

UNIVERZITA PARDUBICE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

BAKALÁŘSKÁ PRÁCE

2017

Ladislav Kotlár

Univerzita Pardubice
Fakulta elektrotechniky a informatiky

Časový spínač
Ladislav Kotlár

Bakalářská práce

2017

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2016/2017

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Ladislav Kotlár**
Osobní číslo: **I13021**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Komunikační a mikroprocesorová technika**
Název tématu: **Časový spínač**
Zadávací katedra: **Katedra elektrotechniky**

Z á s a d y p r o v y p r a c o v á n í :

Navrhněte a realizujte časový spínač řízený mikroprocesorem. Systém dovoluje spínat zařízení zapojené do zásuvky 230 V/6 A po 15minutových intervalech, pomocí tlačítek a displeje lze volit časy zapnutí/vypnutí, kromě toho je pomocí zvláštního tlačítka možné provést sepnutí okamžitě. Napájení systému je zajištěno přímo ze zásuvky, spínání zátěže je prováděno triakem, časování musí být nezávislé na síti (vytvořte zálohovaný zdroj hodin reálného času).

Rozsah grafických prací:

Rozsah pracovní zprávy: **40 stran A4**

Forma zpracování bakalářské práce: **tištěná**

Seznam odborné literatury:

MATOUŠEK, David. Aplikace mikrokontrolérů ATmega644. 1. vyd. Praha: BEN - technická literatura, 2013, ca 200 s. v různém stránkování. ISBN 978-80-7300-492-7.

VÁŇA, Vladimír. Mikrokontroléry ATMEL AVR: programování v jazyce C : popis a práce ve vývojovém prostředí CodeVisionAVR C. 1. vyd. Praha: BEN - technická literatura, 2003, 215 s. ISBN 80-730-0102-0.

Vedoucí bakalářské práce: **Ing. Bc. David Matoušek**
Katedra elektrotechniky

Datum zadání bakalářské práce: **31. října 2016**

Termín odevzdání bakalářské práce: **12. května 2017**



Ing. Zdeněk Němec, Ph.D.
děkan



L.S.



Ing. Jan Pidanič, Ph.D.
vedoucí katedry

V Pardubicích dne 15. listopadu 2016

Prohlášení autora

Prohlašuji, že jsem tuto práci 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.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 28. 8. 2017

A handwritten signature in blue ink, appearing to read 'Kollár'.

jméno a příjmení autora

PODĚKOVÁNÍ

Rád bych poděkoval své rodině za podporu při studiu a mému vedoucímu práce panu Ing. Bc. Davidu Matouškovi, za jeho trpělivost a odborný dohled nad touto prací.

ANOTACE

Tato bakalářská práce se zabývá návrhem a realizací časového spínače. V teoretickém bloku jsou popsány pojmy, které souvisí s časovačem. V praktickém bloku je popsáno stavění, programování a výběr komponent časovače.

.

KLÍČOVÁ SLOVA

Časovač, ATMEGA, DPS, I2C, SPI,RTC, triak

TITLE

The Timer

ANNOTATION

This bachelor work describes the design and implementation the timer. The theoretical part explains concepts, which are related with the timer. The practical part is described building, programming and choiseing componts of of timer.

KEYWORDS

timer, ATMEGA, DPS, I2C, SPI, RTC, triac

1 Obsah

Úvod.....	10
2 teoretický blok	11
2.1 Periferie	11
2.1.1 Tlačítka	11
2.1.2 Display	14
2.2 Napájecí zdroj	18
2.3 Triak a spínání.....	19
2.3.1 Fázově řízený triak.....	19
2.3.2 Cyklově řízený triak.....	20
2.3.3 Galvanicky oddělený spínač střídavého napětí.....	21
2.4 RTC.....	22
3 Praktická část	23
3.1 Obecné řešení	23
3.2 Schémata pro tvorbu DPS	23
3.2.1 Schéma výkonové DPS - zdroje	24
3.2.2 Schéma výkonové DPS - relé	25
3.2.3 Schéma výkonové DPS cela	26
3.2.4 Schéma logické DPS.....	26
3.2.5 Schéma logické DPS - tlačítka.....	27
3.2.6 Schéma Logická DPS – MCU a periferie	29
3.3 Návrhy DPS	30
3.4 Výběr komponent.....	32
3.5 Popis programu	39
3.5.1 Popis kódu programu	39
4 Závěr	42
Použitá literatura.....	43

5	SEZNAM PŘÍLOH.....	45
---	--------------------	----

SEZNAM ILUSTRACÍ A TABULEK

Obrázek 1.	Odporová síť tlačítek.....	12
Obrázek 2.	Bez kmitavé tlačítko.....	13
Obrázek 3.	Připojení zařízení na sběrnici I2C.....	15
Obrázek 4.	Detail připojení vývodu obvodů na sběrnici I2C.....	15
Obrázek 5.	a) Přenos bitů, b) START a STOP stavy.....	16
Obrázek 6.	Ukázka možného zapojení tří Slave obvodů na SPI sběrnici.....	17
Obrázek 7.	Blokové schéma napájecího zdroje.....	18
Obrázek 8.	Zjednodušená struktura a schématická značka triaku.....	19
Obrázek 9.	Obvod pro fázové řízení s triakem.....	20
Obrázek 10.	Princip cyklového řízení.....	20
Obrázek 11.	Obvod pro cyklové řízení s triakem.....	21
Obrázek 12.	Galvanicky oddělený spínač střídavého napětí.....	21
Obrázek 13.	Schéma zdroje oblasti DPS.....	24
Obrázek 14.	Schéma spínací oblasti- relé.....	25
Obrázek 15.	Schéma výkonové DPS.....	26
Obrázek 16.	Schéma logické DPS.....	26
Obrázek 17.	Schéma zapojení tlačítek.....	27
Obrázek 18.	Výsledné hodnoty veličin tlačítek.....	28
Obrázek 19.	Schéma zapojení MCU a periférii.....	29
Obrázek 20.	Preentscreen CAD editoru EAGLE.....	30
Obrázek 21.	Návrh logické DPS.....	30
Obrázek 22.	Návrh výkonové DPS.....	31
Obrázek 23.	Plastový box KP14.....	32
Obrázek 24.	Alfanumerický displej MC1604C-SYL.....	33

Obrázek 25.	Převodník I2C	33
Obrázek 26.	Tlačítko P-DT6RT	34
Obrázek 27.	ATMEGA 328PU + ATMEGA 32A	34
Obrázek 28.	Transformátor.....	35
Obrázek 29.	Chladič	36
Obrázek 30.	Optočlen	37
Obrázek 31.	Triak	37
Obrázek 32.	Foliové kondenzátory	38
Obrázek 33.	DS1302 jeho vnitřní zapojení.....	39
tabulka č.1.	Tabulka pinů a jejich významu	14
tabulka č.2.	Tabulka Teplot	31

SEZNAM ZKRATEK A ZNAČEK

RTC	Real time clock
A/D	Analogově digitální
MCU	Mikroprocesor
SPI	Seriál Peripheral Interface
I2C	Inter-Integrated CircuitV
DPS	Deska plošných spojů

ÚVOD

Běžný uživatel se ve svém životě setká se spínači každý den. Může se jednat o jednoduchý spínač osvětlení v pokoji, budík, nebo nastavení doby praní špinavého prádla v automatické pračce. Zařízení využívající nějaký druh spínače je velké množství.

Cílem této bakalářské práce je navrhnout a zkonstruovat časový spínač řízený mikroprocesorem. Systém dovoluje spínat zařízení zapojené do zásuvky 230V/8A po 15. minutových intervalech. Systém je rozdělen do několika prvků: logického obvodu, výkonového obvodu, RTC, a modulu displeje.

Bakalářská práce je rozdělena na dvě velké bloky, které jsou dále rozděleny do několika úrovní. V teoretickém bloku je vysvětlena činnost jednotlivých stavebních prvků výše zmíněných obvodů.

V praktickém bloku je vysvětlen návrh a tvorba desek plošných spojů pro logický obvod a výkonový obvod. Vybráních konkrétních modulů a následně vysvětlen kód, psaný v jazyce C.

2 TEORETICKÝ BLOK

S ohledem na zadání bakalářské práce. Kde má být navrhnout systém řízený mikroprocesorem. Systém dovoluje spínat zařízení připojené do zásuvky 230V/6A, pomocí tlačítek a displeje lze volit časy zapnutí / vypnutí, spínání je provedeno triakem a je použit zálohovaný zdroj RTC. V teoretickém bloku jsou uvedeny možné způsoby požití výše uvedených periférií

2.1 Periferie

2.1.1 Tlačítka

Tlačítka k mikroprocesoru můžeme připojit dvěma základními způsoby:

- Prvním způsobem je čtení analogové čtení hodnoty napětí procházející obvodem, na kterém jsou tlačítka.
- Druhým způsobem je „digitální čtení“, tlačítko se zapojí buď proti zemi, nebo proti napájení a na pinu mikroprocesoru se vyhodnocuje, v jakém logickém stavu se nacházíme.

Analogové čtení

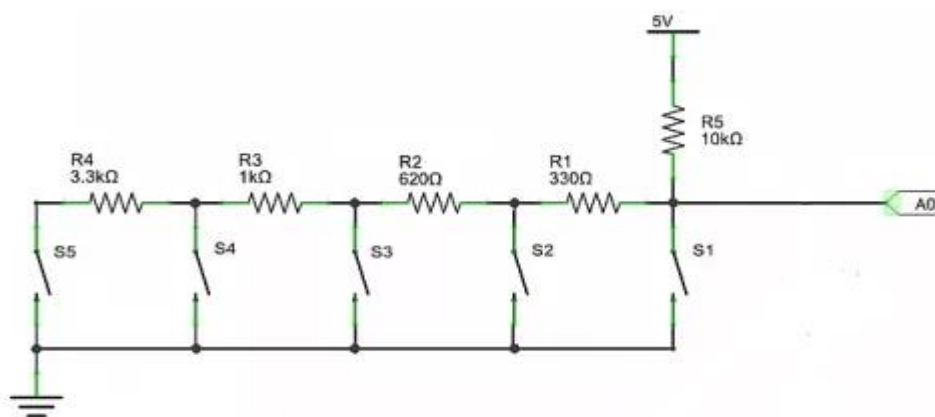
Na obrázku níže je zobrazeno schéma zapojení tlačítek při čtení hodnoty napětí. Jak vidíme, přivedeme k tlačítkům napájení (5V) a zem. Napájení je stejné, jako napájení pro námi zvolený mikroprocesor. Poblíž každého tlačítka je umístěn rezistor, jehož funkci je vytvořit úbytek napětí, při stisknutí tlačítka. Potom vidíme, že ve spodní části obvodu se nachází tzv. uzemňovací rezistor. Úkolem toho rezistoru je, udržet hodnotu napětí rovno nule, pokud není stisknuté žádné tlačítko. Výhodou tohoto zapojení je úspornost pinu mikroprocesoru. Na jeden pin jsme schopni připojit relativně velké množství tlačítek. Jediný faktor, který nás může brzdit v množství tlačítek je dostatečně velký rozdíl hodnot napětí jednotlivých tlačítek. Pokud by rozdíl byl příliš malý tak, by mohli k špatnému vyhodnocení stisknutí tlačítka.

I při analogovém čtení dochází k zákmitům, avšak tyto zákmity jsou zanedbatelné. Nemají vliv na chybnost stisknutí tlačítek. Tyto zákmity bývají pro jistotu řešeny programově pomocí rozmítání čtené analogové hodnoty tlačítka.

Čtení hodnotí napětí je realizováno uvnitř mikroprocesoru, a to pomocí tzv. A/D převodníku. A/D převodník může pracovat ve dvou režimech, které si můžeme zvolit při programování mikroprocesoru. Režimy se nazývají zarovnání doleva a zarovnání doprava.

Pokud zvolíme režim zarovnání doleva, A/D převodník pracuje jako 8. bitový, tedy rozlišuje 256 stavů. Pokud zvolíme režim zarovnání doprava, A/D převodník pracuje jako 10. bitový, tedy rozlišuje 1024 stavů.

Jak vidíme na obrázku níže, rezistory mají různé hodnoty, takže na nich vznikají různé úbytky napětí. Z výše uvedeného jasně vyplývá, že stisknutím různých tlačítek se vygeneruje různá velikost napětí, které budou čteny pomocí MCU a my tak poznáme, které tlačítko bylo stisknuto.



