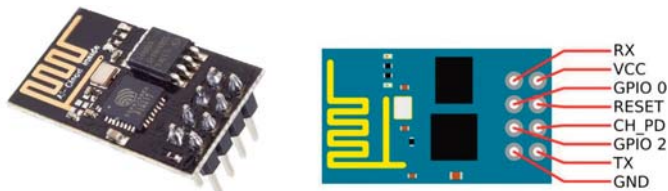
Važeće komande uključuju:

- LON - Uključi LED
- LOFF - Isključi LED

Cilj: Cilj ovog projekta je demonstracija korišćenja Wi-Fi povezivosti na Raspberry Pi Pico.

Povezivost Pico Wi-Fi: Raspberry Pi Pico nema ugrađeni Wi-Fi modul i stoga se ne može povezati sa Wi-Fi mrežom bez interfejsa sa eksternim Wi-Fi modulom. Možda najlakši i najjeftiniji način obezbeđivanja Wi-Fi mogućnosti Pico-u je korišćenjem ESP-01 procesorskog modula. Ovo je mala ploča (pogledajte Sliku 10.1), dimenzija samo 2,7 cm × 1,2 cm, zasnovana na ESP8266 procesorskom čipu, a košta oko 3 USD. ESP-01 ima sledeće karakteristike:

- Radni napon: +3.3 V
- Interfejs: korišćenje jednostavnih AT komandi preko serijskog porta/UART-a
- Integrisani TCP/IP protokolni stek. 802.11 b/g/n
- Ne zahteva spoljne komponente



Slika 10.1: ESP-01 procesorska ploča.

ESP-01 komunicira sa glavnim procesorom putem svojih TX i RX serijskih pinova. To je 8-pinski uređaj sa sledećim nazivima pinova:

VCC: Pin napajanja od +3.3 V

GND: masa

GPIO0: I/O pin. Ovaj pin mora biti povezan na +3.3 V za normalan rad, i na GND za uploadovanje firmvera na čip

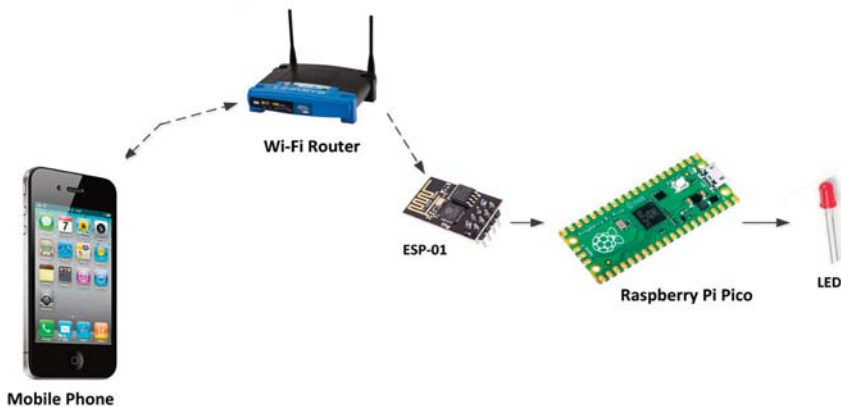
- GPIO2: Opšti I/O pin
- RST: Reset pin. Morate ga povezati na +3.3 V za normalan rad.
- CH_PD: Pin za omogućavanje. Morate ga povezati na +3.3 V za normalan rad.
- TX: Pin za serijski izlaz.
- RX: Pin za serijski ulaz.

ESP-01 pinovi nisu standardno kompatibilni sa prototipskom pločom, stoga je potreban adapter ako se ploča želi pričvrstiti na prototipsku ploču (vidi sliku 10.2).



Slika 10.2: Adapter za prototipsku ploču ESP-01.

Blok dijagram: Slika 10.3 prikazuje blok dijagram projekta.



Slika 10.3: Blok dijagram projekta.

Električna šema: Slika 10.4 prikazuje električnu šemu projekta. Pinovi TX i RX UART 0 na Raspberry Pi Pico se koriste za komunikaciju s ESP-01.

Poglavlje 11 • Bluetooth sa Raspberry Pi Picom

11.1 Pregled

Bluetooth je jedan od najpopularnijih načina za bežičnu razmenu podataka na kratkim udaljenostima. Danas, mnogi elektronski uređaji poput pametnih telefona, laptopova, iPad-a, igračkica, gadžeta, prenosivih uređaja za praćenje zdravlja i tako dalje, svi su opremljeni Bluetooth modulima. Bluetooth koriste mnogi ljudi za deljenje slika i muzičkih datoteka pomoću pametnih telefona. Bluetooth je uparen protokol komunikacije gde oba uređaja moraju omogućiti svoje Bluetooth veze i koristiti isti ključ da bi se povezali jedan sa drugim. Kada je veza uspostavljena, podaci mogu da se šalju u oba smera. Ne morate se brinuti o fizičkoj vidljivosti između uređaja, jer je komunikacija zasnovana na radio-talasima, iako sa ograničenim dometom. Ponekad povezivanje između uređaja može da ne uspe. Trebalo bi obratiti pažnju na sledeće tačke za uspešno uparivanje između uređaja.

