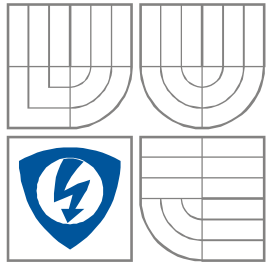


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ
ÚSTAV RADIOELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF RADIO ELECTRONICS

PROGRAMOVATELNÝ IR PŘIJÍMAČ

PROGRAMMABLE IR RECEIVER

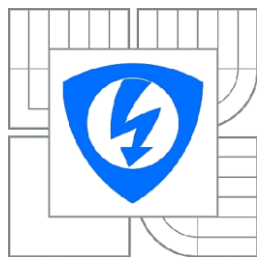
BAKALÁŘSKA PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

DANIEL GÉZE

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. ZBYNĚK FEDRA, Ph.D.



VYSOKÉ UČENÍ

TECHNICKÉ V BRNĚ

Fakulta elektrotechniky

a komunikačních technologií
Ústav radioelektroniky

Bakalářská práce

bakalářský studijní obor
Elektronika a sdělovací technika

Student: Daniel Géze

ID: 115170

Ročník: 3

Akademický rok: 2010/2011

NÁZEV TÉMATU:

Programovatelný IR přijímač

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s protokoly pro IR dálková ovládání (RC5) a s nároky na jejich implementaci v mikrokontroleru. Prostudujte možnosti propojení předpokládaného zařízení s počítačem a vyberte vhodný programovací jazyk pro vytvoření konfigurační aplikace v PC. Navrhněte blokově jednotlivé části zařízení a ovládací aplikace.

Navrhněte a realizujte zapojení s mikrokontrolerem a odpovídajícími periferiemi pro příjem IR dálkového ovládání. Vytvořte rozhraní ve kterém bude možno konfigurovat odezvu na konkrétní kód přijatý přes IR. Zařízení otestujte např. pro ovládání světel, spouštění alarmu a podobně.

DOPORUČENÁ LITERATURA:

[1] BERGMANS, S Philips RC-5 Protocol [online]. SB-Projects, Nizozemí, 2001 - [cit. 20.5.2010].
Dostupné na [www: http://www.sbprojects.com/knowledge/ir/rc5.htm](http://www.sbprojects.com/knowledge/ir/rc5.htm)

[2] MANN, B. C pro mikrokontroléry. BEN - technická literatura, Praha 2003

Termín zadání: 7.2.2011

Termín odevzdání: 27.5.2011

Vedoucí práce: Ing. Zbyněk Fedra, Ph.D.

prof. Dr. Ing. Zbyněk Raida
Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Cieľom tejto bakalárskej práce je návrh a realizácia riešenia pre HW a SW programovateľného IR prijímača. IR prijímač predstavuje logický programovateľný automat, ktorý reaguje na príkazy z diaľkového ovládania vysielané komunikačným protokolom RC6 od firmy PHILIPS. Kódy týchto príkazov možno priradiť k rôznym akciám. Použitím konfiguračnej aplikácie v PC je možné ľubovoľným kódom tlačidiel priradiť pevné kódy príkazov pevne naprogramované vo flexibilnom riadiacom procesore Cypress (CY8C29466). Konfiguračný SW na PC je vytvorený pomocou programovacieho jazyka python. Hlavným účelom tohto celého zariadenia je pomôcť handicapovaným ľuďom.

KLÚČOVÉ SLOVÁ

komunikačný protokol RC6, komunikačný protokol Modbus, programovateľný logický automat, programovateľný IR prijímač, Cypress, python, vývojový diagram, bloková schéma

ABSTRACT

The aim of this bachelor thesis is design and realization of solutions for hardware and software programmable IR receiver. IR receiver is actually a logic programmable automat, which responds to commands from the remote broadcast communications protocol RC6 designed by Philips. Codes of these commands can be assigned to different actions. Using the configuration application in PC by any code of buttons it is possible to order firmly programmed codes in a flexible management processor Cypress (CY8C29466). Configuration software for the PC has been created by using the python programming language. The main purpose of this device is helping for the handicapped people.

KEYWORDS

communications protocol RC6, communications protocol Modbus, programmable logic automat, programmable IR receiver, Cypress, python, flowchart, block diagram

BIBLIOGRAFICKÁ CITÁCIA

GÉZE, D. *Programovatelný IR přijímač*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav radioelektroniky, 2011. 35 s., 13 s. příloh. Bakalářska práce. Vedoucí práce Ing. Zbyněk Fedra, Ph.D.

Prehlásenie

Prehlasujem, že svoju bakalársku prácu na tému "*Programovateľný IR prijímač*" som vypracoval samostatne pod vedením vedúceho bakalárskej práce a s použitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej bakalárskej práce ďalej prehlasujem, že v súvislosti s vytvorením tohto projektu som neporušil autorské práva tretích osôb, hlavne som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a som si plne vedomý následkov porušení ustanovení § 11 a nasledujúcich autorského zákona č. 121/2000 Zb., vrátane možných trestne právnych dôsledkov vyplývajúcich z ustanovení § 152 trestného zákona č. 140/1961 Zb.

