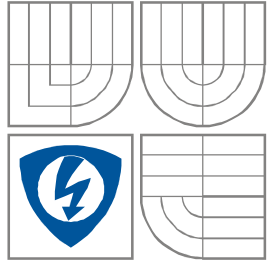
	<p>VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ BRNO UNIVERSITY OF TECHNOLOGY</p>
	<p>FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ ÚSTAV RADIOELEKTRONIKY</p> <p>FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION DEPARTMENT OF RADIO ELECTRONICS</p>

# PŘIJÍMAČ DÁLKOVÉHO OVLÁDÁNÍ

RECEIVER FOR REMOTE CONTROL

## BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

Tomáš Zdařil

### VEDOUCÍ PRÁCE

SUPERVISOR

prof. Dr. Ing. Zdeněk Kolka

BRNO 2009

## **VLOŽIT ORIGINÁL ZADÁNÍ**

(při odevzdávání více kusů závěrečné práce, obsahuje originál zadání jen jedna z nich, ostatní obsahují pouze jeho okopírovanou kopii)

## **ABSTRAKT**

Tato bakalářská práce se zabývá návrhem a konstrukcí zařízení, které bude sloužit jako přijímací modul infračerveného dálkového ovládání. Tímto modulem bude řízen směr otáčení anténního rotátoru. Součástí práce je teoretický rozbor dané problematiky. Dále obsahuje informace o výběru vhodných komponentů, především řídicího mikrokontroléru pro navrhované zapojení, jejich popis a celkové elektrické schéma zapojení. V praktické části jsou obsaženy informace o konstrukci tohoto zařízení včetně veškeré dokumentace a vytvořeného programu pro mikrokontrolér, který je obsažen na přiloženém CD. Na konci dokumentu je uvedeno shrnutí dosažených výsledků práce.

## **KLÍČOVÁ SLOVA**

IR přijímač, anténní rotátor, Bi-phase modulace, RC-5 protokol, mikrokontrolér, ISP programování, AD převodník

## **ABSTRACT**

This bachelor's thesis deals with design and construction facilities that will serve as the receiving module of infrared remote control. This modul will be controlled by the direction of rotation of antenna rotator. Part of this work is theoretical analysis of the issue. It also includes information on selection of appropriate components, primarily for the control microcontroller involvement and their description of the overall wiring diagram. In the practical part contained information on the construction of this facility, including all documentation and set up the program for the microcontroller that is included on the enclosed CD. At the end of the document is a summary of the results of the work.

## **KEYWORDS**

IR receiver, antenna rotator, Bi-phase modulation, RC-5 code, microcontroller, ISP programming, Analog to Digital Converter

ZDAŘIL, T. *Přijímač dálkového ovládání*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 36 s., 4 s. příloh. Vedoucí bakalářské práce prof. Dr. Ing. Zdeněk Kolka

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma Přijímač dálkového ovládání jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce prof. Dr. Ing. Zdeněku Kolkovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne .....

.....

(podpis autora)

# OBSAH

1	Úvod .....	1
2	Teorie IR přenosu .....	2
2.1	Vývoj IR technologie .....	2
2.2	Fyzická vrstva .....	3
2.2.1	Používané rychlosti a formát dat.....	3
2.3	Modulace používané IR zařízeními .....	4
2.3.1	Pulsní modulace .....	5
2.3.2	Pulsně šířková modulace .....	5
2.3.3	Bi-phase modulace .....	6
2.3.4	FSK modulace.....	6
2.4	Přenosové protokoly dálkových ovladačů.....	7
2.4.1	NEC protokol .....	7
2.4.2	RC-5 protokol .....	8
3	Návrh přijímače DO .....	9
3.1	Výběr mikrokontroléru.....	9
3.1.1	Mikrokontrolér ATtiny85 .....	10
3.2	Příjem a dekodování IR signálu .....	11
3.2.1	Přijímač SFH 5110.....	11
3.2.2	Průběhy přijímaného signálu .....	12
3.3	Zapojení.....	14
3.3.1	Popis funkce.....	14
3.3.2	ISP programování.....	15
3.3.3	Volba a výpočet jednotlivých součástí .....	16
4	Realizace zapojení .....	18
4.1	Návrh Desky plošných spojů .....	18
4.2	Osazení a mechanická konstrukce .....	18
4.3	Výpočet parametrů chladiče .....	19
4.4	Propojení modulu přijímače a rotátoru.....	21
5	Program .....	22
5.1	Přerušení (Interrupt).....	22
5.1.1	Natavení registrů pro přerušení PCINT0.....	23
5.2	Využití AD převodníku .....	24
5.2.1	Nastavení AD převodníku .....	25
5.2.2	Převod napětí-úhel natočení.....	27
5.3	Popis programu .....	28
5.3.1	Hlavní program .....	28
5.3.2	Dekodování RC-5 kódu ovladače .....	30
5.4	Porovnání nosných kmitočtů ovladačů.....	33
5.5	Použité pomůcky.....	34
6	Závěr .....	35
7	Použitá literatura: .....	36

## SEZNAM OBRÁZKŮ

Obr. 1.1 Ukázka propojení navrhovaného obvodu s rotátorem.....	1
Obr. 2.1 Příklady přístrojů,kde se užívá infraportu (převzato z [2]).....	2
Obr. 2.2 Signál NRZ,pulzní modulace 3/16 a 1/4délky trvání bitu .....	3
Obr. 2.3 Průběh dat klasického sériového přenosu .....	4
Obr. 2.4 Průběh dat IR přenosu - pulsní modulace 3/16 délky původní doby trvání bitu.	4
Obr. 2.5 Princip pulsní modulace.....	5
Obr. 2.6 Pulsně šířková modulace.....	5
Obr. 2.7 Princip Bi-phase modulace .....	6
Obr. 2.8 FSK modulace (frequency shift keying) .....	6
Obr. 2.9 Příklad datového rámce NEC.....	7
Obr. 2.10 Datový rámec protokolu RC-5 .....	8
Obr. 3.1 Blokové schéma ATtiny 85 (převzato z [4]).....	10
Obr. 3.2 Zapojení vývodů pouzdra obvodu ATtiny85 (převzato z [4]) .....	11
Obr. 3.3 Ukázka některých IR součástek (převzato z [7]).....	11
Obr. 3.4 Blokové schéma a použité pouzdro obvodu SFH 5110 (převzato z [6]) .....	12
Obr. 3.6 Doporučené zapojení ISP pro Atmel AVR.....	15
Obr. 4.1 Hotová osazená deska přijímače připravená k programování pomocí ISP.....	19
Obr. 4.2 Parametry pro výpočet chladiče .....	20
Obr. 4.3 Blokové schéma propojení rotátoru a modulu IR přijímače .....	21
Obr. 5.1 Vývojový diagram hlavního programu .....	29
Obr. 5.2 Princip dekódování kódu dálkového ovladače.....	30
Obr. 5.3 Vývojový diagram dekódování kódu dálkového ovladače.....	32
Obr. 5.4.Datový rámec .....	33

## SEZNAM TABULEK

Tab.3.1 Výběr vhodného mikrokontroléru .....	9
Tab.3.2 Doporučené rozložení pinů pro Atmel AVR .....	15
Tab.5.1 Některé příklady přerušení pro ATtiny85 (převzato z [4]).....	22
Tab.5.2 Převodní tabulka úhel natočení-hodnota ADCL a ADCH po AD převodu.....	27
Tab.5.3 Povelý dálkového ovladače.....	31
Tab.5.4 Porovnání kmitočtů dálkových ovladačů.....	34
Tab.5.5 Seznam použitého software a hardware.....	34

## SEZNAM GRAFŮ

Graf. 3.1 Přijatý kód při stisku tlačítka „vpravo “	13
Graf. 3.2 Přijatý kód při stisku tlačítka „vlevo “	13
Graf. 3.3 Přijatý kód při stisku tlačítka „nahoru “	13
Graf. 3.4 Přijatý kód při stisku tlačítka „dolu “	13

## SEZNAM PŘÍLOH

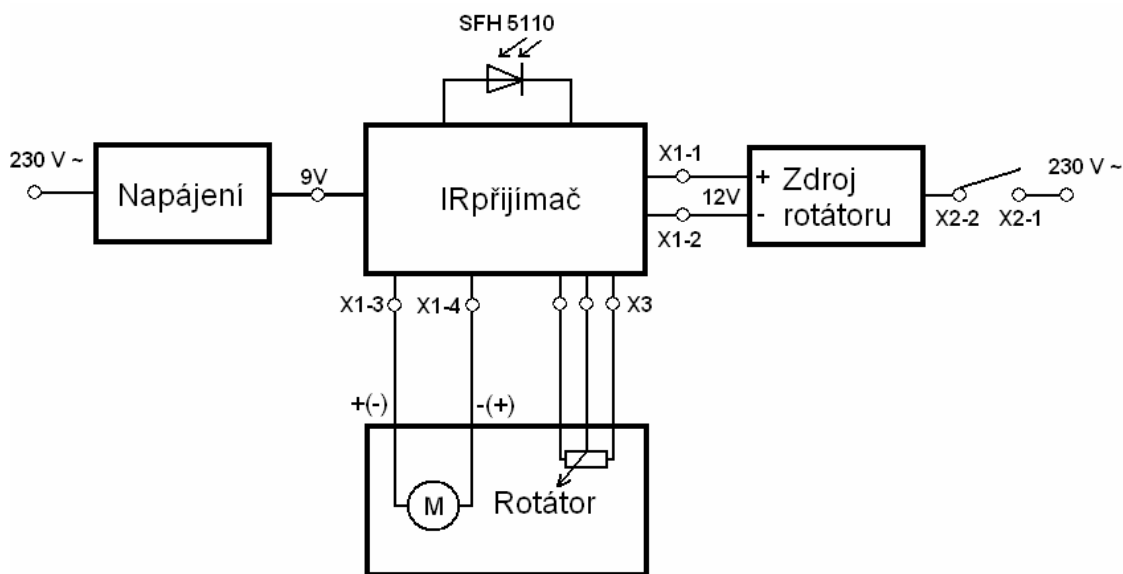
A	NÁVRH ZAŘÍZENÍ .....	37
A.1	Obvodové zapojení... ..	37
A.2	Deska plošného spoje - top (strana součástek).....	38
A.3	Deska plošného spoje - bottom (strana spojů).....	38
A.4	Použité součástky.....	39
B	ČÁSTI ZDROJOVÉHO KÓDU .....	40
B.1	Část hlavního programu.....	40
B.2	Dekódovací algoritmus RC-5 kódu.....	40



# 1 ÚVOD

Cílem této bakalářské práce je rozebrat v teoretické rovině problematiku dálkového ovládání domácích spotřebičů pomocí infračerveného záření. Na základě těchto poznatků navrhnout a zkonstruovat zařízení, které bude složeno jako přijímací modul k dálkovému ovládání anténního rotátoru. Hotový přijímač nahradí původní ovládací analogovou jednotku a s rotátorem bude propojen tak, jak je naznačeno na obr. 1.1. Při návrhu tohoto obvodu bylo požadováno několik specifikací. Napájecí napětí se má pohybovat v rozsahu 7-9V, aby vyhovovalo stabilizátoru napětí 5V. Jako samotný pohon pro otáčení rotátoru bude sloužit spínaný napájecí zdroj 12V. Ten má být elektronicky sepnut při každém požadavku na otočení. Zapojení tedy musí obsahovat takovýto spínač. Je důležité budoucí přijímač osadit svorkovnicemi tak, aby bylo možné ho jednoduše propojit s ovládaným přístrojem.

Základním kamenem zapojení je vhodně vybraný mikrokontrolér, který vyhodnocuje povely z dálkového ovládání. Uživatel bude mít na výběr ze dvou možností otáčení přičemž jeho interval se má pohybovat v rozmezí 0 až 360°. Půjde o natáčení plynulé vlevo i vpravo, a skokové o určitý počet stupňů.



Obr. 1.1 Ukázka propojení navrhovaného obvodu s rotátorem

Celý projekt je rozdělen do několika dílčích částí podle obsahu. V kapitole 2 se čtenář dozví informace, týkající se teorie infračerveného přenosu. Důležitá je pak zejména část zabývající se přenosovými protokoly dálkových ovladačů. Kapitola 3 se nazývá „Návrh přijímače DO“. Zabývá se výběrem hlavních bloků zapojení jakož to mikrokontrolérem, použitým dálkovým ovladačem a nakonec je zde navrženo a popsáno celé zapojení. Následující kapitola 4 popisuje realizaci celého zapojení. Je zde navržena deska plošných spojů a popsána celková mechanická konstrukce včetně návrhu chladiče pro stabilizátor. Program pro mikrokontrolér je popsán a vysvětlen v kapitole 5. V závěrečné 6. kapitole jsou shrnuty a zhodnoceny výsledky celé práce.

## 2 TEORIE IR PŘENOSU

V důsledku velmi rychlého vývoje informačních a komunikačních technologií začal analogový svět nahrazovat svět digitální. Jak už z názvu vyplývá, digitální technologie pracuje s daty v digitální (chceme-li také číslicové) podobě. Aby tato data byla maximálně využívána, je nutné je nějakým způsobem přenášet na určitou vzdálenost. Tento přenos se pak dělí na klasický (pomocí vodičů) a bezdrátový.

Přenos pomocí infračerveného (IR) rozhraní se řadí mezi jednu z nejstarších metod mobilní bezdrátové komunikace. Tuto éru pravděpodobně nastartovala jedna z prvních aplikací moderního bezdrátového přenosu, dálkové ovládání televizních přijímačů. Postupem času však tato metoda začala prostupovat do nejrůznějších odvětví elektroniky. V dnešní době jsou infračerveným rozhraním (chceme-li také infra portem) vybaveny téměř všechny mobilní telefony, nejrůznější periferní zařízení osobních i stolních počítačů, digitální fotoaparáty a kamery a v neposlední řadě samozřejmě dálkové ovladače. Příklady takovýchto zařízení jsou vidět na obr. 2.1.



Obr. 2.1 Příklady přístrojů, kde se užívá infra portu (převzato z [2])

### 2.1 Vývoj IR technologie

Jako první se začal přenos dat v infračerveném pásmu využívat za účelem ovládání nejrůznější spotřební elektroniky. Tato metoda se zdála velice efektivní při dosažení potřebné funkčnosti za přijatelnou cenu. Pro tento účel tedy muselo být navrženo několik způsobů, jak digitální data modulovat aby při přenosu nedocházelo k nežádoucím chybám a aby byl spolehlivý.

Na tuto technologii byly v době vývoje kladeny nejrůznější požadavky. Dostatečně velká přenosová kapacita nízký příkon a hlavně musela být použitelná pro široký rozsah aplikací. V roce 1993 byla založena Infrared Data Association (IrDA), jejímž úkolem bylo vytvořit otevřený IR standard, který by zaručil potřebnou přenosovou kompatibilitu zařízení různých výrobců. Již za rok, tedy v roce 1994, byla vydána první verze IrDA standardu. V současné době je pod hlavičkou IrDA sdruženo mnoho

společností. Mezi ty nejvýznamnější patří Hewlett-Packard, IBM, Sharp, Apple, Canon a další. Tyto společnosti vydaná pravidla a normy respektují a aplikují je na své výrobky.

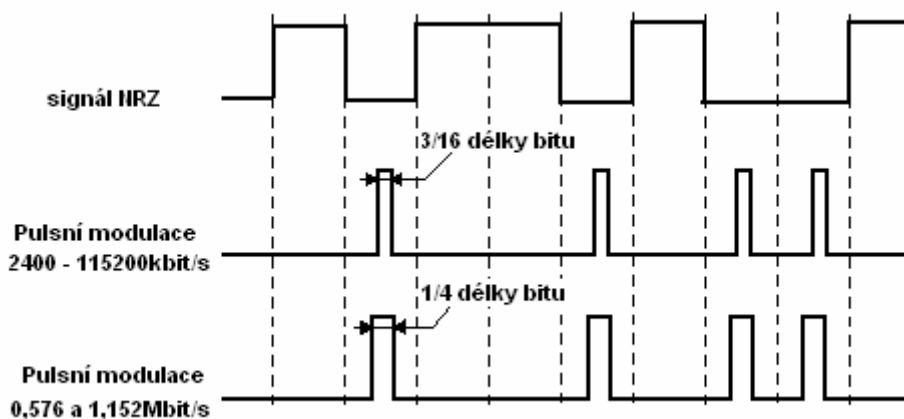
## 2.2 Fyzická vrstva

IrDA zařízení využívá k přenosu infračervené LED diody, jejichž vlnová délka vyzařovaného světla (Infračerveného záření) je  $875\text{nm} \pm$  povolená tolerance  $30\text{nm}$ . Infračervené (infrared) záření je elektromagnetické záření, jehož vlnová délka je větší než vlnová délka viditelného světla, ale menší než mikrovlnného záření. Přijímacími prvky jsou pak PIN diody pracující v generačním režimu. Při dopadu světla na ně se vyrazí elektrony. Ty se pak odvádí do elektrického filtru a ten po sléze propustí pouze frekvence typické pro daný typ IrDA modulační. Mezi energií dopadnutého infra záření a nábojem, který je přijímačem vygenerován platí přímá úměra.