- Proverite da li je Bluetooth uključen na oba uređaja.
- Omogućite otkrivanje vašeg uređaja. Na nekim uređajima možda ćete morati pritisnuti dugme da biste omogućili Bluetooth otkrivanje.
- Proverite da li su dva uređaja blizu jedan drugom.
- Proverite da li su uređaji koji treba da budu upareni kompatibilni jedan sa drugim, npr. njihove verzije su kompatibilne.
- Unesite isti kod za uparivanje na oba uređaja kada od vas to bude zatraženo.

11.2 Bluetooth interfejs

Raspberry Pi Pico nema ugrađeni Bluetooth modul. Moramo koristiti eksterni Bluetooth modul da bismo omogućili Pico-u da komunicira sa drugim uređajima putem Bluetooth-a. Jedna od mogućnosti može biti korišćenje Raspberry Pi računara, ali je jeftinija opcija možda korišćenje serijskog Bluetooth modula, poput HC-06. U narednom odeljku ćemo razviti projekat i naučiti kako da povežemo jeftin Bluetooth modul tipa HC-06 sa našim Raspberry Pi Pico-m.

11.3 Projekat 1: Upravljanje LED diodom sa pametnog telefona korišćenjem Bluetooth-a

Opis: U ovom projektu ćemo slati komande preko Bluetooth veze sa pametnog telefona kako bismo kontrolisali LED diodu koja je povezana sa Raspberry Pi Picom (lako možete zameniti LED diodu sa zvučnikom kako biste električne uređaje mogli daljinski kontrolisati). U ovom projektu, važeće komande su:

- L1 Uključi LED
- L0 Isključi LED

Cilj: Cilj ovog projekta je da pokaže upotrebu jeftinog serijskog Bluetooth modula sa Raspberry Pi Picom.

HC-06 Bluetooth modul

HC-06 je popularan jeftin modul sa 4 pina koji se kontroliše serijski. Raspored pinova je prikazan na slici 11.1.

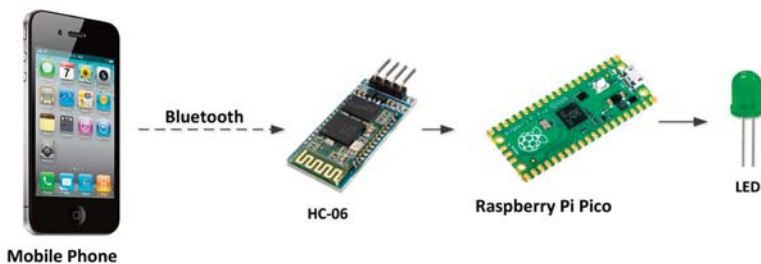


Slika 11.1: HC-06 Bluetooth modul.

HC-06 je modul koji se kontroliše serijski, sa nizom interesantnih specifikacija:

- Napajanje od +3.3 V do +6 V
- Potrošnja struje u neuparenom režimu: 30 mA (10 mA u uparenom režimu)
- Ugrađena antena
- Frekvencijski pojas: 2.40 GHz – 2.48 GHz
- Nivo snage: +6 dBm
- Podrazumevana komunikacija: 9600 boda, 8 bitova podataka, bez pariteta, 1 zaustavni bit
- Pokrivenost signala od oko 10 m (30 ft)
- Sigurnosne funkcije: autentikacija i enkripcija
- Režim modulacije: Gausova frekvencijska modulacija (Gaussian frequency-shift keying)

Blok dijagram: Slika 11.2 prikazuje blok dijagram projekta.



Slika 11.2: Blok dijagram projekta.

Električna šema: Slika 11.3 prikazuje dijagram kola projekta. Pinovi RXD i TXD Bluetooth modula su povezani sa UART 0 pinovima GP0 (TX) i GP1 (RX) Raspberry Pi Pica. LED je povezan sa GP16 (pin 21) preko otpornika za ograničenje struje od 470 oma.