V Brne dňa 27. mája 2011

.....
podpis autora

Pod'akovanie

Ďakujem pedagogickému vedúcemu bakalárskej práce Ing. Zbyňkovi Fedrovi, Ph.D. a odbornému vedúcemu RNDr. Petrovi Fabovi, Ph.D. za účinnú metodickú, pedagogickú a odbornú pomoc a ďalšie cenné rady pri spracovaní mojej bakalárskej práce.

V Brne dňa 27. mája 2011

.....
podpis autora

Obsah

Zoznam obrázkov	VIII
Zoznam tabuliek	IX
Úvod	1
1. Návrh riešenia	2
1.1. Komunikačný protokol diaľkového ovládača	2
1.1.1. Výber komunikačného protokolu diaľkového ovládača	2
1.1.2. Rozbor súčasného stavu problematiky	3
1.1.3. Popis protokolu RC6	3
1.1.4. Návrh na detekciu protokolu RC6.....	6
1.2. Vlastnosti programovateľného IR prijímača.....	12
1.2.1. Návrh HW na základe požadovaných vlastností	14
1.2.2. Voľba SW na základe požadovaných vlastností	14
1.2.3. Popis vlastností mikroprocesora Cypress.....	15
1.3. Konfiguračná aplikácia v PC	15
2. Realizácia IR prijímača - popis hardwarovej časti.....	16
2.1. Stručný popis hlavného riadiaceho prvku mikroprocesora CY8C29466.....	16
2.2. Popis vstupno-výstupných portov	16
2.3. Popis napájacích častí procesora, periférií a prevodníkov sériovej komunikácie.....	16
2.4. Popis hodinového obvodu DS1307, pamäte FRAM a displeja.....	16
2.5. Popis jednotlivých periférií	17
3. Realizácia IR prijímača - popis softwarovej časti.....	18
3.1. Popis SW procesora CY8C29466 IR prijímača	19
3.1.1. Popis použitých vnútorných SW blokov procesora CY8C29466	19
3.1.2. Popis SW modulov procesora CY8C29466	22
3.1.3. Modul cnt_main.asm	22
3.1.4. Modul RC6.asm	23
3.1.5. Modul ir_receiver.asm	23
3.1.6. Modul cnt_sl_ide.asm	24
3.1.7. Modul modbus_config.inc	24
3.2. SW konfiguračného programu	25
3.2.1. Popis modulov konfiguračného programu	25
4. Uživatelský návod	26
4.1. Vytvorenie a zápis konfiguračnej tabuľky	26
4.2. Načítanie a nastavenie času v obvode DS1307 IR prijímača.....	27
4.3. Výpis diagnostických bytov prijatého RC6 protokolu.....	27
4.4. Popis povolených príkazov pri vytváraní konfiguračnej tabuľky.....	27
5. Popis komunikačného protokolu Modbus (μbus).....	30
5.1. Popis použitých telegramov	30
5.1.1. Rozhranie IR prijímač ↔ PC:	30
5.1.2. Rozhranie IR prijímač → displej:	33
6. Experimentálne skúšky zariadenia.....	34
7. Záver	35

Použitá literatúra	36
Zoznam symbolov, veličín a skratiek.....	38
Zoznam príloh	39

Zoznam obrázkov

Obr. 1.1: Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom PHILIPS DVD pri stlačení tlačidla 1	2
Obr. 1.2: LEADER (prevzaté z [1])	3
Obr. 1.3: Informačné a kontrolné bity (prevzaté z [1])	3
Obr. 1.4: Trailer bit (TR) (prevzaté z [1])	4
Obr. 1.5: Kompletný teoretický priebeh protokolu RC6 mode0 s popisom jednotlivých častí .	6
Obr. 1.6: Detekcia signálov kódovaných v komunikačnom protokole RC6.....	7
Obr. 1.7: Detekcia Leadera	10
Obr. 1.8: Detekcia stavu nečinnosti	11
Obr. 1.9: Obecná detekcia (identifikácia) jedného bitu komunikačného protokolu RC6	12
Obr. 1.10: Algoritmus práce programovateľného IR prijímača	14
Obr. 2.1: Blokovaná schéma realizácie.....	17
Obr. 2.2: Schéma zapojenia zopnutia svetla	17
Obr. 2.3: Zapojenie teplomera DS18S20	18
Obr. 2.4: Schéma zopnutia alarmu	18
Obr. 2.5: Schéma zapojenia ventilátora riadeného pomocou PWM	18
Obr. 3.1: Blokovaná schéma návrhu SW programovateľného IR prijímača	19
Obr. 3.2: Nakonfigurované SW bloky programom PSoC Designer	19
Obr. 3.3: Vnútorňá štruktúra pripojenia SW blokov na zbernici a piny procesora	20
Obr. 3.4: Nastavenie globálnych parametrov PSoC Designera jednotlivých SW blokov	21
Obr. 3.5: Konfigurácia pinov procesora	21
Obr. 4.1: Konfiguračná tabuľka vytvorená jazykom Python a jej maska (GUI) programom wxglade	26
Obr. 4.2: Algoritmus zopnutia svetla na obmedzenú dobu	28
Obr. 4.3: Algoritmus riadenia zobrazenia aktuálneho času.....	29
Obr. 5.1: Formát rámca protokolu Modbus (mikrobus) (prevzaté a upravené z [18]).....	30