### 2.2.1 Používané rychlosti a formát dat

Zařízení typu IrDA 1.0 a 1.1 pracují dle normy do vzdálenosti 1m při bitové chybovosti BER  $10^{-9}$  (bit error ratio - poměr chybných bitů ku celkovému počtu přenesených bitů) a maximální úrovni okolního osvětlení 10 klux, což odpovídá přibližně dennímu svitu slunce. Tyto hodnoty platí pouze pro případ nesouososti vysílače a přijímače o 15 stupňů. Pro jednotlivé optické prvky se pak měří výkon do 30 stupňů.

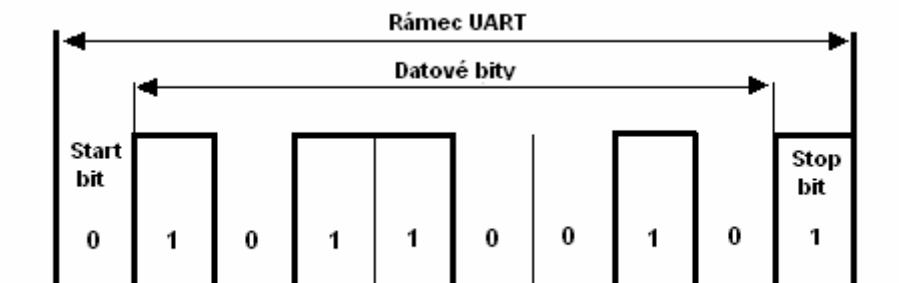
IrDA 1.0 využívá základní rychlosti přenosu od 2400 do 115200 kbit/s a pulsní modulační 3/16 délky původní doby trvání bitu. Norma IrDA 1.1 navíc definuje rychlosti 0,576 a 1,152 Mbit/s. V tomto případě se využívá pulsní modulační o délce 1/4 doby impulsu. Při těchto rychlostech je již nutné signál vysílat synchronně. Signál NRZ na obr. 2.2 je původní datový tok bez modulační.



Obr. 2.2 Signál NRZ, pulsní modulační 3/16 a 1/4 délky trvání bitu

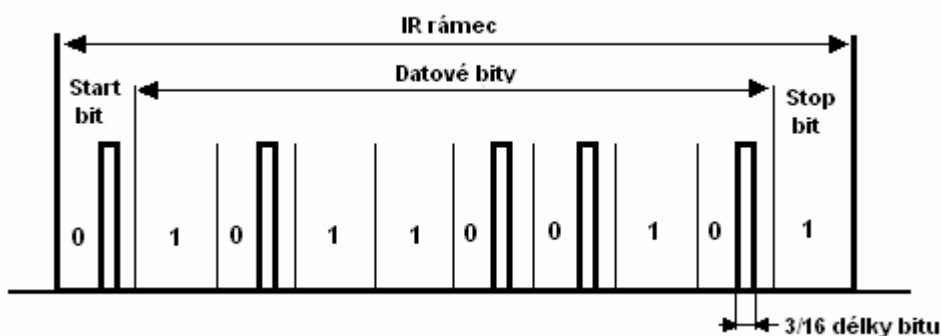
**Formát dat** vysílaných pomocí infračerveného záření se shoduje s klasickým sériovým

přenosem zobrazeným na obr. 2.3. V obou případech se tedy jedná o asynchronní bitové slovo uvozené start bitem, následují samotné datové bity a rámeček je pak ukončen stop bitem. Jednotlivé bity potom představují impulsy, tj. rozsvícení či nerozsvícení vysílací infra diody. Časově pak může být jeden impuls vyjádřen jako  $3/16$  doby impulsu nebo jako impuls s pevnou délkou  $1,63 \mu s$ .



Obr. 2.3 Průběh dat klasického sériového přenosu

Na obr. 2.4 je vidět datový rámeček klasického infra přenosu.



Obr. 2.4 Průběh dat IR přenosu - pulsní modulace  $3/16$  délky původní doby trvání bitu

## 2.3 Modulace používané IR zařízeními

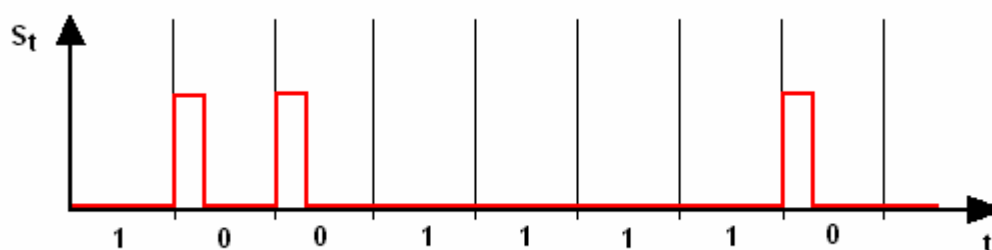
Aby byl přenos dat efektivní a byla zajištěna dostatečná bezpečnost, používá se stejně jako v jiných případech modulace. Data mohou být vysílána mnoha způsoby kódování nosného optického záření. V současné době se používá mnoho modulací, z nichž některé byly přijaty jako standard. Mezi nejčastější čtyři druhy digitálních modulací používaných při přenosu IR zařízení patří:

- Pulsní modulace
- Pulsně šířková modulace
- Bi - phase modulace
- FSK modulace

### 2.3.1 Pulsní modulace

Pulsní modulace patří mezi jednu z nejjednodušších metod kódování dat pro přenos pomocí infračerveného záření. Její princip je znázorněn na obr.2.5 a spočívá v tom, že se daný digitální signál (slovo) rozdělí na určitý počet stejně velkých částí (tzv. časová okna). V těchto jednotlivých oknech se pak v případě logické nuly vyskytuje impuls konstantní délky, který zpravidla zabírá jen určitou část tohoto okna. V případě logické jedničky se zde žádný impuls nenachází.

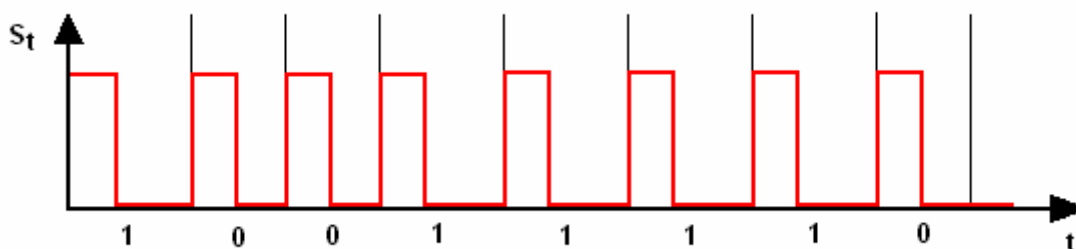
Velmi důležitá je v tomto případě synchronizace hodinových impulsů s hranou přijímaného signálu. V případě, že se delší dobu vysílá větší počet po sobě jdoucích jedniček, může dojít na přijímací straně ke ztrátě synchronismu. Tomuto nežádoucímu jevu se zabráňuje přidáním tzv. *bit-stuffingu*. Po určité době je tedy vyslán jeden nulový bit navíc, který je ovšem poté nutno na přijímací straně opět odstranit.



Obr. 2.5 Princip pulsní modulace

### 2.3.2 Pulsně šířková modulace

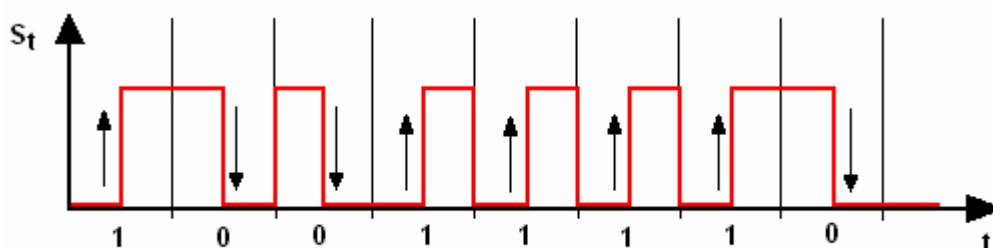
Podstatou kódování dat pomocí pulsně šířkové modulace je opět rozdělení daného slova na určitý počet časových oken, tentokrát však ne stejně dlouhých. V těchto oknech se vždy vyskytuje impuls dané délky. Detekce jednotlivých bitů pak závisí na délce jednotlivých oken. Delší okno odpovídá logické 1, kratší pak logické 0. Stejně jako je tomu u pulsní modulace, i v tomto případě je velmi důležitá synchronizace. Jednotlivá časová okna se synchronizují s náběžnou hranou signálu. Ukázka pulsně šířkové modulace je zobrazena na obr.2.6.



Obr. 2.6 Pulsně šířková modulace

### 2.3.3 Bi-phase modulace

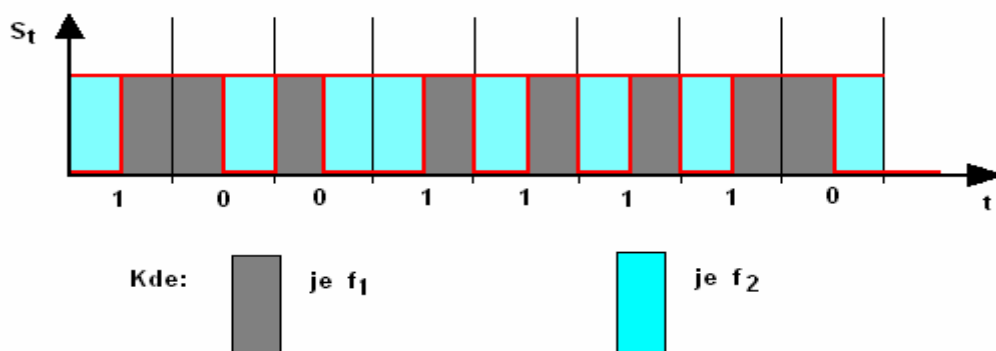
Dalším typem modulace je tzv. Bi-phase modulace. Její princip je znázorněn na obr. 2.7. Daný signál je jako v předchozích případech rozdělen na určitý počet časových oken, jenž mají všechny stejnou velikost. Detekce jednotlivých bitů se provádí následujícím způsobem. Pro jednotlivá časová okna se vždy bere změna úrovně signálu v tomto okně. Je-li kladná (mění se z log 0 na log 1) je bit vyhodnocen jako log. 1, v případě změny z log 1 na log 0 je bit vyhodnocen jako log 0. Tento druh modulace je často užívaný a pro nás důležitý. Příkladem je kód RC-5, který se používá v infračervených dálkových ovladačích.



Obr. 2.7 Princip Bi-phase modulace

### 2.3.4 FSK modulace

FSK (frequency shift keying) na obr. 2.8, je dalším druhem modulace. Patří mezi velice bezpečné metody přenosu. Jednotlivé bity jsou modulovány pomocí dvou rozdílných frekvencí. Velkou nevýhodou této metody je její vysoká pořizovací cena modelovacího zařízení, vyšší energetická náročnost a oproti jiným metodám pomalejší datový tok. Tato modulace se tedy používá výjimečně v systémech s vysokými požadavky na přesnost přenosu.



Obr. 2.8 FSK modulace (frequency shift keying)

## 2.4 Přenosové protokoly dálkových ovladačů

Používání normovaných přenosových protokolů je nutné z toho důvodu, aby se mezi sebou vysílač a přijímač „domluvili“ a nedocházelo k chybám při přenosu. V následujících řádcích si popíšeme a vysvětlíme funkci dvou nejužívanějších protokolů této kategorie. A to:

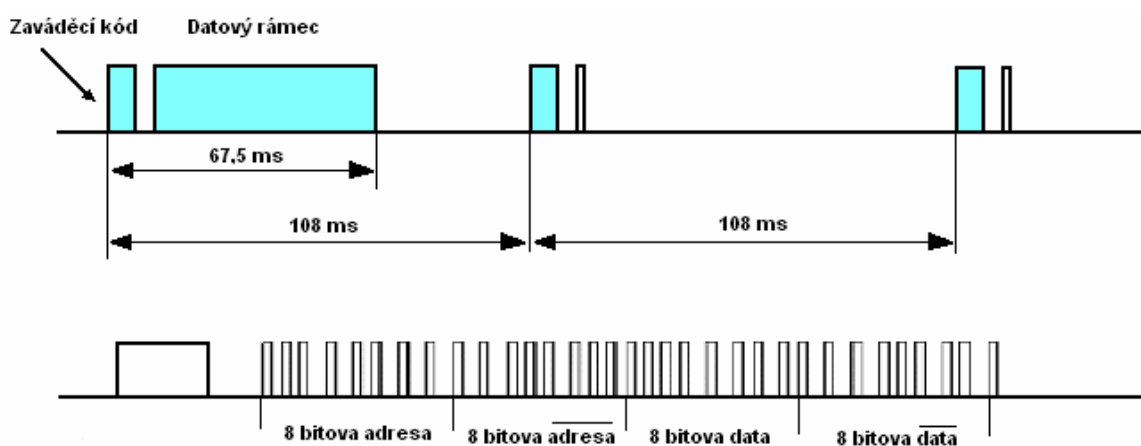
- NEC protokol
- RC-5 protokol

Oba dva byly vyvinuty jako standard pro dálkové ovládání spotřební elektroniky. Důležitý pro nás bude zejména RC-5 protokol který využívá náš dálkový ovladač.

### 2.4.1 NEC protokol

Tento standardní protokol (obr. 2.9), se využívá jak už bylo napsáno výše pro dálkové ovladače. Z důvodu snazšího přenosu signálu se data kódují pomocí pulsně šířkové modulace na nosnou frekvenci 36kHz. Používá se konstantní délka datového rámce společně s pulsně šířkovou modulací. Pro zabezpečení proti chybám se používá redundance. Data se vysílají dvakrát, nejdříve normálně a poté invertovaně.

Nyní si popíšeme datový rámec NEC protokolu. Jako první je vysílán zaváděcí kód tj. burst, který slouží k nastavení citlivosti přijímacího modulu. Má délku 9 ms a je následován pauzou 4,5 ms. Dále jsou vysílány 4 byte které představují adresu, konkrétní příkaz a uvedenou invertovanou redundanci. V případě, že vysíláme daný datový rámec vícekrát (když je na ovladači stisknuto a drženo tlačítko na delší dobu), vysílá se po prvním kompletním rámci jen zaváděcí kód a první bit.

