DC / AC jaudas mērītājs // Atskaite

Kods, atskaite un vairāki pielikumi ir arī pieejami GitHub:  
<https://github.com/Gustavs-K/1.k-2.s-DC-AC-power-meter/>

Kods satur komentārus, kas paskaidros koda darbību daudz dziļākā līmenī.   
Visi komentāri ir angļu valodā.

# Īss aprakts

Projekta galvenais mērķis ir izveidot jaudas mērītāju, kas spēj nomērīt un aprēķināt līdzstrāvas un maiņstrāvas spriegumus, strāvas stiprumus un jaudas.

Projekts iekļauj ESP32, kas pilda ADC un informācijas apstrādes funkcijas un vairākas elektroniskās ierīces, kas pārveido ienākošo spriegumu un slodzes patērēto stāvas stiprumu uz MCU nolasāmu formu un veic citas funkcijas.

# Galvenās funkcijas

Mikrokontrolleris – ESP32\_Devtikc\_V4 – ADC , datu apstrāde, datu izvade caur “serial monitor”,

2x Sprieguma dalītājs – samazina nolasāmo spriegumu, lai iekļautos ESP32 3.3V nolasāmajā diapazonā,

2x Pilna viļņa taisngriezis – pārveido negatīvo maiņstrāvas spriegumu uz pozitīvu un ar ESP32 nolasāmu,

Šunta rezistors – Strāvas stipruma mērīšanai

LED (sarkans) – pārsprieguma indikators,

Poga – DC / AC režīma pārslēgšanai.

# Saturs

[Īss aprakts 1](#_Toc168562252)

[Galvenās funkcijas 1](#_Toc168562253)

[Saturs 2](#_Toc168562254)

[Elektronika 3](#_Toc168562255)

[Voltmetri 3](#_Toc168562256)

[Sprieguma dalītājs 4](#_Toc168562257)

[Šuntes rezistors 4](#_Toc168562258)

[Pilna viļņa taisngriezis 5](#_Toc168562259)

[Poga un LED 5](#_Toc168562260)

[ESP32 5](#_Toc168562261)

[Cenas aprēķins: 6](#_Toc168562262)

[Koda apraksts 6](#_Toc168562263)

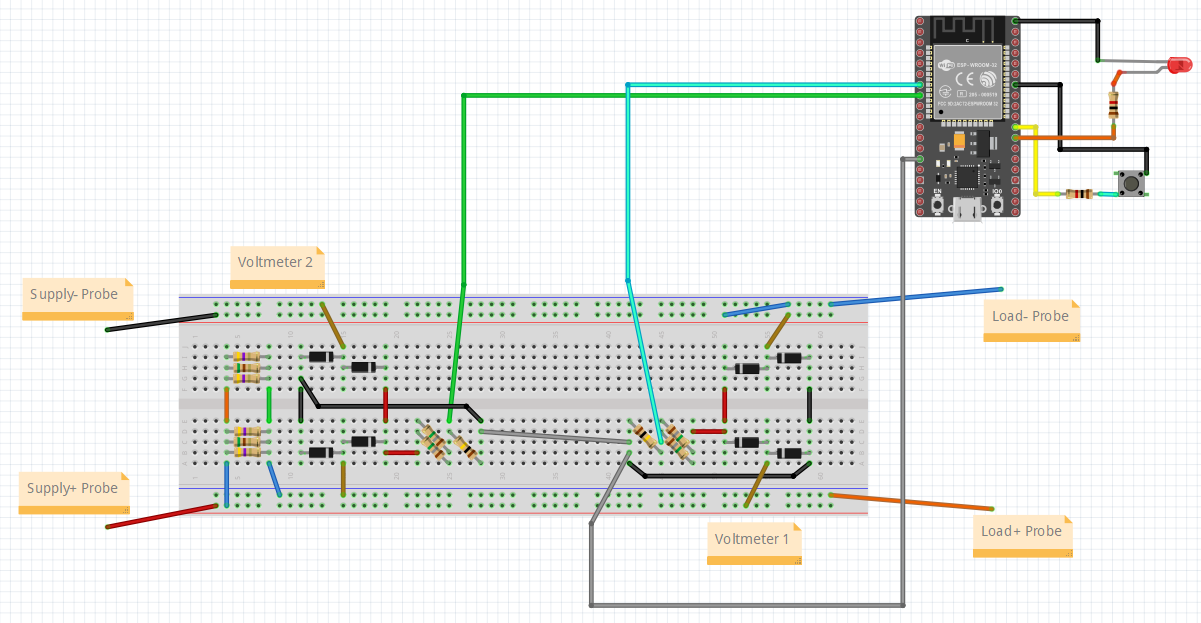
[varResetter() funkcija 6](#_Toc168562264)