Zoznam tabuliek

Tab. 1.1: Prehľad jednotlivých bitov protokolu RC6 mode 0 (prevzaté z [1])	4
Tab. 1.2: Zoznam niekoľkých adries komunikačného protokolu RC5 (prevzaté z [2]).....	4
Tab. 1.3: Zoznam niekoľkých príkazov komunikačného protokolu RC5 (prevzaté z [2])	5

Úvod

Postupom rozvoja techniky a rôznych technológií sa začali používať diaľkové ovládania na princípe prenosu infračerveného žiarenia, ktoré slúžia ako vysielajúce a zabezpečujú komunikáciu s TV prijímačom. Pri stlačení určitého tlačidla sa na vysielajúci na základe komunikačného protokolu (záleží na špecifickom type diaľkového ovládača) vyšle bitová informácia (kód s adresou), pri ktorej sa vykoná určitá funkcia na TV prijímači, pričom každé tlačidlo má pevne zadaný kód pre nejakú funkciu - napr. prepínanie kanálov, ovládanie hlasitosti, vyvolanie funkcie teletext, atď. Existuje veľa firiem, ktoré vyvinuli svoje vlastné komunikačné protokoly, pričom ale mnohé z nich sú vzájomne nekompatibilné. Úlohou bolo nájsť nejaký vhodný komunikačný protokol, ktorý by sa dal pomerne ľahko a bezpečne dekódovať. K dispozícii bolo viacero druhov diaľkových ovládačov, ktorých priebehy boli zosnímané na osciloskope a dôkladne preskúmané.

Hlavný princíp riadiacej aplikácie spočíva v tom, že pomocou diaľkového ovládania sa má riadiť programovateľný IR prijímač, ktorý na jednotlivé príkazy reaguje programovo priradenými akciami – to znamená, že na základe stlačeného tlačidla sa má vykonať podľa vopred stanoveného algoritmu určitý sled operácií, ktorý bude aktivovať príslušné výstupy (aktivácia alarmu, zapnutie LED diódy, relé, atď).

Celá riadiaca aplikácia predstavuje jednoduchý programovateľný logický automat, ktorý prijíma úlohy cez stláčanie tlačidiel na diaľkovom ovládači a reaguje na ne pomocou tzv. priradovacej tabuľky, ktorá sa do IR prijímača dostane pomocou programovacieho zariadenia (PC). Cez tabuľku sa dá ľubovoľnému tlačidlu diaľkového ovládania priradiť jedna zo známych akcií. Zariadenie (programovateľný IR prijímač) môže byť vhodným prostriedkom pre handicapovaných ľudí, ktorí použitím jedného univerzálneho diaľkového ovládania dokážu obsluhovať napr. TV, klimatizáciu, alarm, spínanie svetla, privolanie pomoci prostredníctvom SMS správy, atď. Pri realizácii treba klásť dôraz na tieto nasledovné časti:

- Výber vhodného komunikačného protokolu, jeho analýza, spôsob detekcie
- Návrh programového vybavenia programovacieho zariadenia (PC)
- Návrh komunikačného rozhrania s komunikačným protokolom medzi PC a IR prijímačom
- Návrh IR prijímača – HW a SW
- Návrh niekoľkých periférnych zariadení na otestovanie funkčnosti IR prijímača
- Realizácia prototypu programovateľného IR prijímača a otestovanie jeho spoľahlivej funkčnosti vrátane periférií

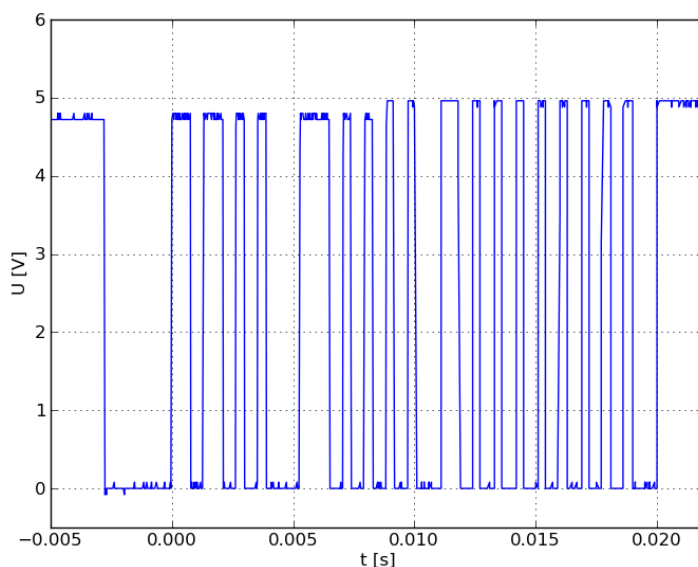
1. Návrh riešenia

V tejto kapitole je popísaný návrh riešenia spomenutých častí z úvodu, pričom sa popíše výber vhodného protokolu, návrhu programového vybavenia – výber vhodných programovacích jazykov, komunikačného rozhrania (USB, sériová linka) pri použití vhodného komunikačného protokolu medzi PC a IR prijímačom, HW a SW riešenie samotného IR prijímača a taktiež aj niekoľkých výstupných periférii na overenie funkčnosti IR prijímača.