1

Obrázek 1. Odporová síť tlačítek

Digitální čtení

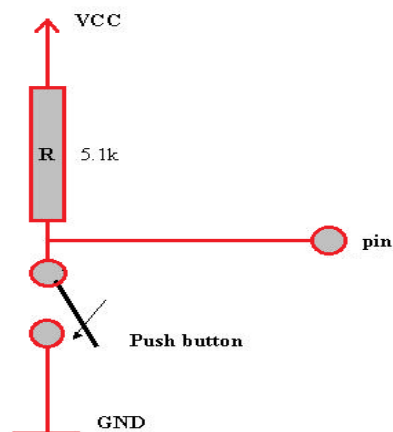
Tlačítko se zapojí:

- proti zemi,
- proti napájení.

Na pinu mikroprocesoru se vyhodnocuje, v jakém logickém stavu se nacházíme. Takto zapojení tlačítko, by mohlo být chybně vyhodnoceno. Při stisknutí tlačítka dochází k tzv. záskmitům, a tlačítko může být vyhodnoceno špatně. (stisknuto vícenásobně, nebo nestisknuto vůbec).

¹ G7smy.co.uk: *MULTIPLE BUTTONS ON ONE PIN WITH AN ARDUINO* [online]. [cit. 2017-08-27]. Dostupné z: <http://www.g7smy.co.uk/2015/09/multiple-buttons-on-one-pin-with-an-arduino/>

Nedostatek tohoto zapojení lze odstranit více způsoby. První způsob je úpravou zapojení.



Obrázek 2. Bez kmitavé tlačítka

Jak vidíme na obrázku výše pro vytvoření bez zamítavého tlačítka stačí přidat do předchozího obvodu dolní propust RC článkem a Schmidtův klopný obvodem.

Výpočet hodnot rezistoru a kondenzátoru je závislý mezní frekvenci námi vytvořené dolní propusti. Volíme frekvenci mezi 50 – 100 Hz. Hodnoty rezistoru a kondenzátoru lze vypočítat pomocí následujícího vztahu.

$$f = \frac{1}{2\pi.R.C}$$

kde,

f – je výsledná mezní frekvence,

π – je Ludolfovo číslo 3,14,

R – je velikost rezistoru,

C - je velikost kondenzátoru.

Druhá možnost ošetření zákmitů je pomocí vhodně napsaného kódu programu. Detekce stisku tlačítka je ošetřena časovou prodlevou a zjištěním, z kterého logického stavu nastává změna. Názorně vysvětleno na příkladu při stisku tlačítka hledáme přechod logických stavů, např. z log. 0 na log. 1. Délka časové prodlevy může být pokaždé jiná.

2.1.2 Display

Většina alfanumerických displejů používá řadič HD44780 od firmy Hitachi. Tento řadič obsahuje znakovou sadu, takže programátor nemusí řídit buzení jednotlivých pixelů. Stačí pouze napsat znak a řadič sám rozsvítí příslušné pixely.

Máme několik možností jak připojit display k mikroprocesoru. Zde budou vysvětleny tři nejpoužívanější možnosti – „klasická“, pomocí rozhraní I2C, pomocí rozhraní SPI,

Klasická

V tabulce uvedené níže, vidíme přehled všech pinů displeje. Blíže si vysvětlíme význam jednotlivých pinů a jejich použití.

Na displeji se nachází 8 pinů, jejichž funkcí je datová sběrnice, dále pin RS rozhoduje, zda se jedná o data nebo instrukce pro řadič, dále pin R/W podle jeho nastavení rozlišujeme mezi zapisováním a čtením dat displeje, a pin E, jehož funkcí je vzorkování dat do ostatní pinů.

Jestliže je pin RS nastaven na logickou 1, potom přenáší znaky k zobrazení nebo k uložení do paměti. V opačném případě jsou přenášeny instrukce.

Displej umožňuje komunikaci po 8 – bitové, nebo 4 - bitové datové sběrnici. Záleží na uživateli, jak je jej zapojí. Zapojením pinů B4,B5,B6,B7 dosáhneme 4 - bitové datové sběrnice. Připojíme-li ke stávajícím pinům piny B0,B1,B2,B3 dosáhneme 8 - bitové datové sběrnice.

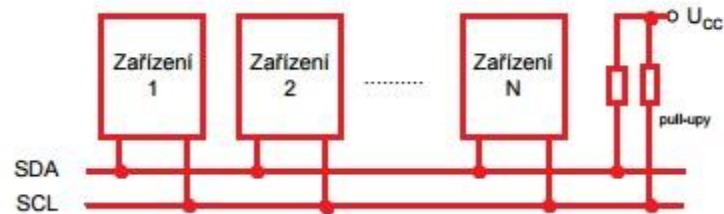
Následující dva způsoby připojení modulu displeje k mikroprocesoru budou vysvětleny pomocí sběrnic.

pin	označení	význam	Pin	Označení	význam
1	VSS	Zem	9	B2	Bit 2
2	VCC	Napájecí napětí	10	B3	Bit 3
3	V0	Nastavení kontrastu LCD	11	B4	Bit 4
4	RS	Výběr registru	12	B5	Bit 5
5	RW	Směr komunikace	13	B6	Bit 6
6	E	Povolení / hodiny	14	B7	Bit 7
7	B0	Bit 0	15	BL+	Anoda pod světlení
8	B1	Bit 1	16	BL-	Katoda pod světlení

tabulka č.1. Tabulka pinů a jejich významu

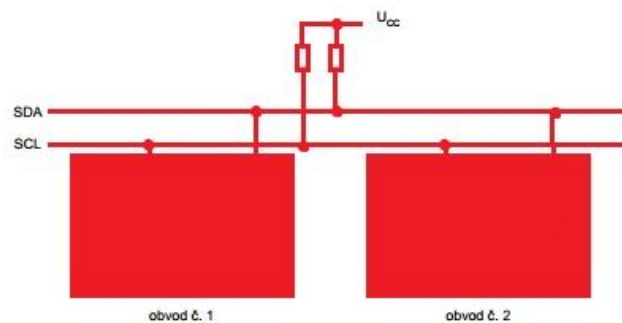
I2C

I2C je dvou vodičová sériová synchronní multimaster sběrnice, která se používá pro připojení periférii, u kterých nepotřebujeme velkou přenosovou rychlost. Tato sběrnice byla vyvinuta firmou Philips.



Obrázek 3. Připojení zařízení na sběrnici I2C

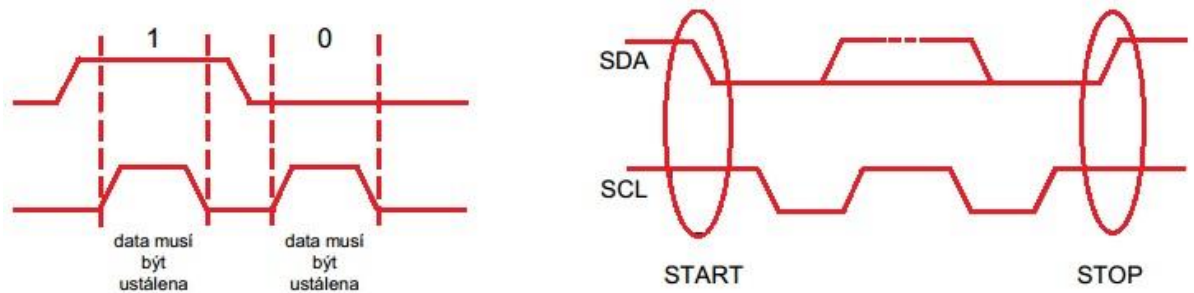
Linka SDA slouží pro přenos dat, linka SCL- jsou hodiny (CLK), které jednoznačně určují kdy přijímač číst stav datové linky. Všechna zařízení připojená přes sběrnici I2C tyto vodiče sdílejí a mohou k nim přistupovat. Obecně by se dalo říci, že je možné, že by mohl kdokoli s kýmkoli komunikovat pomocí tohoto rozhraní. V praxi to není tak jednoduché, narážíme na přílišnou náročnost a komplikovanost na programování.



Obrázek 4. Detail připojení vývodu obvodů na sběrnici I2C

Samotný komunikační protokol není nijak extrémně složitý. Na každé z linek je připojen tzv. zdvihací rezistor (pull-up rezistor), jenž zajistí, že je linka v neutrálním stavu ve stavu log. 1. Všechna zařízení na těchto linkách komunikují pomocí tzv. otevřeného kolektoru. To znamená, že mohou linku pevně stáhnout k zemi a udělat na ní log. 0. Pokud je potřeba linku nastavit do stavu log. 1, všechna zařízení se od ní odpojí a nechají “pull-up” rezistor, aby sběrnici nastavil na napájecí napětí. Díky výše popsanému principu nemůžou vzniknout na sběrnici nebezpečné kolize

Nejjednodušší zpráva se pak skládá ze START sekvence, osmi bitů, acknowledge bitu a STOP sekvence. Příkladovou zprávu vidíme na obrázku níže.



Obrázek 5. a) Přenos bitů, b) START a STOP stavy

I2C Protokol

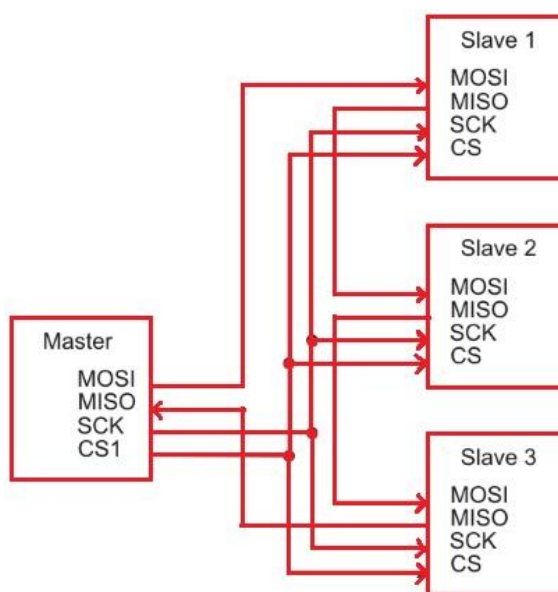
Nejdříve musíme vyřešit otázku adresování. A až poté směr toku dat po sběrnici. První byte odeslaný po START sekvenci, rozhoduje od jakého obvodu a k jakému obvodu budou data směřovat. Prvních sedm bitů z výše jmenovaných osmi, tvoří adresu a osmý bit nastavuje směr toku dat. Devátý bit slouží k potvrzení připravenosti komunikace mezi Slavem a Masterem. Pokud Slave nemůže reagovat, Master nedostane potvrzení (devátý bit bude nastaven na hodnotu log. 1). Slave čeká instrukce od Mastera. Pokud Master vynuluje osmý bít, poskytne Slavovi informaci o tom, že bude vysílat. Vysílání trvá až do konce komunikace. Po obdržení této informace Slave ví, že nijak nemá zasahovat do datové linky, a že má přijímat. Pokud Master odešle osmý bit v hodnotě log. 1, Slave dostane informaci, že má vysílat a master bude přijímat. Komunikace stále pokračuje dokud Master nepotvrdí tzv. acknowledge bit. Pokud je acknowledge bit potvrzen komunikace končí.

I2C Adresování

Jak už bylo popsáno výše adresa periférii připojených přes I2C je sedmibitová. Z tohoto faktu vyplývá, že může existovat pouze 127 unikátních adres, pokud tedy nepočítáme adresu 0. Počet adres je roven počtu druhů zařízení, které lze připojit pomocí I2C k mikroprocesoru. Další problém nastává při připojení více stejných zařízení například RTC DS3231. Ten má pevně přidělenou od výrobce (0x68). Nemůžeme tedy dva takové obvody připojit ke stejné sběrnici. Řešení tohoto problému přichází ze stranu výrobců. Při změně zapojení periferie můžeme nakonfigurovat její adresu, a tak docílit zapojení více stejných periférii na stejnou sběrnici.

