|  |  |  |
| --- | --- | --- |
| UNIZNAK | УНИВЕРЗИТЕТ У НОВОМ САДУ  ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА | FTNZNAK |

**Пројектовање GUI за клавијатуру реализовану помоћу MCU**

дипломска теза (бечелор)

кандидат

Лазар Минчић, EE20/2017

ментор

проф. др Иван Мезеи

мај 2024

|  |  |
| --- | --- |
|  | УНИВЕРЗИТЕТ У НОВОМ САДУ ● **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**  21000 НОВИ САД, Трг Доситеја Обрадовића 6 |
| **КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА** |

Редни број, **РБР**:

Идентификациони број, **ИБР**:

Тип документације, **ТД**:

Тип записа, **ТЗ**: Врста рада, **ВР**: Аутор, **АУ**:

Ментор, **МН**: Наслов рада, **НР**:

Монографска публикација

Текстуални штампани материјал Дипломски рад

**Лазар Минчић**

**Проф. др Иван Мезеи**

**Пројектовање GUI за клавијатуру реализовану помоћу MCU**

Језик публикације, **ЈП**: Српски / ћирилица

Језик извода, **ЈИ**: Српски

Земља публиковања, **ЗП**: Србија

Уже географско подручје, **УГП**: Војводина

Година, **ГО**: **2024**

Издавач, **ИЗ**: Ауторски репринт

|  |  |
| --- | --- |
| Место и адреса, **МА**: | Факултет техничких наука (ФТН), Д. Обрадовића 6, 21000 Нови Сад |
| Физички опис рада, **ФО**:  (поглавља/страна/ цитата/табела/слика/графика/прилога) | **?/?/?/?/?/?/?** |
| Научна област, **НО**: | Електротехника и рачунарство |

Научна дисциплина, **НД**:

Ембедед системи и алгоритми

|  |  |
| --- | --- |
| Предметна одредница/Кључне речи, **ПО**: | **Микроконтролер, Графички интерфејс, Серијска комуникација, QT** |
| **УДК** |  |

|  |  |
| --- | --- |
| Чува се , **ЧУ**: | Библиотека ФТН, Д. Обрадовића 6, 21000 Нови Сад |
| Важна напомена, **ВН**: |  |
| Извод, **ИЗ**: | **У овом раду описана је једна имплементација клавијатуре са лименкама уместо дирки. Користи се микроконтролер ESP32 који комуницира са рачунаром преко серијског порта. Графички интерфејс писан је у QT окружењу. Детаљно је описан начин рада крајањег уређаја, као и начин на који хардвер и софтвер комуницирају.** |
| Датум прихватања теме, **ДП**: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Датум одбране, **ДО**: |  |  |  |
| Чланови комисије, **КО**: | Председник: | **др име презиме, звање** |  |
|  | Члан: | **др име презиме, звање** | Потпис ментора |
|  | Члан, ментор: | **Проф. др Иван Мезеи** |  |

|  |  |
| --- | --- |
|  | UNIVERSITY OF NOVI SAD ● **FACULTY OF TECHNICAL SCIENCES**  21000 NOVI SAD, Trg Dositeja Obradovića 6 |
| **KEY WORDS DOCUMENTATION** |

Accession number, **ANO**: Identification number, **INO**: Document type, **DT**:

Type of record, **TR**: Contents code, **CC**: Author, **AU**: Mentor, **MN**:

Title, **TI**:

Monographic publication

Textual printed material

Bachelor thesis

**Lazar Minčić**

**Prof. dr Ivan Mezei**

**Designing a GUI for a Keyboard Implemented With an MCU**

Language of text, **LT**: Language of abstract, **LA**: Country of publication, **CP**: Locality of publication, **LP**: Publication year, **PY**:

Publisher, **PB**:

Serbian

Serbian

Serbia Vojvodina

**2024**

Author’s reprint

|  |  |
| --- | --- |
| Publication place, **PP**: | Faculty of Technical Sciences, D. Obradovića 6, 21000 Novi Sad |
| Physical description, **PD**:  (chapters/pages/ref./tables/pictures/graphs/appendixes) | **?/?/?/?/?/?/?** |
| Scientific field, **SF**: | Electrical and computer engineering |

Scientific discipline, **SD**:

Embedded systems

|  |  |
| --- | --- |
| Subject/Key words, **S**/**KW**: | **Microcontroller, Graphical user interface, Serial communication, QT** |
| **UC** |  |

|  |  |
| --- | --- |
| Holding data, **HD**: | Library of Faculty of Technical Sciences, D. Obradovića 6, 21000 Novi Sad |
| Note, **N**: |  |
| Abstract, **AB**: | **This paper describes an implementation of a keyboard using cans instead of keys. An ESP32 microcontroller is used, which communicates with a computer via serial port. The graphical user interface is written in QT. The functionality of the end device is described in detail, as well as the way the hardware and software communicate.** |
| Accepted by the Scientific Board on, **ASB**: |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Defended on, **DE**: | |  |  | |  | |
| Defended Board, **DB**: | | President: | **dr ime prezime, zvanje na engleskom** | |  | |
|  | | Member: | **dr ime prezime, zvanje na engleskom** | | Menthor's sign | |
|  | | Member, Mentor: | **Ivan Mezei, Ph.D** | |  | |
|  | | УНИВЕРЗИТЕТ У НОВОМ САДУ ● **ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА**  21000 НОВИ САД, Трг Доситеја Обрадовића 6 | | | Датум: | |
|  | |
| **ЗАДАТАК ЗА ИЗРАДУ ДИПЛОМСКОГ (BACHELOR) РАДА** | | | Лист/Листова: | |
| 1/18 | |

*(Податке уноси предметни наставник - ментор)*

|  |  |
| --- | --- |
| Врста студија: | ✔ Основне академске студије  Основне струковне студије |
| Студијски програм: | **Енергетика, електроника и телекомуникације** |
| Руководилац студијског програма: | **др Милан Сечујски** |

|  |  |  |  |
| --- | --- | --- | --- |
| Студент: | Лазар Минчић | Број индекса: | **ЕЕ20/2017** |
| Област: | Рачунарска електроника | | |
| Ментор: | **др Иван Мезеи** | | |
| НА ОСНОВУ ПОДНЕТЕ ПРИЈАВЕ, ПРИЛОЖЕНЕ ДОКУМЕНТАЦИЈЕ И ОДРЕДБИ СТАТУТА ФАКУЛТЕТА ИЗДАЈЕ СЕ ЗАДАТАК ЗА ДИПЛОМСКИ (Bachelor) РАД, СА СЛЕДЕЋИМ ЕЛЕМЕНТИМА:   * проблем – тема рада; * начин решавања проблема и начин практичне провере резултата рада, ако је таква провера неопходна; * литература | | | |

**НАСЛОВ ДИПЛОМСКОГ (BACHELOR) РАДА:**

**Пројектовање GUI за клавијатуру реализовану помоћу MCU**

**ТЕКСТ ЗАДАТКА:**

1. Увод о коришћеним технологијама

2. Концепт решења

3. Реализација рада, хардвер и програмска подршка

4. Тестирање крајањег уређаја и анализа резултата

5. Дискусија о могућим будућим унапређењима

|  |  |
| --- | --- |
| Руководилац студијског програма: | Ментор рада: |
| др Милан Сечујски | др Иван Мезеи |

Примерак за: - Студента; - Ментора

**Изјава о академској честитости**

Студент: **Лазар Минчић**

Број индекса: **EE20-2017**

Студент **основних академских студија**

Аутор рада под називом: **Пројектовање GUI за клавијатуру реализовану помоћу MCU**

Потписивањем изјављујем:

* да је рад искључиво резултат мог сопственог истраживачког рада;
* да сам рад и мишљења других аутора које сам користио у овом раду назначио или цитирао и наведени у списку литературе/референци који су саставни део овог рада;
* да сам добио све дозволе за коришћење ауторског дела који се у потпуности/целости уносе у предати рад и да сам то јасно навео;
* да сам свестан да је плагијат коришћење туђих радова у било ком облику (као цитата, парафраза, слика, табела, дијаграма, дизајна, планова, фотографија, филма, музике, формула, веб сајтова, компјутерских програма и сл.) без навођења аутора или представљање туђих ауторских дела као мојих, кажњиво по закону (Закон о ауторском и сродним правима, Службени гласник Републике Србије, бр. 104/2009, 99/2011, 119/2012), као и других закона и одговарајућих аката Универзитета у Новом Саду;
* да сам свестан да плагијат укључује и представљање, употребу и дистрибуирање рада предавача или других студената као сопствених;
* да сам свестан последица које код доказаног плагијата могу проузроковати на предати рад и мој статус;
* да је електронска верзија рада идентична штампаном примерку и пристајем на његово објављивање под условима прописаним актима Универзитета.