1.1. Komunikačný protokol diaľkového ovládača

1.1.1. Výber komunikačného protokolu diaľkového ovládača

Bolo zosnímaných osem priebehov pri použití ôsmich diaľkových ovládačov. Analýza sa začala preskúmaním komunikačných protokolov od firmy PHILIPS. V [1] je uvedený popis viacerých komunikačných protokolov. Detailnou analýzou sa ukázalo, že sa nejedná o komunikačný protokol RC5 [2], [3], ale o jeho nástupcu RC6 [1] využívaného v dnešnej dobe častejšie (obr. 1.1). Ostatné zosnímané priebehy sú uvedené v zozname príloh A.



Obr. 1.1: Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom PHILIPS DVD pri stlačení tlačidla 1

Použitie rôznych modelov diaľkových ovládačov poukazuje na rôznu štruktúru a rôzne dĺžky trvania jednotlivých komunikačných protokolov. Na výber komunikačného protokolu boli zvolené nasledovné kritériá:

- musí ho obsahovať aspoň jeden z ovládačov, ktoré boli k dispozícii
- musí existovať podrobný popis tohto protokolu
- musí byť jednoduchými programovacími prostriedkami detekovateľný
- musí mať perspektívu do budúcnosti
- v súčasnej dobe štandard v moderných zariadeniach

Týmto podmienkam vyhoviel protokol RC6 od firmy Philips. Dva z diaľkových ovládačov, ktoré boli k dispozícii, vysielali týmto protokolom. Existuje jediný variant tohto protokolu,

a to RC6 mode0, pričom všetky mód bity sú v úrovni log. 0. V budúcnosti sa zrejme predpokladá rozšírenie tohto protokolu o ďalšie módy.

1.1.2. Rozbor súčasného stavu problematiky

Na príjem a dekódovanie komunikačného protokolu RC6 neboli zistené žiadne práce, k dispozícii sú len práce s komunikačným protokolom RC5 [4], [5], preto sa predkladá vlastný návrh realizácie.

Na princípe programovateľného zariadenia pracujú všetky známe logické automaty. Pri každom musí byť správanie sa automatu vopred určené nejakým konfiguračným programom – tento prípad je špeciálny – IR prijímač čaká na príkaz (povel) z diaľkového ovládania a potom na tento povel reaguje podľa naprogramovanej tabuľky priradenou akciou.

1.1.3. Popis protokolu RC6

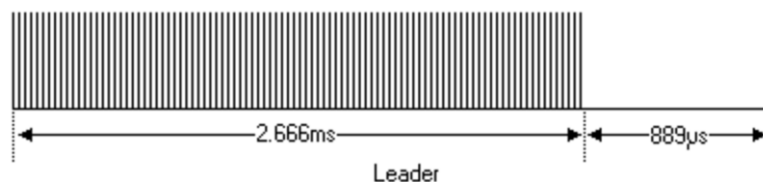
Protokol RC6 vyvinula firma PHILIPS ako nástupcu protokolu RC5. Nosná frekvencia protokolu je 36 kHz. Strieda sa pohybuje v rozmedzí 25 – 50%. Využíva sa tzv. kódovanie typu Manchester (Bi-phase coding). Vysvetlenie princípu kódovania Manchester, pomocou ktorého sú modulované dáta:

Výstupný signál obsahuje medzeru a značku každého bitu. Ak je bit "1", potom je prvá polovica bitu vysoká úroveň a druhá medzera (nízka úroveň). Bit "0" – prvá polovica bitu nízka úroveň a druhá vysoká úroveň.

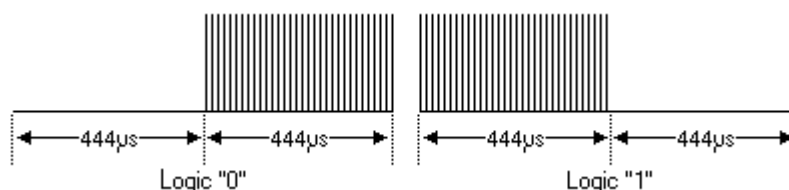
Hlavná časová jednotka je $1t$, ktorá prebehne 16-krát za periódu $\Rightarrow 1/36000 * 16 = 444 \mu s$.