SPI

Rozhraní SPI je nejjednodušší forma sběrnice. Rozhraní SPI (Seriál Peripheral Interface) slouží pro komunikaci mikroprocesorů mezi sebou, nebo s různými periferiemi, těmi mohou například být A/D a D/A převodníky, displeje, SD karty. Přenos SPI je duplexní, tzn., že se data posílají oběma směry zároveň. V některých případech může být simplexní tzn., že se data posílají pouze jedním směrem. Přenos pomocí rozhraní SPI je relativně rychlý, protože rozhraní SPI je synchronní. Všechny obvody sdílejí trojici linek. Hodiny (SCLK), které jsou generovány Masterem. Linku MOSI (Master Out Slave In) po které „tečou“ data z Masteru do Slave a linku MISO (Master In Slave Out) po které se posílají data ze Slave obvodu do Master. Každý Slave má vlastní CS (Chip Select) linku, jejichž funkcí je adresování. Master obsluhuje stejné množství CS linek, jako je Slave obvodů. Slave obvody mají 4 vývody (MOSI, MISO, SCK, CS). Pokud Slave obvod nemá vývod MOSI, potom není schopen data přijímat (pouze jen odesílat). Pokud linka MISO obsluhuje více Slave obvodů, potom musí Master v jednom okamžiku obsluhovat pouze k jeden Slave obvod. K tomuto účelu slouží CS linky ovládané Masterem. Pokud má Slave vstup CS nastaven na hodnotu log. 1, tak je neaktivní a udržuje svůj MISO vývod ve stavu vysoké impedance. Vysoká impedance zapříčiní, že daný Slave obvod nijak neovlivní stav MISO linky. Master pomocí CS linek vybírá, se kterým Slave obvodem chce komunikovat.²

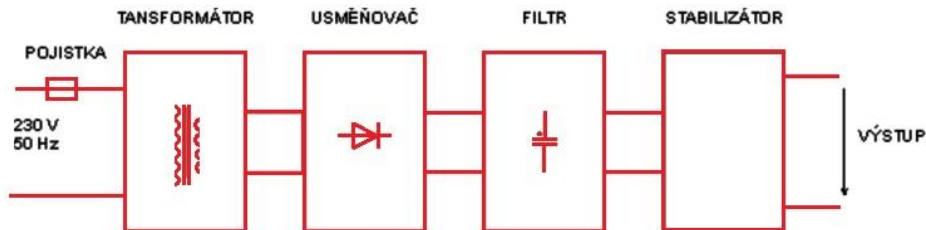


Obrázek 6. Ukázka možného zapojení tří Slave obvodů na SPI sběrnici

² Tajned.cz: SPI rozhraní [online]. [cit. 2017-08-28]. Dostupné z: <http://www.tajned.cz/2016/12/spi-rozhrani/>

2.2 Napájecí zdroj

Napájecí zdroj je jakéhokoliv elektronického zařízení. Zdrojem může být baterie, elektrická síť, nebo solární panel napájený slunečním zářením. Výhodou použití baterie je jednoduchost použití. Pro potřeby této bakalářské práce se dále budeme zabývat napájecím zdrojem na síťové napětí, tedy na 230 V.



Obrázek 7. Blokové schéma napájecího zdroje

Úkolem transformátoru je snížit velikost vstupního síťového napětí a galvanicky oddělit ostatní části napájecího zdroje od síťového rozvodu. Usměrňovač mění střídavé napětí na výstupu transformátoru na stejnosměrné napětí. Toto stejnosměrné napětí vstupuje do filtru, který zajistí, že výstupní napětí neklesá k nule. Na konci obvodu napájecího zdroje máme stabilizátor, který „udrží“ konstantní hodnotu výstupního napětí nezávisle na kolísání vstupu.

Úkolem usměrňovače je přeměna střídavého proudu na stejnosměrný. Podle způsobu činnosti usměrňovače lze rozdělovat na: neřízené, řízené a aktivní. Podle napájecí sítě lze usměrňovače rozdělovat na jednofázové, třífázové a vícefázové. Jednofázové usměrňovače dále dělíme na jednocestné a dvoucestné. Jednocestné se vyskytují ve dvou variantách: s kondenzátorem a bez kondenzátoru. Dvoufázové usměrňovače dále dělíme na uzlový usměrňovač s kondenzátorem a můstkový usměrňovač s kondenzátorem (Graetzův můstek s kondenzátorem).

Úkolem filtru je odstranění zvlnění, které se může objevit na výstupu usměrňovače. Jako filtr lze použít RC nebo LC články. Tyto články jsou filtry typu dolní propust.

Úkolem stabilizátoru je stabilizace výstupního napětí nebo proudu při změnách vstupního napětí a teploty. Ostatní veličiny většinou nehrají vliv na hodnotu výstupního napětí. Mezi veličiny, které by mohli mít vliv na velikost výstupního napětí, řadíme např. stárnutí součástek, elektromagnetické rušení atd.

2.3 Triak a spínání

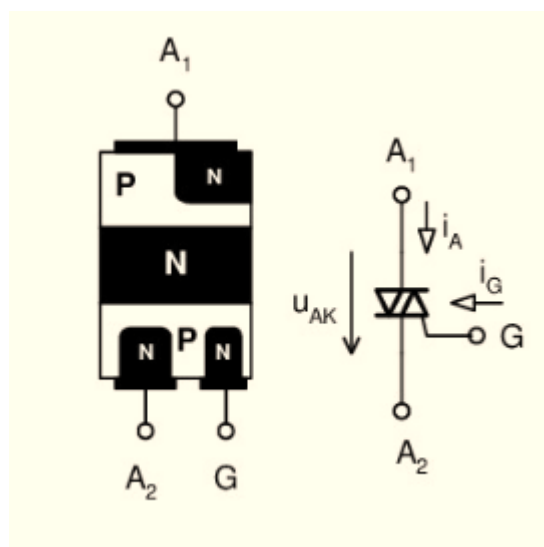
Pro provoz v obou půlperiodách je obecně výhodnější triak. Triak lze zjednodušeně označit jako antiparalelní spojení dvou tyristorových struktur PNP a NPN. Jedna struktura pracuje s kladným napětím a druhá se záporným napětím.

Spínání triaku může být prováděno třemi způsoby:

- Fázově řízený triak,
- Cyklově řízený triak,
- Galvanicky oddělený spínač střídavého napětí.³

2.3.1 Fázově řízený triak

Rozsah fázového řízení je teoreticky $\alpha = 0$ až 2π . Teoreticky proto, e spínací napětí diaku je obvykle okolo 30 V, takže od průchodu napájecího napětí nulou musí vždy uplynout určitý čas. Triakem pak nemůže téci proud po celou periodu, ale vždy po něco kratší interval.⁴

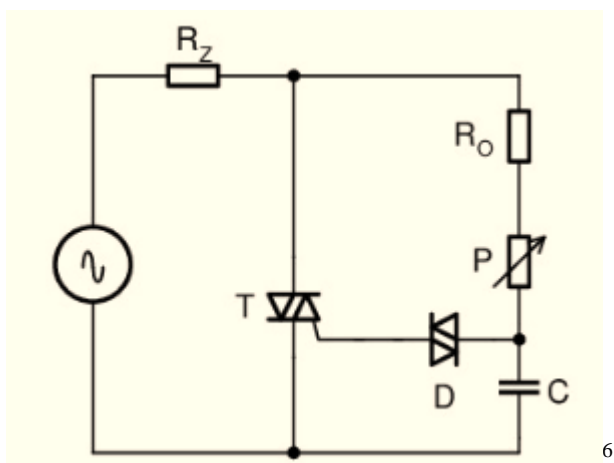


Obrázek 8. Zjednodušená struktura a schématická značka triaku

³ MATOUŠEK, David. *IEOBE - Elektrické obvody*.

⁴ MATOUŠEK, David. *IEOBE - Elektrické obvody*.

⁵ MATOUŠEK, David. *IEOBE - Elektrické obvody*.

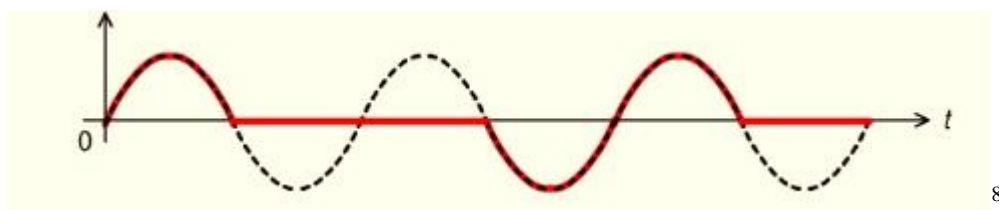


Obrázek 9. Obvod pro fázové řízení s triakem

2.3.2 Cyklově řízený triak

Provoz fázově řízených obvodů je velmi problematický a s ohledem na elektromagnetické rušení EMI. Ke spínání tyristoru nebo triaku nedochází při průchodu nulou, ale obecně při nenulové hodnotě síťového napětí. Fázově řízený regulátor pak vytváří v napájecí síti rušivé impulzy.

Jiný způsob řízení se nazývá cyklově řízení. Ke spínání dochází při průchodu nulou a vždy se propustí určitý celý počet půlperiod.⁷



Obrázek 10. Princip cyklového řízení

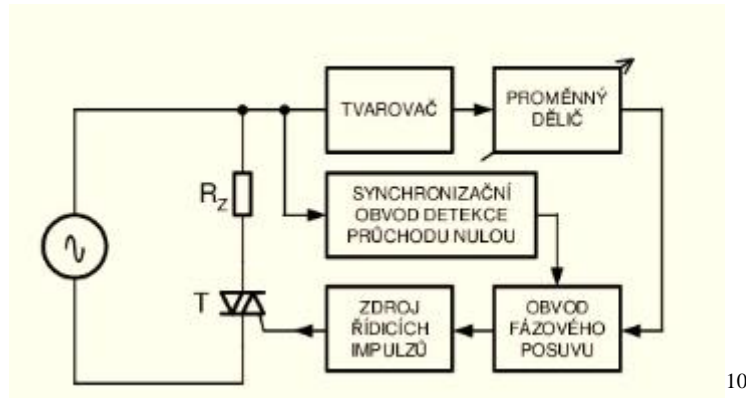
Tvarovač tvaruje vstupní sinusový signál získaný ze sítě na obdélkový průběh. Proměnný dělič propouští jeden impulz ze série impulzů a tím vlastně zajišťuje regulaci. Obvod fázového

⁶ MATOUŠEK, David. *IEOBE - Elektrické obvody*.

⁷ MATOUŠEK, David. *IEOBE - Elektrické obvody*.

⁸ MATOUŠEK, David. *IEOBE - Elektrické obvody*.

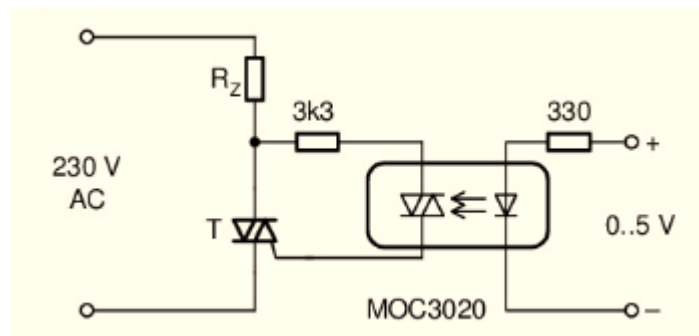
posuvu synchronizuje spouštěcí impuls s okamžikem, kdy napájecí napětí prochází nulou. Takto získaný impuls pak spouští triak.⁹



Obrázek 11. Obvod pro cyklové řízení s triakem

2.3.3 Galvanicky oddělený spínač střídavého napětí

Pro řízení zátěže v síťovém obvodu například z mikroprocesorového systému lze použít zapojení viz obrázek níže. Použitý optočlen MOC3020 má jako výstupní prvek fototriak (spíná při osvětlení řídicí elektrody světlem z integrované LED). Vstupní obvod je navržen pro 5. voltové napájení. Výstupní odvod optočlenu je přes rezistor o odporu 3,3 k napájen přímo z řízeného obvodu.¹¹



Obrázek 12. Galvanicky oddělený spínač střídavého napětí¹²

⁹ MATOUŠEK, David. *IEOBE - Elektrické obvody*.

¹⁰ MATOUŠEK, David. *IEOBE - Elektrické obvody*.