Нови Сад, 27.9.2021. Потпис студента

**САДРЖАЈ**

[1. Увод 1](#_Toc170162937)

[2. Теоријске основе 3](#_Toc170162938)

[3. Опис проблема и концепт решења 6](#_Toc170162939)

[Додатак А 7](#_Toc170162940)

[Литература 8](#_Toc170162941)

**СПИСАК СКРАЋЕНИЦА**

**СПИСАК СЛИКА**

[Слика 2.1 Функционални блок дијаграм ESP32 SoC 4](#_Toc170162016)

[Слика 2.2 Пинаут дијаграм развојне плоче ESP32 DEVKIT V1 5](#_Toc170162017)

### СПИСАК ТАБЕЛА

**No table of figures entries found.**

**ЗАХВАЛНИЦА**

Захваљујем се ментору професору др Ивану Мезеиу, као и породици и пријатељима на пруженој подршци приликом израде овога Рада.

# УВОД

Прва генерација микрокотролера на тржишту се појављује седамдесетих година прошлог века, интегрисајући CPU (енг. *central processing unit*, централна процесорска јединица), меморијски чип и периферије заједно на једном чипу. Међутим, како је број транзистора који је могуће сместити на исту површину чипа растао, тако су се ови уређаји развијали. С тога данашњи микроконтролери имају значајно бржу CPU јединицу, више опција повезивања (укључујући и бежичне опције као што су Bluetooth и Wi-Fi), а истовремено мању потрошњу у односу на своје претходнике.

У овоме раду биће описани хардверски и софтверски концепти који се могу искористити при изради клавијатуре са лименкама уместо дирки. Постоји доста начина за реализацију овакве машине, на пример, коришћењем Raspberry Pi рачунара са одговарајућим пропратним хардвером, HAT-ом (енг. *hardware attached on top*, хардвер причвршћен од горе) у комбинацији са Raspbian оперативним системом и QT платформом за Linux оперативне системе. Међутим, у овоме раду описује се реализација поменуте машине корићшењем једног од многих систем на чипу микроконтролера из фамилије „ESP32“, повезаног на лични рачунар (енг. *personal computer*) преко серијског USB (енг. *universal serial bus*) порта. У том случају, софтверски део могуће је пројектовати на QT платформи за Windows оперативне системе, са обзиром на то да је Windows најпопуларнији оперативни систем за личне рачунаре.

Једна од мана изабраног решења је повећана комплексност програма приликом комуникације MCU и PC, као и немогућност бежичне комуникације, о чему ће више бити речи у поглављу \_\_.

Микроконтролери (енг. *microcontroler unit*, MCU) су самоодрживи системи на чипу, погодни за ембедед имплементације. Као „рачунари у малом“, масовно се производе и користе у разним секторима индустрије, те је потражња за њима веома висока. Компоненте на чипу од којих су MCU састављени такође су релативно слабе рачунарске снаге и перформанси, што даље доприноси њиховој ниској цени. У време писања овог текста, микроконтролер марке Silicon Labs могуће је купити за невероватних 0,89 €, што је отприлике 100 динара [1]. Захваљујући својој ниској цени и изобиљу, MCU постају приступачни масама и тиме дају могућност за иновације у различитим областима употребе. Ова чињеница за последицу има и то да свако ко чита овај текст веома лако може да рекреира уређај за који је писан GUI (енг. *graphical user interface*), што је детаљно наведено у поглављу \_\_.

Мотиви за склапање овакве машине као и за даљи развој њене пропратне софверске апликације су бројни, с обзиром на то да се апликација може написати на разне начине, у зависности од потреба крајањег корисника. Уређај може бити коришћен како као играчка, тако у едукационе и практичне сврхе, нпр. при компоновању или у образовним установама које се баве музиком, као пример примене микроконтролера, серијске комуникације, писања у QT програмском окружењу...

