



ФЕА

Катедра КСТ

Дипломна Работа на тема:

“Проектиране на микроконтролерен модул с цел използване на AVR/PIC асемблер”

Дипломант:

Веселин Станчев

ФН: 614872

спец. КСТ

Научен Ръководител:

доц. д-р инж. Мария Маринова

Цел на дипломната работа

- Да бъде проектиран микроконтролерен модул с цел използване на AVR-базиран микроконтролер и PIC-базиран микроконтролер. Трябва да бъде осъществена комуникация между двата микроконтролера на асемблерно ниво. Модулът ще може да бъде използван за учебни и университетски цели. Проектът е изцяло open-source с цел по-доброто бъдещо развитие.

Поставени задачи

- Обзор на съществуващи микроконтролерни модули с AVR/PIC базирани микроконтролери
- Изисквания към проекта
- Анализ на микроконтролерите които ще бъдат използвани за микроконтролерния модул
- Анализ на асемблерните инструкции които ще бъдат използвани
- Софтуерни инструменти за проектиране на микроконтролерния модул и програмиране
- Проектиране на микроконтролерния модул
- Програмиране на микроконтролерите
- Реализиране на кода като библиотека
- Постигнати резултати. Бъдещо развитие

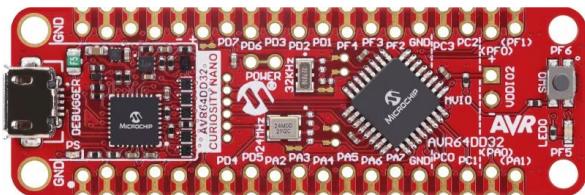
1.Обзор на съществуващи микроконтролерни модули



ANAVI MACRO PAD 2 - базиран на ATtiny 85 – AVR RISC архитектура



ANAVI LIGHT CONTROLLER - базиран на ESP32 – RISC архитектура



AVR64DD32 Curiosity Nano - базиран на AVR64DD32 – PIC базиран



Raspberry Pi Pico - базирана на RP2040 – RISC архитектура

Характеризирани са 4 микроконтролерни модули. Направена е характеристика на предметната област както и класификация на микроконтролерите.

За всеки микроконтролер е направено изследване, което обхваща:

- обща характеристика на микроконтролерният модул
- микроконтролерите, които поддържа
- комуникационните протоколи, които поддържа
- интерфейсите, които поддържа
- асемблери, на които може да бъде програмирани микроконтролерите

2. Изисквания към проекта

Поставени са изисквания към софтуерната и хардуерна част на проекта.

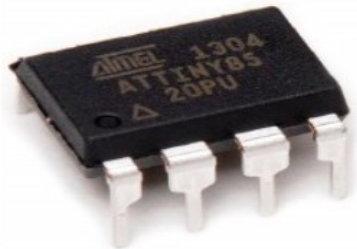
За хардуерната част трябва:

- Да се изберат подходящи микроконтролери чрез които да бъде изграден микроконтролерният модул
- Да се изберат интерфейси, с които ще разполага модула
- Да се избере софтуер за проектирането на модула

За софтуерната част трябва:

- да бъдат избрани асемблери съвместими с архитектурата на микроконтролерите
- да бъде реализиран протокол за комуникация между микроконтролерите
- трябва да бъде избран протокол за комуникация, спрямо който да бъде написан кодът
- кодът за микроконтролерите трябва да бъде реализиран чрез функции - labels

3. Анализ на микроконтролерите, които ще бъдат използвани за микроконтролерния модул



AtTiny-85

ATTiny85 разполага със следните възможности:

- 8Kb Flash памет
- 512 байта EEPROM памет
- комуникация през универсален сериен интерфейс (USI)
- възможност за I2C комуникация чрез пиновете PB0,PB2



PIC10F320

PIC10F320 разполага със следните възможности:

- 128 байта памет
- 8 битов таймер
- ADC-конвертер
- комуникация през вътрешен интерфейс (ICSP)

4. Анализ на асемблерните инструкции, които ще бъдат използвани (ISA)

За ATtiny85 са достъпни следните видове инструкции:

- аритметико-логически инструкции
- инструкции за разклоняване на потока от данни
- инструкции за работа с битове
- инструкции за пренос на данни

За PIC10F320 са достъпни следните видове инструкции:

- инструкции за разклоняване на потока от данни
- инструкции за работа с битове
- инструкции за работа с байтове

5. Софтуерни инструменти за проектиране на микроконтролерният модул и програмиране



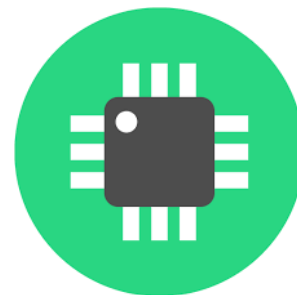
Git е разпределена система за контрол на версиите с отворен код, която може да бъде използвана за проследяване на промените за всеки тип файл.



