

UNIVERZITA PARDUBICE
Fakulta elektrotechniky a informatiky

VELKOFORMÁTOVÉ HODINY

Michal Hroch

Bakalářská práce
2022

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2020/2021

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Michal Hroch**
Osobní číslo: **I18074**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Řízení procesů**
Téma práce: **Velkoformátové hodiny s teploměrem**
Zadávací katedra: **Katedra řízení procesů**

Zásady pro vypracování

Cílem práce je vytvoření digitálních hodin se zobrazením pomocí velkého 7seg displaye (dodá vedoucí práce). Hodiny budou periodicky přepínat mezi zobrazením hodin a zobrazením teploty. Hodiny mohou být řízeny CPU AVR/Atmel Cortex M0+ doplněné obvodem RTC. Napájení řídicí části by mělo být zálohováno bateriemi pro zamezení ztráty časové informace. Hodiny budou vybaveny ISP rozhraním a ovládacími prvky.

Rozsah pracovní zprávy: **30-40**
Rozsah grafických prací:
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

- [1] Matoušek D., Práce s mikrokontroléry Atmel AVR, BEN technická literatura, Praha 2006, ISBN: 80-7300-209-4
[2] Joseph Yiu., The Definitive Guide to the ARM Cortex-M3 (Embedded Technology) 1st Edition, ISBN-13: 978-0750685344
[3] Herout Pavel., Učebnice jazyka C – 1. díl, Kopp, ISBN-13: 978-80-7232-383-8

Vedoucí bakalářské práce: **Ing. Martin Dobrovolný, Ph.D.**
Katedra elektrotechniky

Datum zadání bakalářské práce: **27. listopadu 2020**
Termín odevzdání bakalářské práce: **14. května 2021**

L.S.

Ing. Zdeněk Němec, Ph.D.
děkan

Ing. Daniel Honc, Ph.D.
vedoucí katedry

V Pardubicích dne 29. ledna 2021

Prohlášení

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 13. 5. 2022

Michal Hroch

PODĚKOVÁNÍ

Chtěl bych poděkovat svému vedoucímu práce za pomoc a rady při řešení zadání a spoustě cenných rad při psaní práce. Dále bych chtěl poděkovat svým kamarádům, kteří mi zajistili potřebnou logistiku při řešení plošného spoje v době covidové.

V neposlední řadě chci poděkovat rodině za podporu v době mého studia.

V Pardubicích dne 13. 5. 2022

Michal Hroch

ANOTACE

Bakalářská práce se zabývá návrhem a realizací velkoformátových sedmissegmentových hodin. Byl proveden rozbor možných řešení. Hodiny byly sestrojeny a oživeny.

KLÍČOVÁ SLOVA

ESP32, sedmissegmentový displej, hodiny, teplota

TITLE

THE DIGITAL CLOCK

ANNOTATION

The bachelor thesis deals with the design and implementation of large-format seven-segment lessons. An analysis of possible solutions was performed. The clock was built and revived.

KEYWORDS

ESP32, seven-segment display, clock, temperature

OBSAH

	Seznam zkratk	9
	Seznam ilustrací	10
	Seznam tabulek	11
	Úvod	12
1	Úvod do problematiky	13
1.1	Cíle práce	13
1.2	Postup práce	13
1.3	Prvotní architektura výrobku	14
2	Analýza možných řešení	15
2.1	Zálohování času	15
2.2	Měření teploty	15
2.3	Způsoby řízení displejů	15
2.3.1	Přímé řízení	15
2.3.2	Multiplexní řízení	16
2.3.3	Řízení displeje pomocí posuvného registru	16
2.3.4	Řízení displeje pomocí driverů	16
2.3.5	Vlastní řešení	17
2.4	Získání času	18
2.4.1	Network Time Protokol	18
2.4.2	Rádiové stanice DCF77	18
2.4.3	Polohový systém GPS	18
2.4.4	Řešení	18
3	Návrh zařízení a výběr komponent	19
3.1	Návrh architektury zařízení	19
3.2	Mikrokontrolér	19
3.3	ESP32	19
3.3.1	Procesor	19
3.3.2	Wifi modul	20
3.3.3	RTC	20
3.3.4	Režimy napájení	20
3.4	Rozšíření vstupů	21
3.4.1	Expandér PCF 8575	21

3.5	RTC obvod	22
3.6	Teplotní čidlo	22
3.7	Propojení částí pomocí sběrnice I2C	23
3.7.1	Komunikace na sběrnici	23
3.7.2	Řízení sběrnice zařízením master	24
3.7.3	Adresování	24
3.7.4	Přenosové rychlosti	24
4	Praktická část	26
4.1	Blokové schéma zařízení	26
4.2	Popis jednotlivých celků	27
4.2.1	Napájecí část	27
4.2.2	Řídicí část	28
4.2.3	Sekce expandérů	29
4.2.4	RTC	30
4.2.5	Sedmisegmentové displeje	30
4.3	Deska plošných spojů	31
4.4	Ovládání displejů	32
4.5	Programová část	34
4.5.1	Sekce Main Loop	34
4.6	Oživení, ověření funkce, optimalizace	37
4.7	Budoucí rozšíření	37
5	Výrobek	37
5.1	Stav výrobku	38
6	Závěr	39
	Použitá literatura	40
	Přílohy	42

SEZNAM ZKRATEK

ACK	Acknowledgement code	Potvrzovací signál
AP	Access Point	Přístupový bod
BSS	Basic Service Set	Infrastrukturní síť
CPU	Central Processing Unit	Centrální procesorová jednotka
CS	Chip Select	Výběr čipu
DCF	Rádiová stanice	
DNS	Domain Name System	System doménových jmen
ICMP	Internet Control Message Protocol	Protokol řídicích zpráv
IDE	Integrated Development Environment	Vývojové prostředí
I2C	Inter-Intergrated Circuit	Interní datová sběrnice
LED	Light-Emitting Diode	Elektroluminiscenční dioda
NPN	Bipolární tranzistor	
NTC	Negative Temperature Coefficient	Negastor
PCB	Printed Circuit Board	Deska plošných spojů
RTC	Real Time Clock	Hodiny reálného času
STA	Stations	Stanice
SCL	Serial Clock	Sériové hodiny
SDA	Serial Data	Sériová data
TCP/IP	Transmission Control Protocol/Internet Protocol	Sada komunikačních protokolů
Wifi	Wireles Fidetily	Bezdrátová komunikace

SEZNAM ILUSTRACÍ

Obr. 1.1 – Blokové schéma výrobku	13
Obr. 1.2 – Prvotní návrh architektury	14
Obr. 2.1 – Posuvný registr 74HC595	16
Obr. 2.2 – Driver MAX7219	17
Obr. 2.3 – Konverze (Marco, 2013).....	17
Obr. 4.1 – Konečné blokové schéma výrobku.....	26
Obr. 4.2 – Schéma napájecí části	27
Obr. 4.3 – Provedení napájecí části na PCB	28
Obr. 4.4 – Schéma řídicí části.....	28
Obr. 4.5 – Použitý expandér ve schématu	29
Obr. 4.6 – Použitý expandér na PCB	29
Obr. 4.7 – Modul DS3231	30
Obr. 4.8 – Připojení sedmissegmentových displejů	31
Obr. 4.9 – Vyleptaný a osazený plošný spoj.....	32
Obr. 4.10 – Návrh plošného spoje	32
Obr. 4.11 – Umístění segmentů na displeji.....	33
Obr. 4.12 – Vývojový diagram Init části	34
Obr. 4.13 – Vývojový diagram Main Loop části.....	36
Obr. 5.1 – Dokončený výrobek.....	38

SEZNAM TABULEK

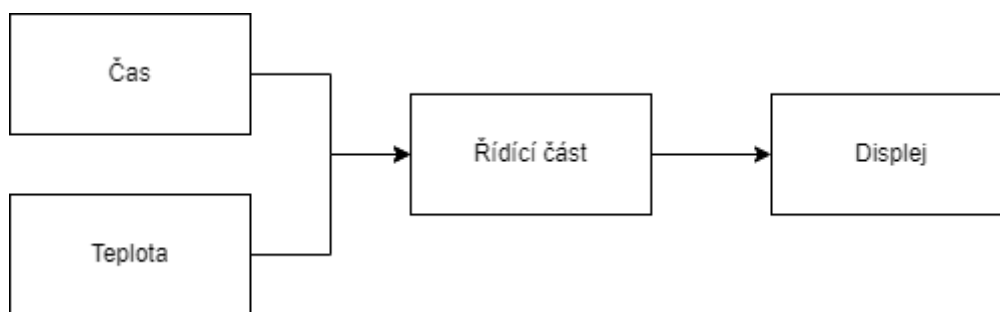
Tabulka 4.2 – Řídicí tabulka pro ovládání displejů	33
--	----

ÚVOD

Cílem práce je vybrat vhodné komponenty pro digitální sedmissegmentové hodiny, jejich následný návrh a realizace. Dalším bodem práce je vytvořit program, který dokáže získat přesný čas a periodicky ho zobrazovat spolu s teplotou.

1 ÚVOD DO PROBLEMATIKY

Digitální hodiny jsou zařízení, která mohou zobrazovat čas, datum nebo třeba i teplotu. Doma jsou využívány pro spoustu věcí například na troubě pro pečení, na lednici nebo třeba jen na obyčejné probuzení do práce. Tento výrobek byl vybrán, protože jsem ho chtěl použít pro své osobní účely. Dalším impulzem mimo jiné bylo získání sedmisegmentových displejů od vedoucího práce.