Protokol RC6 obsahuje nasledovné bity:

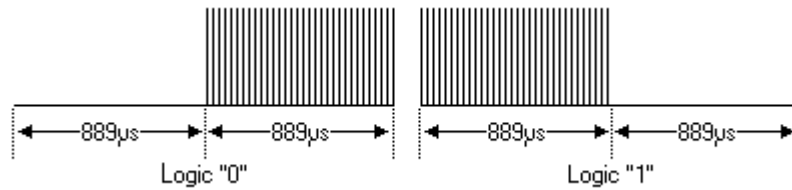
- LEADER (LB) – na nastavenie úrovne IR prijímacej jednotky, jeho vysoká úroveň je $6t$ (obr. 1.2)
- Informačné a kontrolné bity – obsahujúce high level a low level v perióde $1t$ (obr. 1.3)
- Trailer bit (TR) – obsahuje high level a low level v perióde $2t$ a mení sa po každom novom stlačení tlačidla oproti predchádzajúcemu, aby sa mohlo rozlíšiť, či ide o ďalší príkaz (obr. 1.4)
- Štart bit (SB) – vždy aktívny v log. 1
- $mb2 - mb0$ (mód bity) – pri uvedenom protokole nadobúdajú vždy úroveň log. 0
- Voľný signál – nie sú odosielané žiadne dáta a pre prijímač je dôležité tento signál detekovať na konci procesu dekódovania komunikačného protokolu RC6 a zabrániť prípadnej chybe pre príjem dát.



Obr. 1.2: LEADER (prevzaté z [1])



Obr. 1.3: Informačné a kontrolné bity (prevzaté z [1])



Obr. 1.4: Trailer bit (TR) (prevzaté z [1])

Diaľkové ovládače PHILIPS používajú momentálne len mód (režim) protokolu RC6 - mode0
Štruktúra a následnosť bitov RC6 mode 0 je uvedená v tab. 1.1:

Tab. 1.1: Prehľad jednotlivých bitov protokolu RC6 mode 0 (prevzaté z [1])

LB	SB	Mb2 ... mb0	TR	a7 ... a0	c7 ... c0	Voľný signál
Hlavička				Adresné bity	Riadiace bity	dĺžky 6t

Na obr. 1.5 je uvedený kompletný teoretický invertovaný priebeh príkazu. Leader na začiatku je dôležitým bezpečnostným prvkom, pretože filtruje zahájenie detekcie RC6 kódu náhodnými krátkymi fluktuáciami impulzami. K bezpečnostným prvkom patria aj štart bit a mód bity, pretože ich hodnoty sú pevne stanovené a ak to tak nie je, treba hneď ukončiť detekciu protokolu. Trailer umožňuje zmenou svojej hodnoty pri každom novom stlačení obslúžiť zvlášť príkazy, pri ktorých je dôležitá dĺžka stlačenia tlačidla (napr. pri zmene hlasitosti) a naopak (napr. pri prepnutí kanála). Adresný byte (tab. 1.2) určuje typ zariadenia – diaľkové ovládania k dispozícii boli od TV – adresa 0 a DVD prehrávača – adresa 4. Riadiaci byte (tab. 1.3) predstavuje kód príkazu. Adresu a kód pri danom diaľkovom ovládaní nemožno ovplyvniť, pri našom IR prijímači je len možnosť k nim priradiť kód výkonného povelu. Na konci priebehu je stav nečinnosti, ktorý možno tiež priradiť k bezpečnostným prvkom, ktoré musia byť splnené pre bezpečný príjem príkazu.

Tab. 1.2: Zoznam niekoľkých adries komunikačného protokolu RC5 (prevzaté z [2])