Avra е инструмент за компилиране на avr assembler.



GNU Make е система за автоматизация на компилацията или тестването на програма. Част е от GNU проекта.



LibrePCB е сет от инструменти за проектиране на микроконтролерни модули

6. Проектиране на микроконтролерния модул

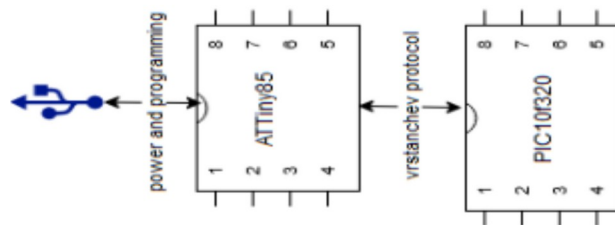
За реализацията на микроконтролерният модул са избрани микроконтролерите:

ATTiny85, PIC10F320

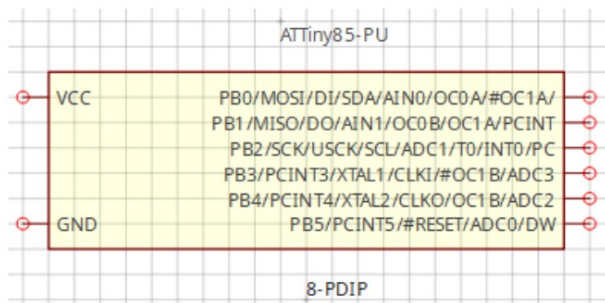
Интерфейси, с които ще разполага микроконтролерния модул:

USB за захранване, vrstanchev протокол за комуникация

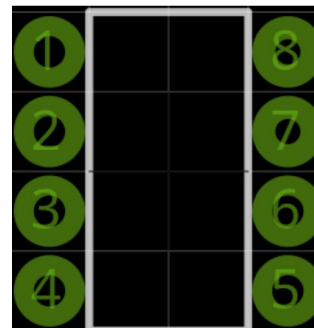
Блокова схема на микроконтролерния модул