Poglavlje 12 • Korišćenje digitalno-analognih pretvarača (DAC)

12.1 Pregled

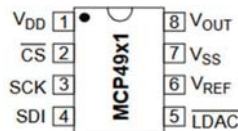
DAC pretvarači se koriste za pretvaranje digitalnih signala u analogni oblik. Takvi pretvarači imaju mnogo primena u digitalnoj obradi signala (DSP) i digitalnim kontrolnim aplikacijama. Na primer, možemo generisati talase pisanjem programa i zatim pretvarati ove talase u analogni oblik i generisati ih iz našeg digitalnog računara. Takođe nam je potreban DAC ako želimo da povežemo zvučnik ili neki drugi uređaj koji radi sa analognim naponima sa našim Raspberry Pi Pico.

Raspberry Pi Pico nema ugrađeni ADC pretvarač i stoga se mora koristiti spoljni DAC čip kako bi se generisali analogni signali. U ovom poglavlju ćemo naučiti kako koristiti popularni DAC čip (MCP4921) sa našim Raspberry Pi Pico kako bismo generisali neke jednostavne signalne talase.

12.2 MCP4921 DAC

Pre nego što počnemo da koristimo MCP4921, vredi detaljno pogledati njegove karakteristike i način rada. MCP4921 je 12-bitni DAC koji funkcioniše sa SPI interfejsom. Slika 12.1 prikazuje raspored pinova ovog čipa. Osnovne karakteristike uključuju:

- 12-bitnu operaciju
- Podršku za radni takt od 20 MHz
- Vreme smirivanja od 4.5 μ s
- Ulaz za eksterni referentni napon
- 1 \times ili 2 \times pojačanje, sa kontrolom
- Napajanje od 2.7 V do 5.5 V
- Temperaturni opseg od -40 °C do +125 °C



Slika 12.1: Microchip MCP4921 DAC.

Za više detalja, pogledajte informacije na:

<http://ww1.microchip.com/downloads/en/devicedoc/21897b.pdf>

Opisi pinova:

- Vdd: napon napajanja
- CS: izbor čipa (aktivno-nisko)
- SCK: SPI takt
- SDI: SPI ulazni podaci
- LDAC: koristi se za prenos podataka iz ulaznog registra na izlaz (aktivno-nisko)
- Vref: referentni ulazni napon
- Vout: analogni izlaz
- Vss: masa

U ovom projektu ćemo koristiti MCP4921 sa pojačanjem od 1 (tj. jediničnim pojačanjem). Kao rezultat toga, sa referentnim naponom od 3.3 V i konverzijom podataka od 12 bita, najmanja promena jednog bita (LSB) na DAC-u će biti $3300 \text{ mV} / 4096 = 0.8 \text{ mV}$.

12.3 Projekat 1: Generisanje pravougaonog signala sa amplitudom ispod +3.3 V

Opis: U ovom projektu ćemo koristiti DAC da generišemo pravougaoni signal sa frekvencijom od 500 Hz (period = 2 ms) i 50% radnog ciklusa (tj. vreme uključenosti = 1 ms, vreme isključenosti = 1 ms). Izlazni napon će biti 2 V_{peak} (primetićete da ovo ne bi moglo biti postignuto bez korišćenja DAC-a, pošto je maksimalni izlazni napon pina +3.3 V).

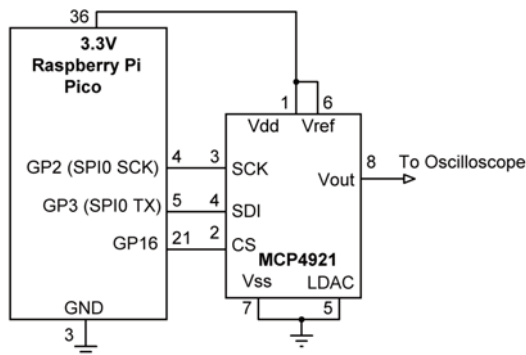
Cilj: Cilj ovog projekta je da pokažemo kako se DAC čip može povezati sa Raspberry Pi Pico.

Blok dijagram: Slika 12.2 prikazuje blok dijagram projekta.



Slika 12.2: Blok dijagram projekta.

Električna šema: Električna šema projekta prikazana je na slici 12.3. Pinovi GP3 (SPI0 TKS) i GP2 (SPI0 SCK) su povezani na MCP4921 pinove SDI i SCK, respektivno. CS MCP4921 se kontroliše odvojeno od GP16 (pin 21). Izlaz DAC-a je povezan sa osciloskopom.



Slika 12.3: Šema kola projekta.

Podaci se upisuju u DAC u 2 bajta. Niži bajt određuje D0:D8 digitalnih ulaznih podataka. Viši bajt sastoji se od sledećih bitova:

- D8:D11 bitovi D8:D11 digitalnih ulaznih podataka
- SHDN 1: aktivan (izlaz je dostupan), 0: isključuje uređaj
- GA kontrola pojačanja izlaza. 0: pojačanje je 2×, 1: pojačanje je 1×