¹¹ MATOUŠEK, David. *IEOBE - Elektrické obvody*.

¹² MATOUŠEK, David. *IEOBE - Elektrické obvody*.

2.4 RTC

RTC (Real Time Clock) jak vypovídá zkratka, jedná se o hodiny reálného času. Měřit čas můžeme i bez použití RTC , ale použití RTC má tyto výhody:

- Nízká spotřeba,
- Vyšší přesnost vůči jiným řešením,
- Ušetří systémové prostředky.

3 PRAKTICKÁ ČÁST

V tomto bloku je vysvětlena konstrukce a programové řešení časového spínače

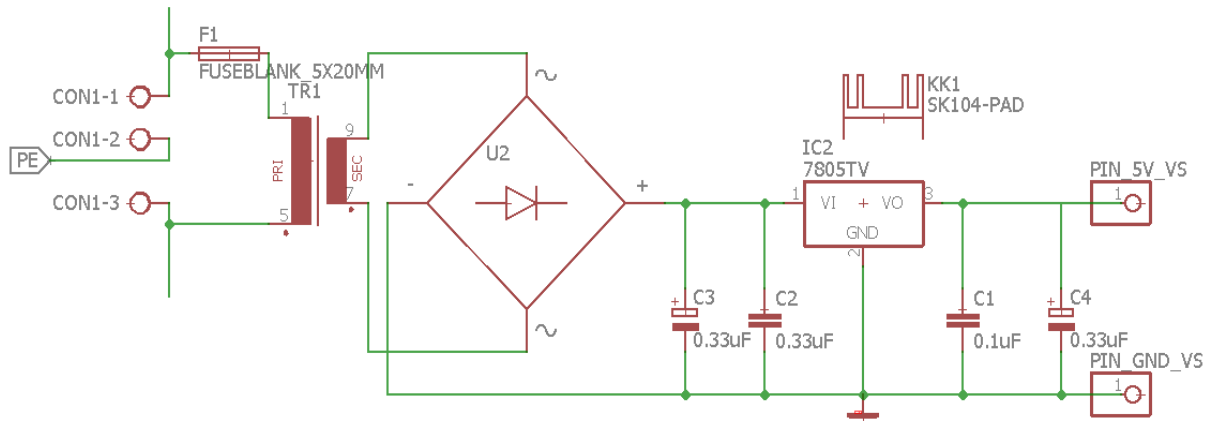
3.1 Obecné řešení

Celá aplikace je řízená mikroprocesorem ATMEGA 32A od firmy ATMEL. Při zapnutí výrobku dochází nejdříve k inicializaci komponent (A/D převodníku, RTC, LCD) a poté se na 4. řádkovém alfanumerickém displeji objeví uvítací dialog. Uvítací dialog zmizí po uplynutí předem zvoleného časového intervalu. Interval byl zvolen 1s. Po této akci se objeví první tzv. Základní menu, u kterého má uživatel možnost nastavení dalších akcí, jako jsou výběr modu spínání zátěže, spínání zátěže, a přesun do dalších menu. Tyto menu mají jiný design než základní menu a slouží k nastavení RTC, dob spínání, zobrazení tabulky spínání a vymazání paměti. K nastavení těchto akcí slouží šestice tlačítek, které jsou barevně rozlišené. Barvy tlačítek bylo zvoleny pro lepší rozpoznatelnost jejich funkce. Pro jednoduché úpravy programu, byl vyveden ISP konektor. K správné obsluze výrobku bude v přílohách obsažen manuál pro uživatele.

3.2 Schémata pro tvorbu DPS

Výrobek se skládá ze dvou větších bloků. Prvním blokem je výkonová deska plošných spojů. Výkonová deska plošných spojů je rozdělena na dvě části (zdrojová a spínací oblast-relé) pomocí galvanického oddělení. Důvodem galvanického oddělení, je že zdrojovou a spínací oblastí prochází napětí o různých velikostech. Druhým blokem je logická deska plošných spojů.

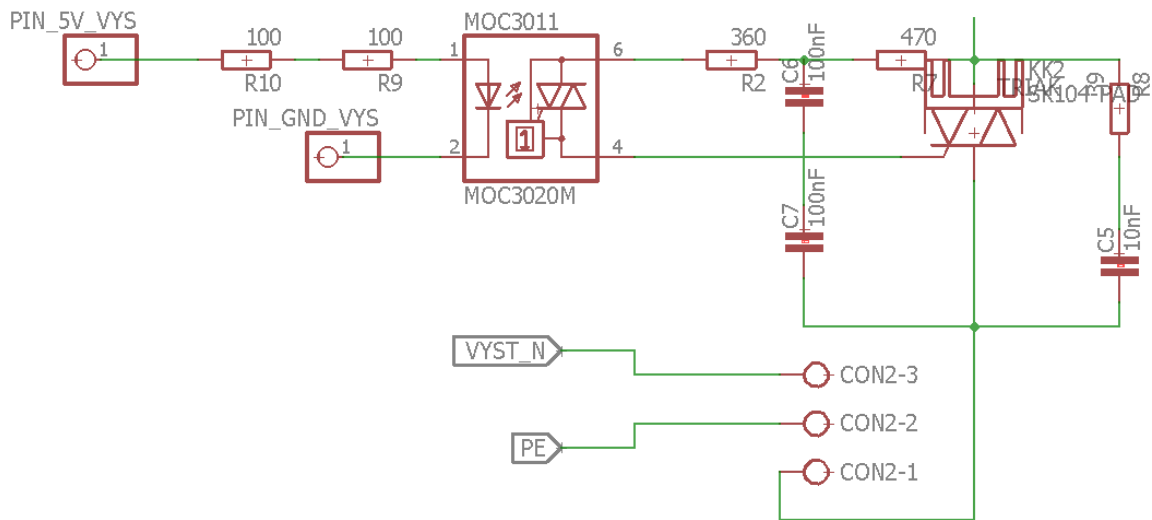
3.2.1 Schéma výkonové DPS - zdroje



Obrázek 13. Schéma zdroje oblasti DPS

Obvod je napájen ze sítě, proto byla zvolena jako vstupní konektor šroubovací svorkovnice. Tato svorkovnice má rozstup pinů 5 mm a podle údajů udávaných výrobcem, je tato svorkovnice dimenzovaná na napětí 250V a procházejícím proudem 10A. Tyto parametry jsou více než dostačující pro potřeby výrobku (zadání 230V , 6A). Vstupní proud a napětí prochází dále obvodem přes pojistku, která je dimenzovaná na 250V a špičkový proudový až 35A , což je opět více než dostačují, protože obvod je napájen ze sítě. Pokud by tedy došlo k výpadku pojistek v elektroinstalaci, může zásuvkou procházet špičkový proud maximálně 16 A. Pojistka výdrží téměř dvojnásobný proud ve špičce. Transformátor změní velikost napětí z 230V na 9V. Graetzův můstek převede střídavý proud na stejnosměrný, k usměrnění dochází pomocí čtveřice diod. Stabilizátor stabilizuje hodnotu výstupu na 5V, tato hodnota napětí je stálá a neovlivnitelná. Nepůsobí na ní změny okolí, například změna teploty, tlaku nebo malá změna napětí na vstupu. U stabilizátoru se nachází chladič, který odvádí teplo vzniklé na stabilizátoru. Kolem stabilizátoru vidíme čtveřici kondenzátorů. Ty mají blokovací a odrušovací funkci. Piny označeny 5V_VS a GND_VS slouží k napájení mikroprocesoru a dalších komponent nacházejících se na logické DPS.

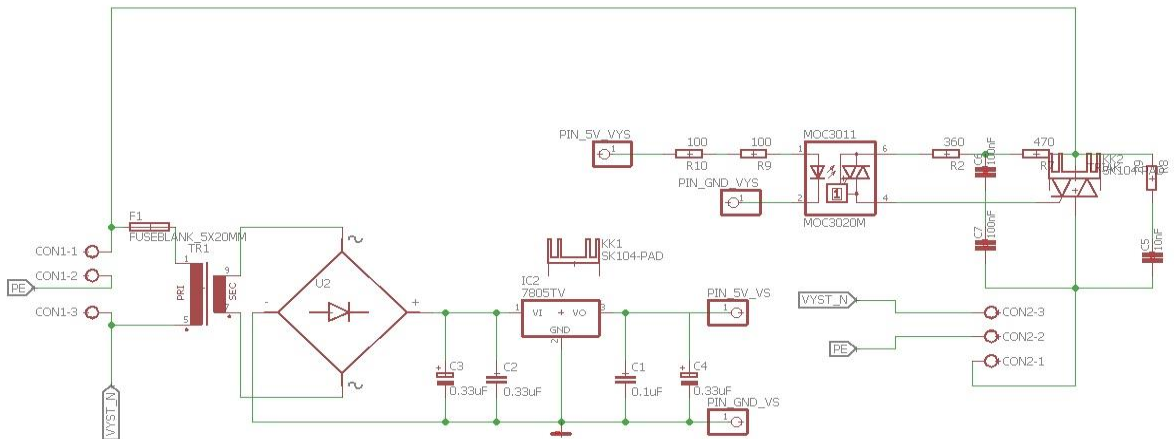
3.2.2 Schéma výkonové DPS - relé



Obrázek 14. Schéma spínací oblasti- relé

Piny 5V_VYS a GND_VYS jsou vstupními piny pro tuto část obvodu. Slouží k přivedení spínaného napětí, vzniklého spínáním mikroprocesoru. Toto napětí má hodnotu 5V, tato hodnota je pro naše použití příliš vysoká. Použitý optočlen potřebuje pro svůj chod 3V a proud 10mA, pokud by tyto hodnoty napětí a proudu nebyly dodrženy, došlo by k poškození optočlenu. K snížení hodnoty napětí a proudu jsou použity odpory R9 a R10. Na výstupu optočlenu je zvednuta hodnota napětí a proudu z výše udávaných hodnot na 1A a 250V. Rezistor R9 a kondenzátor C9 slouží k přemostění triaku. Rezistor R7 a kondenzátory C6, C7 slouží k přemostění vazebního členu. Ke správné funkci triaku musí být na něj přivedena fáze. Tato část obvodu je ukončena svorkovnicí. Tato svorkovnice má rozstup pinů 5 mm a podle údajů udávaných výrobcem, je tato svorkovnice dimenzovaná na napětí 250V a procházejícím proudem 10A. Tyto parametry jsou více než dostačující pro potřeby výrobku (zadání 230V, 6A).

3.2.3 Schéma výkonové DPS cela

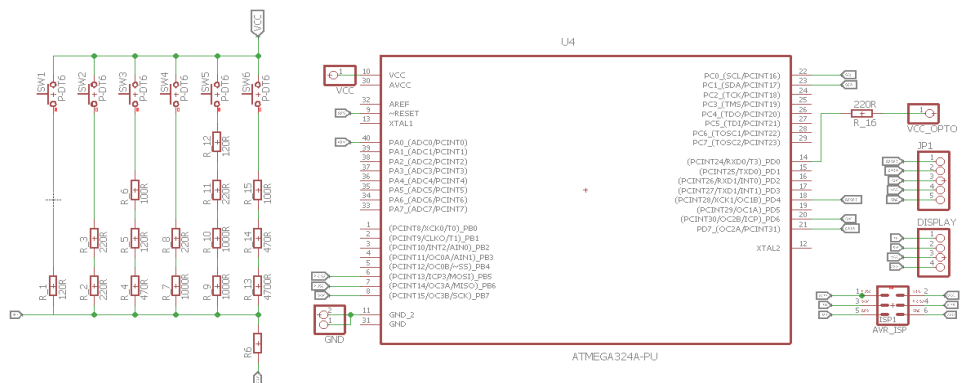


Obrázek 15. Schéma výkonové DPS

Na schématu výše vidíme propojení obou částí výkonové dps. Jedna se o propojení vodičů mezi vstupem a výstupem. Pokud by k tomuto propojení vodičů nedošlo, tak by triak nemohl správně pracovat.