Obr. 1.1 – Blokové schéma výrobku

1.1 CÍLE PRÁCE

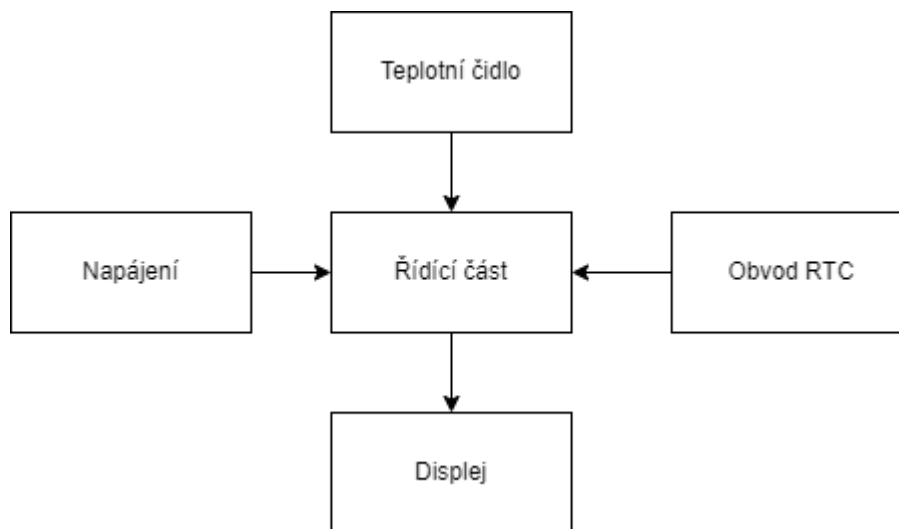
Cílem je vytvoření funkčních hodin, které lze připevnit na zeď. Zobrazují aktuální čas a teplotu. Čas je získáván a automaticky aktualizován z ntp serveru. Změnu zimního času na letní čas je nutno provádět přeprogramováním. Střídání mezi zobrazením času a zobrazením teploty je pouze přepnutím zobrazení. Je třeba, aby projekt obsahoval teplotní čidlo, které půjde umístit do dostatečné vzdálenosti od hodin, aby jejich případné teplo neovlivňovalo teplotu měření. Napájení výrobku bude provedeno pomocí adaptéru. Dalším cílem je nepřesáhnout rozměrem desky plošných spojů velikost displejů.

1.2 POSTUP PRÁCE

Bylo potřeba provést rešerši, jak lze tyto displeje ovládat, nejlépe řešit zálohování a měření teploty. Poté byly vybrány součástky, které byly třeba použít. Dalším bodem bylo složení obvodu na nepájivém poli, kde se použila první verze kódu. Poté bylo vytvořeno schéma a deska plošných spojů ve vývojovém prostředí KiCad. Deska byla vyleptána, osazena a oživena. Nakonec byl nahrán program a doupraven na již hotové desce plošných spojů.

1.3 PRVOTNÍ ARCHITEKTURA VÝROBKU

Zařízení je tvořené z napájecí části, řídicí části, teplotního čidla a modulu obvodu RTC. Modul obvodu RTC lze zálohovat pomocí baterie. Napájení je řešeno externě. Řídicí část je použita pouze jako logická část výrobku. Teplotní čidlo je digitální, pro snazší práci se zobrazováním a dostačující přesností.



Obr. 1.2 – Prvotní návrh architektury

2 ANALÝZA MOŽNÝCH ŘEŠENÍ

2.1 ZÁLOHOVÁNÍ ČASU

Tento problém je převážně řešen pomocí záložního zdroje. Jedná se zpravidla o akumulátor, který je schopen po omezenou dobu dodávat energii potřebnou pro provoz celého mikrokontroléru a dalších součástí. Nevýhodou je, že pokud dojde záložní zdroj a původní je stále neobnověn, dojde ke ztrátě času. Jako další řešení bývá použito zálohování pomocí obvodu reálného času, který má na sobě připojenou baterii. Díky tomu je obvod schopný si uchovat časovou informaci.

2.2 MĚŘENÍ TEPLoty

Měření teploty je nedílnou součástí mnoha dalších projektů. Používá se ve sklenících, ale také ve výrobních závodech. Pro její měření se převážně nyní používají digitální senzory. Je to především díky jejich dostupnosti a snazší práci s nimi. Takovéto teploměry mohou být i zataveny, což zajistí jejich vodotěsnost. Analogové senzory jsou obvykle přesnější, ale práce s nimi je složitější. U analogových čidel teploty se používají převážně termistory NTC, kterým s vyšší teplotou klesá odpor. Mezi jejich nevýhody patří kalibrace a nutnost přesného měření odporu.

Mezi oblíbené součástky patří i digitální DS18B20. Lze ji připojit pomocí OneWire a je s ním snadná práce. Dokáže měřit teplotu i vlhkost. Použít lze od -55 °C ~ $+125\text{ °C}$ s přesností $\pm 0,5\text{ °C}$. Pro potřeby mé práce, kde se bude měřit pouze pokojová teplota, je to zbytečně moc přesné. Místo toho jsem zjistil, že lze měřit teplotu pomocí již vlastněného modulu DS3231, který budu chtít použít ve svém výrobku. Proto tento modul využiji i na měření teploty.

2.3 ZPŮSOBY ŘÍZENÍ DISPLEJŮ

2.3.1 Přímé řízení

Jedná se o naprosto nejzákladnější a nejjednodušší řízení sedmissegmentového displeje. Nejdříve jsou připojeny všechny vývody displeje k vývodům mikrokontroléru. Tím lze řídit každý jednotlivý segment displeje pomocí jednoho pinu. Značná nevýhoda tohoto řízení je velká spotřeba pinů mikrokontroléru. Na každý digit by bylo potřeba 8 vývodů, pokud není počítána i tečka. Další velkou nevýhodou je proudové zatížení pinů mikrokontroléru (Tajned, 2021).

2.3.2 Multiplexní řízení

Tento způsob je hlavně používán, když je potřeba připojit víc digitů. Napřed se propojí všechny stejné segmenty jednotlivých displejů. Díky tomu je potřeba pouze 8 vývodů, které určují nastavení segmentů. Následně je potřeba ještě jeden pin za každý použitý digit, který se připojí na tranzistor. Tento tranzistor má za úkol spínat napájení pro jednotlivé displeje z dané skupiny. Po takovémto zapojení probíhá řízení tak, že se nastaví hodnota na prvním digitu a pustí se napájení pouze do téhož displeje. Poté je digit vypnut, nastavena hodnota další pozice a ta je rozsvícena. Pokud je tento postup prováděn dostatečně rychle, vytvoří se iluze, že vše svítí najednou. V tomto případě se doporučuje frekvence minimálně 50 HZ, ideálně ale více jak 60 HZ (Tajned, 2021).

2.3.3 Řízení displeje pomocí posuvného registru

Posuvný registr je součástka, která je schopna sériové komunikace. Pro ovládání už stačí 8bitová součástka. Takový obvod bývá připojen pouze třemi vodiči k mikrokontroléru a dalšíma dvěma k napájení a zemi. Použití posuvného registru umožňuje uvolnit další piny na mikrokontroléru, protože jich nepotřebuje tolik. Většinou se jedná o hodnotu kolem 4 pinů. Hlavní výhodou je omezení proudového zatížení pinů mikrokontroléru (Tajned, 2021).

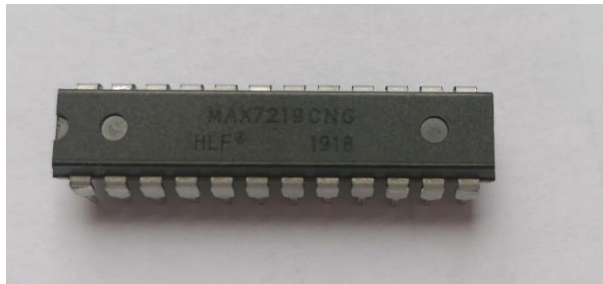


Obr. 2.1 – Posuvný registr 74HC595

2.3.4 Řízení displeje pomocí driverů

Další možností je použití driverů. Jedná se o součástku, která je už vyráběna pro řízení sedmisegmentových displejů. Většinou už mají vstupy pro jednotlivé segmenty a digity. Stačí je připojit pouze dvěma piny z mikrokontroléru. Programování s použitím těchto driverů je také

značně jednoduší. Lze použít už předem vytvořenou knihovnu pro daný driver. Nevýhodou může být malá možnost proudového zatížení (Agnihotri, 2021).



Obr. 2.2 – Driver MAX7219

Naprostá většina jich je tvořena pro displeje se společnou katodou. Takovéto součástky jdou ovšem použít i pro displeje se společnou anodou. Vyžaduje to ovšem drobné úpravy kódu a zapojení. Jedná se především o přepojení jednotlivých segmentů na piny pro digity a napájení digitů na piny pro segmenty. Následně se musí otočit tabulka pro nastavování čísel o 90° (Marco, 2013).