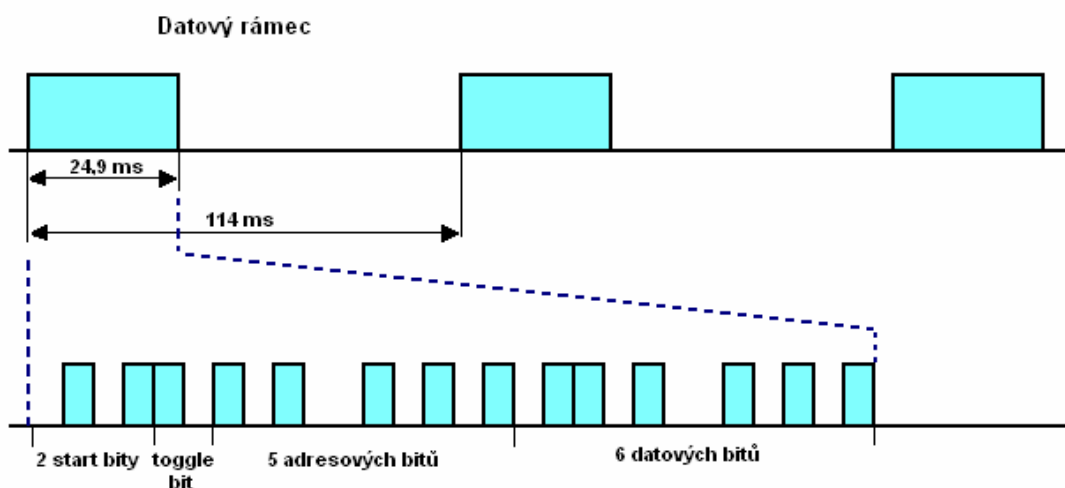


Obr. 2.9 Příklad datového rámce NEC

## 2.4.2 RC-5 protokol

Asi nejpoužívanější protokol pro dálkové ovládání infračerveným zařízením je právě RC-5. Byl navržený firmou Philips pro ovládání přístrojů spotřební elektroniky. Používá Bi-phase kódování na nosný kmitočet 36kHz. Jeho takovou zvláštností je, že využívá netypickou strukturu rámce s délkou 6-ti datových bitů. Každým stiskem klávesy je vyslán datový rámec o délce 114 ms, jehož příklad je vidět na obr. 2.10. Informace o stisku tlačítka však zabírá pouze 24,9 ms tohoto rámce. Jelikož je počet bitů 14, časová délka jednoho je pak 1,778 ms. Složení takového rámce je následující:

- 2 start bity - Přenos rámce začíná dvěma start bity, které mají vždy logickou úroveň 1
- toggle bit - Tento bit slouží pro identifikaci držení tlačítka na dálkovém ovladači. Stiskneme-li a držíme tlačítko, nemění se. Po uvolnění a následném opětovném stisknutí je tento bit invertován a při podržení se vysílá jeho nová hodnota. Tato funkce slouží například pro přepínání tv programu, kdy při stisknutí a držení příslušného tlačítka dojde k přepnutí právě o jeden kanál.
- 5 adresových bitů - Slouží k rozeznání konkrétního ovládaného zařízení. Teoreticky tak lze ovládat  $2^5$ , tj. až 32 různých zařízení jedním ovladačem
- 6 datových bitů - Každému tlačítku na dálkovém ovladači je přiřazen kód, jenž je reprezentován právě těmito bity. Nebývá však pravidlem, že kódy odpovídajících si tlačítek různých výrobců jsou shodné.



Obr. 2.10 Datový rámec protokolu RC-5



## 3 NÁVRH PŘIJÍMAČE DO

Tato část projektu je věnována převážně praktickým informacím, souvisejících s návrhem přijímače dálkového ovládání, který bude řídit anténní rotátor. Nejprve je pro tuto aplikaci vybrán z velkého množství dostupných obvodů vhodný mikrokontrolér. Následuje stručný popis jeho základních parametrů. Dále jsou zde sepsány informace, týkající se přijímače infra signálu SFH 5110, jakožto další klíčové součásti celého zapojení. Čtenář zde ještě nalezne informace o použitém ovladači a jeho průbězích. Na konci celé kapitoly je popsána funkce navrženého zapojení.

### 3.1 Výběr mikrokontroléru

V současné době je na našem trhu dostupné velké množství nejrozličnějších mikrokontrolérů od různých výrobců. Mezi nejznámější z nich patří například Intel, Freescale nebo Atmel. Pro navrhovanou aplikaci je možné nalézt větší množství vhodných řídicích obvodů. Jelikož sem se již setkal s obvody od firmy Atmel, zaměřím se tedy při výběru právě na jejich výrobky.

Atmel vyrábí integrované obvody již od roku 1984. Mezi jeho nejvýznamnější počiny patří například obvody architektury 8051, dále řada AT91SAM 32-bit a novější Atmel AVR. Zkratka AVR značí vlastní architekturu procesorů Atmel. Ty se rozdělují na 8 a 32 bitové. Jsou uzpůsobeny tak, aby se daly velice efektivně programovat za pomoci jazyka C a samozřejmě také v Asembleru. Procesory AVR se dále dělí podle své výbavy, výkonu a použitým periferiím do skupin ATtiny a ATmega. Jak už sám název napovídá, skupina Tiny obsahuje obvody určené pro jednodušší aplikace. Jsou však dostatečně vybaveny a proto jsem se při výběru zaměřil do jejich řad. Základní požadavky pro výběr byly:

- alespoň 4 vstupně/výstupní piny
- externí přerušení
- dostatečná paměť pro program
- AD převodník pro zpětnou vazbu
- případně interní oscilátor

Tab.3.1 Výběr vhodného mikrokontroléru

Obvod	Počet I/O pinů	ADC /počet kanálů	Paměť pro program [kB]	Pouzdro
ATtiny44	12	ano/8	4	DIL14
ATtiny84	12	ano/8	8	DIL14
<b>ATtiny85</b>	<b>6</b>	<b>ano/4</b>	<b>8</b>	<b>DIL8</b>

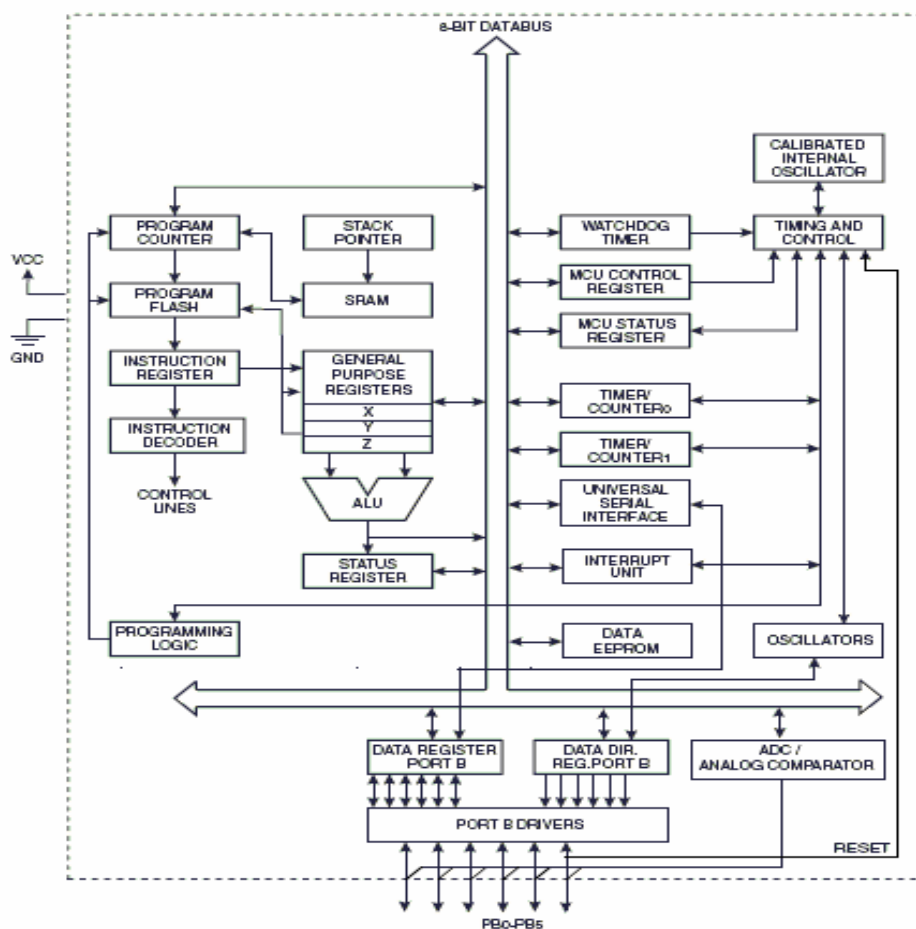
Z obvodů uvedených v tab.3.1 byl nakonec vybrán ATtiny85. Obsahuje sice jen 6 I/O pinů, avšak je plně vybaven nejrozličnějšími periferiemi a pro toto zapojení dostačující.

### 3.1.1 Mikrokontrolér ATtiny85

Obvod ATtiny85 vyrábí firma Atmel. Patří do skupiny AVR 8bit RISC. Tento mikrokontrolér najde uplatnění především v jednodušších aplikacích. Je usazen v pouzdře DIL8 a pro případ SMD montáže si jej můžeme vybrat také v pouzdře SO8.

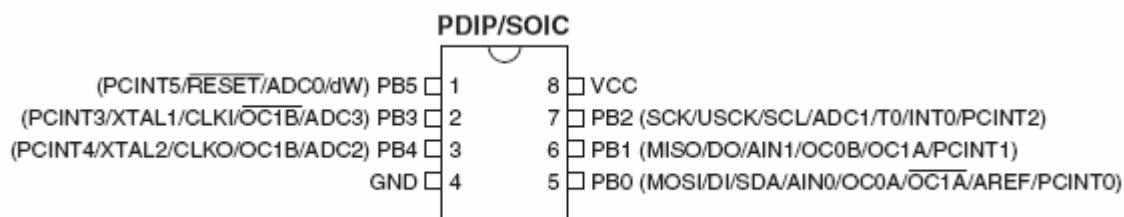
Zde jsou základní vlastnosti tohoto obvodu [4], jehož blokové schéma si můžeme prohlédnout na obr.3.1.

- Nízkopříkonový AVR 8-bitový mikrokontrolér
- RISC architektura
- Až 120 instrukcí vykonávaných v jednom strojovém cyklu
- 32 osmi bitových registrů
- Programová paměť 8kByte
- Paměť pro data a konstanty EEPROM 512Byte
- Dva 8-bitové čítače/časovače s PWM
- 10-bitový AD převodník - 4 kanálový
- Interní oscilátor
- 6 I/O pinů
- Napájení 2,7-5,5 V



Obr. 3.1 Blokové schéma ATtiny 85 (převzato z [4])

Zapojení jeho vývodů je vidět na obr.3.2. Jelikož je obvod umístěn v pouzdře DIL8, je k dispozici pouze 6 vstupně/výstupních pinů.



Obr. 3.2 Zapojení vývodů pouzdra obvodu ATtiny85 (převzato z [4])

### 3.2 Příjem a dekódování IR signálu

Abychom mohli plně využívat výhod dálkového ovládání pomocí IR ovladače, je nutné tento signál před následným zpracováním nejprve přijmout a dekódovat. K tomuto účelu se vyrábí celá řada nejrůznějších součástek od různých výrobců. Například firmy Hewlett Packard, IBM, Texas instrument a jiné jich vyrábí celou řadu. Máme na výběr z variant samostatných přijímačů (přijímací infra dioda +dekodér), ale i jako páry vysílač-přijímač. Některé příklady takovýchto součástek si můžeme prohlédnout na obr.3.3.



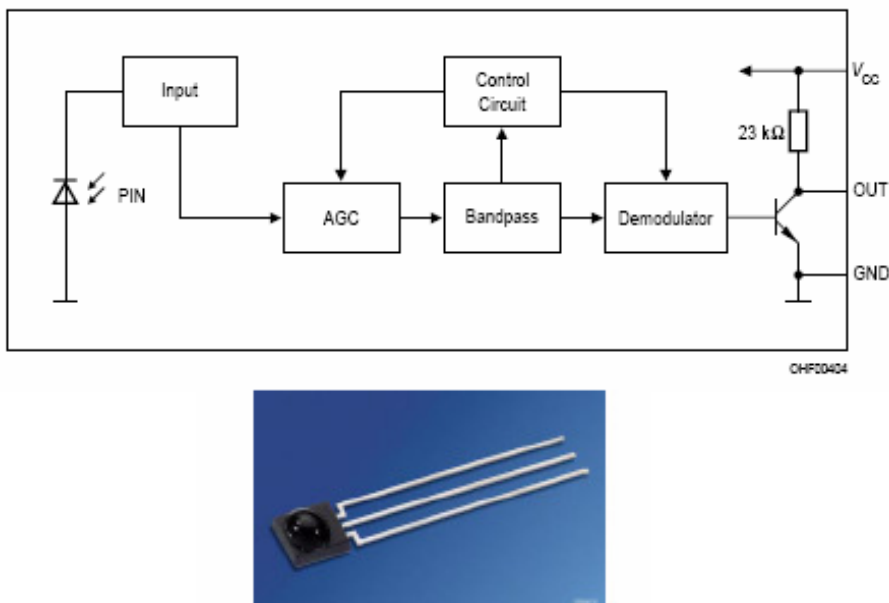
Obr. 3.3 Ukázka některých IR součástek (převzato z [7])

#### 3.2.1 Přijímač SFH 5110

Přijímač SFH 5110 je součástka sloužící, jak už název napovídá, pro příjem a dekódování povelů infračerveného záření. V jediném pouzdře je spojena funkce přijímací diody, zesilovače, limitéru, kmitočtového filtru a demodulátoru. Daný signál je po přijmutí diodou zesílen, a prohnán kmitočtovým filtrem. Ten zamezí průchodu neúčinného signálu, na něž je nastaven (zejména denního světla). Limitérem je přizpůsoben na požadovanou úroveň a následně demodulován.

Výstupem takového přijímače je klasický datový tok RS232 protokolu (sériový tok dat). Pak už stačí pouze výstup tohoto obvodu napojit na příslušný ovládací prvek (např. mikrokontrolér) a data dále zpracovávat.

Na obr.3.4. vidíme vnitřní zapojení a pouzdro takového přijímače. Z důvodu, aby se zamezilo přijímání neúčinného signálu, a minimalizovala se chybovost, je celý obvod zalit do černého plastového pouzdra.



Obr. 3.4 Blokové schéma a použité pouzdro obvodu SFH 5110 (převzato z [6])

### 3.2.2 Průběhy přijímaného signálu

Jako zdroj IR signálu a k ovládání přijímacího modulu slouží dálkový ovladač Logitech Harmony 525. Je to univerzální ovladač, který je řízen pomocí internetu. Během chvilky do něj můžeme „nahrát“ funkce jakéhokoli dálkového ovladače. Než jej tedy k řízení začneme používat, učiníme tak. Jelikož chceme řídit pouze směr otáčení anténního rotátoru vpravo a vlevo (přičemž na výběr bude z možností jemného otáčení, nebo skokově o určitý úhel), využijí se z jeho mnoha tlačítek využívat pouze 4. Tyto funkce jsou z logického hlediska přiřazeny křížovému ovladači tlačítkům „vpravo, vlevo, nahoru a dolů“. Těm odpovídají jejich ekvivalenty „Vol+, Vol-“, a „Ch.+“, „Ch.-“. Viz obr.3.5. a tab.5.3.

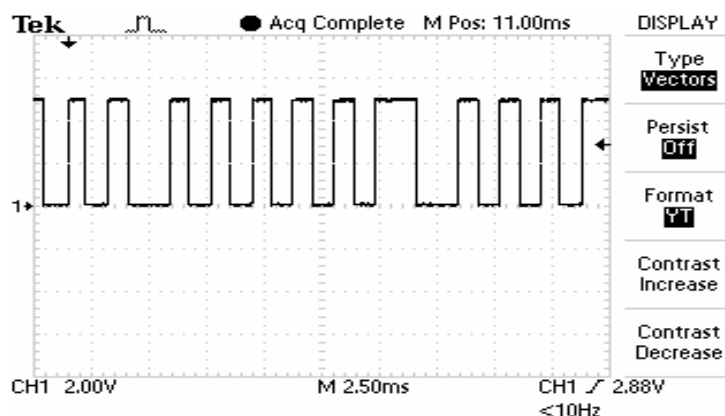
Každé tlačítko dálkového ovladače je reprezentováno určitým kódem. Jedná se o posloupnost bitů, jejíž průběh je dán použitým přenosovým protokolem a použitým typem modulace. V případě ovladače Logitech Harmony 525 se jedná se o 14-ti bitový RC-5 kód (viz.2.4.2), jehož průběhy byly změřeny pomocí osciloskopu. Pro tlačítka „vpravo, vlevo, nahoru a dolů“ jsou kódy tlačítek následující:

- vpravo (Vol.+): 1 1 0 1 1 1 1 1 1 0 1 1 1 1
- vlevo (Vol.-): 1 1 0 1 1 1 1 1 1 0 1 1 1 0
- nahoru (Ch.+): 1 1 1 1 1 1 1 1 1 0 1 1 1 1
- dolů (Ch.-): 1 1 0 1 1 1 1 1 1 0 1 1 1 0

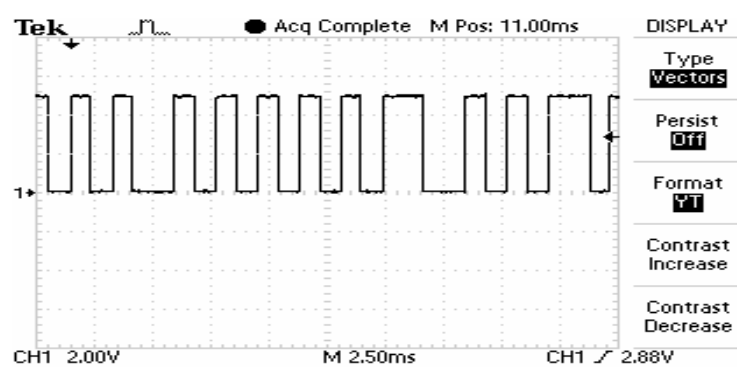


Obr. 3.5 Použitý dálkový ovladač  
(převzato z [9])

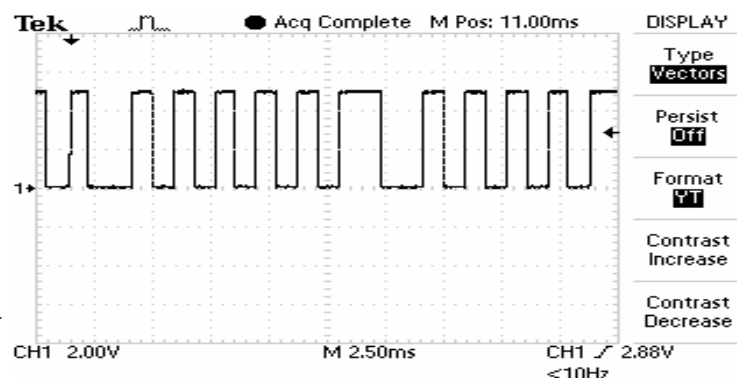
Jak jsme se v grafech 3.1 až 3.4. přesvědčili, jedná se skutečně o kód typu RC-5. Začíná dvěma start bity logické úrovně „1“. Následuje toggle bit, měnící se s každým dalším stiskem tlačítka, 5 adresových bitů 6 a datových bitů, které jsou rozhodující pro každé tlačítko. Datový rámec nemá oproti teoretickým předpokladům délku přesně 24,8888ms. Ta byla změřena na osciloskopu a činí 24,4ms.