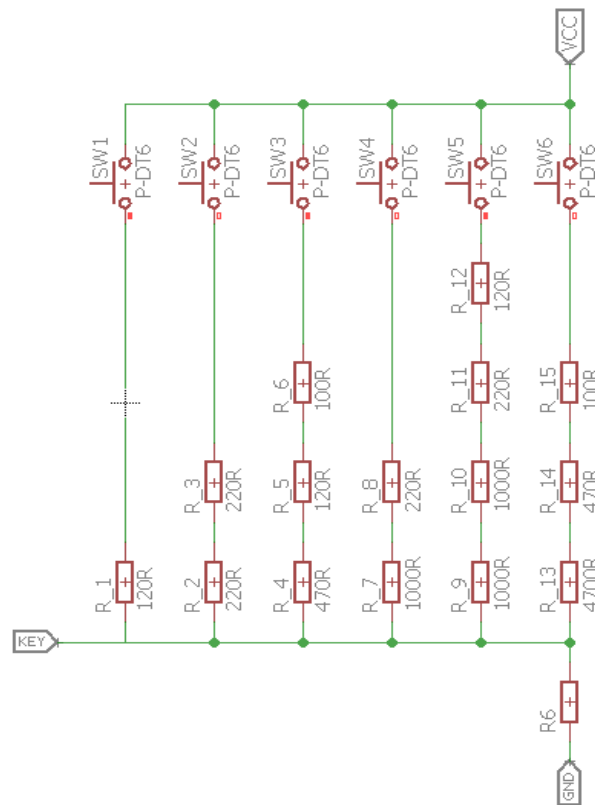
3.2.4 Schéma logické DPS

Schéma logické DPS není nijak funkčně rozděleno jako schéma výkonové DPS, ale pro lepší čitelnost viz obrázky níže bude pro rozdělení také na dvě oblasti a to na oblast tlačítek a oblast samotného mikroprocesoru.



Obrázek 16. Schéma logické DPS

3.2.5 Schéma logické DPS - tlačítka



Obrázek 17. Schéma zapojení tlačítek

Schéma logické DPS tvoří šestice tlačítek a odporová síť. Tlačítka jsou připojena k napájecímu napětí VCC o velikosti 5V a uzemněná pomocí vodiče GND. Tento celek je připojen k MCU pomocí vodiče KEY. Poblíž každého tlačítka je umístěn rezistor, jehož funkci je vytvořit úbytek napětí, při stisknutí tlačítka. Potom vidíme, že ve spodní části obvodu se nachází tzv. uzemňovací rezistor. Úkolem toho rezistoru je, udržet hodnotu napětí rovno nule, pokud není stisknuté žádné tlačítko. Výhodou tohoto zapojení je úspornost pinu mikroprocesoru. Na jeden pin jsme schopni připojit relativně velké množství tlačítek. Jediný faktor, který nás může brzdit v množství tlačítek je dostatečně velký rozdíl hodnot napětí jednotlivých tlačítek. Pokud by rozdíl byl příliš malý tak, by mohlo dojít k špatnému vyhodnocení stisknutí tlačítka. Čtení hodnot napětí je realizováno uvnitř mikroprocesoru, a to pomocí tzv. A/D převodníku.

Vypočet rezistoru.

Uzemňovací rezistor je zvolen. Jeho velikost je $1k \Omega$. Vypočet rezistorů je pomocí rovnice:

$$U_1 = U_2 \cdot \frac{R_0}{R_0 + R_1}$$

kde,

U_1 - zvolený úbytek napětí,

U_2 - napájecí napětí 5V,

R_0 - vypočítaný odpor,

R_1 - uzemňovací odpor.

V textu níže je detailně popsán způsob vypočtu rezistorů připadajícím k jednotlivým tlačítkům. Vysvětlení je provedeno na prvním tlačítku. Výpočet u ostatních tlačítek je obdobný. Výsledky budou zapsány v tabulce níže.

$$0,7 = 5 \cdot \frac{R_0}{R_0 + R_1}, \text{ provedeme operaci dělení (dělíme 5V)}$$

$$\frac{0,7}{5} = \frac{R_0}{R_0 + R_1}, \text{ upravena rovnice po dělení}$$

$$0,14 = \frac{R_0}{R_0 + R_1}, \text{ výsledná rovnice po dělení, provedeme roznásobení jmenovatelem}$$

$$0,14 \cdot R_0 + 140 = R_0, \text{ výsledná rovnice po roznásobení jmenovatel,}$$

$$140 = 0,86 \cdot R_0, \text{ výsledná rovnice po rozdělení neznáme o hodnot}$$

$$\frac{140}{0,86} = R_0, \text{ výsledná rovnice po provedení operace dělení. } R_0 = 162,79 \Omega, \text{ výsledný odpor.}$$

Vypočítána hodnota rezistoru je $162,79 \Omega$. Rezistor takovéto hodnoty neseženeme. Musíme tedy hledat rezistor, který má hodnotu co nejbližší této hodnotě. Zvolil jsem rezistor 120Ω .

Zvol. úbytek napětí[V]	Výsledný odpor [Ω]	Zvolený odpor [Ω]	V úbytek napětí [V]
1,4	388,88	$220 + 220 = 440$	1,53
2,1	724,13	$470 + 120 + 100 = 690$	2,04
2,8	1272,72	$1000 + 220 = 1220$	2,74
3,5	2333,33	$1000 + 1000 + 222 + 120 = 2340$	3,5
4,2	5250	$4700 + 470 + 100 = 5270$	4,2

Obrázek 18. Výsledné hodnoty veličin tlačítek

3.2.6 Schéma Logická DPS – MCU a periferie

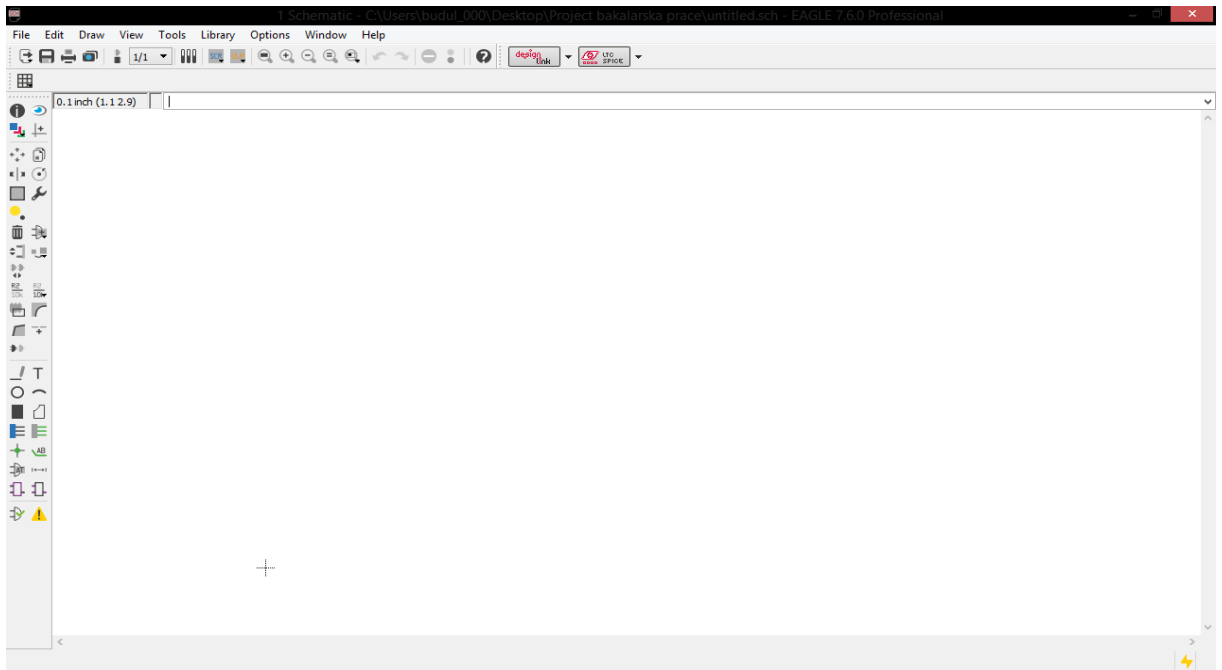


Obrázek 19. Schéma zapojení MCU a periferií

Na obrázku výše uvedeném obrázku. Můžeme vidět schéma zapojení MCU a jednotlivých periferií. MCU je napájen prostřednictvím pinů VCC a GND_2 a je na něj přivedeno napětí o velikosti 5V. Dále se zde nachází programovací ISP konektor. Tento konektor slouží k propojení MCU s PC a umožňuje následné programování MCU. Konektor ISP se skládá s šestice vodičů MISO, MOSI, SCK, RES, GND a VCC. Vodič VCC slouží k napájení. Vodič GND je zemnicí vodič. SCK je vodič hodinového taktu. RES slouží k restartování MCU. MISO je vodič vstupních dat ze SLAVE zařízení. A MOSI je vodič výstupních dat z MASTER zařízení. Konektor označen jako DISPLAY slouží pro připojení 4. řádkového alfanumerického displeje. Tento display je připojen k MCU pomocí rozhraní I2C, jehož princip funkčnosti je podrobněji vysvětlen v teoretické BK práci. Konektor označen jako JP1 slouží pro připojení RTC. Vodič označený jako RESET slouží k restartování obvodu RTC. Vodič označený jako CLK slouží k přivedení hodinového taktu. Vodič označený jako DATA je vstupně / výstupní datový vodič. Vodič označený jako VCC slouží připojení napájení (5V).

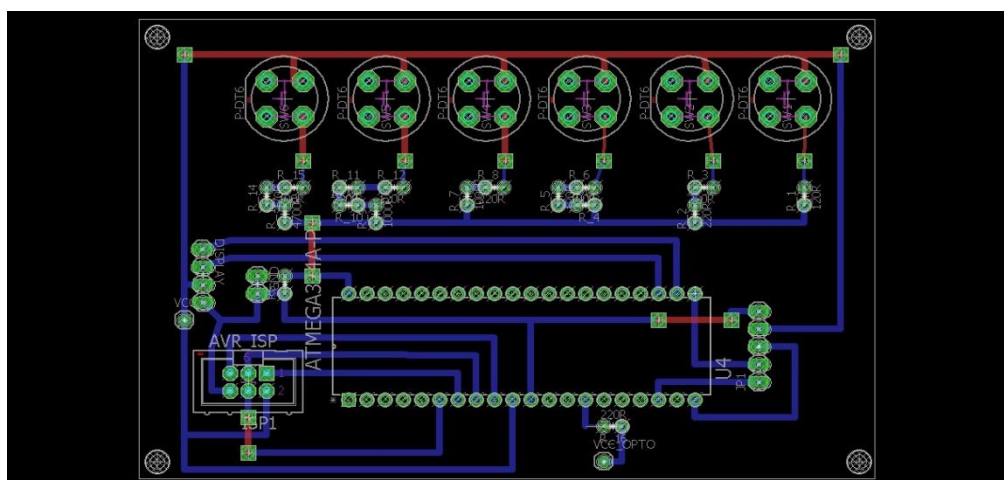
3.3 Návrhy DPS

Návrhy DPS byly vytvořeny v CAD editoru EAGLE. Ten editor poskytuje zdarma tzv. Freeware verzi, která obsahuje všechny funkce jako placená verze, avšak je omezena velikost navržené DPS. Toto omezení se nijak neprojevilo na navržených DPS. Na obrázku níže vidíme CAD editor EAGLE ve verzi 7.6.0.