У наставку ће бити објашњени детаљи прво хардверске, а потом и софтверске архитектуре овога пројекта, неке од потенцијалних примена овога уређаја, као и предлози за даљи развој софтверске апликације.

НАВЕДИ ПОГЛАВЉА НА КРАЈУ

# ТЕОРИЈСКЕ ОСНОВЕ

* 1. Софтверски алати
     1. QT

QT је вишеплатформски апликациони интефејс коришћен за развој GUI и апликација које могу бити покренуте на разним софтверским и хардверским платформама као што су Linux, Windows, macOS, Android, ембедед системи, и други [][]. QT сам по себи није програмски језик, већ програмско окружење написано у језику C++. MOC (*meta-object compiler*) препоцесор проширује C++ језик са функцијама јединственим QT-ју, као што су нпр. сигнали и слотови. QT долази са својим интегрисаним окружењем за развој (енг. Integrated Development Environment, IDE) под именом QT Creator. Овај IDE садржи алате корисне при развијању апликације и уклањању грешака. QT Creator је доступан у четири издања, Community, Indie Mobile, Professional и Enterprise. За потребе овог рада коришћено је прво, Community издање, с обзиром на то да је оно једино са лиценцама отвореног кода, док је за остала издања неопходна куповина одговарајуће лиценце пре коришћења. Многобројне QT библиотеке и класе развијене за рад са периферним уређајима, мултимедијама, сликама и звуком, један су од разлога зашто је баш ова платформа изабрана за израду овог пројекта.

* + 1. Arduino IDE

Ardino IDE (eng. *Integrated Development Environment*) је уређивач текста посебно намењен за Arduino развојне плоче. Омогућава отпремање написаног програма на претходно изабрану плочу. Подржане су и плоче из ESP32 фамилије, у колико се у подешавањима инсталира одговарајући додатак. Могућ је увид у податке које плоча шаље рачунару.

* 1. Микроконтролер ESP32

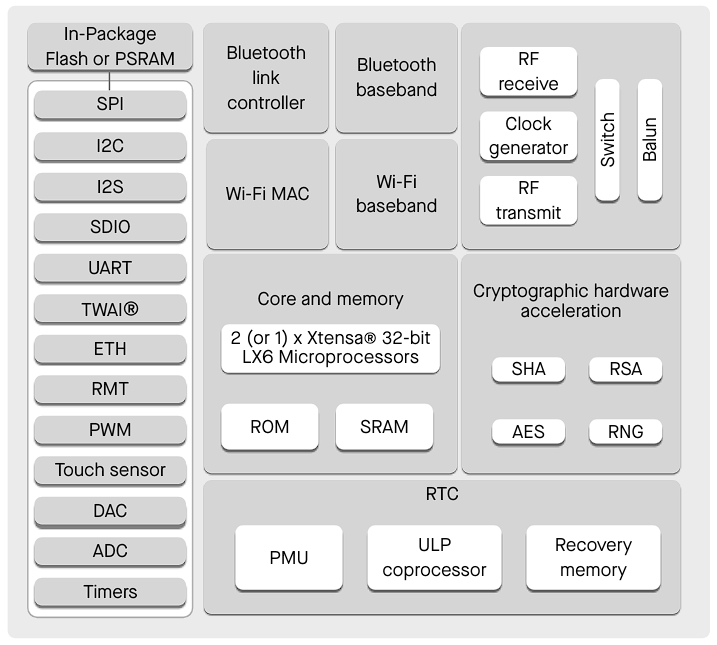
Кључна компонента коришћена у изради решења је микроконтролер кинеског произвођача Espressif Systems, имена ESP32. Нека од побољшања овог MCU у односу на своје претходнике су:

* робустан дизајн, са оперативном температуром од-40°C to +125°C
* веома ниска потрошња енергије,
* високи ниво интеграције, и
* хибридни Wi-Fi & Bluetooth микрочип[].