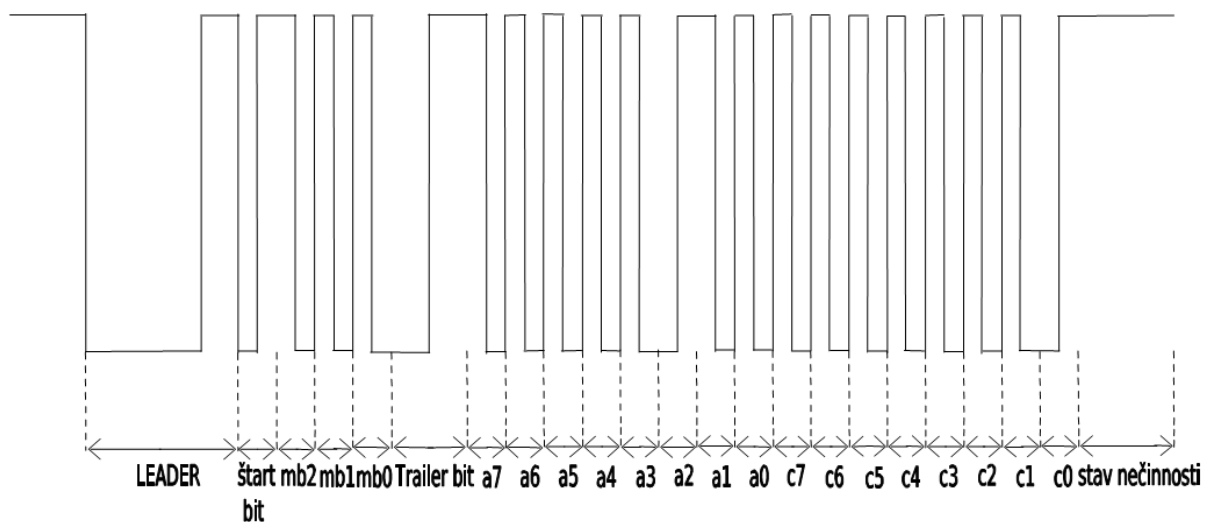
RC5 – Address	Device
\$00 – 00	TV1
\$01 – 01	TV2
\$02 – 02	Teletext
\$03 – 03	Video
\$04 – 04	LV1
\$05 – 05	VCR1
\$06 – 06	VCR2
\$07 – 07	Experimental
\$08 – 08	Sat1
\$09 – 09	Camera
\$0A – 10	Sat2
\$0B – 11	
\$0C – 12	CDV
\$0D – 13	Camcorder
\$0E – 14	
\$0F – 15	
\$10 – 16	Pre-amp
\$11 – 17	Tuner
\$12 – 18	Recorder1
\$13 – 19	Pre-amp
\$14 – 20	CD player
\$15 – 21	Phono
\$16 – 22	SatA

\$17 – 23	Recorder2
\$18 – 24	
\$19 – 25	
\$1A – 26	CDR
\$1B – 27	
\$1C – 28	
\$1D – 29	Lightning
\$1E – 30	Lightning
\$1F - 31	Phone

Tab. 1.3: Zoznam niekoľkých príkazov komunikačného protokolu RC5 (prevzaté z [2])

RC5 – Command	TV Command	VCR Command
\$00 – 00	0	0
\$01 – 01	1	1
\$02 – 02	2	2
\$03 – 03	3	3
\$04 – 04	4	4
\$05 – 05	5	5
\$06 – 06	6	6
\$07 – 07	7	7
\$08 - 08	8	8
\$09 - 09	9	9
\$0A - 10	-/--	-/--
\$0C – 12	Standby	Standby
\$0D – 13	Mute	
\$10 – 16	Volume +	
\$11 – 17	Volume -	
\$12 – 18	Brightness +	
\$13 – 19	Brightness -	
\$20 – 32	Program +	Program +
\$21 – 33	Program -	Program -
\$32 – 50		Fast Rewind
\$34 – 52		Fast Forward
\$35 – 53		Play
\$36 – 54		Stop
\$37 – 55		Recording

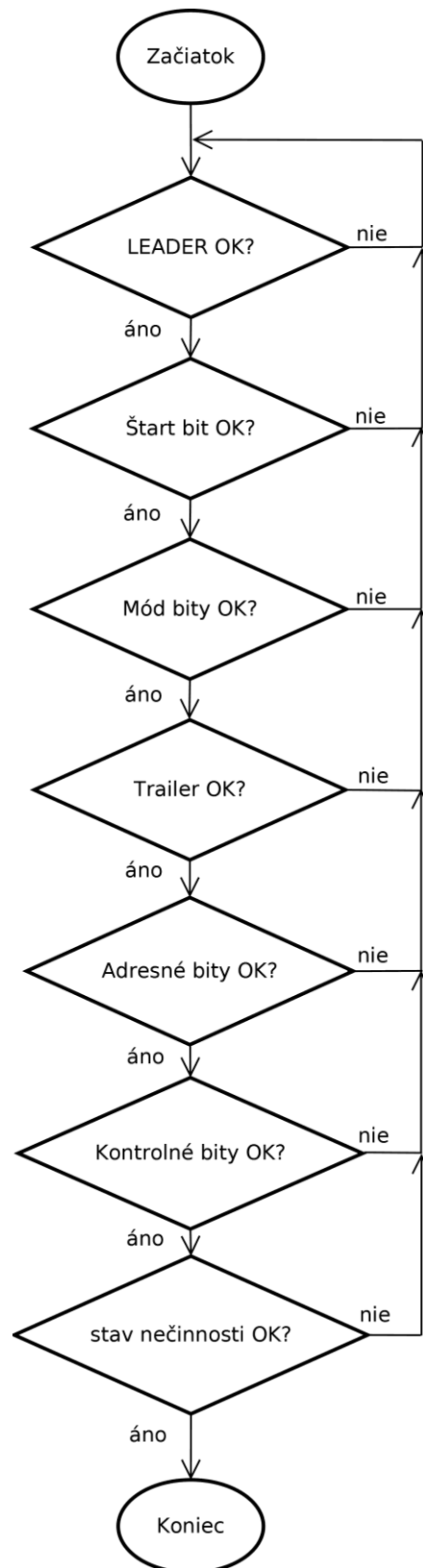
Keďže je RC6 protokol rozšírenejší oproti RC5, tak sa pri ňom zrejme uplatnia tie isté príkazy vzhľadom na jednotlivé adresy.