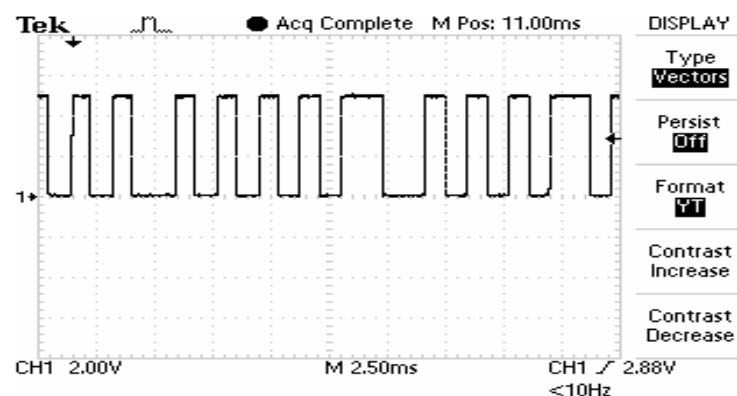
Graf. 3.1 Přijatý kód při stisku tlačítka „vpravo“



Graf. 3.2 Přijatý kód při stisku tlačítka „vlevo“



Graf. 3.3 Přijatý kód při stisku tlačítka „nahoru“



Graf. 3.4 Přijatý kód při stisku tlačítka „dolu“

### 3.3 Zapojení

Navrhovaný přijímač bude sloužit k pohodlnému dálkovému ovládní anténního rotátoru a po jeho dokončení se uloží namísto stávající ovládací jednotky. Jeho funkcí je řízení směru otáčení stejnosměrného motoru vlevo a vpravo (a to po jemných krocích, nebo skokově o určitý úhel). Z technického hlediska půjde o přepínání polaroty napájení na motoru rotátoru za pomoci dvou stejnosměrných relé. Ty jsou řízeny mikrokontrolérem ATtiny85. Celé zařízení je konstruováno pro napájecí napětí 5V. Pro pohon rotátoru se použije 12 V spínaný zdroj.

#### 3.3.1 Popis funkce

Schéma zapojení je vidět v příloze A1. Na konektor J1 je připojeno transformované a usměrněné napětí v rozsahu 7 až 9V. To je přes diodu D1 přivedeno na stabilizátor IC2 5V a společně s kondenzátory C1 až C4 (filtrační kondenzátory) zajišťuje napájení celého obvodu. Dioda LED1 s odporem R1 slouží pouze ke kontrole zapnutého/vypnutého stavu. Hlavní částí celého zapojení je mikrokontrolér Atmel řady AVR ATtiny85. Pro příjem infra signálu je použit přijímač a dekodér v jednom pouzdře SFH 5110 popsany v kapitole 3.2.1. Výstup tohoto obvodu je připojen na pin PB0 mikrokontroléru IC1. Hodnoty odporu R5 a kondenzátoru C5 jsou voleny podle doporučeného katalogového zapojení výrobce.

Spínání motoru a přepínání jeho chodu vlevo a vpravo je řešeno pomocí dvou relé ovládaných obvodem IC1. Na pin PB4 mikrokontroléru je přes bázevý odpor R2 a spínací tranzistor T1 připojeno relé K1. To přepíná polaritu napětí z 12 V externího spínaného zdroje na svorkách X1-3 a X1-4 (sloužících pro připojení stejnosměrného motoru rotátoru). Samotný zdroj se připojí na svorky X1-1 a X1-2. Jelikož toto relé obsahuje 2 přepínací kontakty, a po připojení napájení motoru by docházelo k samovolnému „otáčení“ vlevo, nebo vpravo je nutné zapojení osadit ještě jedním spínacím kontaktem. K tomu to účelu je na pin PB3 připojeno přes spínací tranzistor T2 relé K2. To, jelikož má k dispozici 2 přepínací kontakty, slouží zároveň jako řízený elektronický spínač pro externí spínaný napájecí zdroj rotátoru. Jeho svorky jsou ve schématu označeny X2-1 a X2-2. Spínaný zdroj i v pohotovostním režimu odebírá určitý minimální proud a dochází tak k zbytečné spotřebě energie. Celkový mechanismus přepínání polaroty je vidět ve spodní části schématu.

Aby bylo možné programově řešit snímání, popřípadě nastavování, úhlu natočení o určitý počet stupňů, musí zapojení obsahovat jakousi zpětnou vazbu. Tu tvoří přesný referenční zdroj napětí 2,56V (dioda D4 a rezistor R6). Ten bude přes svorkovnici X3 napojen na potenciometr zabudovaný v rotátoru. Na něm vzniká podle úhlu natočení (odporu potenciometru) úbytek napětí, který je snímán ADC převodníkem (ADC1-pin PB2) a převeden na jeho digitální obraz. ATtiny85 obsahuje 10-bitový ADC převodník s možností 4 multiplexovaných vstupů. Rotátor se bude otáčet v rozmezí 0-360° a tudíž bude potřeba minimálně 360 stavů. Jelikož je převodník 10-bitový je možno snímat až 1024 stavů, tak je plně dostačující. Zapojení dále obsahuje ISP konektor sloužící k pohodlnému programování přímo v aplikaci. Je označen SV1. Více o ISP programování v kapitole 3.3.2.

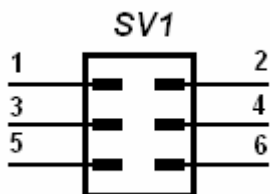
Kondenzátor C6 je pouze jako odrušovací kapacita motoru. Dioda LED2 bude sloužit k signalizaci příjmu infra kódu.

Protože vnitřní schéma obvodu ATTiny85 obsahuje interní pull-up rezistory, jsou k I/O pinům PB3 a PB4 mikrokontroléru IC1 připojeny paralelně rezistory R7 a R8. Ty slouží k tomu, aby při programování obvodu přímo v aplikaci (start po resetu) nedocházelo k samovolnému přepnutí pinu do logické úrovně „H“ a tím k „cvakání“ kontaktů relé.

### 3.3.2 ISP programování

V dnešní době je programování mikroprocesorů za pomoci klasických programátorů již zastaralé. Například u mikrokontrolérů ATMEL jej nahrazuje již běžné programování pomocí rozhraní ISP. Zkratka ISP (In System programming) znamená programování přímo v aplikaci. Není tedy nutné při každém přeprogramování mikrokontrolér vyjmát z obvodu a vkládat do patice programátoru. Ten se pouze pomocí ISP rozhraní propojí s naší aplikací, která obsahuje programovaný procesor. Obvody programovatelné metodou ISP tedy využívají ISP rozhraní. To dle [7] obsahuje:

- Napájecí napětí: **VTG** (+Ucc) a **GND** (pro případ že programátor nemá vlastní napájecí napětí)
- Datové signály: **MOSI** (*Master Out – Slave In*) – sériový vstup dat do mikroprocesoru  
**MISO** (*Master In – Slave Out*) – sériový výstup dat z mikroprocesoru  
**SCK** (*Serial clock*) – sériové hodiny (synchronizace komunikace)  
**/RST** Reset-nulování mikroprocesoru (převzato z [4])



Obr. 3.6 Doporučené zapojení ISP pro Atmel AVR

Jedná se o 6-ti pinový konektor který je vidět na obr. 3.6. Je vhodné, dodržovat standardní rozložení jednotlivých pinů tak, jak je napsáno v tab.3.2.

Tab.3.2 Doporučené rozložení pinů pro Atmel AVR

Vývod 1	MISO	Vývod 2	VTG (Ucc)
Vývod 3	SCK	Vývod 4	MOSI
Vývod 5	RST	Vývod 6	GND

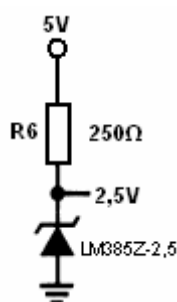
### 3.3.3 Volba a výpočet jednotlivých součástek

**Stabilizátor napětí (IC2):** Pro napájení celého obvodu je použit stabilizátor napětí 78M05. Jelikož se v zapojení vyskytují 2 relé, a předpokládá se celkový odběr proudu přijímače větší než 100mA, byl zvolen v pouzdře TO220. To umožňuje odběr do proudu až do hodnoty 500mA za předpokladu použití chladiče. Bude tedy nutné brát při návrhu desky plošných spojů v potaz chlazení obvodu pevným chladičem.

**Mikrokontrolér (IC1):** Viz. Kapitola 3.1.1.

**Přijímač SFH 5110:** Slouží pro příjem a dekódování IR povelů z dálkového ovládání. Vyžaduje napájení 4,5 až 5,5 V. Hodnoty odporu R5 a kondenzátoru C5 jsou voleny dle doporučení výrobce. Dále viz. 3.2.1.

**Referenční zdroj 2,56V (D4):** Pro tuto funkci se jako nejvhodnější zdál obvod LM385Z-2,5. Výrobce doporučuje do série přiřadit rezistor. Jeho hodnota se vypočítá dle vztahu (3.2):



Podle katalogu má téct obvodem LM385 proud  $I=10\text{mA}$ . Jelikož úbytek napětí na něm je 2,5V, pak na R6 je:

$$U_{R6} = U - 2,5 = 5 - 2,5 = \underline{2,5V} \quad (3.1)$$

pak hodnotu R6 vypočteme dle vztahu:

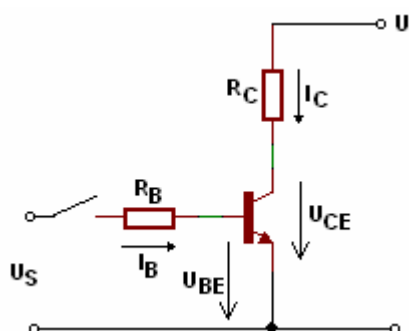
$$R6 = \frac{U_{R6}}{I} = \frac{2,5}{10^{-3}} = \underline{\underline{250\Omega}} \quad (3.2)$$

**Relé pro přepínání chodu (K1):** Zde padla volba na G5V2-5 OMRON. Obsahuje 2 přepínací kontakty. Je ovládáno stejnosměrným napětím 5V. Odpor jeho cívky je 50Ω.

**Relé pro spínání chodu (K2):** Má 2 funkce. Slouží jako spínač chodu motoru a zároveň jako elektronický vypínač napájecího zdroje rotátoru. Jelikož bude toto relé spínat kontakt 230V AC musí být na to výkonově přizpůsobeno. Vhodné je například MILLINSPOT H820F05C. Obsahuje 2 přepínací kontakty a je ovládáno DC napětím 5V. Odpor jeho cívky je 47 Ω. Maximální spínaný proud 5A.

**Tranzistory T1 a T2:** Jsou klasické NPN tranzistory BC546B v pouzdře TO092 zapojené ve spínacím režimu.

**Výpočet hodnot bazových odporů R2 a R3 pro spínací tranzistory T1 a T2:**



Obecné schéma zapojení tranzistoru ve spínacím režimu je vidět na obrázku.

**Výpočet R2 ( $R_B$ ):**

$R_C = 50\Omega$  (odpor cívky relé),  $U_{CE} = 0,4V$ ,

$U_{BE} = 0,7V$ ;  $U = 5V$ ;  $U_S = 5V$



$$I_C = \frac{U - U_{CE}}{R_C} = \frac{5 - 0,4}{50} = \underline{0,092A} . \quad (3.3)$$

Aby byl tranzistor v saturaci a tedy spolehlivě fungoval jako spínač, volí se  $I_B$  asi 20 krát menší než  $I_C$ . Jeho hodnotu vypočteme dle vztahu (3.4):

$$I_B = \frac{I_C}{20} = \frac{0,092}{20} = \underline{4,6mA} . \quad (3.4)$$

Známe-li velikost proudu do báze  $I_B$ , pak dle (3.5) vypočteme hodnotu  $R_2$ :

$$R_2 = \frac{U - U_{BE}}{I_B} = \frac{5 - 0,7}{4,6 \cdot 10^{-3}} = \underline{\underline{934,8\Omega}} \quad (3.5)$$

**Obdobně vypočteme hodnotu  $R_3$ :**

$R_C = 47\Omega$  (odpor cívky relé),  $U_{CE} = 0,4V$ ,  $U_{BE} = 0,7V$   $U = 5V$ ,  $U_S = 5V$

$$I_C = \frac{U - U_{CE}}{R_C} = \frac{5 - 0,4}{47} = \underline{97,9mA}$$

$$I_B = \frac{I_C}{20} = \frac{97,9 \cdot 10^{-3}}{20} = \underline{4,89mA}$$

Pak hodnotu  $R_3$  vypočteme ze vztahu (3.8):

$$R_3 = \frac{U - U_{BE}}{I_B} = \frac{5 - 0,7}{4,89 \cdot 10^{-3}} = \underline{\underline{879,3\Omega}} \quad (3.8)$$

Z řady E12 tedy volíme:  $R_2 = 1k\Omega$ ,  $R_3 = 820\Omega$

**ISP konektor (SV1):** Možností je více. V tomto zapojení je použit typ ML6 6-ti pinový s klíčem aby nedocházelo k záměně polarity.

Dále schéma obsahuje různé pasivní součástky. Jejich hodnoty byly zvoleny dle doporučení katalogových zapojení výrobců, obvykle užívaných hodnot a dalších typických zapojení. Výsledné navržené schéma zapojení je příloze A1 na konci práce.

## 4 REALIZACE ZAPOJENÍ

V předchozí části projektu je popsána problematika přenosu dat pomocí infračerveného záření. Na základě těchto informací bylo navrženo schéma zapojení přijímače dálkového ovládání pro řízení modulu anténního rotátoru. Další část projektu je zaměřena více prakticky. Zabývá se návrhem desky plošných spojů (DPS) v programu Eagle. Výsledná navržená DPS je vyrobena a osazena součástkami. Tato část také obsahuje informace o návrhu chladiče pro stabilizátor napětí, který je nesmírně důležitý pro celkovou správnou funkci celého zapojení. Na závěr kapitoly jsou zde sepsány informace, jak propojit navržený obvod s přístrojem, který bude ovládat.

### 4.1 Návrh Desky plošných spojů

Přijímač je postaven na jedné jednostranné desce plošných spojů o rozměrech 65\*89 mm. Velikostně je situována tak, aby se v případných dalších krocích konstrukce mohla vejít například do krabičky U-KM29, kterou lze zakoupit v síti prodejen GM Electronic. Výsledná navržená DPS je vidět v příloze A3, a je vytvořena metodou spojitých čar s použitím klasických součástek. Při jejich rozmístování na budoucí DPS bylo nutné brát v potaz fakt, že pomocí relé K2 bude spínán síťový přívod napětí 230VAC pro spínaný zdroj. Tomuto je deska uzpůsobena dodržení bezpečné izolační vzdálenosti mezi síťovou částí (relé K2 a svorkovnice X2) a ostatními částmi desky.