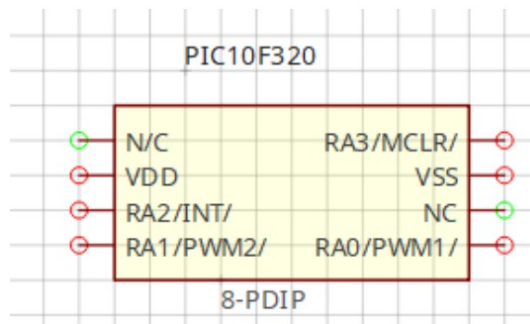
Символи на компонентите



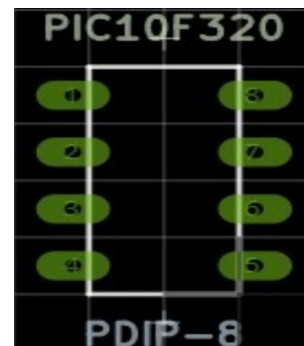
Символ ATtiny-85



Footprint ATtiny-85

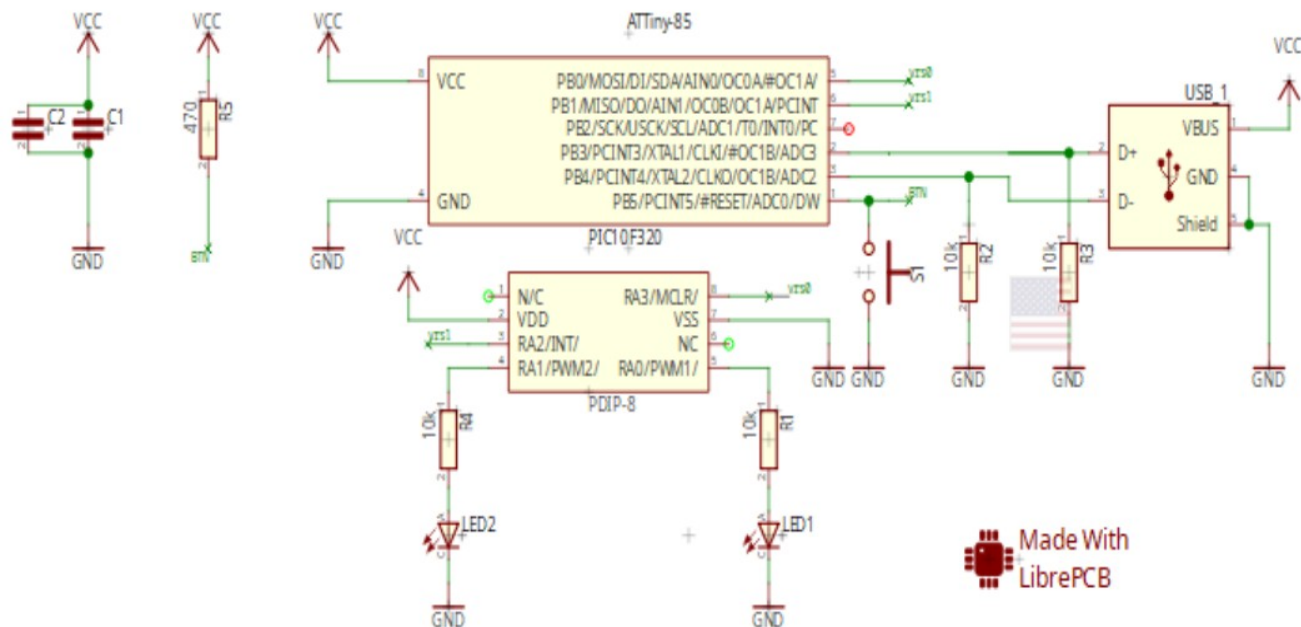


Символ PIC10F320



Footprint PIC10F320

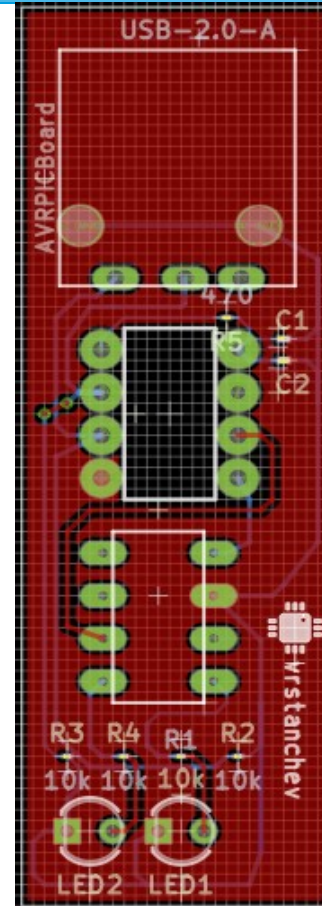
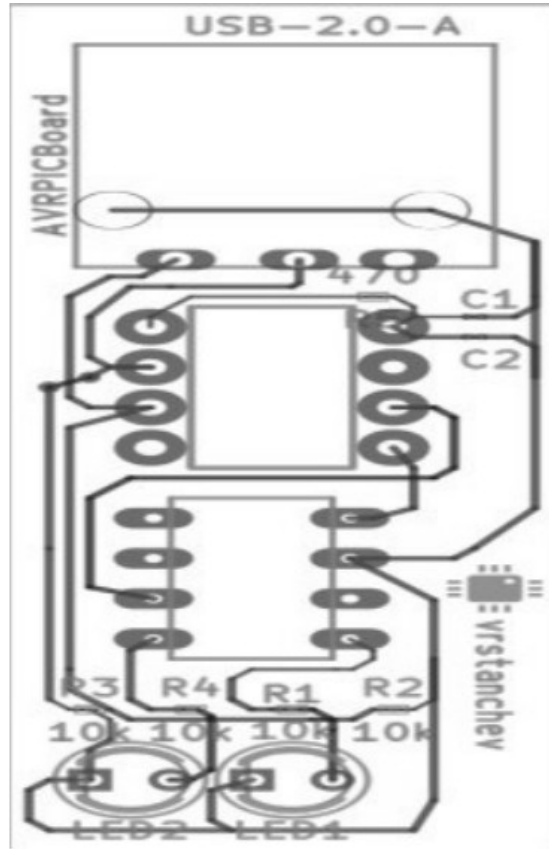
Схема на микроконтролерния модул



Made With
LibrePCB

Интерфейс	Свързаност	Използвани пинове
USB	Attiny85 ↔ USB	PB3, PB4
vrstanchev	Attiny85 ↔ PIC10F320	PB0, PB1 → RA2, RA3

Разположение на модула



7. Програмиране на микроконтролерите

Избрани са следните асемблери:

- AVR Assembler
- PIC Assembler
- Избран е протокол за комуникация `vrstanchev`
- кодът за ATTiny85 ще бъде във функцията `att_func()`
- кодът за PIC10F320 ще бъде във функцията `pic_func()`
- функциите ще бъдат съобразени с протокола `vrstanchev`