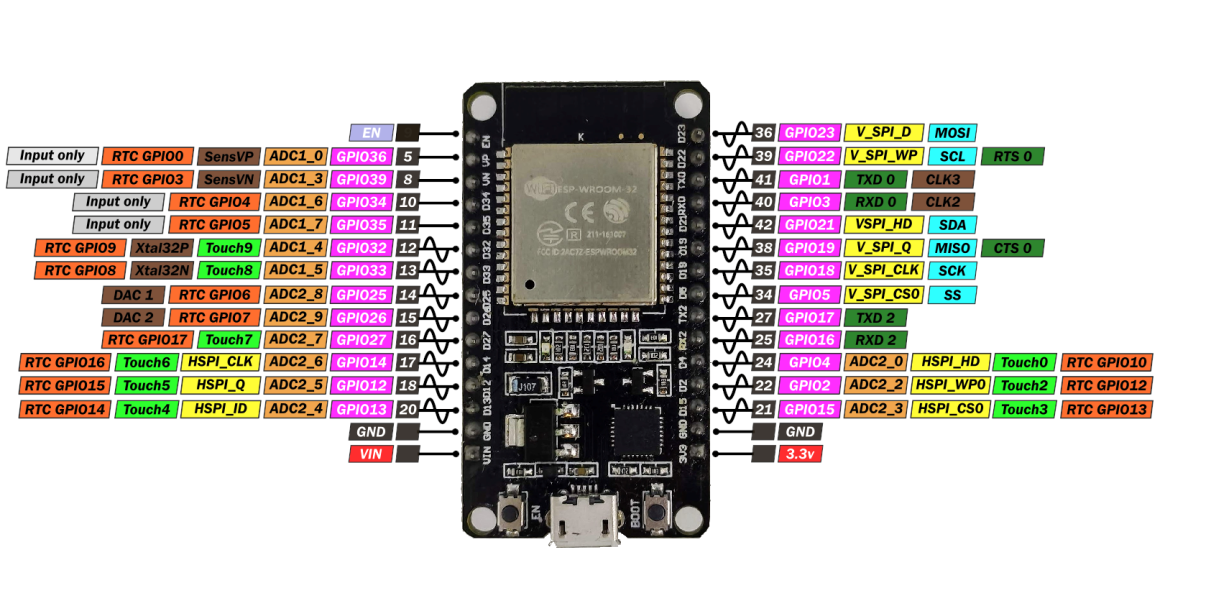
За потребе пројекта коришћена је једна од ESP32 развојних плоча, конкретно ESP32 DEVKIT V1 добављача DoIT, због лакшег повезивања са рачунаром и осталим коришћеним компонентама (потенциометар, led диоде). Особине од интереса овог MCU су:

* Xtensa® двојезгарни 32-битни LX6 микропроцесор
* 448 KB ROM, 520 KB SRAM меморије
* 34 програмабилних GPIO (*general-purpose input/output*)
* 12-битни SAR ADC (*successive-approximation-register*) (*analog-to-digital converter*)
* два 8-битна DAC (*digital to analogue converter*)
* 10 сензора на додир
* три UART интерфејса (*universal asynchronous receiver-transmitter*)
* LED PWM до 16 канала (*light-emitting diode*)( *Pulse Width Modulation*)[].

Конкретан модул у употреби је ESP-WROOM-32, у чијој се основи налази ESP32-D0WDQ6 микрочип. Функционални блок дијаграм ESP32 серије SoC може се видети на слици [], док је пинаут развојне плоче у употреби (ESP32 DEVKIT V1) приказан на слици [].