Společná katoda – max7219

A-g: Led diody na digitu

	g	f	e	d	c	b	a
Digit1	0	1	1	1	1	1	1
Digit2							
Digit3							
Digit4							
Digit5							
Digit6							
Digit7							
Digit8							

90°

Digit 1-x: Pořadové číslo digitu

Společná anoda – max7219

A-g: Pořadové číslo digitu

	g	f	e	d	c	b	a
Digit1							1
Digit2							1
Digit3							1
Digit4							1
Digit5							1
Digit6							1
Digit7							0
Digit8							

Digit 1-x: Led diody na digitu

Obr. 2.3 – Konverze (Marco, 2013)

2.3.5 Vlastní řešení

Ze všech možných řešení ovládání by bylo nejlepší ovládání přes driver, ale nepodařilo se mi sehnat žádný, který by byl schopen urdit takto velké displeje. Proto jsem se rozhodl vytvořit si vlastní driver. Toho bych chtěl docílit tak, že všechny displeje budou adresovatelné

a půjde na nich nastavit již předdefinovaný znak. K tomu je potřeba rozšíření výstupů. Proto budu hledat expandér, který mi umožní adresování s dostatečným počtem výstupů.

2.4 ZÍSKÁNÍ ČASU

2.4.1 Network Time Protokol

Jedná se o internetový protokol (ve zkratce NTP), který je používán k synchronizaci času. Spadá pod jednu z nejstarších částí TCP/IP. Výhodou používání NTP je dostačující přesnost a jednoduchost získání času. Při programování je ovšem nutné nastavit offset podle zeměpisné polohy. Tato technologie také samostatně nezmění zimní čas na letní (Kirvan, 2022).

2.4.2 Rádiové stanice DCF77

Jde o rádiové stanice, které vysílají časový signál, podle kterého se synchronizují například hodiny. Jedná se o velmi přesnou informaci. Poskytne přesný čas, datum, a dokonce zajistí i změnu času mezi zimním a letním. Jedná se o mnohem přesnější řešení než NTP. Pro použití je třeba přijímat tento rádiový signál. Tato technologie by obnášela pořízení dalšího modulu pro příjem (Poupa, 2002).

2.4.3 Polohový systém GPS

Další možností je použití systémů GPS nebo GLONASS. Tyto systémy mají přesný čas, protože jsou navázané na mezinárodní atomový čas. Pro získání této časové informace je potřeba zakoupit anténu pro příjem signálu (Papouch, 2019).

2.4.4 Řešení

Do tohoto výrobku jsem zvolil použití NTP. Pro potřeby výrobku se jedná o dostačující přesnost. Toto řešení neobsahuje žádný modul navíc, a tudíž není zvýšena potřebná velikost plošného spoje ani cena.

3 NÁVRH ZAŘÍZENÍ A VÝBĚR KOMPONENT

3.1 NÁVRH ARCHITEKTURY ZAŘÍZENÍ

Návrh byl poměrně intuitivní. Bylo potřeba řídicí jednotce dodat napájení pro řídicí mikrokontrolér a zároveň dostatečné napájení pro samotné segmenty hodin. Z důvodu bezpečnosti a po předchozích zkušenostech zvolím v napájecí části použití modulů step-down, díky kterým nedojde k přepětí řídicí části. Tyto moduly, pouze s jiným rozsahem, budou použity také na napájení displeje hodin. Dále v napájecí části bude zakomponován diodový můstek. Ten zabrání připojení opačné polarity napájení. Následovat bude řídicí mikrokontrolér. Ten bude zajišťovat komunikaci s Wifi. Vzhledem k počtu digitů bude potřeba řešit rozšíření pinů z řídicího mikrokontroléru a jejich následné ovládání. Další část jsou samotné displeje. Ty budou připojeny k těmto rozšířením. Jedná se o sedmisegmentové displeje se společnou anodou. Každý z těchto displejů má osm vývodů pro jednotlivé segmenty a dva pro napájení. Dále bude k řídicí části připojen obvod RTC a teplotní čidlo. Nakonec bude třeba aktivní prvky propojit.

3.2 MIKROKONTROLÉR

U mikrokontroléru byla především požadovaná možnost připojení na Wifi a velký počet pinOut výstupů, na které by byla možnost přímo připojit displeje. Dále byla potřeba možnost rozšíření výstupů přes jakékoliv sběrnice. Jedna z možností bylo Arduino UNO, které by se případně rozšířilo o moduly pro Wifi a Bluetooth, spolu s ESP32. ESP32 bylo zvoleno především kvůli integrované Wifi a Bluetooth, vnitřnímu RTC obvodu a pěti režimům spotřeby, které by se daly využít i v případném rozšíření nebo upravení výrobku. Vzhledem k již vyleptané anténě na Wifi dojde k úspoře místa na desce plošných spojů. Tento mikrokontrolér také obsahuje stejný počet vývodů pro I2C jako Arduino UNO, které bude potřeba pro rozšíření vstupů pro možnosti ovládání.

3.3 ESP32

3.3.1 Procesor

Srdcem mikrokontroléru ESP32 je jeho procesor. Jedná se o Xtensa LX6. Tento procesor je dvoujádrový, 32bitový a jeho výkon je až 600 DMIPS. V tomto řešení bylo použito

poměrně známé ESP32, kvůli přítomnosti Wifi modulu a zabudovanému obvodu reálného času a možnostem nastavování spotřeby (Expressif, 2021).

3.3.2 Wifi modul

Wifi modul mikrokontroléru ESP32 pracuje na standardech 802.11 b/g/n a implementuje sady protokolů TCP/IP. Také podporuje základní síťovou topologii, jenž se nazývá BSS – Basic Service Set.

Zařízení, která se připojí k Wifi síti, se označují jako Stations (STA). Spojení se sítí je zprostředkováváno pomocí přístupových bodů. Ty se označují AP. Tento modul se může chovat jako Station, tak i jako Wifi hotspot, označovaný jako Soft AP. Oba tyto stavy se dějí pod dohledem DCF. Jedná se o distribuční koordinační funkci, která primárně brání kolizím na síti. Řízení spotřeby v těchto situacích je řešeno s minimální interakcí s hostitelem, aby se minimalizovala aktivní doba komunikace (Expressif, 2021).

3.3.3 RTC

Hodiny mohou mít pět různých zdrojů. Jedná se o externí 32 kHz krystalové hodiny, o externí krystalové hodiny vydělené čtyřmi, interní RC oscilátor (obvykle 150kHz), interní 8 MHz oscilátor, interní 32,25 kHz hodiny. Pokud je mikrokontrolér v normálním režimu spotřeby a CPU potřebuje rychlejší přístupy, mohou být vybrány externí, vysokorychlostní hodiny nebo interní 8 MHz oscilátor. Pokud mikrokontrolér je v režimu nízké spotřeby, vybere se jeden ze zbývajících možností zdrojů. Tyto hodiny mohou sloužit pro případné budoucí změny, kdy bude zálohována celá řídicí část a čas bude uchováván zde (Expressif, 2021).

Tyto obvody však nebyly nakonec použity. Byl využit obvod DS3231. Jeho hlavní výhoda spočívá v již existujícím zálohování pomocí baterie CR2032.

3.3.4 Režimy napájení

Mikrokontroler má propracovanou technologii napájení, která umožňuje pět režimů. V prvním, který je označován jako aktivní, je vše používáno bez omezení. Druhý mód se nazývá Modem-sleep. V tomto režimu není Wifi, Bluetooth a rádio používáno. CPU není omezeno a hodiny jsou nastavitelné. Spotřeba se v tomto stavu pohybuje v závislosti na nastavených hodinách od 25 mA do 68 mA. Další možností je nastavení light-sleep. Jedná se o situaci, při které je CPU pozastaveno. RTC a přidružené periferie, stejně jako koprocesor, pracují. K probuzení dojde při jakémkoliv eventu. Spotřeba si při tomhle režimu pohybuje

kolem 0,8 mA. Předposledním režimem je Deep-sleep. Zde je napájené hlavně RTC a přidružené periferie. Spotřeba záleží na několika faktorech. Pokud je napájena jen RTC část, tak se spotřeba pohybuje kolem 10 μ A. Pokud k tomu jsou napájeny i senzory s koprocesorem, spotřeba stoupne na 150 μ A. Posledním režimem je Hibernace. V té je napájen jen RTC časovač, který je případně může probudit čip z hibernace. Spotřeba je 5 μ A (Expressif, 2021).

3.4 ROZŠÍŘENÍ VÝSTUPŮ

Rozšíření výstupů bylo potřeba vyřešit, aby šlo připojit displeje a ovládat je. Proto byl hledán expandér, který by šlo ideálně napájet z již vybraného ESP32, měl připojení přes I2C a alespoň dalších 8 výstupů. Vybral jsem expandér PCF8575, který byl přímo popsán na stránkách o ESP32. Jedná se o obvod, který lze připojit pomocí rozhraní I2C přes pin SCL, přenášející hodinový signál a pin SDA, jenž je vodičem datovým. Tento modul poskytuje rozšíření o 16 pinů navíc, čímž umožní řídit dva digity displejů. Je zde potřeba napájení od 2,5 V po 5,5 V. Tohle rozmezí výborně sedí pro napájení z ESP32 přes pin s 3,3 V. Při tomto napětí dochází ke spotřebě maximálně 10 μ A. Možnost adresování je přes 3 hardwarové adresní piny, které umožní připojení až osmi zařízení na rozhraní I2C. Navíc jsem již tyto expandéry vlastnil, takže jsem se rozhodl je přímo využít v této práci (Nextewo, 2016).