Na levé krajní straně DPS nalezneme svorkovnici X1, sloužící pro přepínání polarity na motoru rotátoru a konektor J1 pro připojení napájecího napětí. Propojení svorkovnic X1 a X2 s relé K1 a K2 je realizováno dostatečně širokými cestami. A to z toho důvodu, že zde budou spínány proudy v řádu až jednotek ampér. Programovací konektor SV1 je situován do spodní části desky přímo pod integrovaný obvod. Zjednoduší se tím jeho propojení s IO1. Blokovací kondenzátory C3, C4 a C7 se umísťují dle doporučení co nejbližší integrovaným obvodům. Obzvláště pak C7 přímo k napájecímu pinu 8 obvodu IC1 z důvodu, aby nedocházelo k zákrmitům napájecího napětí. IR přijímač může být osazen přímo, zasunut do patice, nebo spojen s deskou propojovacími kablíky. Je důležité, aby zem tohoto obvodu byla spojena se zemí desky. Zjednoduší se tím pak jeho případné umístění na libovolnou stranu krabičky dle polohy přípravku. Rozmístění dalších „klasických“ součástek již nevyžaduje speciální pozornost a jsou tedy umístěny tak, jak to bylo možné.

Nakonec bylo nutné v návrhu zajistit chlazení stabilizátoru, sloužícího pro napájení celého přípravku. To je realizováno pevným hliníkovým chladičem DO1A pro pouzdro TO220. Má výkon 25K/W a je umístěno pod stabilizátorem, se kterým je spojen pomocí šroubku. Vypočet jeho parametrů je popsán v kapitole 4.3. Aby se ještě více zvětšila tepelná vodivost mezi pouzdem stabilizátoru napětí a chladičem, je vhodné použít na tento přechod malé množství teplo vodivé pasty.

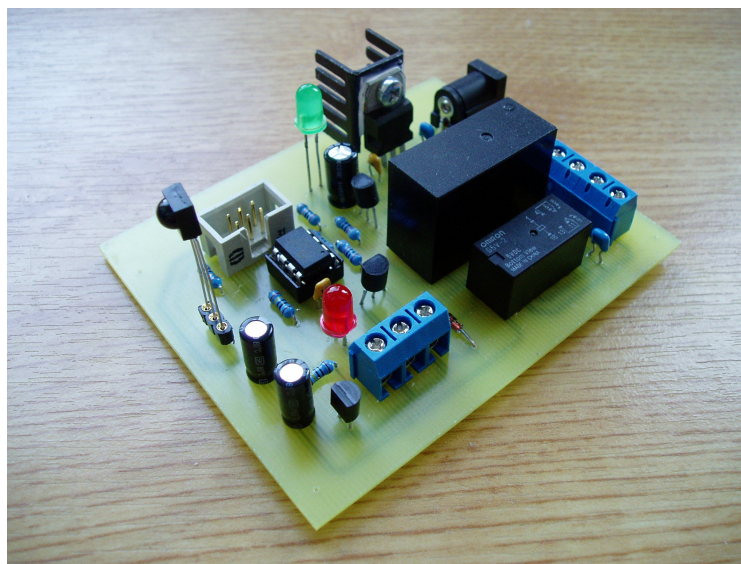
### 4.2 Osazení a mechanická konstrukce

Osazení navržené desky není složité. Nejprve je nutné vyvrtat díry pro jednotlivé součástky. Pro ty klasické, jako rezistory a kondenzátory, se doporučuje použít vrták o

průměru 0,8mm. Díry pro svorkovnice, konektory a relé je nutné zvětšit na průměr 1,2mm-1,5mm dle vývodů. Máme-li vyvrtány všechny díry, je dobrým zvykem DPS ze strany spojů nějak ochránit před oxidací vznikající postupem času. Můžeme k tomu použít nejrůznějších laků nebo například postačí trochu kalafuny rozpuštěné v technickém lihu.

Nyní je deska připravena k osazení. Jako první doporučuji zapájet pasivní součástky, svorkovnice a konektory. Po té diody, tranzistory a nakonec integrované obvody. Při pájení je potřeba dodržet správnou polaritu některých součástek. Především jsou to elektrolytické kondenzátory a diody. Dále pozor na polaritu tranzistorů! V dalším kroku vypočteme dle kapitoly 4.3. velikost vhodného chladiče a ten přišroubujeme k stabilizátoru. Nakonec doporučuji vyvrtat v rozích otvory přibližně průměru 3mm, které usnadní připevnění desky do krabičky. Na obr. 4.1 je vidět osazená deska připravená k programování za pomoci ISP programátoru.

Před připojením napájecího napětí zasuneme integrovaný obvod do patice. Pozor při tom na polaritu, klíč obvodu musí směřovat vzhůru! Nyní můžeme připojit napájení. Měla rozsvítit indikační dioda LED1, která nám signalizuje zapnutí obvodu. Pokud se tak nestane, zkontrolujeme napájení a odstraníme případné chyby.



Obr. 4.1 Hotová osazená deska přijímače připravená k programování pomocí ISP

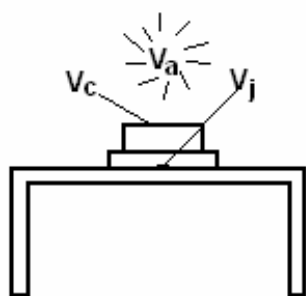
### 4.3 Výpočet parametrů chladiče

Chlazení polovodičových prvků je velmi důležité. V případě větších odebíraných proudů nezbytné a zvyšuje účinnost součástky. V následujících řádcích je sepsán jednoduchý postup jak pomocí několika katalogových hodnot zjistit velikost plochy potřebného chladiče pro stabilizátor napětí.

U každé součástky, kterou je potřeba chladit, uvádí výrobce maximální výkonovou ztrátu. Jedná se o takovou hodnotu výkonu, kterou je schopna předat kvalitnímu chladiči. Chladičů se vyrábí velké množství druhů od klasických pevných, aktivních (větráčky) až po vodní chladiče. Při běžné funkci vzniká na čipu součástky přebytečné

teplo, které se značí  $V_j$ . To přechází hned přes několik tepelných odporů (čip-pouzdro, pouzdro-chladič a chladič-teplota okolí), které brání jeho vyzaření do okolí. Aby se součástka nepřehřála musejí být tyto odpory co nejmenší. Tepelný odpor mezi čipem součástky a jeho pouzdrem je dán výrobou a nezměníme jej. Co však ovlivnit můžeme je tepelný odpor mezi pouzdrem a chladičem. Snažíme se, aby byla styčná plocha co největší. Můžeme použít teplotně vodivé silikonové pasty. Tepelný odpor mezi chladičem a okolím minimalizujeme natřením chladiče matnou černou barvou. Prospívá také umístění chladiče do horizontální polohy (lepší proudění vzduchu)

Navrhovaný chladič pro stabilizátor napětí zobrazený na obr.4.2, je řešen jako klasický kovový chladič chlazený pouze okolním vzduchem.



Použité konstanty:

C1.....materiálová konstanta: Al = 1, Cu = 0,75

C2.....vyzařovací konstanta: pro Al = 2 mm je rovna 0,43  
svisle 0,5 vodorovně

Va.....teplota okolí: klasicky 45°C

Vc.....teplota pouzdra

Vj.....maximální teplota přechodu Si = 200°C, Al = 100°C

K1.....vnitřní tepelný odpor

K2.....vnější tepelný odpor (převzato z [11])

Obr. 4.2 Parametry pro výpočet chladiče

Než začneme se samotným výpočtem plochy chladiče, tak si nejprve určíme hodnotu maximálního ztrátového výkonu [11]:

$$P = (U_1 - U_2) \cdot I_{OUT\max} = (9 - 5) \cdot 0,5 = 2W \quad (4.1)$$

kde:  $U_1$ ...vstupní napětí stabilizátoru

$U_2$ ...výstupní napětí stabilizátoru

$I_{OUT\max}$  ....maximální odebíraný proud

### 1) Výpočet teploty pouzdra $V_c$ [11]

$K1 = 3^\circ C/W$  (určeno z katalogu),  $P = 0,5A$  (maximální odběr přijímače)

$$V_c = V_j - (K1 \cdot P) = 200 - (3 \cdot 2) = 194^\circ C \quad (4.2)$$

### 2) Výpočet vnějšího odporu $K2$ [11]

$$K2 = \frac{V_c - V_a}{P} = \frac{194 - 45}{2} = 74,5^\circ C / W \quad (4.3)$$

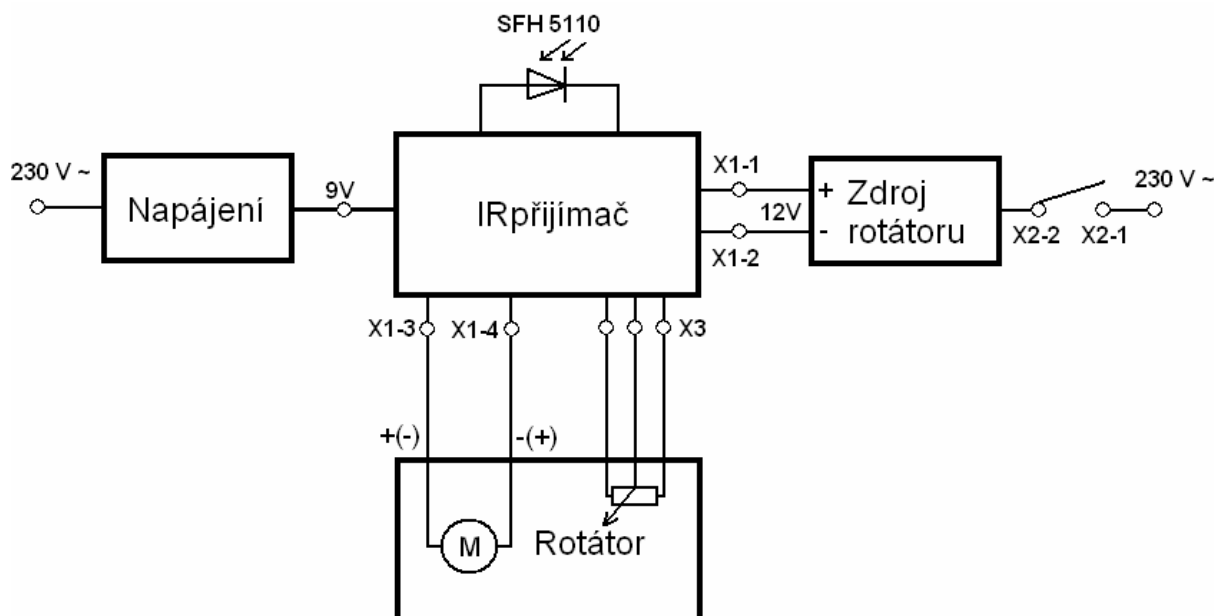
### 3) Plocha chladiče [11]

$$S = \frac{C1 \cdot 7,6 \cdot 10^2 \cdot C2}{K2 - C1 \cdot 1,73} = \frac{1 \cdot 7,6 \cdot 10^2 \cdot 0,43}{74,5 - 1 \cdot 1,73} = 4,49 cm^2 \quad (4.4)$$

Dle výpočtu by měla být plocha chladiče asi  $4,5\text{cm}^2$ . Může mít tedy rozměry např.  $15 \times 30\text{ mm}$  a můžeme jej ohnout do U.Z katalogu výrobce byl zvolen chladič DO1A, který s plochou  $6,49\text{cm}^2$  dostatečně vyhovuje požadavkům.

## 4.4 Propojení modulu přijímače a rotátoru

Blokové schéma propojení přijímacího modulu s rotátorem je vidět na obr .4.3. Přijímač bude napájen ze síťového zdroje s výstupním napětím 7-9V. Ke svorkám X1-1 a X1-2 je připojen spínaný 12 V zdroj sloužící pro pohon stejnosměrného motoru rotátoru. Pomocí mechanismu přepínání relé K1 a K2 je na svorkách X1-3 a X1-4 přepínána polarita  $+(-)$  a  $- (+)$ . K těmto svorkám je připojen samotný motor sloužící k otáčení vlevo (a vpravo). Jelikož je osa přístroje mechanicky spojena s interním zabudovaným potenciometrem, dochází současně při otáčení z jedné krajní polohy do druhé i k točení tohoto potenciometru. Tím se nastavuje jeho odpor. Z referenčního zdroje 2,5V je přiváděno napětí na svorky X3. Jeho úbytek je pak opět připojen ke svorkám X3 a přiveden na mikrokontrolér. Této skutečnosti je využito a je popsána v kapitole.5.2. Jelikož se dá říct, že i v klidovém stavu (když je přijímač zapnut a připraven k přijímání povelů) spotřebovává spínaný napájecí zdroj rotátoru nějakou energii, je přijímač vybaven dalšími svorkami označenými X2. Ty fungují jako elektronický spínač. Programově je řešen tak, že při každém platném přijatém povelu k točení se nejprve sepne a spustí napájecí zdroj. Pak se provede příslušný úkon a poté se opět vypne. Tím se na dobu nečinnosti zdroj odpojí a šetří tak energii. Čidlo pro příjem IR povelů z dálkového ovladače je z praktických důvodů usazeno v patici. V případě potřeby je možno jej umístit na libovolnou stranu a s paticí přijímače jej propojit vodiči. Pozor ovšem na polaritu! Zem obvodu spojit se zemí desky.



Obr. 4.3 Blokové schéma propojení rotátoru a modulu IR přijímače

## 5 PROGRAM

Přijímač dálkového ovládání je sice navržen a zkonstruován, ale bez příslušného ovládacího programu pro mikrokontrolér nemůže být využit. V následujících řádcích jsou popsány nejdůležitější body při vytváření programu. V kapitole 5.1. nalezneme důležité informace týkající se externího přerušení, kterého je využíváno při dekódování kódu dálkového ovladače. Je zde uveden především výčet registrů včetně jejich popisu a konkrétního nastavení pro vytvářenou aplikaci. Dále zde najdeme informace potřebné pro pochopení problematiky AD převodníku. Stručný soupis jeho vlastností a konkrétní popis nastavení pro řešení problematiky přesného nastavování úhlu natočení rotátoru.

### 5.1 Přerušení (Interrupt)

Někdy nastane případ, že potřebujeme, aby program reagoval na nějakou určitou vnitřní nebo vnější událost. Tou událostí může být například přetečení čítače/časovače, dokončení převodu AD převodníkem, změna logické úrovně na pinu procesoru a.t.d. V okamžiku, kdy procesor této události vyhoví, nastane tzv. přerušení. Při něm se tedy normální běh programu přerušuje a CPU pokračuje ve výkonu programu na jiném místě vykonáním příslušné obslužné rutiny. Po jejím dokončení se program vrátí na místo, kde došlo k přerušení.

Každé přerušení má svůj tzv. vektor přerušení. Je to adresa programu, kam CPU skočí při vyvolání konkrétního přerušení. Vektorů přerušení je hned několik a každý procesor má své vlastní. Záleží na perifériích, kterými je daný procesor vybaven. Nalezneme je v datasheetu, v sekci interrupts. V tab. 5.1. jsou vidět některé příklady možných zdrojů přerušení pro procesor ATtiny85 [4].

Tab. 5.1 Některé příklady přerušení pro ATtiny85 (převzato z [4])

Vektor č.	Programová adresa	Zdroj	Definice
1	0x0000	RESET	Externí reset
2	0x0001	INT0	Externí požadavek na přerušení 0
3	0x0002	PCINT0	Požadavek na přerušení při změně úrovně pinu
4	0x0003	TIMER1_COMPA	TIMER1 shoda s A
...	...	...	...
14	0x000D	USI_START	Start USI
15	0x000E	USI_OVF	Přetečení USI

Této události bude využito jako hlavního principu při dekódování kódu dálkového ovladače. Možností je několik a v následující kapitole je popsán jeden z nich.

### 5.1.1 Nastavení registrů pro přerušení PCINT0

Pro práci s přerušením jsou u procesoru AVR vyhrazeny určité registry. Jejich výčet nalezneme v datasheetu pro každý mikrokontroler. Jejich nastavením určujeme, při jaké vnější/vnitřní události dojde k přerušení. Vnější vstupy přerušení jsou ovládány pomocí registrů **MCUCR**, **GIMSK** a **GIFR**.