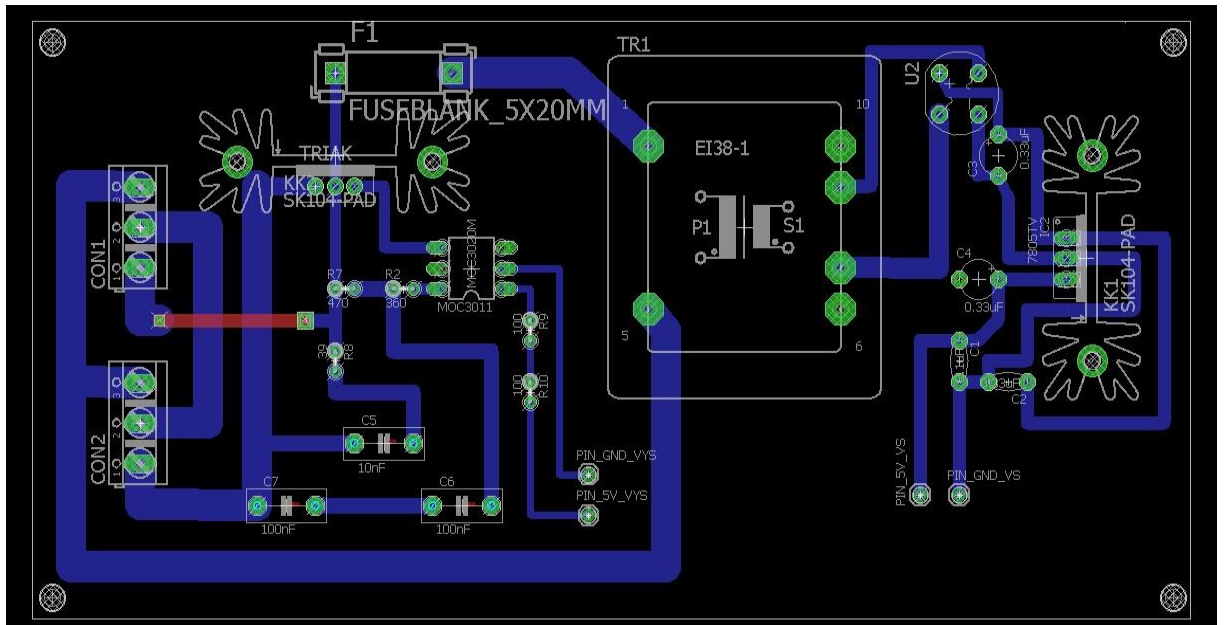
Obrázek 20. Preentscreen CAD editoru EAGLE

Na obrázku níže vidíme schéma pro logickou desku plošných spojů. Deska je navržena jako dvouvrstvá a to z důvodu designu výrobku. Na přední straně se nachází tlačítka a zbytek obvodu se nachází z druhé strany. Vrtací otvory mají velikost 3 mm.



Obrázek 21. Návrh logické DPS

Na obrázku níže vidíme, návrh výkonové desky plošných spojů. Spoje byly navrženy s ohledem na protékající proud a napětí. Stejně tak byly dodrženy i izolační mezery mezi vodiči. Tloušťka vodičů, kterými protéká navrhovaných 8A je 5mm a šířka mezer mezi vodiči kterými protéká napětí 230V je také 5mm.



Obrázek 22. Návrh výkonové DPS

Pod obrázky návrhů DPS je uvedena tabulka vysvětlující vztah mezi šířkou spoje a maximálním dovoleným protékajícím proudem.

Šířka spoje (mm)	Největší oteplení folie©		
	30	50	70
Dovolený proud (A)			
0,8	2,0	2,9	3,4
1,0	2,4	3,1	3,8
1,5	3,0	3,9	4,5
2,5	4,3	5,6	6,8
3,0	5,0	7,0	8,6
5,0	6,0	8,0	10,0

tabulka č.2. Tabulka Teplot

3.4 Výběr komponent

Začal jsem s výběrem boxu, ve kterém se bude budoucí výrobek nacházet. Zvolil jsem plastovou krabičku KP14 (Z17). Při výběru jsem hleděl na pořizovací cenu, velikost a materiál.



13

Obrázek 23. Plastový box KP14

Krabička je dostatečně vysoká pro umístění displeje a tlačítek pod ním a zároveň dostatečně dlouhá, aby se do ní vešla delší výkonová deska plošných spojů. Materiál, ze kterého je krabička vyrobena byl zvolen plast jednoduší mechanickou práci (vrtání otvorů).

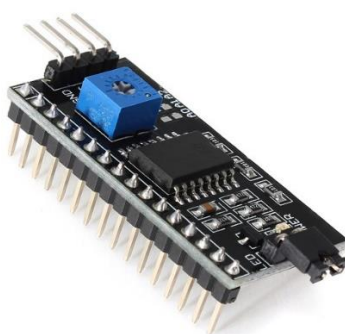
Z důvodu velkého množství položek obsažených v menu a v pod menu programu výrobku. Jsem zvolil 4 řádkový alfanumerický displej MC1604C-SYL. Tento displej obsahuje řadič HD44780 od firmy Hitachi. U tohoto displeje je možné použít komunikaci po 8bitech, po 4 bitech, nebo použít externí převodník. Já jsem si zvolil převodník I2C. Důvodem použití I2C převodníku je úspora volných pinů, pro budoucí upgrade výrobku.

¹³ Gme.cz: *Krabička plastová KP14* [online]. [cit. 2017-08-27]. Dostupné z: <https://www.gme.cz/krabicka-plastova-kp14-z-17>



14

Obrázek 24. Alfanumerický displej MC1604C-SYL



15

Obrázek 25. Převodník I2C

Dále jsem vybral oválná tlačítka P-DT6RT. Výhodou těchto tlačítek je, že mají i na hmatník, který se již nemusí kupovat zvlášť. Tlačítka byla připojena na jeden pin a při jejich stisknutí se přes A/D převodník čte hodnota napětí, které prochází logickým obvodem při jejich stisknutí.

¹⁴ *Sledujcenu.cz: MC1604C-SYL* [online]. [cit. 2017-08-27]. Dostupné z: <https://www.sledujcenu.cz/p/mc1604c-syl/>

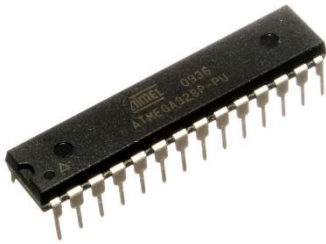
¹⁵ *Sanity.cz: I2C převodník* [online]. [cit. 2017-08-27]. Dostupné z: <http://www.santy.cz/moduly-c22/arduino-lcd-1602-16x2-modul-shield-mega-nano-i2c-redukce-seriovy-serial-2560-i52/>



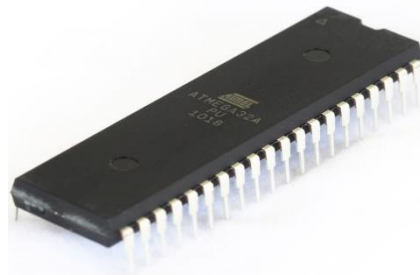
16

Obrázek 26. Tlačítko P-DT6RT

Dále jsem vybíral mikroprocesor. Nejdříve jsem zvolil mikroprocesor ATmega 328 pu. Ten byl zvolen na základě poměru cenu/výkon. Po obtížích, které byly způsobeny špatnou obsluhou přerušení jsem nakonec přešel na mikroprocesor ATmega 32A. Kde problém s přerušením jsem vyřešil pomocí většího počtu portů.



17



18

Obrázek 27. ATMEGA 328PU + ATMEGA 32A

Další komponenta v pořadí je transformátor. Na vstup transformátoru (primární vinutí) je přivedeno 230V napětí. Podle zatížení se na výstupu transformátoru objeví buď 9V (v případě zatíženého sekundárního vinutí), nebo 13V (v případě nezatíženého sekundárního vinutí).

¹⁶ Tlačítkový spínač. Gme.cz [online]. [cit. 2017-08-27]. Dostupné z: <https://www.gme.cz/tlacitkovy-spinac-p-dt6rt>

¹⁷ ATmega328. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-08-27]. Dostupné z: <https://en.wikipedia.org/wiki/ATmega328>

¹⁸ Protostack: atmega32A [online]. [cit. 2017-08-27]. Dostupné z: <https://protostack.com.au/shop/microcontrollers/atmega32a-pu-atmel-8-bit-32k-avr-microcontroller/>

Nevýhodou tohoto transformátoru jsou větší rozměry. Tato nevýhoda je vykoupena vysokou pohltivostí daného transformátů.



19

Obrázek 28. Transformátor

Další komponentou je chladič. Vybral jsem chladič s označením V7143. Největší množství vyřazeného tepla se objeví na stabilizátoru. Výpočet teploty je proveden podle následující rovnice:

$$W = U \cdot I \cdot t$$

kde,

W – práce,

U – napětí (dle potřeby může být rozdíl napětí),

I – proud procházejícím součástkou,

t- čas.

Jak vidíme na rovnici popsané výše, stačí nám pouze vypočítat práci vynaloženou na průchod proudu vodičem. Práce vynaložená na průchod proudu vodičem je totiž rovna množství vyřazeného tepla. Toto teplo nazýváme Joul-Lencovo teplo. Po dosazení konkrétních hodnot

¹⁹ Gme.cz: EI transformátor do DPS HAHN BV EI 382 1189 (1x9) [online]. [cit. 2017-08-27]. Dostupné z: <https://www.gme.cz/ei-transformator-do-dps-hahn-bv-ei-382-1189-1x9>

za výše popsané proměnné nám vyjde teplota stabilizátoru 3 Watty. Tento chladič uchladí až 10 Wattů. Jak vidíme výběr chladiče je více než naddimenzován.



20

Obrázek 29. Chladič

Další komponentou je optočlen. Mnou vybraný optočlen nese označení MOC3011M. Tento optočlen poskytuje na výstupu proud 1A a napětí 250V. Spínací proud potřebný po sepnutí optočlenu je od 10mA do 60mA. Mikroprocesor dává na výstup svých portů mnohem větší proud, proto je potřeba použít předřadné odpory, které nám protékající proud zmenší na námi požadovanou hodnotu. Vypočet předřadných odporů provedeme pomocí Ohmova zákona:

$$I = \frac{U}{R}$$

kde,

I – proud protékající obvodem,

U – napětí,

R – velikost odporu.

Tento vzorec si upravíme na následující tvar: $R = \frac{\Delta U}{I}$ kde,

I – proud protékající obvodem,

ΔU – diferenční napětí,

R – výsledný odpor.

²⁰ Gme.cz: chladič V7143 [online]. [cit. 2017-08-27]. Dostupné z: <https://www.gme.cz/v7143>



21

Obrázek 30. Optočlen

Další komponentou je triak. Mnou zvolený triak nese označení BT137/800. Tento triak poskytuje na svém výstupu až 8A/250 V, což je pro vypracování této bakalářské práce více než dostačující.



22

Obrázek 31. Triak

²¹ *Gme.cz: MOC3011M* [online]. [cit. 2017-08-27]. Dostupné z: <https://www.gme.cz/moc3011m>

²² *Official.cz: Triak BT137/800* [online]. [cit. 2017-08-27]. Dostupné z: <https://www.official.cz/z10989-triak-bt137-800>

Dalšími komponentami jsou foliové kondenzátory. Zvolil jsem 10 nF kondenzátor označený CF6-10N/J RM5 (zakoupen jeden). A dále dva 100 nF kondenzátory označené CF6-100N/J. Tyto kondenzátory jsou uzpůsobeny tak, aby jimi mohlo procházet vysoké napětí. V našem případě jsou uzpůsobeny na 400V, což je plně dostačující. Obvodem totiž bude procházet 230V. Kondenzátory vždy volíme tak, aby byly uzpůsobeny k průchodu minimálně dvojnásobné hodnoty napětí, než jaká bude obvodem procházet.



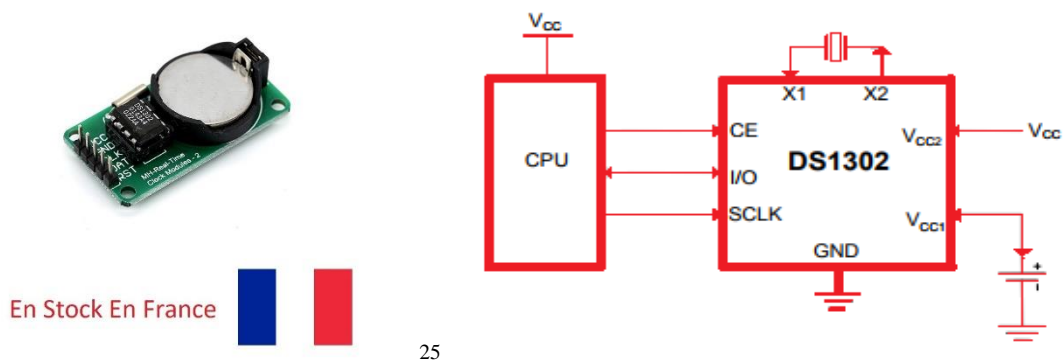
23 24

Obrázek 32. Foliové kondenzátory

Další a také poslední komponentou je RTC. Mnou zvolený obvod je označený DS1302. Jedná se o RTC, které komunikuje pomocí samostatného protokolu. Jedná se o jakýsi binární protokol, kdy ke komunikaci dochází pomocí nastavení pinů, na které je RTC připojeno. Zapojení, které tento obvod používá je vidět na obrázku níže.