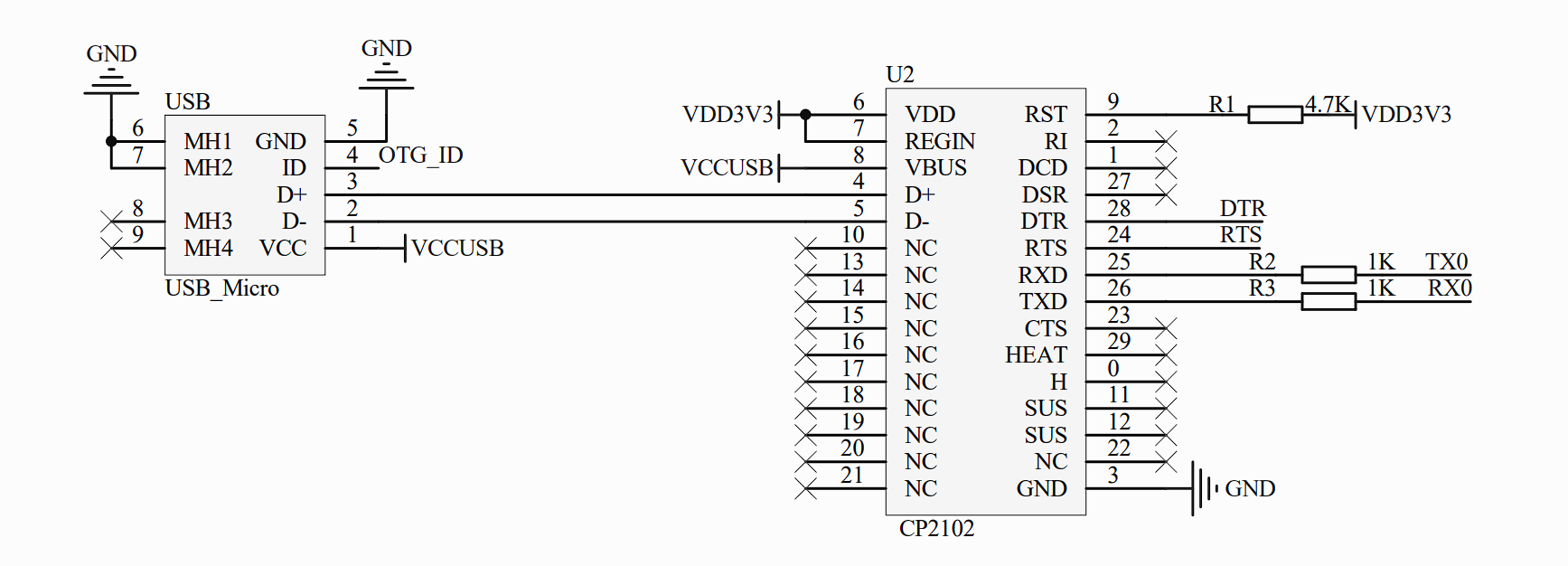
Слика 2.1 Функционални блок дијаграм ESP32 SoC []



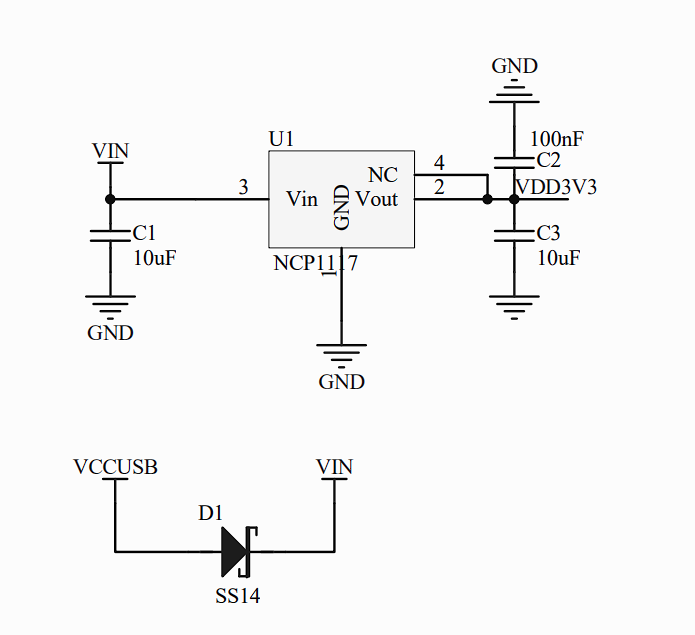
Слика 2.2 Пинаут дијаграм развојне плоче ESP32 DEVKIT V1 []

* + 1. UART серијски модул

Универзални асинхрони пријемник / одашиљач или UART (universal asynchronous receiver-transmitter) је начин комуникације између два хардверска уређаја (на пример, два микроконтролера, микроконтролера и рачунара). Комуникација се одвија преко стандардних асинхроних серијских интерфејса []. ESP32 чип поседује три UART порта, сваки са пропратним регистрима за подешавање брзине преноса података, дужине податка изражене у битима, броја стоп битова, бита парности итд. Од ова три UART порта на располагању, на развојној плочи у употреби у овом пројекту (DEVKIT V1) један од њих (конкретно UART0) повезан је са USB-UART мостом, што олакшава комуникацију микроконтролера са рачунаром као и отпремање програма и одткањање грешака. Конктретно се користи CP2102 модул моста, а детаљна шема приказана је на слици 2.3. USB порт рачунара снабдева развојну плочу одређеним напоном Vccusb, који углавном износи око 5V. USB кабел такође спаја масе два уређаја, што је битно да би напонски нивои података који се шаљу могли бити исправно очитани на прималачкој страни. Регулатори напона унутар CP2102 као и на развојној плочи гарантују излазни напон од 3.3V са толерантим маргинама грешке, тако да се ESP32 чип и периферије могу безбедно напајати (види слику 2.4).