V našem případě je požadováno, aby docházelo k externímu přerušení PCINT0 vždy, když se změní hodnota logické úrovně na pinu PB0 (kam připojen výstup z IR přijímače). Provedeme tedy následující nastavení:

#### Registr GIMSK ( General Interrupt Mask Register ) –masky přerušení

V tomto registru povolíme externí přerušení reagující na změnu logické úrovně na určitém pinu. Konkrétně se jedná o bit **PCIE** (Pin Change Interrupt Enable) a učiníme tak jeho nastavením do logické úrovně „H“

Rozložení jednotlivých bitů v registru:

Bit	7	6	5	4	3	2	1	0
	-	INT0	PCIE	-	-	-	-	-
Počáteční hodnota	0	0	0	0	0	0	0	0

Registr **GIMSK** tedy naplníme hodnotou:

0	0	1	0	0	0	0	0
---	---	---	---	---	---	---	---

#### Registr PCMSK ( Pin Change Mask Register )

Jelikož je zvoleno přerušení generující se při změně logické úrovně, musíme ještě v registru **PCMSK** nastavit příslušný bit. Jedná se o bity PCINT0 – PCINT5.

Rozložení jednotlivých bitů v registru:

Bit	7	6	5	4	3	2	1	0
	-	-	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0
Počáteční hodnota	0	0	0	0	0	0	0	0

Protože je signál z výstupu přijímače SFH5110 přiváděn na pin PB0 mikrokontroléru (PCINT0), dle kapitoly 3.1.1. musíme nastavit bit PCINT0 na hodnotu „H“

Tedy registr **PCMSK** naplníme hodnotou:

0	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---

## SREG (Status Register)

Status registr, nebo také stavový registr, je jedním z nejdůležitějších registrů procesorů AVR. Obsahuje informace o výsledcích různých aritmetických a jiných operací. Navíc obsahuje bit sloužící k globálnímu povolení všech přerušení. Jedná se o 8-bitový registr. Rozložení jeho jednotlivých bitů je vidět na obrázku:

Bit	7	6	5	4	3	2	1	0
	I	T	H	S	V	N	Z	C
Počáteční hodnota	0	0	0	0	0	0	0	0

Význam:

- I - globální povolení/zakázání přerušení
- T - transfer bit-používaný při kopírování bitu
- H - příznak přetečení-nastaví se pokud dojde k přetečení mezi nižší a vyšší polovinou výsledku
- S - druhý doplněk
- V - přeplnění v druhém doplňku
- N- příznak záporného výsledku
- Z - nulový výsledek
- C - příznak přetečení (převzato z [12])

Aby mohlo dojít k obslužení externího přerušení, je nutné v registru SREG ještě nastavit globální povolovací bit na hodnotu „H“. To se provedeme zapsáním instrukce *sei( )* do smyčky hlavního programu. Zpravidla se tato instrukce píše jako poslední. Tímto se naplní registr SREG hodnotou:

1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

Vše je pak připraveno ke skoku do obsluhy přerušení.

K zápisu příslušných bitů registrů do log. úrovně „H“ v smyčce hlavního programu je výhodné použít operace bitového posuvu. Neovlivňuje se tím stav ostatních bitů registru. Tento zápis může mít tvar například:

```
GIMSK = GIMSK | (1<<PCIE); //nastav bit PCIE registru GIMSK do „H“
```

## 5.2 Využití AD převodníku

Jako většina mikroprocesorů AVR, obsahuje i ATtiny85 zabudovaný AD převodník. Pomocí něj je možné například měřit napětí, ale i jiné veličiny, které se dají na napětí převést. Jedná se o převodník typu s postupnou aproximací, který obsahuje 4 multiplexované vstupy (PB2, PB3, PB4 a PB5). Dva z nich dokonce s možností nastavování zisku. To dává uživateli na výběr z celé řady způsobů, jak jej využít. Na další stránce je výčet jeho základních parametrů.



Základní parametry (převzato z [4]):

- AD převodník s postupnou aproximací
- Nastavitelné rozlišení až 10 bitů
- Čas převodu v rozmezí 65-250  $\mu$ s
- Možnost 4 multiplexovaných vstupů
- Nastavitelná vnitřní reference 1.1V nebo 2.56V
- Různé možnosti spouštění
- Možnost skoku do přerušení po dokončení převodu

Jak je již psáno výše (v kap.3.3.1) při otáčení rotátoru dochází zároveň ke změně odporu zabudovaného potenciometru.Úbytek napětí na něm je přiváděn na vstup AD převodníku (pin PB2) mikrokontroléru a je z něj získávána jeho digitální podoba.Ta je pak následně zpracována procesorem,vyhodnocena a odpovídá úhlu natočení.

Interval 360° natáčení byl rozdělen do 12 stejných dílů (po 30°).Po vyslání odpovídajícího kódu z ovladače dojde k sepnutí relé.Na pinu PB2 se začíná měnit hodnota napětí, která je ihned převáděna.Dojde-li k poklesu,resp.nárůstu této hodnoty o předem definovaný díl ,procesor ji vyhodnotí jako nadefinovaný skok.Následně pak pošle na příslušný pin log. L.Tím se vypne relé a zastaví se otáčení.

### 5.2.1 Nastavení AD převodníku

AD převodník je obdobně jako přerušení řízen několika speciálními registry.Celý jejich výčet včetně popisu funkcí nalezneme v datasheetu obvodu a nebude zde dále rozebírán.Popsány budou pouze ty,které je potřeba nastavit pro vytvářenou aplikaci.

#### Registr ADMUX

Tento registr slouží k nastavení referenčního napětí (bity REFS0-REFS2).Dále k zarovnání výsledku převodu (ADLAR) a výběru převáděného kanálu (MUX3-MUX0).Zde provedeme dle datasheetu následující nastavení.Zvolíme vnitřní referenci 2,56V a vybereme kanál ADC1 (pin PB2) na kterém bude prováděn převod (MUX0).Ostatní bity ponecháme ve výchozím nastavení.

Rozložení jednotlivých bitů v registru:

Bit	7	6	5	4	3	2	1	0
	REFS1	REFS0	ADLAR	REFS2	MUX3	MUX2	MUX1	MUX0
Počáteční hodnota	0	0	0	0	0	0	0	0

Registr ADMUX naplníme hodnotou:

1	0	0	1	0	0	0	1
---	---	---	---	---	---	---	---

## Registr ADCSRA

Je dalším klíčovým registrem, sloužící pro nastavování AD převodu. Řídí celý proces a informuje nás o jeho výsledcích. Aby převodník fungoval dle výchozích požadavků, musíme v tomto registru nejprve provést následující nastavení:

- Bitem ADEN zapneme/vypneme převodník
- Pomocí bitů ADPS2-ADPS0 nastavíme dle datasheetu (kapitola AD převod) obvodu interní před-děličku hodinového signálu tak, aby se frekvence pohybovala v intervalu 50-200kHz. Vypočteno z (5.1)
- Povolíme převod na vybraném kanálu bitem ADCS
- Ostatní bity ponecháme v původním nastavení.

Rozložení jednotlivých bitů v registru:

Bit	7	6	5	4	3	2	1	0
	ADEN	ADCS	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0
Počáteční hodnota	0	0	0	0	0	0	0	0

Při frekvenci 8MHz nastavíme interní před-děličku na 64.

$$f_{CLK_{ADC}} = \frac{f_{CPU}}{64} = \frac{8 \cdot 10^6}{64} = \underline{125000Hz} \quad (5.1)$$

Tímto získáme pro AD převodník hodinový signál o frekvenci 125KHz.

Registr **ADCSRA** naplníme hodnotou:

1	1	0	0	0	1	1	0
---	---	---	---	---	---	---	---

## Registr ADCL a ADCH

Do tohoto registru se zapisují výsledky převodu a od ostatních se liší zejména tím, že se skládá ze dvou 8 bitových registrů. Výsledek operace je zde zapsán 10 bity a zarovnán standardně doprava (bit ADLAR v registru ADCSRA = 0) nebo případně doleva (ADLAR = 1). Zbytek bitů je není využit.

Pro jednoduchý převod platí následující vztah

$$ADC = \frac{V_{in} \cdot 1024}{V_{REF}} \quad (5.2)$$

kde:  $V_{in}$  ..... Vstupní (převáděné) napětí  
 $V_{REF}$  ... Referenční napětí (v našem případě 2,56V)  
1024....10 bitové rozlišení převodníku

Výsledek operace je vždy po dokončení převodu zapsán do registru ADCL a ADCH. V programu použijeme pro čtení hodnot po AD převodu příkaz *ADCW*.

Rozložení jednotlivých bitů v registru pro ADLAR = 0:

Bit	15	14	13	12	11	10	9	8
ADCH	-	-	-	-	-	-	ADC9	ADC8
ADCL	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0
Bit	7	6	5	4	3	2	1	0

## 5.2.2 Převod napětí-úhel natočení

Chceme snímat hodnotu natočení v rozsahu 0-360° a každých 30° vypnout relé. Referenční napětí je nastaveno na vnitřní  $V_{REF} = 2,56V$ . Rozlišení převodníku 10-bitové. Daný interval je rozdělen na 12 dílů po 30° a je sestavena převodní tabulka:

Tab.5.2 Převodní tabulka úhel natočení-hodnota ADCL a ADCH po AD převodu

Úhel natočení [°]	$R_{pot} [\Omega]$	U [V]- převáděné	Hodnota převodu v ADCW Dec	Hodnota převodu v ADCW Bin
0	0	0	0	0000000000
30	417	0,213	85	0001010101
60	833	0,426	170	0010101010
...	...	...	...	...
330	4583	2,346	938	1110101010
360	5000	2,560	1024	1111111111

Dle (5.2) se vypočte pro jednotlivá analogová napětí jejich digitální obraz a zapíše do tab.5.2. Z ní lze vyčíst, že při změně úhlu o 30° se sníží (resp. zvýší) převáděné napětí asi o 0,213V, což je ve výsledku převodu hodnota asi 85 DEC. To znamená, že se při kladné (resp. záporné) změně v registrech ADCL a ADCH o tuto hodnotu, rotátor otočí o 30°. Pro čtení převáděné hodnoty vložíme do programu následující funkci [14], která vrací hodnotu po převodu (*ADCW*). V programu je pak nutné obě krajní polohy natočení, resp. jejich hodnoty napětí změřit a krok 30° korigovat upravením hodnoty kroku po AD převodu.

```

unsigned int get_volt(void) //vrací výsledek převodu
{
    ADCSRA = ADCSRA | (1<<ADSC); // start převodu
    while (!(ADCSRA & (1<<ADIF))); //čeká na převod
    ADCSRA = ADCSRA & 0xFF;
    return ADCW; //vrací hodnotu ADCW (převzato z [14])
}

```

## 5.3 Popis programu

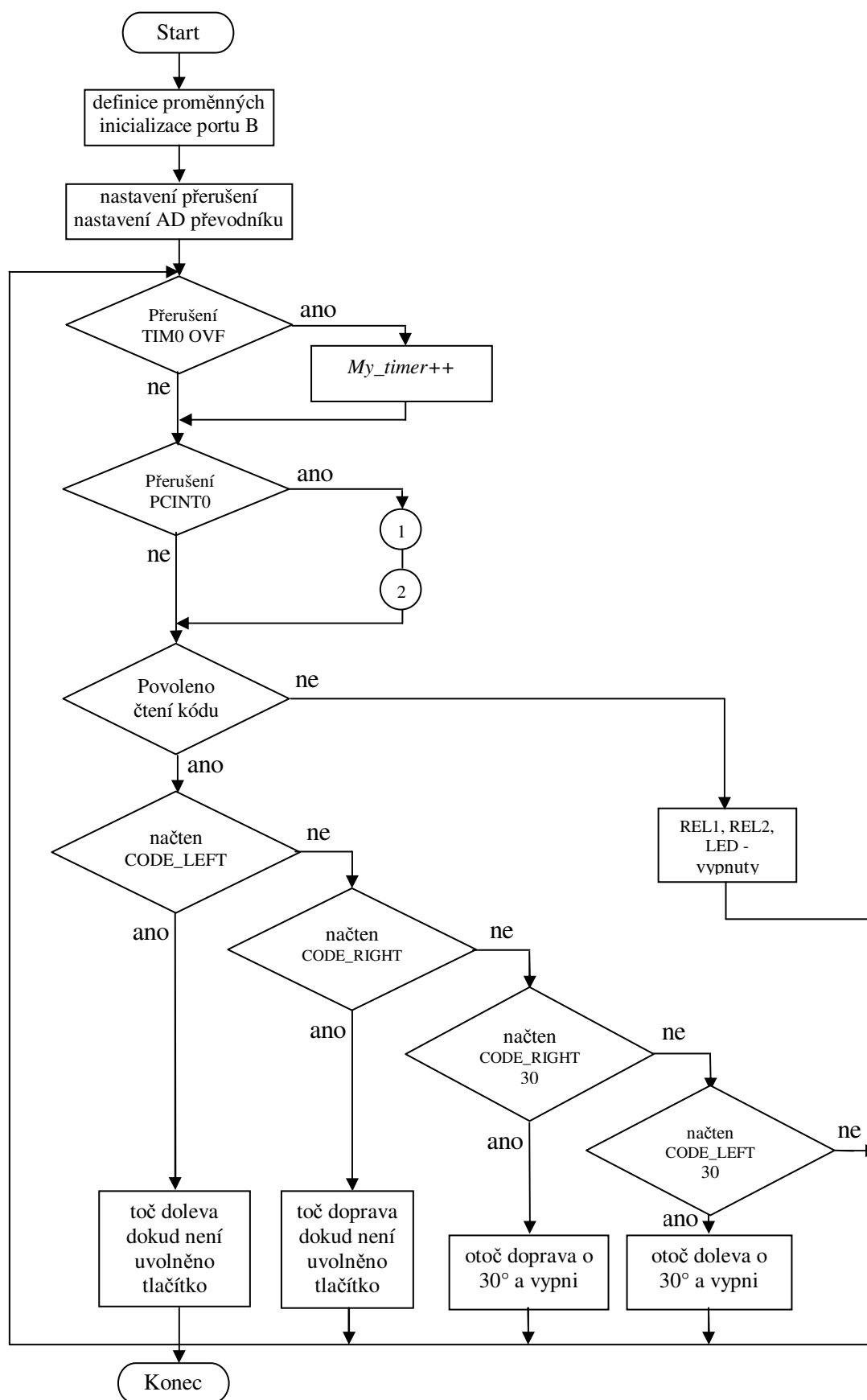
Celý program pro mikrokontrolér je napsán v jazyce C, což přináší své výhody i nevýhody. Jeho hlavní výhodou je, že je výsledný program mnohem kratší a jeho vytváření pohodlnější. Na druhou stranu je však mnohem náročnější na paměť a jeho výkon je pomalejší. Programátor tak ztrácí úplnou „kontrolu“ na průběhem svého programu. Jako vývojový prostředek byl zvolen volně dostupný nástroj od společnosti ATMEL, AVR studio.

### 5.3.1 Hlavní program

Princip hlavního programu je poměrně jednoduchý a celý jeho běh je naznačen ve vývojovém diagramu na obr. 5.1. Jeho část nalezneme včetně všech komentářů v příloze B1. Celý program je pak na přiloženém CD. Na začátku jsou umístěny funkce pro správný běh. Nejprve je nadefinována proměnná *value*, inicializuje se port B a nastaví se, které piny budou pracovat jako vstupní (slouží pro čtení dat z infra přijímače) a výstupní (ovládání-relé). Následuje nastavení externího přerušení a přerušení při přetečení interního čítače/časovače viz. kapitola 5.1.1. Dále je nastaven dle kapitoly 5.2.1. zabudovaný AD převodník. V registru SREG se všechna přerušení povolí nastavením bitu I do log. „H“ a spustí se časovač. Program pak běží v nekonečném cyklu a čeká, na změnu logické úrovně na pinu PB0. Ta signalizuje přijetí kódu z ovladače. V případě, že tak nastane, program skočí do obsluhy přerušení PCINT0. Tam se pomocí algoritmu postupně dekóduje přijatý kód a ten je pak uložen do určité proměnné. Program pak skočí opět do hlavní smyčky, kde se výsledek algoritmu porovnává s hodnotou nadefinovanou v hlavní části programu. Rovná-li se tyto hodnoty, provede se určitá posloupnost příkazů. Ta zajistí nastavení příslušných výstupních pinů mikrokontroléru do logických úrovní „H“ a tím sepnutí příslušných relé. V opačném případě je program připraven k přijetí a testování dalšího kódu.