3.4.1 Expandér PCF 8575

Tento expandér je šestnácti bitový a má 16 vstupně výstupních pinů. Připojení je možné přes sběrnici I2C. Díky možnostem sběrnice, které jsou popsány níže, tento obvod za cenu připojení k hodinovému signálu a signálovému vodiči poskytne dalších 16 pinů. Ty lze následně volitelně použít. Tento modul dokáže pracovat i s kolísavým napětím, které může být od 2,5 V až do 5,5 V. Díky svému provedení je také vhodný na řízení led displejů s velkými protékajícími proudy. Tento expandér také vyniká svojí spotřebou. V nečinnosti potřebuje maximálně pouze 10 μ A. Zároveň má poměrně velkou pracovní frekvenci. Je schopen pracovat na 400 kHz, což je fast mode na sběrnici I2C. Také disponuje hardwarovým nastavením adresy. Jedná se o tři hardwarové piny ze spodní strany součástky, díky kterým je možno připojit až 8 těchto zařízení. Tento obvod jsem vybral především proto, že jsem již několik kusů vlastnil a měl jsem s nimi velmi dobré zkušenosti. Proto jsem je využil na vytvoření vlastního ovládání jednotlivých pinů displeje (Texas Instruments, 2020).

3.5 RTC OBVOD

Při výběru obvodu RTC jsem především chtěl obvod, který dokáže udržet čas i po výpadku napájení díky baterii. Ta je součástí obvodu. Dále mezi parametry v tomto případě patřila cena a dostupnost knihoven. Ohledně připojení jsem hledal především I2C rozhraní. Z množství obvodů jsem nakonec vybral modul DS3231. Jako výchozí adresu má tento obvod 0x57. Vybraný modul jsem používal již dříve a měl jsem s ním velmi dobré zkušenosti ohledně udržení času po vypojení napájení. Čas udrží díky výměnné CR2032, která je běžně dostupná. Jedná se o baterii, která má napětí 3,6 V s kapacitou 220 mAh. U tohoto obvodu se udává zpoždění kolem dvou minut za rok. Navíc se jedná o tepelně kompenzovaný krystal, a tudíž se obvod nezpozdí tolik, jako jiný, nekompensovaný obvod (Last Minute Engineers, 2021).

3.6 TEPLOTNÍ ČIDLO

Při výběru byly použity jako kritéria cena a připojení. Přesnost nebyla prioritní, protože bylo předpokládáno, že čidlo bude používáno pro měření pokojových teplot, a proto je přesnost na celé stupně dostačující. Tuto přesnost splňuje naprostá většina dostupných a běžně používaných senzorů. Mezi favority se nakonec dostalo čidlo DS18B20.

Jedná se o velmi rozšířené čidlo i s knihovnou. Má slušnou přesnost a velmi velký teplotní rozsah. Ani cena rozhodně není překážkou. Připojení tohoto senzoru se provádí přes rozhraní 1 – Wire. Nakonec jsem se pro toto čidlo nerozhodl a místo toho jsem použil teplotní senzor již na modulu DS3231. K této změně došlo především proto, že jsem nechtěl DS3231 připojit na desku plošných spojů přímo, aby případné teplo, které by na desce vzniklo, ještě více nezpozdilo krystalový oscilátor. Největším problémem byla malá přesnost tohoto senzoru, která je udávaná ± 3 °C. Tento problém byl vyřešen tak, že jsem zapojil tento modul, abych ověřil jeho přesnost. Po změření a porovnání pomocí termostatu v místnosti jsem zjistil, že na rozsahu 15 °C až 25 °C má maximální nepřesnost $\pm 0,5$ °C. To již byla přijatelná nepřesnost, a tak byl tento obvod použit i pro měření teploty. Jelikož jsem tento modul již předem vlastnil, tak nebylo nutno už zakupovat další. Navíc využití stejného modulu, stejně jako pro čas, umožnilo opět nerozšiřovat desku plošných spojů nad cílenou velikost (Last Minute Engineers, 2021).

3.7 PROPOJENÍ ČÁSTÍ POMOCÍ SBĚRNICE I2C

Sběrnice I2C je jedna z nejrozšířenějších sběrnic dnešní doby. Je možné ji najít jak v osobním počítači, tak i v mobilech. Jedná se o sériovou sběrnici vyvinutou firmou Philips. Hlavní výhodou této sběrnice je to, že veškerá komunikace probíhá pouze po dvou připojených vodičích. Jelikož se jedná o poměrně starou sběrnici, existuje základní a rozšířená verze. V základní verzi se adresuje pouze sedmibitově. U rozšířené verze je to dokonce deset bitů. Díky tomu lze připojit velké množství dalších čipů na společnou sběrnici (HW server, 2000).

3.7.1 Komunikace na sběrnici

Pro komunikaci je používán hodinový signál, díky kterému jde o přenos synchronní. Dále se jedná o poloduplexní přenos, v jeden okamžik může pouze jedno zařízení na sběrnici vysílat, ovšem není podmínkou, že pouze jediné zařízení poslouchá. U této sběrnice se také nepoužívá speciální signál CS, kterým se vybírá zařízení jako slave. Je to kvůli tomu, že každému uzlu lze přiřadit jedinečnou adresu. Komunikace probíhá přes dva vodiče. První z nich je signálový SDA, který slouží pro oboustranný přenos dat. Druhý je SCL. Jedná se o vodič, který ostatním zařízením posílá hodinový signál. Maximální délka vodičů je určena podle jejich nejvyšší přípustné kapacity 400 pF. Tyto vodiče tvoří otevřený kolektor. Oba jsou připojeny na dva pull – up rezistory. Díky tomu v momentě, kdy jsou uzly odpojeny, dochází ke zvýšení napětí na obou koncích na úroveň logické jedničky. Zde je logická jednička definována jako klidový stav. V tomto stavu může sběrnice zůstat neomezenou dobu. Pokud nastane komunikace, tak datové bity jsou posílány na SDA. Tato hodnota se ovšem nesmí změnit, pokud hodnota SCL není logická nula. Toto je porušeno pouze tehdy, jedná-li se o Start a Stop. Jelikož tato sběrnice má rozdílné maximální frekvence, jsou definovány rozdílné minimální doby setrvání vodiče SCL na úrovni logické jedničky a nuly. Při komunikaci na této sběrnici také dochází k synchronizaci hodin. Toho je docíleno tak, že se SCL nastaví na logickou jedničku a ostatní zařízení odměří dobu až od okamžiku, kdy tento vodič dosáhne hodnoty logické jedničky. To samé se děje s logickou nulou, nejdříve se nastaví na SCL logická nula, ale sledující zařízení odměří dobu až skutečně po dosažení hodnoty logické nuly. Toto umožňuje pomalejším zařízením pozdržet přenos od vysílacího zařízení tím, že podrží logickou nulu na vodiči SCL (HW server, 2000).

3.7.2 Řízení sběrnice zařízením master

Celé řízení sběrnice vede vždy jedno zařízení – master. Díky tomu na sběrnici nebude docházet ke kolizím. Ostatní zařízení pracují jako slave. Nemohou řídit sběrnici, nemůžou přijímat ani odesílat data, pokud o to nejsou masterem požádáni. Pro informace, které zařízení typu slave potřebuje komunikovat hlavně kvůli vnějším událostem, existují tři řešení. Prvním je neustálé dotazování se na čidlo. V takovém případě se jedná o polling. Řadič se může po každém daném úseku zeptat čidla na stav všech připojených zařízení. V závislosti na odpovědích může provést příslušnou reakci na již zmíněný stav. Může se například jednat o zvýšenou teplotu prostoru, která by mohla spustit ventilátor. Druhou možností je, že má zařízení typu slave další speciální vodič, který slouží k informování řadiče o tom, že zařízení potřebuje komunikovat. Toto řešení lze považovat za obcházení této sběrnice. Poslední možností je řešení pomocí multi – master. Jak už název napovídá, jedná se o síť s více mastery. Takovéto řešení může být značně produktivnější, ale dochází u něj ke zvýšení nákladů a provedení může být značně složité. Největším problémem ovšem je zabránění kolizí a případně jejich řešení. Jednou z možností pro řešení těchto kolizí je hierarchie jednotlivých uzlů (HW server, 2000).

3.7.3 Adresování

Každé zařízení na dané sběrnici, které je ve stavu slave, může dostat svoji jedinečnou adresu. Díky tomu lze poté jednoznačně určit, s jakým uzlem chce master komunikovat. Pro menší počet připojených zařízení se používá sedmibitová adresa. S takovou adresou lze rozlišit až 128 zařízení. Při větším počtu zařízení lze použít desetibitovou adresu, která je schopna pokrýt až 1024 zařízení. Obě tato čísla ale jsou ve skutečnosti menší, protože obsahují již adresy, které jsou rezervované pro vysílání příkazů. Lehkou nevýhodou desetibitového adresování je, že některá již starší čidla nemusí tento způsob adresace podporovat (Tišnovský, 2009).