²³ Gme: *Foliový kondenzátor CF6-10N/J RM5* [online]. [cit. 2017-08-27]. Dostupné z: <https://www.gme.cz/foliovoy-kondenzator-cf6-10n-j-rm5>

²⁴ Didaktik electronic s.r.o.: *CF6-100N/J* [online]. [cit. 2017-08-27]. Dostupné z: <http://www.didaktikelectronic.sk/cf6-100n-j.html>



25

Obrázek 33.DS1302 jeho vnitřní zapojení

3.5 Popis programu

3.5.1 Popis kódu programu

Výsledný program se skládá z více částí. Tyto části, nazýváme knihovny. Každá knihovna se skládá ze dvou souborů. Tyto soubory musí mít stejný název. Liší se svou příponou např. knihovna.c a knihovna.h . Soubor s příponou c obsahuje příkazy, funkce a struktury, které jsou potřebné pro správný běh programu. Soubor s příponou h obsahuje definice těchto příkazů, funkcí a struktur. Výsledný program se skládá z main a těchto knihoven:

- I2C,
- Menu,
- ADC,
- DS1302.

Výsledný kód programu je dlouhý a má mnoho částí. Z tohoto důvodu budou v tomto úseku vysvětleny nejdůležitější části kódu. Celý kód bude k nahlédnutí v přílohách.

Knihovna IC2

Tato knihovna obstarává komunikaci MCU a použitého I2C převodníku. Komunikační „přenosová“ rychlost je nastavena na 100 kHz. Nejčastěji používané funkce pro následné použití jsou LCD_init, LCD4bity a LCD_RAD_1234.V textu níže je přehled funkcí, které můžeme najít v I2C knihovně a jejich následné vysvětlení.

²⁵ *Ebay.com: Module DS1302 RTC with battery Real Time Clock Module for arduino raspberry DIY* [online]. [cit. 2017-08-27]. Dostupné z: <http://www.ebay.com/itm/Module-DS1302-RTC-with-battery-Real-Time-Clock-Module-for-arduino-raspberry-DIY-/263122365134?hash=item3d4350aace:g:1ZQAASwH4NZgyg0>

```

#include "I2C.h" → Volání souboru s deklaracemi
void TWI_inic() → Funkce pro inicializaci I2C, obsahuje nastavení
přenosové rychlosti TWI
uint8_t TWI_zapis(uint8_t adresa,uint8_t data) → Funkce pro kontrolu dat a
nastavení adresy
void pisLCD4bity(uint8_t d,uint8_t rs) → Funkce pro výpis na LCD, posílá
po 4 bytech.
void pisLCD8bitu(uint8_t d,uint8_t rs) → Funkce pro výpis na LCD, posílá
po 8 bytech.
void LCD_inic() → inicializace LCD
#define LCD_RAD_1 0x00 → Definice adresy prvního řádku displeje
#define LCD_RAD_2 0x40 → Definice adresy druhého řádku displeje
#define LCD_RAD_3 0x10 → Definice adresy třetího řádku displeje
#define LCD_RAD_4 0x50 → Definice adresy čtvrtého řádku displeje

```

Knihovna ADC

Tato knihovna obstarává obsluhu kláves. Klávesy jsou čteny pomocí analogové hodnoty. Aby bylo nedošlo k chybnému přečtení tlačítek, bylo použito rozmítání. Rozmítaná hodnota je 9. Každé tlačítko je řešeno jako jeho naměřená hodnota + - rozmítaná hodnota.

```

void ADinit() → Funkce pro inicializaci A/D převodníku.
int index_mazani = 0;
void PrectiKlavesy(void) → Funkce obstarávající čtení analogové hodnoty
stisknutých kláves.
ADCSRA |= 0b01000000 → Spuštění A/D převodníku
if (analog_value>10) → Podmínka obstarávající ošetření klidové hodnoty
nestisknutých kláves. Hodnota nestisknutého
tlačítka je rovna nule. Kdyby nebyla použita
podmínka >10, program vykoval stále.
if((analog_value <= TL_OK+ ROZMITANI) && (analog_value >= TL_OK -
ROZMITANI))
→ Podmínka kontrolu stisku tlačítka ok. Kontrola je prováděna pomocí
přičtení a odečtení rozmítací hodnoty (9).

```

Knihovna DS1302

Tato knihovna je převrátu z webu²⁶. Nejvíce používanými funkcemi jsou ds1302_set_time, ds1302_get_time, ds1302_init() a struktura ymdhms_type mytime.

ds1302_get_time(struct ymdhms_type *ATime) → Funkce pro získání hodnoty času. Funkce obsahuje přepočítání z BCD na desítkovou soustavu.
ds1302_set_time(struct ymdhms_type *ATime) → Funkce pro čtení hodnot z registrů. Funkce obsahuje přepočítání z desítkové na BCD soustavu.
struct ymdhms_type → Struktura obsahuje proměnné rok, měsíc, den, hodina, minuta, sekunda. Všechny proměnné jsou datového typu char.
ds1302_init() → slouží k inicializaci použitého RTC.

Knihovna menu

Tato knihovna menu se skládá z dvou funkcí, které obsahují sled příkazů pro výpis jednotlivých menu, dialogů a dalších textových řetězců. Dále je zde obsazen přepočítání pro indexy pro řádky (viz přehledová tabulka spínání). Dále je zde obsazen přepočítání času podle pozice indexu v poli.

MENU_INTRO(void) → Funkce slouží pro kontrolu inicializace RTC a pro výpis inicializace komponent a uvítacího dialogu.
ZobrazMenu(void) → Funkce slouží k výpisu všech menu a vykonání výše zmíněných přepočítání.
if(pole[index_pole]==0)printf("%2i:%2i%2i:%2i>OFF<",index_pole/4,((index_pole%4)*15),(index_pole+1)/4,(((index_pole+1)%4)*15));
Elseprintf("%2i:%2i%2i:%2i>ON<",index_pole/4,((index_pole%4)*15),(index_pole+1)/4,(((index_pole+1)%4)*15));
→ ukázka výpočtu hodin z indexu pole. Pomocí modulo 4 a násobením 4, získáme minuty. Pomocí dělení 4, získáme hodiny

prikazLCD(LCD_CURZB);
if ((index_pole>=0)&&(index_pole<16));
if ((index_pole>=16) &&(index_pole<32))
prikazLCD(LCD_SETAD+LCD_RAD_3+index_pole-16);
if ((index_pole>=32) && (index_pole<48))
prikazLCD(LCD_SETAD+LCD_RAD_4+index_pole-32);
→ ukázka výpočtu pozice kurzoru pro jednotlivé řádky.

Main

²⁶ Atmel community: DS1302 Timekeeping Chip Library [online]. [cit. 2017-08-27]. Dostupné z: <http://community.atmel.com/projects/ds1302-timekeeping-chip-library>

Main je část programu, ve které dochází k volání všech výše jmenovaných knihoven, a také k volání funkcí uvnitř těchto knihoven. Dále zde můžeme nalézt příkazy pro ukládání do EEPROM .

```
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>
#include <stdint.h>
#include <stdio.h>
#include "ADC.h"
#include "I2C.h"
#include "menu.h"
#include "ds1302.h"
#include <avr/eeprom.h> → naikludování knihoven

#define read_eeprom_array(address,value_p,length) eeprom_read_block ((void
*)value_p, (const void *)address, length)
#define write_eeprom_array(address,value_p,length) eeprom_write_block
((const void *)value_p,(void *)address, length) → makra pro práci s eeprom

void SpinaconOFF(void)
→ Funkce zajišťující kontrolu času a sepnuti nebo vypnuti výstupu.
ADCSRA = 0b10001011; → reference - AUcc, zarovnění doleva, ADC0
ADMUX = 0b01100000; → zapnutí A/D převodníku, povolení přerušeni, f =
125kHz
```

4 ZÁVĚR

Cílem práce bylo navrhnout a zkonstruovat časový spínač řízený mikroprocesorem.

Z vytyčených podmínek zadání byly splněny následující podmínky.

- Použití MCU ATMEGA32A
- Spínání zátěže 6 A / 230 V
- Ovládání pomocí tlačítek
- Zálohovaný zdroj reálného času.
- Okamžité spínání
- Použití 4. řádkového displeje

Z výše popsaných podmínek vidíme, že byly splněny všechny podmínky, které byly stanoveny při stanovení Bakalářské práce.

V teoretickém bloku jsou popsány informace, které jsou nezbytné pro pochopení a realizaci tohoto výrobku.

V praktickém bloku je popsán výběr komponent, postup vytváření výrobku a jeho naprogramování.

Časový spínač byl úspěšně sestaven a splňuje veškeré body zadání. Byl osazen mikroprocesorem atmega32A ,který má velké množství pinů. Mnou navržené řešení, umožnilo ponechat velké množství pinů prázdných pro případný upgrade výrobku. Bylo by zajímavé přidat teplotní senzor a z daného výrobku vytvořit jakýsi termostat pro pokojové spínání elektrického vytápění. Nebo bychom mohli přidat GSM modul, a dovolit výrobek nastavit na delší vzdálenost např. pro využití na chatě.

POUŽITÁ LITRATURA

VÁŇA, Vladimír. Mikrokontroléry ATMEL AVR - Programování v jazyce C: popis a práce ve vývojovém prostředí CodeVisionAVR C. Praha: BEN technická literatura, 2003.ISBN80-7300-102-0.

MATOUŠEK, David. Aplikace mikrokontrolérů ATmega644. Praha: BEN - technická literatura, 2013. ISBN 978-80-7300-492-7.

Základy ovládání mikrokontroléru ATmega(16/32) 14.díl – Dvouřádkový LCD displej. Tajned.cz[online]. [cit. 2017-08-22]Dostupnéz:<http://www.tajned.cz/2014/11/zaklady-ovladani-mikrokontroleru-atmega-14-dil-dvouradkovy-lcd-displej/>

In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-08-26]. Dostupné z: https://cs.wikipedia.org/wiki/D%C4%9Bli%C4%8D_nap%C4%9Bt%C3%AD

Tajned.cz: I2C – Relativně jednoduše [online]. [cit. 2017-08-28]. Dostupné z: <http://www.tajned.cz/2016/10/i2c-relativne-jednoduse/>

Tajned.cz: SPI rozhraní [online]. [cit. 2017-08-28]. Dostupné z: <http://www.tajned.cz/2016/12/spi-rozhrani/>

Usměrňovač. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-08-28]. Dostupné z: <https://cs.wikipedia.org/wiki/Usm%C4%9Br%C5%88ova%C4%8D>

Tajned.cz: Základy ovládání mikrokontroléru ATmega(16/32) 9.díl – A/D převodník (2.část)[online]. [cit. 2017-08-28]. Dostupné z: <http://www.tajned.cz/2013/12/zaklady-ovladani-mikrokontroleru-atmega-8-dil-ad-prevodnik-2-cast/>

Dělič napětí. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-08-28]. Dostupné z: https://cs.wikipedia.org/wiki/D%C4%9Bli%C4%8D_nap%C4%9Bt%C3%AD

MATOUŠEK, David. *IMTEE*.

MATOUŠEK, David. *IEOBE - Elektrické obvody*.