Слика 2.3 Шематик USB дела развојне плоче

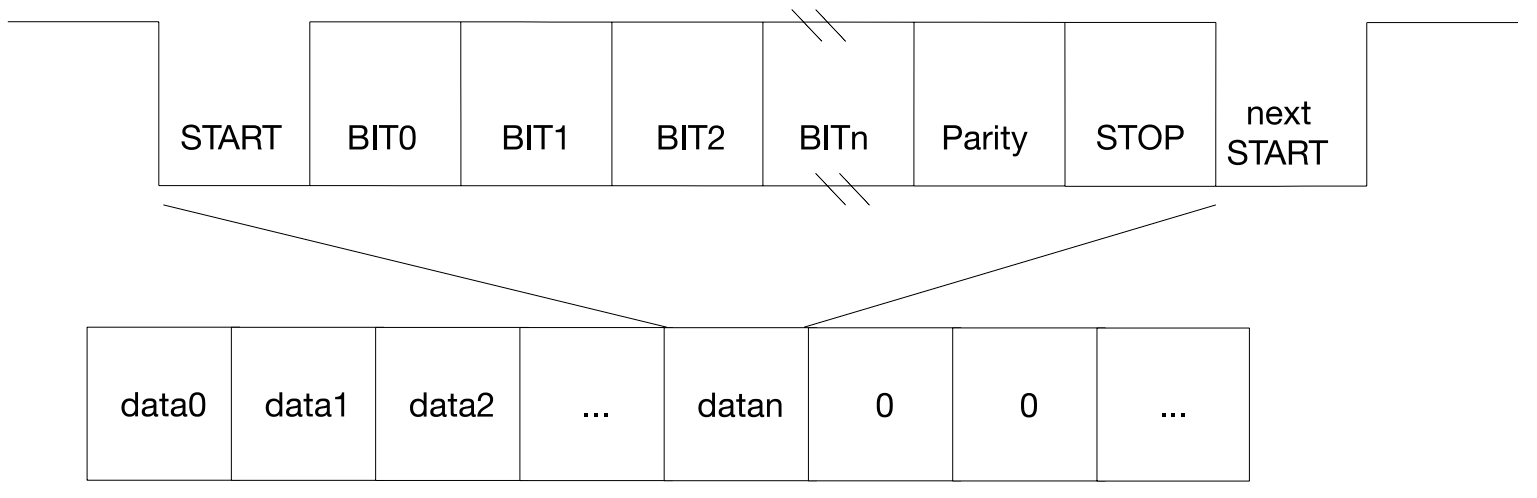


Слика 2.4 Шематик дела за напајање развојне плоче

UART модул унутар ESP32 нуди full duplex хардверске комуникацијске могућности, што значи да се подаци могу и слати и примати у исто време. Неке од особина UART портова на ESP32 су:

* програмабилна брзина преноса
* 1024 x 8-битна RAM меморија дељена између сва три порта UART0, UART1 и UART2 тј. њихових одговарајућих FIFO (*first in first out*) пријемних /одашиљчких регистара
* подржава слање 5/6/7/8 бита
* 1/1.5/2 стоп битова
* подршка за бит парности
* подршка хардверске и софтверске контроле протока