3.7.4 Přenosové rychlosti

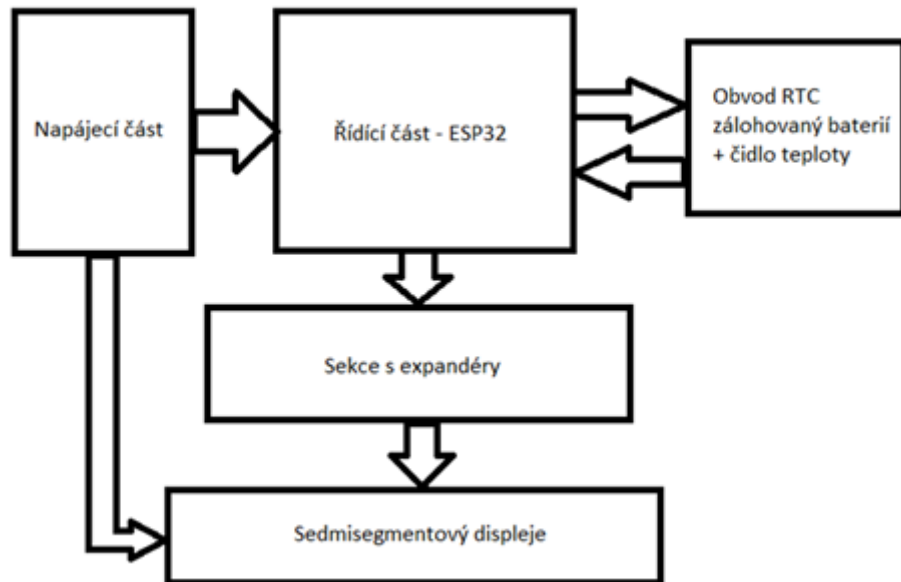
Mezi výhody sběrnice I2C nepochybně patří standardizované rychlosti přenosu. Ty vycházejí z frekvence hodinového signálu. Do těchto rychlostí se započítávají i potvrzovací bity. První přenosovou rychlostí je low speed mode. Jde o nejpomalejší standardní přenosovou rychlost, která je 10 kbps. Poněkud rychlejší je standard mode, který dosahuje až 100 kbps. Čtyřikrát rychlejší, tudíž 400 kbps je fast mode. Tyto tři rychlosti jsou jedny z nejrozšířenějších. Dále jsou rychlosti 1 Mbps, která se nazývá fast mode +. A nejrychlejší je high speed mode,

který dosahuje až 3,4 Mbps. Ve všech těchto rychlostech je datový bajt přenášen minimálně v osmi hodinových cyklech, kde v devátém hodinovém cyklu je přenesen potvrzovací bit ACK. Díky tomu je reálná rychlost přenosu užitečných dat trochu nižší (Dudáček, 2002).

4 PRAKTICKÁ ČÁST

4.1 BLOKOVÉ SCHÉMA ZAŘÍZENÍ

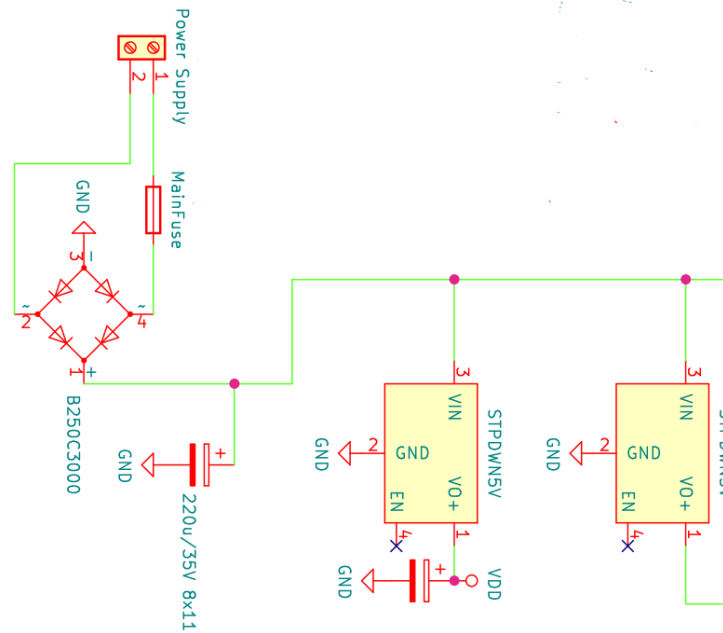
Při vytváření blokového schéma bylo vycházeno z návrhu architektury zařízení.



Obr. 4.1 – Konečné blokové schéma výrobku

4.2 POPIS JEDNOTLIVÝCH CELKŮ

4.2.1 Napájecí část



Obr. 4.2 – Schéma napájecí části

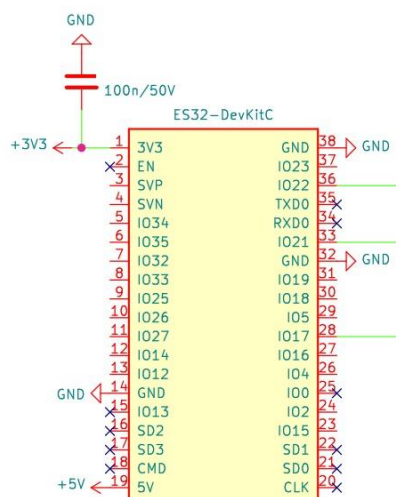
Napájecí část je tvořena samostatnými moduly. Byly použity moduly stepdown, které zajistí při minimálních rozměrech vysokou účinnost. Z důvodu předpokládané spotřeby jsem zvolil 2 moduly. Jeden bude zásobovat displeje, druhý řídicí část. Dále je napájecí sektor vybaven LED diodami, které signalizují napětí +5 V, napájení displejů a napájení řídicí části. Tyto diody byly přidány pro snadnější detekci případné chyby. Také je zde přidán Graetzův můstek, který upraví případnou opačnou polaritu připojeného napětí. Tato část neobsahuje napájecí adaptér. Ten se připojuje externí. Dále bylo potřeba vyřešit spotřebu pro dimenzování stepdown modulů. Největší spotřebu zde jistě mají displeje, kterými maximálně protéká 30 mA na segment. Při počtu osmi segmentů i s tečkou na digitu, je maximální proud na digit 240 mA. Všechny dohromady tedy mají maximálně skoro 1,5 A. Vybral jsem tedy 4 V – 24 V modul, který zvládne až 3 A. Tento modul je použit i u sekce řídicí části. Tam jsem odhadl maximální odběr do 1 A. Jedná se ovšem o maximální hodnoty, běžná pracovní hodnota u displejů je 10 mA, a tudíž lze použít 18W adaptér, ideálně ale výkonnější.



Obr. 4.3 – Provedení napájecí části na PCB

4.2.2 Řídicí část

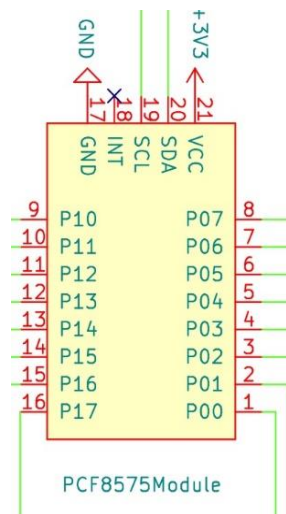
Tento sektor je tvořený pomocí ESP32, které ovládá obvod RTC a expandéry.



Obr. 4.4 – Schéma řídicí části

S obvodem RTC probíhá komunikace přes I2C. Tím je získána teplota a čas, pokud se nepovedla jeho aktualizace. Aktuální čas je získáván ze serveru NTP, přes Wifi modul řídicí jednotky. Z pohledu řídicí části a displeje se tato část chová pouze jako logická, zatímco veškeré spínání je prováděno v sekci expandérů. Tím je zajištěna další ochrana proti případným přepětím. Celá řídicí část je připojena na PCB patice, které umožňují rychlou výměnu případného poškozeného ESP32 za nové, bez nutnosti odpájení z desky plošných spojů.

4.2.3 Sekce expandérů



Obr. 4.5 – Použitý expandér ve schématu

Expandéry jsou připojeny přes I2C sběrnici. Každý má nastavenou odlišnou adresu pro jednoznačnou komunikaci s řídicí částí. Jde o adresy 0x20, 0x21 a 0x22. Každý ze tří expandérů ovládá 16 tranzistorů, kterými jsou spínány jednotlivé segmenty dílčích částí displeje. Každý expandér je zapojen do PCB patice pro jeho rychlou a nenáročnou výměnu v případě poškození.