#### Příklad vyhodnocení povelu „Točení vpravo-skokově o 30 °“

- Přijímač byl připojen k napájecímu napětí => spuštěn hlavní program
- Definovaly se proměnné a inicializoval se port B
- Nastavila se potřebná přerušení včetně globálního povolení v SREG a bylo provedeno nastavení AD převodníku; spustil se časovač
- Program běží v nekonečné smyčce; po každém přetečení č/č je vyvoláno přerušení;
- Čeká se na změnu úrovně na pinu PB0-přijetí kódu dálkového ovladače
- kód je přijat a pomocí externího přerušení postupně vyhodnocen; je povoleno jeho čtení
- Je přijatý kód shodný s hodnotou *CODE\_LEFT* - ne, proto se testuje dál
- Je přijatý kód shodný s hodnotou *CODE\_RIGHT* - ne, testuje se dál
- Je přijatý kód shodný s hodnotou *CODE\_RIGHT\_30* - ano
- Provedou se příslušné instrukce-otočení doprava o 30° a následné vypnutí
- Program je připraven k příjmu dalšího kódu

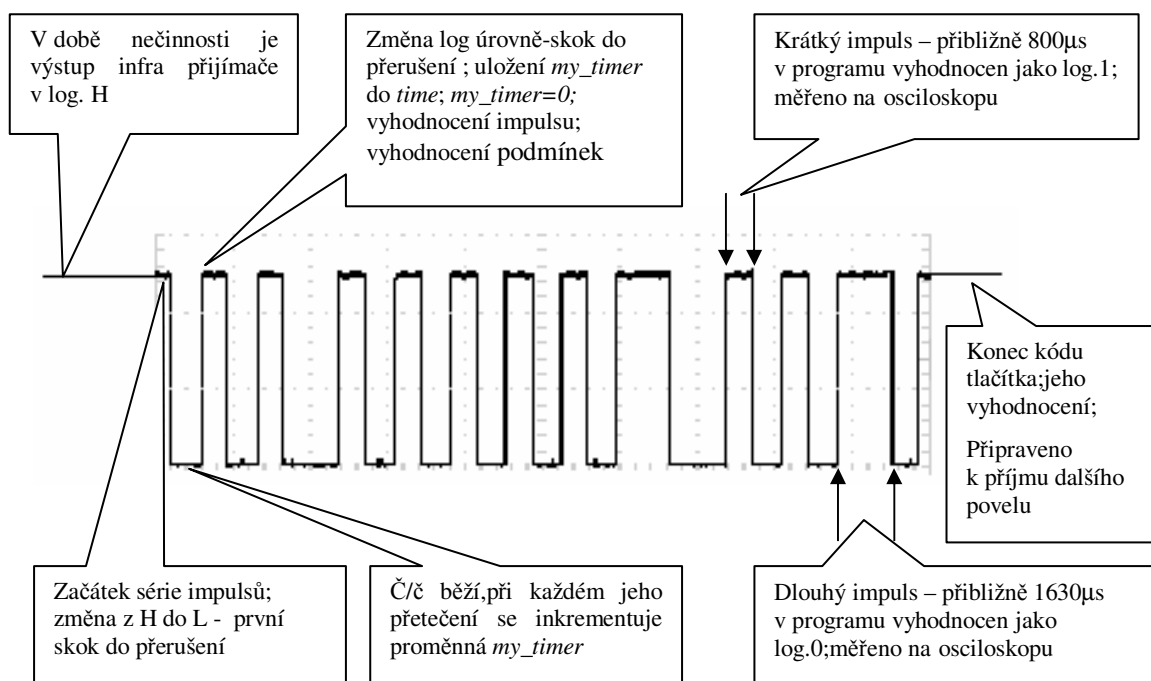


Obr. 5.1 Vývojový diagram hlavního programu

### 5.3.2 Dekódování RC-5 kódu ovladače

Jak už bylo napsáno v teoretické části, ovládání je realizováno pomocí ovladače využívající RC-5 přenosový protokol. Způsobů, jak rozpoznat kódy jednotlivých tlačítek je hned několik. Pro tento účel jsem vytvořil algoritmus který pracuje následovně:

Základním principem, zobrazeným na obr. 5.2, je měření délky jednotlivých impulsů za pomoci interního čítače/časovače. Ty se pak postupně ukládají do pomocné proměnné a na konci cyklu se porovnávají s nadefinovanými hodnotami.



Obr. 5.2 Princip dekódování kódu dálkového ovladače





V době, kdy není na ovladači zmáčknuto žádné tlačítko, je výstup infra přijímače SFH5110 v logické úrovni "H" a program běží v hlavní smyčce. Zde je povoleno externí přerušení na změnu logické úrovně. Zároveň je povoleno přerušení při přetečení interního čítače/časovače. Po zmáčknutí příslušného tlačítka je vyslán odpovídající kód. Při první změně logické úrovně (z log H do log L) nastane přerušení. Nejprve se nadefinuje pomocná proměnná *time*, která slouží pro porovnání a vyhodnocování jednotlivých impulsů. Do této proměnné se uloží hodnota z *my\_timer*, která se po sléze vynuluje. Nyní se testují jednotlivé podmínky. Vzhledem k tomu, že se jedná o první změnu logické úrovně, a v *time* je uložena hodnota neodpovídající žádné z těchto podmínek, program skočí zpět do hlavní smyčky. Čeká se na další změnu (tentokrát z log. L do log. H) a čítač/časovač stále běží. Po každém jeho přetečení (při frekvenci 8MHz vždy po 32µs) dojde ke skoku do obsluhy přerušení TIM0\_OVF. Tam se inkrementuje hodnota pomocné proměnné *my\_timer* a program skočí opět do hlavní smyčky. Nyní přijde další změna log. úrovně (L do H) a postup se opakuje. Tentokrát je však do *time* uložena hodnota z *my\_timer*, která odpovídá délce měřeného impulsu. Proměnná *my\_timer* se vynuluje a začíná se testovat.

Pohybuje-li se načtená hodnota v intervalu 600-900 $\mu$ s (což odpovídá 18-28 přetečeními č/č), proměnná *code* se nejprve posune o jeden bit doleva. Na pozici nultého bitu se pak uloží log.1. a *step\_rotation* inkrementuje svoji hodnotu (přičte 1). Je-li naopak v intervalu 1400-2100 $\mu$ s (43-65 přetečení), *code* se pouze posune opět o jeden bit a na nultou pozici se neuloží nic, resp. uloží se log 0. a *step\_rotation* také inkrementuje svoji hodnotu. Při každém dalším příchodu přerušení (změně logické úrovně) se celý postup opakuje. Do *code* se tak postupně ukládají jednotlivé bity. Protože je celý kód každého tlačítka tvořen 23 takovými bity a ukládá se pouze posledních 8, je zbytek je přemazán. Tudíž nejsou uloženy start bity, toggle bit a adresa ovladače. To má výhodu v tom, že je teoreticky možné používat i jiný ovladač, který dodržuje normu kódování RC-5 a používá odpovídající kódy tlačítek.

Dojde-li k naplnění *step\_rotation* hodnotou 23 (přečtení celého kódu), povolí se jeho čtení a *code* se uloží do *code\_2*. Nastane skok do hlavního programu, kde se vyhodnotí výsledky. Vývojový diagram naznačující dekodování RC-5 kódu dálkového ovladače je vidět na obr. 5.3.

V hlavním programu jsou vyhodnocovány 4 povely. Jejich souhrn se stručným popisem je vidět v následující tabulce:

Tab.5.3 Povely dálkového ovladače

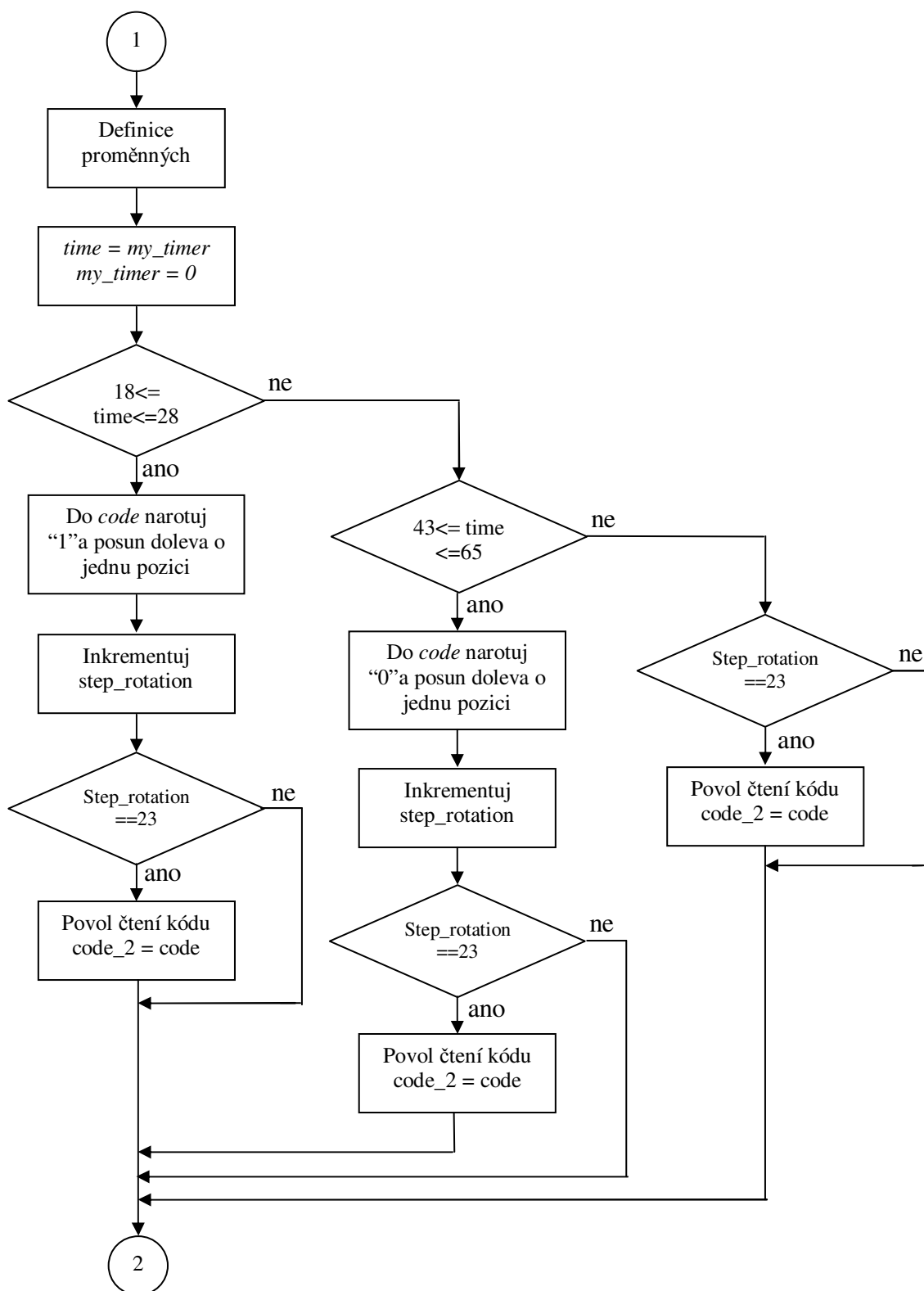
Povel	Jeho kód	Symbol na ovladači	Popis
CODE_LEFT (vlevo)	FE7A		Točení vlevo-jemně
CODE_RIGHT (vpravo)	FE7E		Točení vpravo-jemně
CODE_LEFT_30 (dolu)	F9FE		Točení vlevo-skokově o 30°
CODE_RIGHT_30 (nahoru)	F9FA		Točení vpravo-skokově o 30°

### Točení vlevo - jemně; Točení vpravo-jemně

Je-li přijatý kód programem vyhodnocen jako CODE\_LEFT respektive CODE\_RIGHT, dojde k sepnutí relé. Při zmáčknutí a následném držení dochází k točení doleva (doprava). Úkon je ukončen v okamžiku, kdy uživatel uvolní na ovladači příslušné tlačítko, nebo dojde k otočení do krajní polohy.

### Točení vlevo-skokově o 30°; Točení vpravo-skokově o 30°

Je-li naopak vyslaný kód ovladače vyhodnocen programem jako CODE\_LEFT\_30 případně CODE\_RIGHT\_30. Dojde k sepnutí odpovídajících relé na dobu, než nastane otočení o úhel 30°.



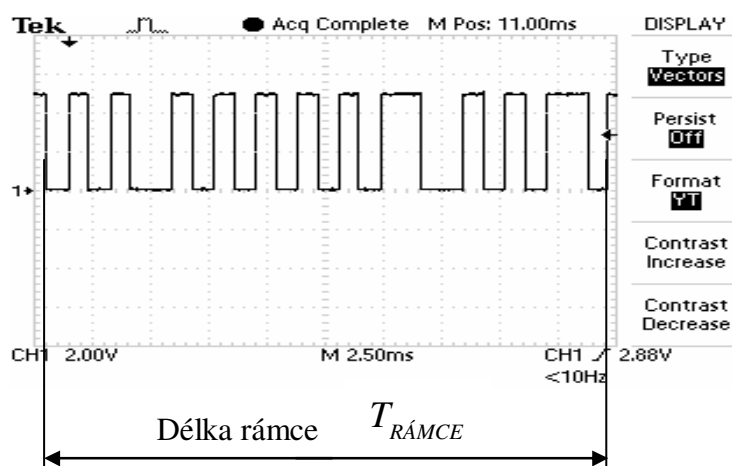
Obr. 5.3 Vývojový diagram dekódování kódu dálkového ovladače



## 5.4 Porovnání nosných kmitočtů ovladačů

Při vytváření algoritmu pro dekódování RC-5 kódu dálkového bylo zjištěno, že délka rámce použitého ovladače úplně přesně neodpovídá teoretickým poznatkům. Tedy ani nosná frekvence modulace není přesně  $f_0 = 36\text{kHz}$ . Odchylka od přesného udávaného kmitočtu  $f_0$  je zřejmě způsobena použitím nekvalitních RC oscilátorů. Svůj vliv na tuto chybu má zajisté i teplotní rozsah.

Z tohoto důvodu jsem se rozhodl změřit a porovnat několik ovladačů od různých výrobců z hlediska této frekvence. Výsledky jsou sepsány v tab.5.4.



Obr. 5.4. Datový rámec

### Postup:

Nejprve byly za pomoci osciloskopu změřeny a zapsány do tab.5.4. délky rámců (viz obr.5.4.) všech testovaných dálkových ovladačů. Víme, že každý bit přenosového protokolu RC-5 je přenášen jako 32 pulsů o teoretické nosné frekvenci  $f_0 = 36\text{kHz}$ . Takovýchto bitů obsahuje každý rámec 14. Délka rámce je 24,8888ms. Platí tedy:

$$14 \text{ bitů} \cdot 32 \text{ pulsů} \cdot 2 \cdot \frac{1}{f_0} = 24,8888 \text{ms}$$

Z tohoto lze dle vztahu (5.4) vypočítat nosnou frekvence  $f_0$  jednotlivých ovladačů:

$$f_0 = \frac{32 \cdot 2 \cdot 14}{T_{\text{RÁMCE}}} = \frac{896}{24,4 \cdot 10^{-3}} = \underline{\underline{36,721 \text{kHz}}} \quad (5.4)$$

kde: 32·2...počet pulsů jednoho modulovaného bitu (impuls a mezera)  
 14.....počet bitů rámce  
 $T_{\text{RÁMCE}}$ ...délka rámce

Tab.5.4 Porovnání kmitočtů dálkových ovladačů

Ovladač	Použitý protokol	Změřená délka rámce [ms]	Vypočtená frekvence [kHz]	Použití
Logitech Harmony 525	RC-5	24,4	36,721	Universální
SENCOR SPT 310	RC-5	25,1	35,697	CD přehrávač
OVP RC 2024	RC-5	23,3	38,455	Televize
PHILIPS	RC-5	23,0	38,957	Televize

## 5.5 Použité pomůcky

V tab.5.5. je uveden všechny software a hardware, který byl užít při vytváření tohoto projektu.