- трябва да не се допуска излишно обръщение към регистрите
- трябва да бъде изложена логиката на компилацията
- трябва да бъде показана компилацията
- трябва кодът да бъде реализиран и като софтуерна библиотека поради:
- бъдещото развитие на проекта
- използване за различни проекти

Логика на кода

Софтуерната част на проекта се състои от функциите :

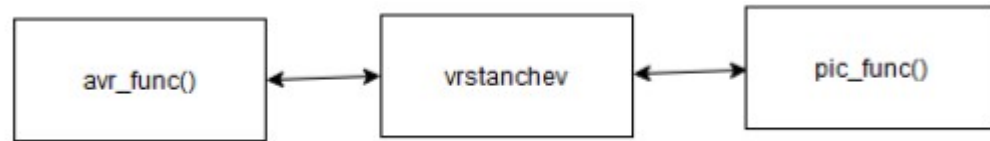
- att_func
- pic_func
- vrstanchev

Във функцията att_func ще бъде описано:

- в съответствие със схемата прави пиновете PB0 и PB1 на ATTINY-85 да бъдат изходни за сигнала, който достига до PIC10F320, след което са прави обръщение към избрания комуникационен протокол
- биват записани стойности в регистри, след като пиновете са направени изходни.
- Тези стойности след обръщането към протокола биват обработени и изпратени към
- другия микроконтролер.

Протокол за комуникация

- Във функцията `pic_func` спрямо описаното в схемата пиновете RA2 и RA3 са входни. Чрез директивата `call` се обръщаме към протокола, за да получим данните.
- Реализиран е самостоятелен протокол за комуникация между микроконтролерите, подобен на UART. Функциите са реализирани в съответствие със схемата на микроконтролерния модул. След като съответните пинове бъдат конфигурирани се прави обръщение към лейбъла на протокола.



Използвани пинове относно протокола

Компонент	Използвани пинове
ATTiny85	PB0 , PB1
PIC10F320	RA2, RA3
LED1, LED2	RA0, RA1

Сравнение между UART и самостоятелния протокол `vrstanchev`

Протокол	UART	VRSTANCHEV
Възможности	Half duplex, full duplex	Half duplex
Реализиран чрез	C	Assembler

- `.include "tn85def.inc"`

```
.equ pb0_out=0b00000001
.equ pb1_out=0b00000010
.cseg
.org 0x00
att_func:
sbi r16, pb0_out
sbi r17, pb1_out
out DDRB, r16
out PORTB,r16
out DDRB, r17
out PORTB,r17
```

Възможности за бъдещо развитие на проекта

1. Разширяване на възможностите на микроконтролерния модул - хардуерна част:

- добавяне на допълнителни интерфейси към модула
- разработване и използване на допълнителни помощни модули към основния
- реализиране на различни версии на модула чрез AVR-AVR
PIC-PIC

2. Разширяване на възможностите на микроконтролерния модул - софтуерна част:

- адаптиране на библиотеката и реализираните функции за друга архитектура (ISA)
- разширяване на възможностите на комуникационния протокол vrstanchev

Насочеността на проекта е изцяло практическа.

Проектираният микроконтролен модул може да бъде използван за университетски и учебни

цели. Може да бъде използван при:

- провеждане на лабораторни упражнения
- изучаване на GNU Tools
- изучаване на взаимодействие между два асемблера
- програмиране на AVR базирани микроконтролери
- изучаване на протоколи за комуникация
- Постигнатите резултати са обобщени в статия по темата на дипломната. Изпратена е за преглед и публикуване в списание с индексирание в SCOPUS. Налична е в Research Gate

Заклучение

В рамките на сегашната дипломна работа беше разгледана целта:

Да бъде проектиран микроконтролерен модул с използване на AVR-базиран микроконтролер и PIC-базиран микроконтролер с цел комуникация между двата микроконтролера на асемблерно ниво.

- Изискванията към софтуерната и хардуерна част на проекта бяха изпълнени.
- Реализиран е самостоятелен протокол за комуникация между микроконтролерите.
- Поставените цели бяха успешно изпълнени.
- Целта на дипломната работа е постигната.
- Дипломната работа ще бъде допълнително развита, чрез използване на някои от посочените възможности.

ИЗТОЧНИЦИ

[1.1] ANAVI Macro Pad 2 DataSheet

[1.2] ANAVI Light Controller DataSheet

[1.3] avr64dd32 DataSheet

[1.4] rp2040 board DataSheet

[2] ATTiny85 Datasheet

[3] PIC10F320 Datasheet

[4] Linux Man Pages

[5] LibrePCB Manual

[6] Designing Microcontroller module with target to use AVR/PIC assembler