Obr. 4.6 – Použitý expandér na PCB

4.2.4 RTC

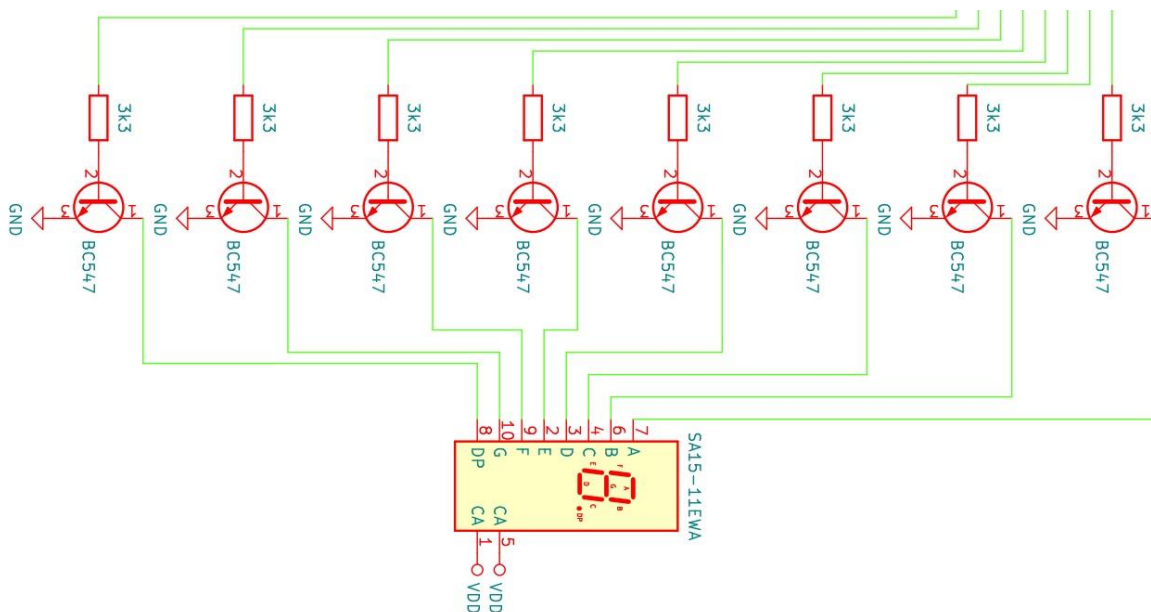
Modul DS3231 byl zvolen jako obvod reálného času a zároveň i teplotní čidlo. Jedná se o součástku, jejíž komunikace probíhá přes I2C sběrnici. Jelikož umístění modulu nebylo plánováno přímo na desce plošného spoje, je z ní vyveden pomocí 4 vodičů. Tento počet vodičů je dán počtem pinů pro připojení na sběrnici I2C. Jedná se o napájení, zem, hodinový signál a datový signál. Tento modul obsahuje teplotně kompenzovaný krystalový oscilátor, který zajistí menší zpoždění vlivem teplot. Také jsem tento obvod použil, protože jsem ho již vlastnil a do výrobku se mi vzhledem k požadavkům na přesnost hodil. Práce s ním je poměrně jednoduchá díky veřejně dostupné knihovně.



Obr. 4.7 – Modul DS3231

4.2.5 Sedmisegmentové displeje

Sedmisegmentové displeje jsou červené barvy a se společnou anodou. Podle dokumentace procházející proud jedním rozsvíceným segmentem je 30 mA. Na plně rozsvícený jeden displej je tedy potřeba 240 mA. Proud samozřejmě může být nižší a tím bude snížen i jas displeje. Za dostatečný je považován už 10 mA. Mezi každým párem digitů se nachází dvě červené LED diody, pro vytvoření oddělení hodin, minut a vteřin.



Obr. 4.8 – Připojení sedmissegmentových displejů

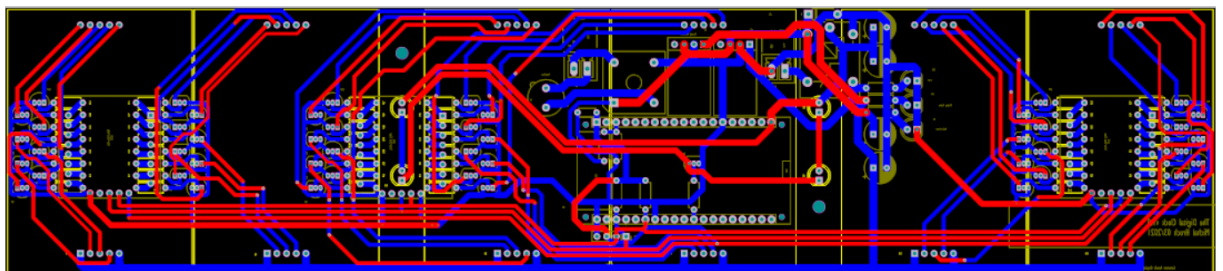
4.3 DESKA PLOŠNÝCH SPOJŮ

Návrh a výroba desky plošných spojů se skládaly ze dvou částí. V první části bylo vytvořeno schéma. Poté následovalo zkoušení a ověřování funkcí na nepájivém poli při použití jen jednoho displeje. Pokračovalo to postupným přidáváním jednotlivých digitů spolu s expandéry, zvětšováním kódu knihovny pro ovládání a adresování jednotlivých digitů až po celkovou úpravu programu. Po částečném oživení na nepájivém poli přišlo na řadu vytvoření samotného tištěného spoje. Vzhledem k limitovanému prostoru a množství součástek bylo zvoleno navrhnutí oboustranného plošného spoje. Ten byl navržen pro vývodové součástky. Sekce napájení byla umístěna nad řídicí jednotku na zadní straně, aby světelná signalizace stavu napájení vizuálně nepřekážela. Byly v ní použity dva moduly step-down, kondenzátory a příprava na budoucí rozšíření v podobě powerbanky. Z řídicí části byly vyvedeny cesty pro expandéry a obvod RTC. Z expandérů byly vyvedeny cesty pro velké množství tranzistorů, které byly umístěny tak, aby jejich vývody končily pod jednotlivými digity spolu s rezistory. Dále jsou zde umístěny dvě LED diody, které oddělují digity. Pro obvod RTC nebylo na desce

zvoleno pouzdro odpovídající jeho velikosti, aby bylo možno tento obvod umístit na místo, kde je potřeba měřit teplotu, bez nutnosti přesouvat celé hodiny.



Obr. 4.9 – Vyleptaný a osazený plošný spoj



Obr. 4.10 – Návrh plošného spoje

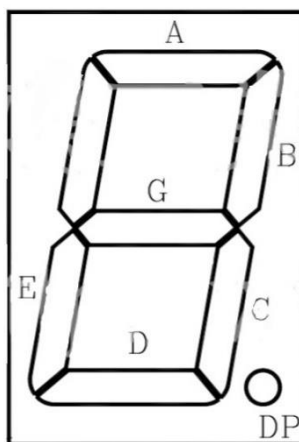
4.4 OVLÁDÁNÍ DISPLEJŮ

Aby bylo možno na displejích zobrazovat znaky, bylo potřeba vytvořit knihovnu, ve které se bude nacházet kombinace všech použitelných variant pro každý displej. Tyto znaky, označené jako zobrazovaná hodnota, se skládají z osmi segmentů na digitu, které bylo třeba nastavit. Jednotlivé označení segmentů vychází s datasheetu daných displejů a v kódu jsem značení dodržel, abych předešel případným nesrovnalostem. Ve vytvořené knihovně platí, že pokud je uvedeno pro danou zobrazovanou hodnotu na segmentu HIGH, tak daný segment je rozsvícen.

Tabulka 4.1 - Řídící tabulka pro ovládání displejů

Zobrazovaná hodnota	Segment							
	A	B	C	D	E	F	G	DP
0	HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	LOW	LOW
1	LOW	HIGH	HIGH	LOW	LOW	LOW	LOW	LOW
2	HIGH	HIGH	LOW	HIGH	HIGH	LOW	HIGH	LOW
3	HIGH	HIGH	HIGH	HIGH	LOW	LOW	HIGH	LOW
4	LOW	HIGH	HIGH	LOW	LOW	HIGH	HIGH	LOW
5	HIGH	LOW	HIGH	HIGH	LOW	HIGH	HIGH	LOW
6	HIGH	LOW	HIGH	HIGH	HIGH	HIGH	HIGH	LOW
7	HIGH	HIGH	HIGH	LOW	LOW	LOW	LOW	LOW
8	HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	HIGH	LOW
9	HIGH	HIGH	HIGH	LOW	LOW	HIGH	HIGH	LOW
Vypnutý digit	LOW	LOW	LOW	LOW	LOW	LOW	LOW	LOW
Znak stupně	HIGH	HIGH	LOW	LOW	LOW	HIGH	HIGH	LOW
C	HIGH	LOW	LOW	HIGH	HIGH	HIGH	LOW	LOW

Na Obr. 4.11 je uvedeno rozložení jednotlivých segmentů. Lze si všimnout, že na displejích se nachází i tečky, které nejsou využívány, ale jsou již připraveny na případné budoucí použití, například pro datum (Bright LED Electronics Corp, 2022).



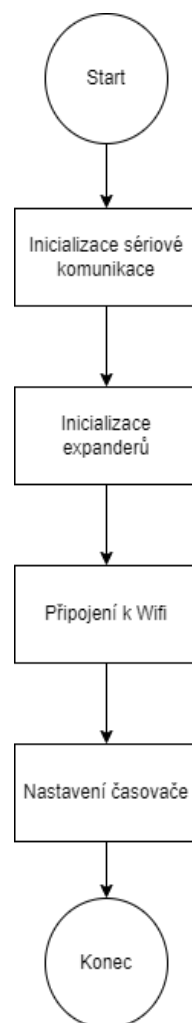
Obr. 4.11 – Umístění segmentů na displeji

Každý pár displejů je přes NPN tranzistory připojen na jeden expandér, který je připojen přímo k mikrokontroléru. Ten adresuje jednotlivým expandérům nastavení jejich výstupů a díky tomu se otevrou nebo zavřou tyto tranzistory, které ovládají jednotlivé segmenty na digitech.

4.5 PROGRAMOVÁ ČÁST

Celý kód je psaný v programu Arduino IDE 1.8.19. Program má dvě části, kde v části Init dochází k inicializaci funkcí a v části Main Loop k zobrazování času a teploty.