Real time clock. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-08-28]. Dostupné z: https://en.wikipedia.org/wiki/Real-time_clock

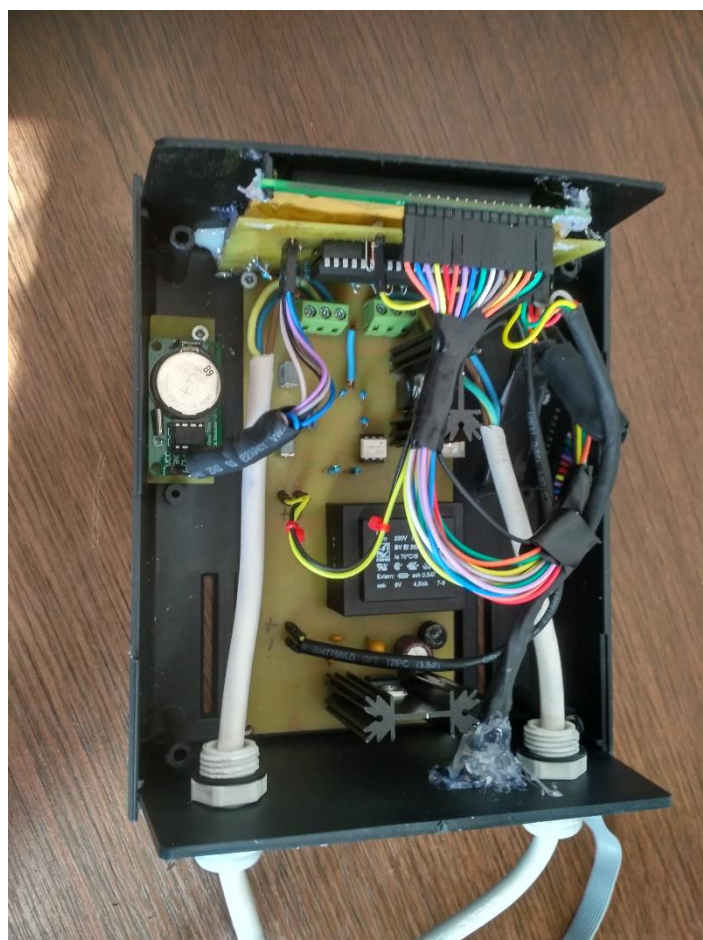
5 SEZNAM PŘÍLOH

Příloha A – výsledný časový spínač

Příloha B – manual

Příloha C- CD

Příloha A – Výsledný časový spínač



Obrázek č. 1 Kompletně zapojený časový spínač pohled ze shora



Obrázek č. 2 ISP konektor, pohled ze zadu

Příloha B - Manuál

Na obrázku níže vidíme zařízení vypnutém stavu. Nyní si popíšeme význam jednotlivých tlačítek.

- Tlačítko označené jako č. 1 - mód ZAPNUTO / VYPNUTO.
- Tlačítko označené jako č. 2 - mód MANUAL / AUTOMAT
- Tlačítko označené jako č. 3 - EXIT
- Tlačítko označené jako č. 4 - ŠÍPKA VLEVO / NAHORU
- Tlačítko označené jako č. 5 - ŠÍPKA VPRAVO / DOLU
- Tlačítko označené jako č. 6- OK



obrázek č. 1.

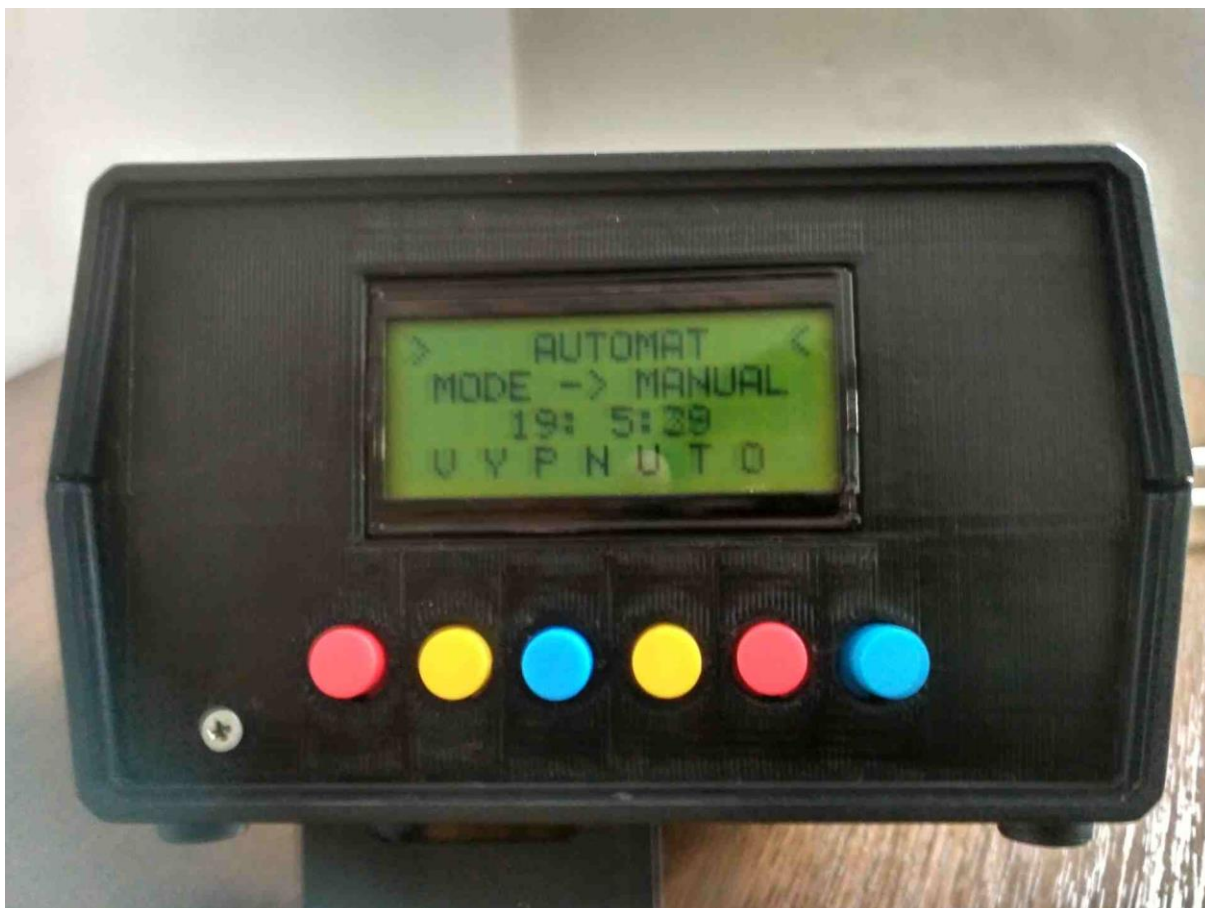
Výrobek je ošetřen proti nevhodnému použití tlačítek. Nelze používat dva nebo více tlačítek zároveň.

Menu

Základní menu

Na obrázku níže vidíme základní menu, které se objeví po zapnutí a inicializaci výrobku. V tomto menu máme dva možnosti výběru námi požadovaných akcí.

- volba modu AUTOMAT / MANUAL



obrázek č. 2.

Automat / Manuál

Po stisku tlačítka č. 2 má uživatel možnost výběru volby modu sepnutí zátěže.

- Manuální spínání
- Automatické spínání

Manuální spínání

Uživatel zjistí, že byla vybraná možnost manuálního spínání prostředním prvním řádku displeje. Na displeji se v prvním řádku objeví nápis manuál. K tomuto nápisu směřují dvě šipky. V modu manuál má uživatel možnost okamžitého spínání zátěže bez nastavení času. A to prostřednictvím stisku tlačítka č. 1.

- Objeví-li se na čtvrtém řádku displeje nápis vypnuto, zátěž sepnuta není, viz obrázek č. 3.

- Objeví-li se na čtvrtém řádku displeje nápis zapnuto, zátěž je sepnuta, viz obrázek č. 4.



obrázek č. 3.



obrázek č. 4.

Automatické spínání

Uživatel zjistí, že byla vybraná možnost automatického spínání prostředním prvním řádku displeje. Na displeji se v prvním řádku objeví nápis automat. K tomuto nápisu směřují dvě šipky. V módu automat má uživatel možnost provést volbu nabídky nastavení. Mezi jednotlivými nabídkami nastavení se uživatel pohybuje pomocí tlačítek č. 4 a č. 5. Výběr nabídky nastavení je proveden pomocí stisku tlačítka č. 6. Mezi tyto nabídky nastavení patří:

- Časovače
- Tabulky
- RTC
- Vymazání tabulky

Na obrázku č. 5 vidíme první z výše jmenovaných nabídek nastavení – nastavení časovače. Toto menu má jiný design než základní menu. Tento design bude dodržen i u ostatních menu.



Obrázek č. 5

Stiskem tlačítka č. 6 se uživatel dostane do tabulky nastavení, kterou můžeme vidět níže na obrázku č. 6. Tabulka se skládá z 15. minutových intervalů. U každého časového intervalu je zobrazena indikace stavu sepnutí zátěže.

- ON - zátěž je zapnuta.
- OFF - zátěž je vypnuta.

Každý interval je zapsán na samotném řádku. Tabulka obsahuje 96 řádku a díky tomu umožňuje uživateli nastavit spínání zátěže na dobu 1 dne. V tabulce nastavení se uživatel pohybuje pomocí stisku tlačítek č. 4 a č. 5. Výběr stavu spínání zátěže uživatel provede pomocí stisku tlačítka č. 1. Uložení do paměti je provedeno pomocí stisku tlačítka č. 2. Stiskem tlačítka č. 3 může uživatel vyjet z této nabídky.



Obrázek č. 6

Na obrázku č. 7 vidíme druhou z výše jmenovaných nabídek nastavení – tabulky. Stiskem tlačítka č. 6 se dostaneme do přehledové tabulky, kterou vidíme na obrázku č. 8.



Obrázek č. 7

Tabulka je navrhnutá formou dvojkovej logiky a značí průběh spínání zátěže po dobu 24 hodin.

- logická 1 znamená pravdu – výstup je sepnutý
- Logická 0 znamená nepravdu – výstup není sepnutý

Každý znak chápeme jednotlivě a značí indikaci 15 minutového časového intervalu. Na jeden řádek se vejde 16 znaků, takže uživatel se prostřednictvím jednoho řádku dozví informaci o průběhu 4 hodin. Na obrázku níže vidíme 3 řádky znaků, takže jednoduchým výpočtem víme, že se jedná o indikaci 12 hodin. Z výše zmíněných informací jasně vyplývá, že tabulka je dělena na 2 části:

- 12:00 až 24:00
- 00:00 až 12:00

Tato informace je zobrazena na prvním řádku displeje viz obrázek č. 8.

Tabulkou lze procházet pomocí stisku tlačítek č. 4 a č. 5. V tabulce lze provést změnu stavu sepnutí výstupu pomocí stisku tlačítka č. 1. Uživatelem provedené změny stavu sepnutí lze uložit do paměti pomocí stisku tlačítka č. 2. Stiskem tlačítka č. 3 může uživatel vyjet z této nabídky.



Obrázek č. 8

Na obrázku č. 9 vidíme třetí z výše jmenovaných nabídek nastavení – RTC. Stiskem tlačítka č. 6 se dostaneme do nabídky nastavení aktuálního času, kterou vidíme na obrázku č. 10.



Obrázek č. 9

Stiskem tlačítek č. 4 a č. 5 volíme mezi volbou nastavení hodin nebo minut. Nastavení sekund je pro potřeby výrobku zbytečné. Zvolenou volbu potvrdíme pomocí tlačítka č. 2. Touto akcí se uživatel dostane do nabídky, kterou vidíme na obrázku č. 11.



Obrázek č. 10

V této nabídce má uživatel možnost přímého nastavení aktuálního času. Uživatel pozná, jestli nastavuje hodiny nebo minuty prostřednictvím druhého řádku. U nastavované jednotky času se vyskytuje dvojice ukazatelů. Změna hodnoty nastavované jednotky, probíhá stiskem tlačítek č. 4 a č. 5. Uložení nastavené hodnoty času probíhá pomocí stisku tlačítka č. 1. Stiskem tlačítka č. 3 může uživatel vyjet z této nabídky.



Obrázek č. 11

Na obrázku č. 12 vidíme čtvrtou a také poslední z výše jmenovaných nabídek nastavení – Vymazání tabulky. Stiskem tlačítka č. 6 se dostaneme do nabídky nastavení aktuálního času, kterou vidíme na obrázku č. 10.



Obrázek č. 12

Na obrázku č. 13 vidíme následující dialog, ve kterém si uživatel volí mezi vymazáním paměti nebo nevymazáním paměti přístroje. Vymazání paměti je provedeno stiskem tlačítka č. 1. Nevymazání přístroje je provedeno stiskem tlačítka č. 1.



Obrázek č. 13

Příloha C – CD

Obsah CD:

DPS_BP1 – Schéma a návrh výkonové DPS

DPS_BP2 – Schéma a návrh logické DPS

KotlárL_ČasovySpinač_2017.pdf – Vlastní text práce