Tab.5.5 Seznam použitého software a hardware

Nástroj	Verze	Popis
Microsoft Word	2003	Tvorba textu
Eagle	5.3.0	Návrh schématu a plošného spoje
AVR studio	4.16	Vytváření programu pro mikrokontrolér
ASIX UP	2.41	Nahrání programu do mikrokontroléru
ASIX PRESTO	-	Programátor USB

## 6 ZÁVĚR

V první části bakalářské práce jsem sepsal teoretické informace důležité pro návrh přijímače dálkového ovládání. Ten slouží pro řízení směru otáčení anténního rotátoru. Zjistil jsem, že pro zabezpečení přenosu za pomoci infračerveného dálkového ovládání je potřeba data modulovat. Použitý dálkový ovladač využívá RC-5 přenosový protokol. Jeho jednotlivé bity jsou modulovány za pomoci Bi-phase modulace na nosnou frekvenci 36kHz. V laboratoři jsem změřil a v grafech 3.1 až 3.4 zobrazil průběhy signálu pro tlačítka, která jsou využívána pro otáčení. Je z nich vidět, že se skutečně jedná o nejpoužívanější protokol RC-5.

V další části jsem se zabýval návrhem takového přijímače. Celé zapojení je řízeno jedním mikrokontrolérem. Ten byl vybrán na základě požadavků sepsaných v kapitole 3.1. Jedná se o obvod ATMEL AVR ATtiny85. Důležité je, že obsahuje externí přerušování, dále AD převodník (pro zpětnou vazbu) a má dostatečnou paměť pro program. Dále jsem vybral vhodný obvod pro příjem kódu z dálkového ovladače. Typ SFH 5110 (viz. 3.2.1) splňoval všechny požadavky, protože obsahuje několik bloků v jednom pouzdře a na jeho výstupu je již připraven signál pro další zpracování. Po výběru těchto dvou hlavních částí přijímače jsem v programu Eagle navrhnul konečné schéma zapojení (viz. příloha A.1). Zapojení je z důvodu pohodlnějšího programování osazeno konektorem ISP.

Následuje část, ve které je popsána samotná realizace přijímače. Nejprve jsem v programu Eagle navrhnul desku plošných spojů. Je vytvořena metodou spojitých čar. Má rozměry 64,8 x 88,9 mm. Při návrhu jsem upustil od možnosti rozlitého společného kontaktu země přes celou desku (polygonu). Tu jsem pouze v návrhu zvýraznil větší šířkou vodiče (z důvodu možného rušení). Dále bylo nutné oddělit dostatečnou izolační mezerou část, která spíná síťové napětí 230V (relé K2 se svorkovnicí X2), od zbytku obvodu. Zařízení jsem osadil součástkami a dle 4.3 vypočetl plochu potřebného chladiče. Ta vyšla asi 4,5 cm<sup>2</sup>. Z katalogu jsem vybral typ DO1A, který s plochou 6,49 cm<sup>2</sup> vyhovuje. Chladič jsem přišrouboval ke stabilizátoru. Výsledný výrobek je vidět na obr. 4.1.

V poslední, 5. kapitole jsem se zabýval vytvořením programu pro mikrokontrolér, který rozpozná přijatý kód ovladače a na základě jeho vyhodnocení provede příslušnou operaci (točení vlevo, vpravo). Jeho části jsou obsaženy v příloze B na konci dokumentu. Celý program je pak uložen na připojeném CD. Základem je dekódovací algoritmus pro rozpoznání RC-5 kódu ovladače (viz 5.3.2 a vývojový diagram na obr. 5.3.) Přijímač je za pomoci programu také schopen otáčet rotátorem vždy o určitý úhel (nastaveno na 30°) při zmáčknutí příslušného tlačítka na DO. Na konci této kapitoly jsem praktickým měřením průběhů ovladačů různých výrobců změřil jejich nosné frekvence a porovnal v tab. 5.4. Ta se u ovladačů Logitech Harmony 525 a SENCOR SPT 310 pohybovala kolem 36kHz. U OVP RC 2024 a Philips to byla hodnota okolo 38kHz.

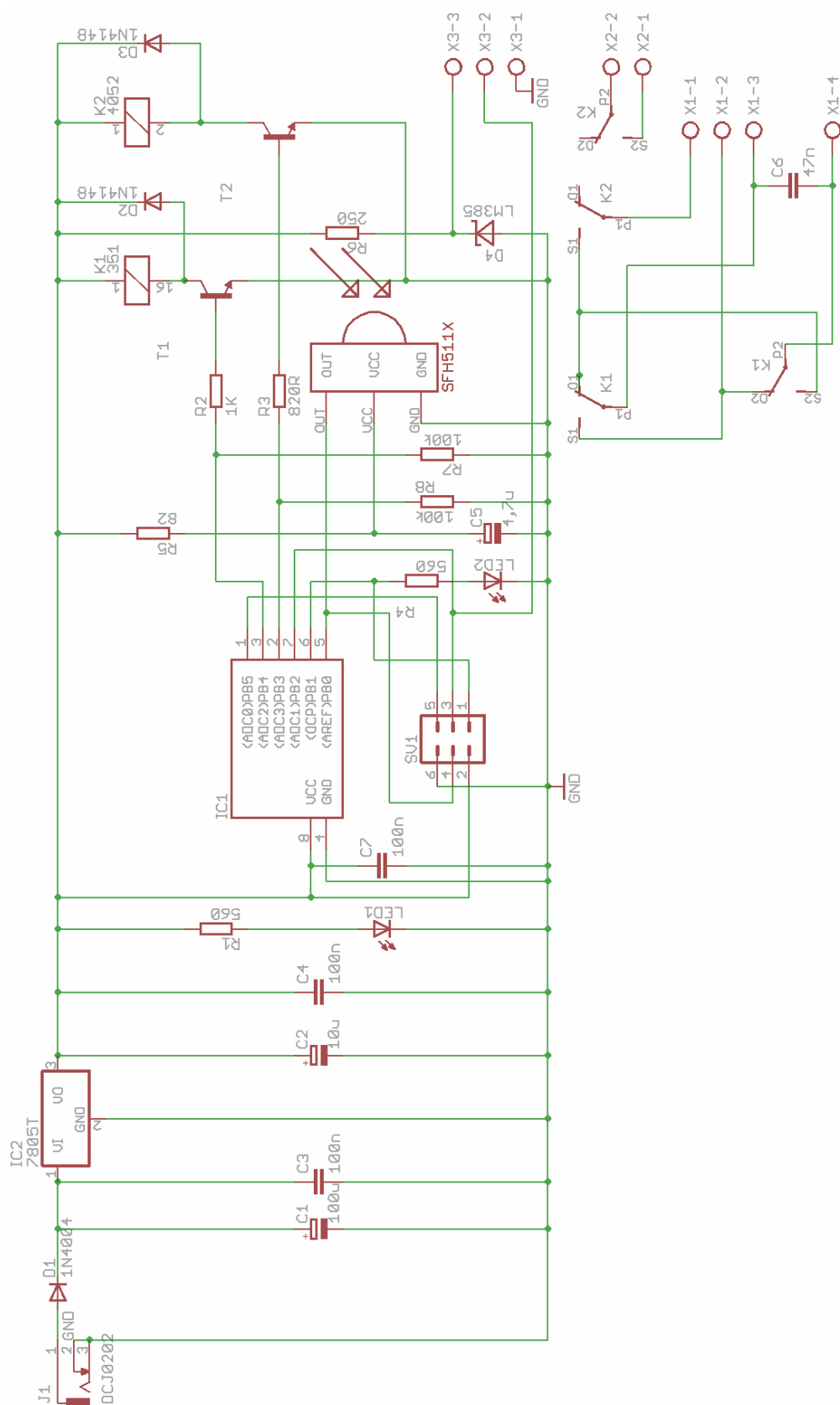
Navržený a vyrobený přijímač pro dálkové ovládání by jistě našel v praxi i jiné uplatnění, než to, kvůli kterému byl zkonstruován. Napadá mě například dálkové spínání domácích spotřebičů a podobně.

## 7 POUŽITÁ LITERATURA:

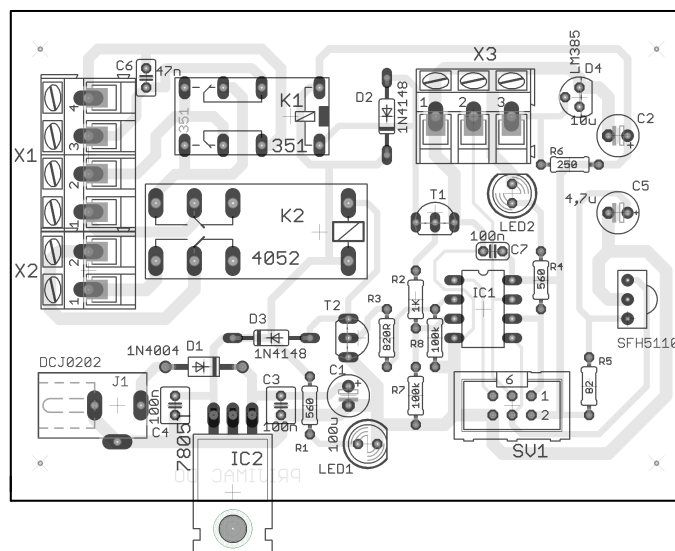
- [1] MATOUŠEK, D. *Práce s mikrokontroléry ATMEL AVR - 3. díl - uP a praxe*. 1.vyd. Praha: BEN - technická literatura, 2003.
- [2] *Http://office.microsoft.com* [online]. 2003 , 12.dubna 2004 [cit. 2008-11-12]. Dostupný z WWW: <<http://www.microsoft.com>>.
- [3] Váňa, V. *Mikrokontroléry ATMEL AVR : popis procesoru a instrukční soubor*. 1. vyd. Praha : BEN - technická literatura, 2003. 336 s
- [4] *www.atmel.com : Datasheet* [online]. 1998 [cit. 2008-12-08]. Dostupný z WWW: <<http://www.atmel.com>>.
- [5] *Http://www.irda.org/* [online]. 1998 [cit. 2009-01-03]. Dostupný z WWW: <<http://www.irda.org/>>.
- [6] *Http://www.datasheetcatalog.com/* [online]. 2005 [cit. 2008-12-15]. Dostupný z WWW: <[http://www.datasheetcatalog.com/datasheets\\_pdf/S/F/H/5/SFH5110-40.shtml](http://www.datasheetcatalog.com/datasheets_pdf/S/F/H/5/SFH5110-40.shtml)>.
- [7] *Http://hw.cz/* [online]. 2000 [cit. 2009-01-03]. Dostupný z WWW: <<http://hw.cz/>>.
- [8] PINKER, J. *Mikroprocesory a mikropočítače*. 1. vyd. [s.l.] : BEN, 2004. 160 s.
- [9] *Www.logitech.com* [online]. 2000[cit. 2009-05-25]. Dostupný z WWW: <http://www.logitech.com/index.cfm/67/4725&cl=cz,cs>
- [10] MIŠUREC, J., ZEMAN, V., ŠTĚPÁN, M. *Konstrukce elektronických zařízení - návrh plošných spojů*. Skriptum VUT Brno: 2003. s. 1-72.
- [11] *WWW.pandatron.cz : elektronický magazín* [online]. 2000-2009 [cit. 2009-05-22]. Dostupný z WWW: <[http://pandatron.cz/?219&vypocet\\_chladice](http://pandatron.cz/?219&vypocet_chladice)>.
- [12] *WWW.programujte.cz : zaměřeno na informační technologie* [online]. 2004-2009 [cit. 2009-05-22]. Dostupný z WWW: <<http://programujte.com/index.php?rubrika=70-elektrotechnika&sekce=179-avr>>.
- [13] Váňa, V. *Atmel AVR programování v jazyce C*. Praha : BEN - technická literatura, 2003. 216 s., 1 CD-ROM.
- [14] KARAS, Ondřej. *Http://www.kvetakov.net/* [online]. Pardubice : 2006 [cit. 2009-05-24]. Dostupný z WWW: <<http://www.kvetakov.net/clanky/avr/>>. ISSN 1803-1021.

# A NÁVRH ZAŘÍZENÍ

## A.1 Obvodové zapojení

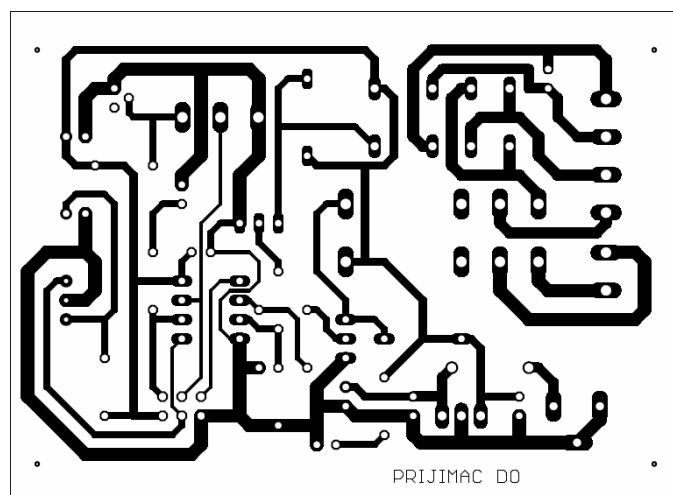


## A.2 Deska plošného spoje – top (strana součástek)



Rozměr desky 64,8 x 88,9 [mm], měřítko M1:1

## A.3 Deska plošného spoje – bottom (strana spojů)



Rozměr desky 64,8 x 88,9 [mm], měřítko M1:1

## A.4 Použité součástky

Označení	Hodnota	Popis
C1	100u / 16V	Elektrolytický kondenzátor
C2	10u / 16V	Elektrolytický kondenzátor
C3,C4	100n	Keramický kondenzátor
C5	4,7u / 16V	Elektrolytický kondenzátor
C6	47n	Keramický kondenzátor
C7	100n	Keramický kondenzátor
D1	1N4004	Universální dioda
D2,D3	1N4148	Universální dioda
D4	LM385	Referenční zdroj
IC1	ATTINY85PU	Mikrokontroler ATMEL
IC2	7805T-T0220	Stabilizátor napětí
IR	SFH 5110	IR přijímač
J1	DCJ0202	Napájecí konektor
K1	G5V2-5 OMRON	DC Relé
K2	H820F05C MILLINSPOT	DC Relé
LED1 , LED2	LED 5mm	Led dioda
R1,R4	560R	Rezistor
R2	1K	Rezistor
R3	820R	Rezistor
R5	82R	Rezistor
R6	250R	Rezistor
R7,R8	100k	Rezistor
SV1	ML6	Programovací konektor
T1,T2	BC546B-NPN-TO92	Tranzistor NPN
X1	AK300/4	Svorkovnice
X2	AK300/2	Svorkovnice
X3	AK300/3	Svorkovnice

## B ČÁSTI ZDROJOVÉHO KÓDU

### B.1 Část hlavního programu

```
int main(void)
{
    unsigned int value; //definice proměnné pro AD převod

    DDRB = (1<<PB3)|(1<<PB4); //nastavení pinu PB a PB4 jako vstupní
    GIMSK = GIMSK | (1<<PCIE); //přerušování PCIE povoleno
    PCMSK = PCMSK | (1<<PCINT0); //pin PB0
    TCCR0B = TCCR0B | (1<<CS00); //spustí časovač, bez před děličky
    TIMSK = TIMSK | (1<<TOIE0); //povolí přerušování č/č
    // nastavení převodníku
    ADMUX = (1<<REFS2)|(1<<REFS1)|(1<<MUX0); //reference 2,56V,
    //kanál MUX0
    ADCSRA = (1<<ADEN)|(1<<ADPS2)|(1<<ADPS1); //zapne převod,
    //nastaví před děličku

    sei(); //globální povolení přerušování

    while(1) { //nekonečná smyčka
        .....
    }
```

### B.2 Dekódovací algoritmus RC-5 kódu

```
ISR(PCINT0_vect) //obsluha externího přerušování PCINT0
{
    unsigned int time; //proměnná pro porovnání délky impulsu

    time = my_timer; //obsah my_timer se uloží do time
    my_timer = 0; //vynulování my_timer
    TCNT0 = 0;

    if(time >= LOG1_DOWN && time <= LOG1_UP) { //je-li time 18-28
        code = (code | 0b00000001) << 1; //do code narotuje "1"
        step_rotation++; //inkrementace step_rotation
    }
    else if(time >= LOG0_DOWN && time <= LOG0_UP) { //je-li time 43-65
        code = (code << 1); //do code narotuje "0"
        step_rotation++; //inkrementace step_rotation
    }
    if(step_rotation==23) { //pokud je step_rotation==23
        code_read = CODE_READ_ENABLE; //je povoleno čtení kódu
        code_2 = code; //do code_2 uložena hodnota z code
    }
}
```