V sekci Init dochází především k inicializaci. Nejprve se inicializuje sériová komunikace a poté komunikace s expandéry. Poté se zařízení připojí k Wifi, povolí a nastaví časovač. Zde jsem z části využil veřejných knihoven jako je PCF8575.h, WiFi.h nebo Wire.h. Jedná se o knihovny, které jsou vytvořené pro práci při těchto příležitostech.



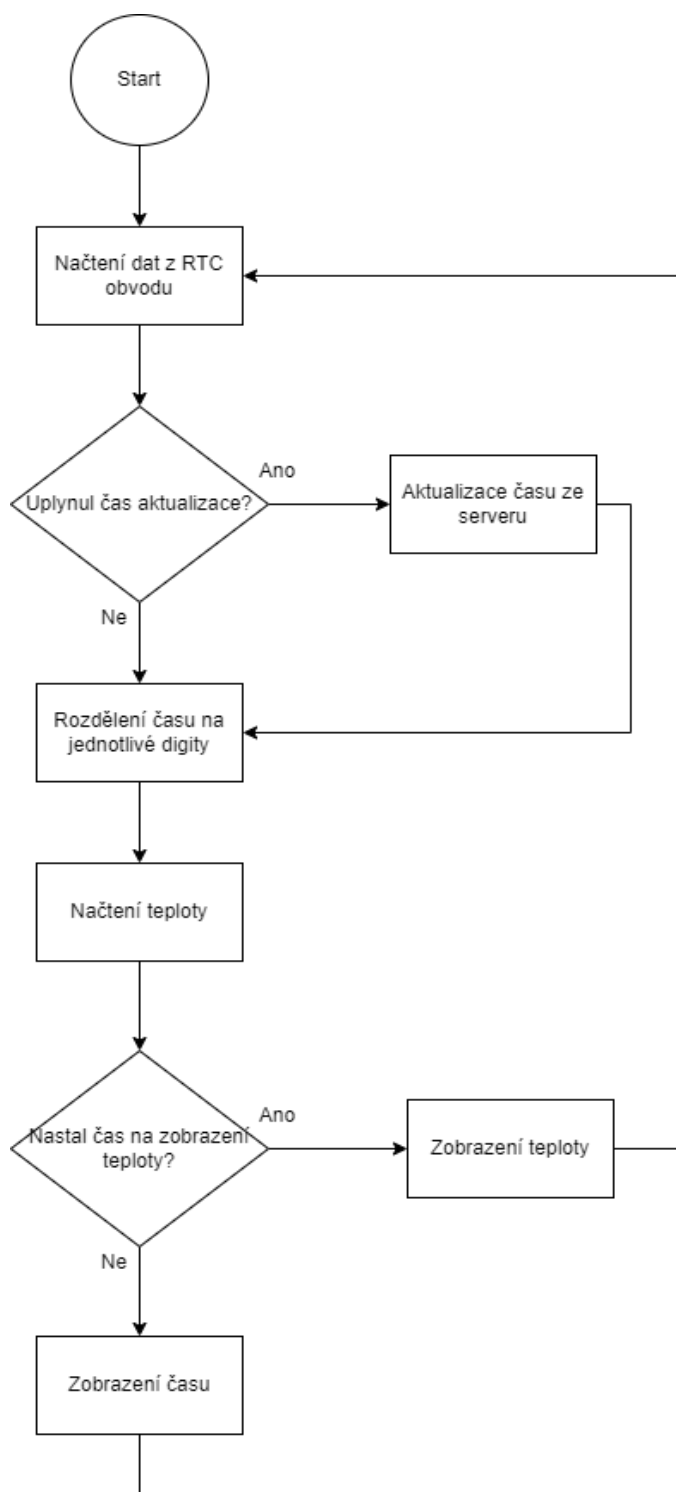
Obr. 4.12 – Vývojový diagram Init části

4.5.1 Sekce Main Loop

V této sekci se nachází nekonečná smyčka, která udržuje zařízení v chodu. Tato sekce nastává po konci sekce Init. Nejprve se zde načtou data z obvodu RTC. Následuje podmínka, která porovnává hodnotu Counteru, jenž počítá uběhnutý čas, s hodnotou SyncSec. Ta určuje čas mezi aktualizacemi. Pokud je Counter stejný nebo větší, nastává aktualizace času ze serveru.

V mém případě se jedná o server `cz.pool.ntp.org`. Pokud Counter nedosáhl ještě hodnoty pro aktualizaci času, data z RTC se rozdělí pro jednotlivé digity. Pak nastává načtení teploty z obvodu DS3231 a následné omezení teplotního rozsahu. To je zde zakomponováno ze dvou důvodů. Prvním je, že tyto hodiny zobrazují pouze kladná čísla. Znak pro mínus není v knihovně pro segmenty naprogramován. Druhým důvodem je, že zobrazují pouze dva znaky hodnot teploty. Hodiny jsou určeny na pokojové použití a trojmístné číslo zde není předpokládáno. Následuje další podmínka, která řeší, zda již nastal čas pro zobrazení teploty. Pokud nastal, zobrazí se teplota a celý program se vrátí na začátek k načtení dat z obvodu RTC.

Pokud tento čas ještě nenastal, zobrazí se pouze čas a program se stejně jako v prvním případě vrátí na začátek k načtení dat z obvodu RTC.



Obr. 4.13 – Vývojový diagram Main Loop části

4.6 OŽIVENÍ, OVĚŘENÍ FUNKCE, OPTIMALIZACE

První pokus o oživení proběhl až po celkovém zapájení. Následně byl nahrán program pro zobrazování času. Pro ověření funkce byly hodiny připojeny na laboratorní zdroj a udržovány v nepřetržitém chodu jeden den. Po zkontrolování přesnosti byl čas totožný se zobrazovaným na serveru. Při kontrole teploty se ani ta téměř nelišila od teploty, kterou ukazoval termostat v místnosti. Následně byly hodiny vystaveny znovu celodennímu běhu, během kterého došlo pomocí zdroje k náhodnému vypínání a znovu zapínání. Během této zkoušky již byly nalezeny nedostatky, jako třeba zpoždění nebo vynulování hodin. U části kódu došlo tedy k přepsání, což vyřešilo tyto problémy.

4.7 BUDOUCÍ ROZŠÍŘENÍ

Pro budoucí rozšíření je již na desce plošného spoje implementována odbočka pro připojení velké powerbanky, která by zvládla při výpadku proudu napájet nejen řídicí část s obvodem RTC, ale i samotné displeje. Dalším budoucím rozšířením může být vylepšení čtení teploty na bezdrátové, kdy čidlo by bylo venku a zasílalo by jen svůj údaj. Toto by šlo provést přes již vyvedenou sběrnici I2C, kam lze připojit přijímač a čidlo opatřit vysílačem. Případně čidlo připojit na síť a data by byla poskytnuta podobně jako časový údaj.

5 VÝROBEK

Výrobek je samostatná deska plošných spojů, která je tvořena napájecí částí, obvodem RTC, řídicím zařízením ESP32, třemi expandéry a šesti digity pro zobrazování.

Pro fungování výrobku je třeba ho doplnit vhodným zdrojem na napájení. Pro mé potřeby jsem využil laboratorní zdroj. Dále je potřeba v případě přemístění výrobku přepsat jeho přihlašovací údaje k Wifi a případně poupravit posun času vzhledem k časovému pásmu.

5.1 STAV VÝROBKU

Výrobek je dokončen a plně funkční. Obsahuje obvod RTC, který díky vestavěné baterii není závislý na napájení a který zároveň plní funkci teplotního čidla, mikrokontrolér ESP32 pro řízení hodin a šest velkých 57 mm sedmissegmentových displejů.



Obr. 5.1 – Dokončený výrobek

6 ZÁVĚR

Cílem této bakalářské práce bylo vytvoření velkoformátových hodin, které budou střídavě ukazovat teplotu s časem a budou schopny čas udržet i po ztrátě napájení. V průběhu realizace se mi povedlo navrhnout a sestrojít zařízení, které obsahuje řídicí mikrokontrolér, zobrazovací jednotky a modul reálného času. Ten je využit i jako teploměr. Deska plošných spojů byla navržena tak, aby nepřesáhla velikost digitů, následně byla vyleptána, osazena, oživena a ozkoušena její funkce. Deska dále obsahuje dva otvory o velikosti 2 mm na připevnění ke zdi. Hodiny lze pohodlně položit na rovný povrch a nespádnou díky rozteči mezi displeji a deskou plošného spoje. Všechny cíle tak byly splněny.