[inputReader() funkcija 6](#_Toc168562265)

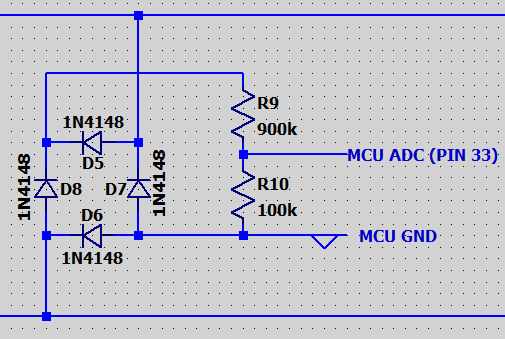
[dataProcessor() funkcija 7](#_Toc168562266)

[outputPrinter() funkcija 7](#_Toc168562267)

# Elektronika



## Voltmetri

Voltmetri sastāv no pilna viļņa taisngrieža, sprieguma dalītāja un ESP32 analogās ieejas. V1 mēra kopējo ķēdes spriegumu un V2 mēra spriegumu pēc šuntes, lai varētu aprēķināt tās sprieguma kritumu. 

Oriģināli bija iedomāta sistēma ar vairākiem režīmiem, kurā katrs režīms pielietotu savu detaļu kombināciju. Šī ideja tika atmesta, lai vienkāršotu jaudas metra pielietošanu un izveidi, bet tādējādi arī zaudējot mazliet precizitāti un ieviešot apgrūtinājumus datu apstrādē.

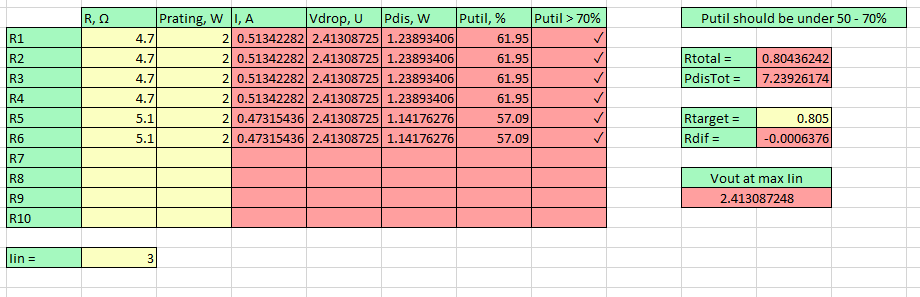
## Sprieguma dalītājs

Spieguma dalītājs sastāv no diviem 1.8M Ω rezistoriem (kopā 900k Ω) un viena 100k Ω rezistora. Šīs vērtības kopā izveido izejas pret ieejas sprieguma attiecību 1/10, kas palielina ESP32 nolasāmo diapazonu no 3.3V uz 33V.

Lielās rezistoru vērtības tika izvēlētais, lai samazinātu voltmetru ietekmi uz mērāmās ķēdes darbību.

## Šuntes rezistors

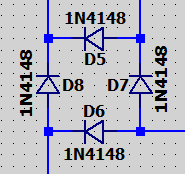
Šuntes rezistors sastāv no sešiem, paralēli novietotiem, rezistoriem, kuru vērtības ir 4x 4.7 Ω un 2x 5.1 Ω. Oriģināli 0.805 Ω šunte bija iedomāta, bet, pieejamo rezistoru vērtību dēļ, 0.8044 Ω bija tuvākā iegūstamā kopējā pretestība, kas spētu izdalīt pietiekami daudz jaudas un izmantotu pēc iespējas mazāku rezistoru daudzumu. 0.805 Ω tika izvēlēts, lai pie maksimālā nomērāmā spieguma krituma (3.3V) strāvas stiprums būtu apmēram 4 ampēri.   
Meklējot rezistoru kombināciju, maksimālais plānotais strāvas stiprums tika samazināts līdz 3 ampēriem.