Obr. 1.5: Kompletný teoretický priebeh protokolu RC6 mode0 s popisom jednotlivých častí

1.1.4. Návrh na detekciu protokolu RC6

Z hore uvedeného popisu jednotlivých častí protokolu prirodzene vyplýva aj zostavenie vývojového diagramu (vid' obr. 1.6) na detekciu kódovaných signálov obsahujúcich štruktúru komunikačného protokolu RC6.



Obr. 1.6: Detekcia signálov kódovaných v komunikačnom protokole RC6

Z uvedeného vývojového diagramu je vidieť, že detekcia jednotlivých častí musí prebiehať v poradí ako na obr. 1.5 zľava doprava. Ak pri detekcii niektorej časti nastane chyba, treba skočiť na začiatok a znovu čakať na príchod Leadera. Iba po bezchybnom prijatí všetkých častí protokolu možno pristúpiť k vykonaniu prijatého príkazu. Návrh nasledovnej filozofie detekcie komunikačného protokolu RC6 (obr. 1.5):

- Čakať na vzostupnú (nábežnú) hranu Leadera a po jej príchode naštartovať časovač
- Čakať na zostupnú (dobežnú) hranu Leadera, po jej príchode zastaviť časovač a zistiť dĺžku trvania prvej fázy Leadera, ak je dĺžka trvania nesprávna, tak je potrebné ukončiť detekciu a v prípade korektnosti (správnosti) je možné pokračovať v detekcii
- Čakať na vzostupnú hranu štartovacieho bitu a od nej sa odpichnúť nastavovaním časových intervalov snímania signálu tak, aby boli vždy v strede jednotlivých fáz postupne všetkých bitov protokolu
- Nakoniec skontrolovať stav nečinnosti po prijatí všetkých bitov protokolu

Sled časových intervalov použitých pri detekcii RC6 protokolu (štart je vzostupná hranu štartovacieho bitu):

- 222us – skok na stred 1.fázy štartovacieho bitu
- 444us – skok na stred 2.fázy štartovacieho bitu
- 444us - skok na stred 1.fázy bitu mb2
- 444us - skok na stred 2.fázy bitu mb2
- 444us - skok na stred 1.fázy bitu mb1
- 444us - skok na stred 2.fázy bitu mb2
- 444us - skok na stred 1.fázy bitu mb0
- 444us - skok na stred 2.fázy bitu mb0
- 666us – skok na stred 1.fázy trailer bitu
- 888us – skok na stred 2.fázy trailer bitu
- 666us - skok na stred 1.fázy bitu a7
- 444us - skok na stred 2.fázy bitu a7
- 444us - skok na stred 1.fázy bitu a6
- 444us - skok na stred 2.fázy bitu a6
- 444us - skok na stred 1.fázy bitu a5
- 444us - skok na stred 2.fázy bitu a5
- 444us - skok na stred 1.fázy bitu a4
- 444us - skok na stred 2.fázy bitu a4
- 444us - skok na stred 1.fázy bitu a3
- 444us - skok na stred 2.fázy bitu a3
- 444us - skok na stred 1.fázy bitu a2
- 444us - skok na stred 2.fázy bitu a2
- 444us - skok na stred 1.fázy bitu a1
- 444us - skok na stred 2.fázy bitu a1
- 444us - skok na stred 1.fázy bitu a0
- 444us - skok na stred 2.fázy bitu a0
- 444us - skok na stred 1.fázy bitu c7
- 444us - skok na stred 2.fázy bitu c7
- 444us - skok na stred 1.fázy bitu c6
- 444us - skok na stred 2.fázy bitu c6