POUŽITÁ LITERATURA

- Agnihotri, N. 2021. Digital clock using Arduino, 7-segments, and MAX7219. *Engineers Garage* [online]. [cit. 2021-02-18]. Dostupné z: <https://www.engineersgarage.com/digital-clock-using-arduino-7-segments-and-max7219/>.
- BS-AD16RD. *Bright LED Electronics Corp.* [online]. [cit. 2022-03-23]. Dostupné z: <https://www.gme.cz/data/attachments/dsh.512-092.1.pdf>.
- Digital Clock Project Using 8 Digit 7 Segment MAX7219 Module. *Instructables Circuits* [online]. [cit. 2021-03-06]. Dostupné z: <https://www.instructables.com/Digital-Clock-Project-Using-8-Digit-7-Segment-MAX7>.
- DIY Digital Clock With 7 Segment LED Display. *Instructables Circuits* [online]. [cit. 2021-03-06]. Dostupné z: <https://www.instructables.com/DIY-Digital-Clock-With-7-Segment-LED-Display/>.
- Diy Seven Segment Display Clock, and MAX7219. *Arduino* [online]. [cit. 2021-02-18]. Dostupné z: https://create.arduino.cc/projecthub/e_s_c/diy-seven-segment-display-clock-a246f0.
- DUDÁČEK, K. 2002. Sériová rozhraní SPI, Microwire, I2C a CAN. *Západočeská univerzita* [online]. [cit. 2022-04-9]. Dostupné z: http://home.zcu.cz/~dudacek/NMS/Seriova_rozhrani.pdf.
- ESP32 and PCF8575 I/O expander example. *ESP32 learning* [online]. [cit. 2022-04-9]. Dostupné z: <http://www.esp32learning.com/code/esp32-and-pcf8575-i-o-expander-example.php>.
- ESP32 Series. *Espressif Systems* [online]. [cit. 2022-03-12]. Dostupné z: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf.
- ESP32 – WROOM. *Octopus Lab* [online]. [cit. 2022-04-9]. Dostupné z: <https://www.octopuslab.cz/esp32/>.
- GPSETH, GPSRS: Čas z GPS pro vaše zařízení. *DPS Elektronika od A do Z* [online]. [cit. 2021-02-18]. Dostupné z: <https://www.dps-az.cz/soucastky/id:68745/gpseth-gpsrs-cas-z-gps-pro-vase-zarizeni>.
- Interface DS3231 Precision RTC Module with Arduino. *Last Minute Engineers* [online]. [cit. 2022-04-23]. Dostupné z: <https://lastminuteengineers.com/ds3231-rtc-arduino-tutorial/>.
- Introduction of Basic Service Set (BSS). *GeeksforGeeks* [online]. [cit. 2022-03-12]. Dostupné z: <https://www.geeksforgeeks.org/introduction-of-basic-service-set-bss/>.
- LIŠKA, R. Vrstvy TCP/IP. *České vysoké učení technické v Praze* [online]. [cit. 2022-03-20]. Dostupné z: <http://www-troja.fjfi.cvut.cz/~liska/unix/node18.html>.
- MARCO, D. Drive Max7219/Max7221 with common anode displays. *Marco-difeo* [online]. [cit. 2021-03-04]. Dostupné z: <https://marco-difeo.de/2013/02/21/drive-max7219max7221-with-common-anode-displays/>.
- Model TCP/IP. *Internet a jeho služby* [online]. [cit. 2022-03-20]. Dostupné z: http://ijs2.8u.cz/index.php?option=com_content&view=article&id=14&Itemid=120.
- Network Time Protocol (NTP). *Tech Target* [online]. [cit. 2022-03-12]. Dostupné z: <https://www.techtarget.com/searchnetworking/definition/Network-Time-Protocol>.

- PCF8575. *Texas Instruments* [online]. [cit. 2022-03-12]. Dostupné z: <https://www.ti.com/product/PCF8575>.
- PCF8575 – Remote 16-bit I/O expander for ESP8266 – I2C-bus – P1. *Nextevo* [online]. [cit. 2022-03-12]. Dostupné z: <http://www.esp8266-projects.com/2016/06/p1-pcf8575-remote-16-bit-io-expander.html/>.
- POUPA, M. Vysílač časového signálu a normál. frekvence DCF77. *Západočeská univerzita* [online]. [cit. 2022-04-9]. Dostupné z: <http://home.zcu.cz/~poupa/pcf77.html>.
- Stručný popis sběrnice I2C a její praktické využití k připojení externí eeprom 24LC256 k mikrokontroléru PIC16F877. *HW server* [online]. [cit. 2021-07-22]. Dostupné z: <https://vyvoj.hw.cz/navrh-obvodu/strucny-popis-sbernice-i2c-a-jeji-prakticke-vyuziti-k-pripojeni-externi-eeeprom-24lc256>.
- TIŠNOVSKÝ, P. Komunikace po sériové sběrnici I2C. *Root* [online]. [cit. 2021-07-22]. Dostupné z: <https://www.root.cz/clanky/komunikace-po-seriove-sbernici-isup2supc/>.
- Základy řízení sedmissegmentových displejů bez speciálních obvodů. *Tajned* [online]. [cit. 2021-05-13]. Dostupné z: <http://www.tajned.cz/2015/04/zaklady-rizeni-sedmissegmentovych-displeju-rizeni-bez-specialnich-obvodu/>.
- Základy řízení sedmissegmentových displejů pomocí obvodu 74HC595. *Tajned* [online]. [cit. 2021-05-13]. Dostupné z: <http://www.tajned.cz/2015/04/zaklady-rizeni-sedmissegmentovych-displeju-rizeni-pomoci-obvodu-74hct595/>.
- 7-segment Display Counter. *Electronics Tutorials* [online]. [cit. 2021-05-13]. Dostupné z: <https://www.electronics-tutorials.ws/counter/7-segment-display.html>.

PŘÍLOHY

Příloha A – CD

Příloha B – Schéma

Příloha k bakalářské práci

Velkoformátové hodiny

Michal Hroch

CD

OBSAH

1. Text bakalářské práce ve formátu PDF
2. Úplný zdrojový kód aplikace pro mikrokontrolér

Příloha k bakalářské práci

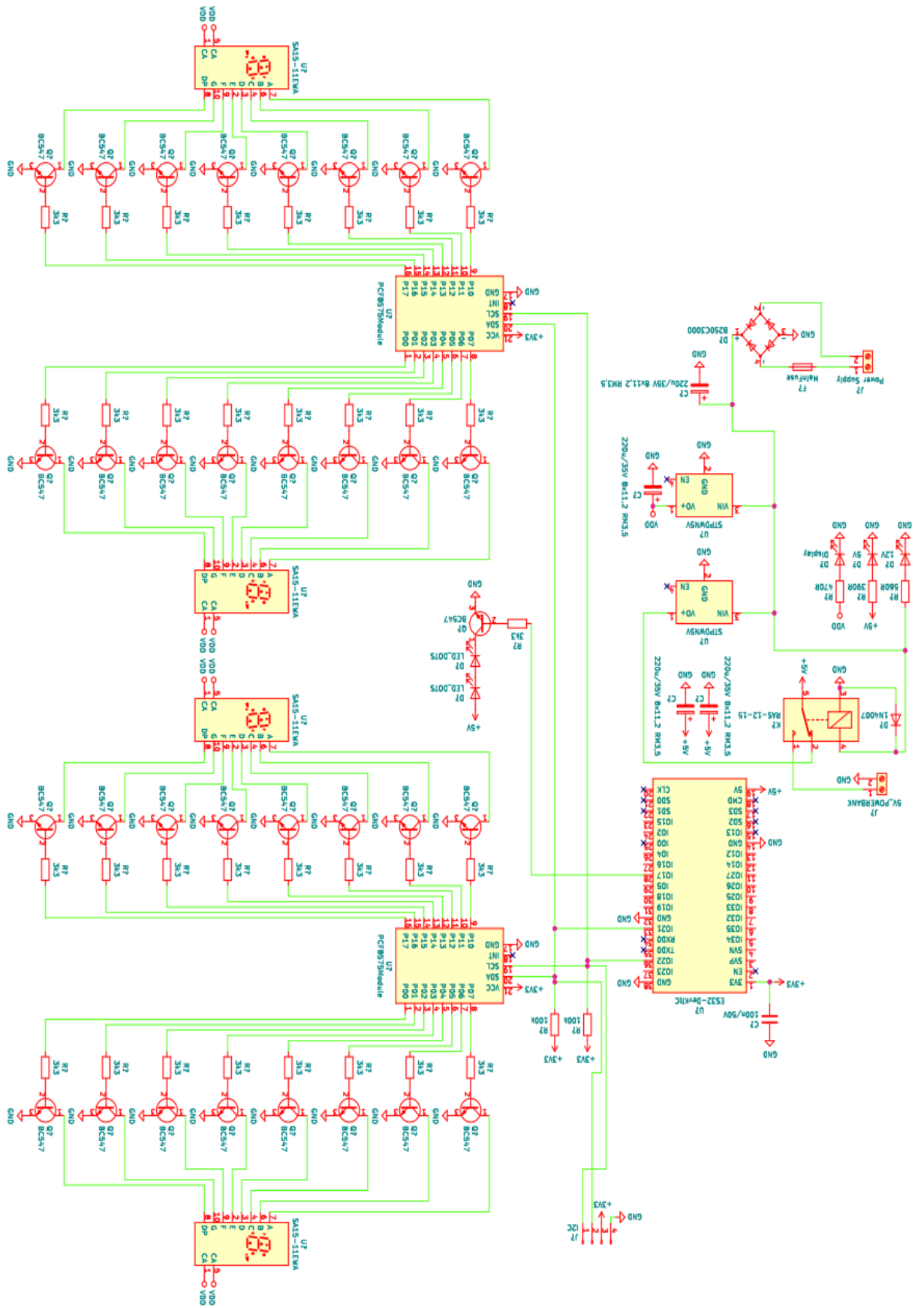
Velkoformátové hodiny

Michal Hroch

Schéma

SEZNAM ILUSTRACÍ

Obrázek 1 - Schéma.....	1
-------------------------	---



Obrázek 1 - Schéma