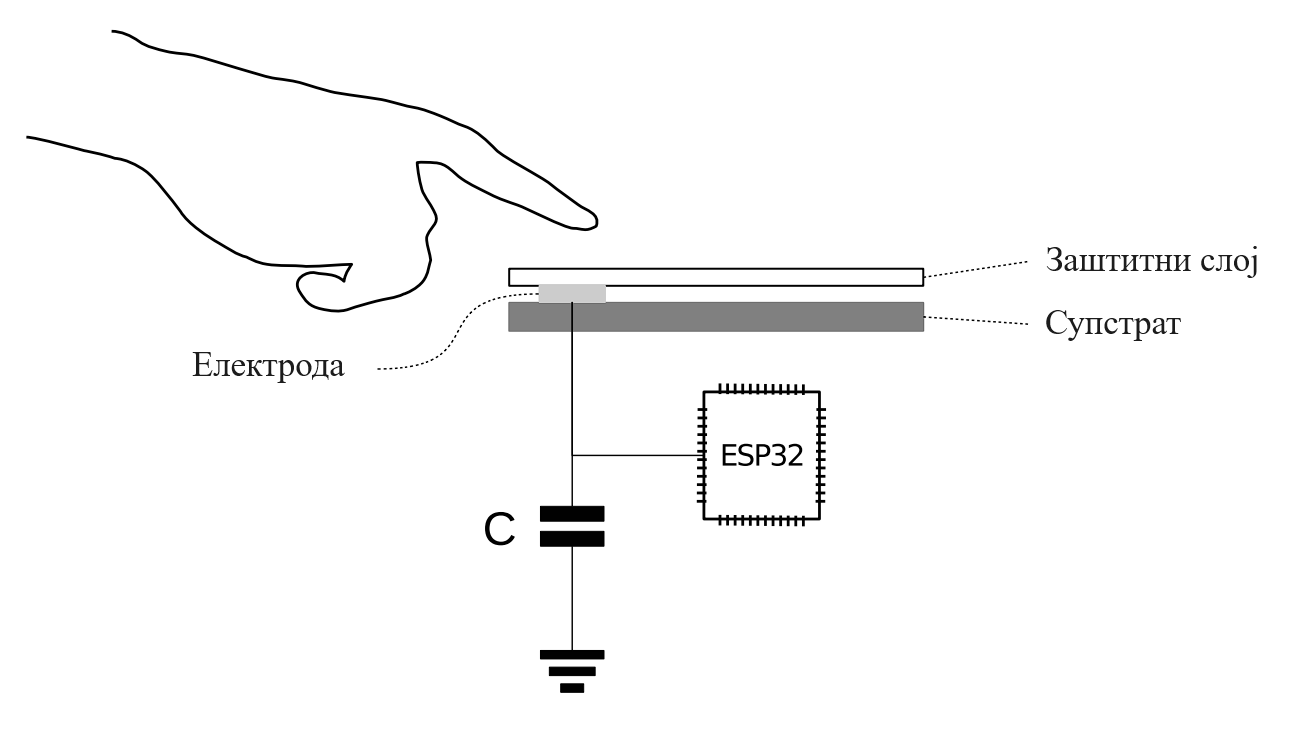
Подаци преко UART интерфејса шаљу се асинхроно, дакле без clock сигнала, што значи једна конекција / жица мање, али има за последицу неопходне начине синхронизације између два уређаја (пакет података постаје дужи због старт и стоп битова, битова парности итд). У случају full duplex начина комуникације довољне су две конекције за трансфер података, и додатна жица за масу, што смањује трошкове у поређењу са неким другим видовима комуникације (нпр, паралелна комуникација захтева конекцију за сваки бит који се шаље). Пример пакета података на UART интерфејсу може се видети на слици 2.5. Бит парности је опцион, и користи се за отклањање грешака које се могу појавити при преносу. Старт бит даје до знања пријемнику да пристижу подаци са предајничке стране који треба да буду прочитани, док стоп бит означава границу, крај пакета податка. С обзиром на то да се clock сигнали не преносе, потребно је да и пријемник и предајник раде на истој фреквенцији да не би дошло до грешака у читању и губљења података.



Слика 2.5 Грађа UART пакета података

* + 1. Капацитивни сензор додира

Сваки од сензора додира на ESP32 направљени су на супстрату са електродом између заштитног слоја (види слику 2.6). Притисак се детектује мерењем промене капацитивности пре и након притиска сензора.



Слика 2.6 Сензор додира

Неке од важнијих карактеристика сензора додира су:

* до 10 сензора на додир
* комбиновањем ових сензора могуће је детектовати додир веће површине или више тачака
* контролише га FSM (*finite state machine*)
* може да ради у режиму чувања енергије.

# ОПИС ПРОБЛЕМА И КОНЦЕПТ РЕШЕЊА

**ДОДАТАК А**Листинг кода (врв. неће требати)

# ЛИТЕРАТУРА