Izveidojot voltmetra maketu, maksimālais strāvas stiprums samazinājās līdz 1 ampēram, maketēšanas plates un vadu savienojumu dēļ.

## Pilna viļņa taisngriezis

Pilna viļņa taisngriezis ir izveidots no 1N4148 diodēm. Šī detaļa tiek izmantota, lai pārveidota maiņstrāvas negatīvās vērtības uz pozitīvām, kuras ESP32 var nolasīt.

Mērot līdzstrāvu šis taisngriezis darbojas kā pret polaritātes aizsardzība.

Šis taisngriezis nesatur filtra kondensatoru, jo tas apgrūtinātu datu apstrādi un mērījumu veikšanai tas nav nepieciešams.

Viens no taisngrieža lielākajiem mīnusiem ir diožu sprieguma kritums, kas iespaido gan līdzstrāvas, gan maiņstrāvas mērījumu precizitāti.

## Poga un LED

Pogas galvenā funkcija ir DC / AC režīma pārslēgšana. Šis režīms maina apstrādes laikā veiktās funkcijas un izvadītos datu.

LED indikators norāda uz pārspriedumu uz ESP32 analogajām ieejām. Šī funkcija tiek veikta ar koda palīdzību. Ekstrēmos gadījumos indikators varētu nestrādāt kā iecerēts.

## ESP32

ESP32 ADC ir ļoti ierobežots. Kaut arī tas ir 12 bitu ADC, tā mērījumi var būt ievērojami atšķirties, sevišķi pie maksimālajām un minimālajām vērtībām, kur mērījumi sāk ievērojami diverģēt no īstā sprieguma.

ESP32 datu apstrādes spējas ir ļoti labas, divu kodolu un 80 – 240 MHz takts frekvences dēļ. Vidējais koda cikla laiks ir ≈80 ms, no kura ≈60 ms ir analogo datu nolasīšana un ≈10µs datu apstrāde.

## Cenas aprēķins:

ESP32-DEVKITC-S1 – 17.13 EUR

1N4148 diode – 0.04 EUR x8

Rezistori :

4.7 Ω, 2W – 0.09 EUR x4

5.1 Ω, 2W – 0.20 EUR x2

1k Ω – 0.09 EUR x2

100k Ω – 0.09 EUR x2

1.8M Ω – 0.01 EUR x4

LED (sarkans) – 0.06 EUR

Maketēšanas plate – 4.83 EUR x2

Kopējā cena – ≈28.33 EUR

# Koda apraksts

Galvenais cikls sastāv ko 4 galvenajām funkcijām, starp kurām tiek nomērīti šo funkciju darbības laiki.

## varResetter() funkcija

Šī funkcija pārmaina visu aprēķināto mainīgo vērtības uz 0, lai nodrošinātu, ka iepriekšējie cikli neietekmē šī cikla aprēķinus.

## inputReader() funkcija

inputReader() galvenokārt nolasa un saglabā V1 un V2 analogās vērtības, pārbauda un veic pogas funkcijas un pārbauda un ieslēdz pārsprieguma LED.

## dataProcessor() funkcija

Šī funkcijas apstrādā nolasītos datus, lai iegūt vērtības, kuras nav iespējams tieši nolasīt. Dati, kurus šī funkcija aprēķina, ir atkarīgi no režīma. DC režīmā tiek aprēķināts vidējais strāvas stiprums un jauda. AC režīmā tiek aprēķinātas RMS vērtības priekš V1, V2, I, īstā jauda, šķietamā jauda, reaktīvā jauda un jaudas faktors.

## outputPrinter() funkcija

outputPrinter() izvada visas izvadāmo mainīgo vērtības serial monitora. Līdzīgi dataProcessor(), šī funkcijas izvadītā informācija ir atkarīga no darbības režīma.