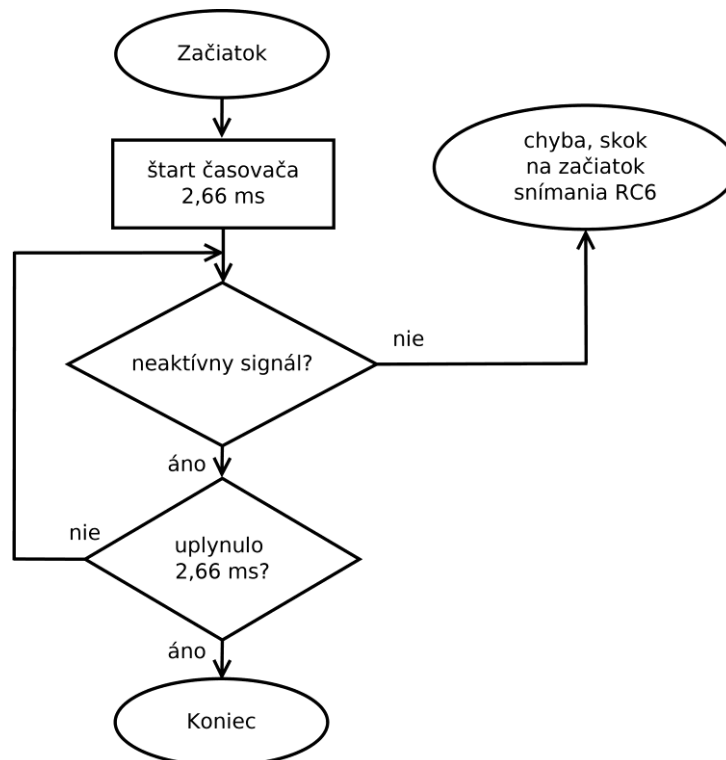
- 444us - skok na stred 1.fázy bitu c5
- 444us - skok na stred 2.fázy bitu c5
- 444us - skok na stred 1.fázy bitu c4
- 444us - skok na stred 2.fázy bitu c4
- 444us - skok na stred 1.fázy bitu c3
- 444us - skok na stred 2.fázy bitu c3
- 444us - skok na stred 1.fázy bitu c2
- 444us - skok na stred 2.fázy bitu c2
- 444us - skok na stred 1.fázy bitu c1
- 444us - skok na stred 2.fázy bitu c1
- 444us - skok na stred 1.fázy bitu c0
- 444us - skok na stred 2.fázy bitu c0
- 444us – skok za začiatok stavu nečinnosti
- 2660us - skok na koniec stavu nečinnosti

Detekciu Leadera popisuje vývojový diagram (obr. 1.7), pričom musí byť dodržaná jeho vysoká úroveň 6t a nízka úroveň 2t.



Obr. 1.7: Detekcia Leadera

Aby mal komunikačný protokol RC6 korektnú a bezpečnú detekciu všetkých bitov, je bezpodmienečne nutné ošetriť aj priebeh stavu nečinnosti (obr. 1.8) po prijatí všetkých bitov.



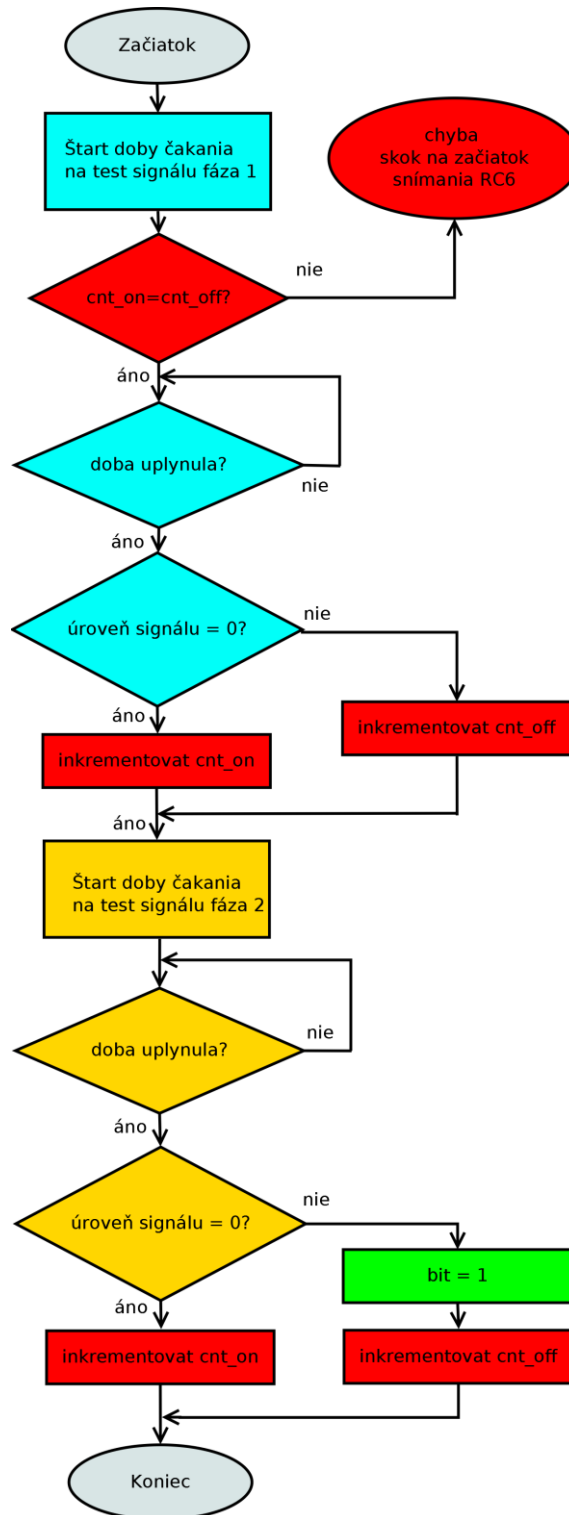
Obr. 1.8: Detekcia stavu nečinnosti

Nasledujúci vývojový diagram (obr. 1.9) predstavuje obecnú detekciu (identifikáciu) jedného bitu komunikačného protokolu RC6, pričom sú jednotlivé časti z dôvodu vyššej prehľadnosti uvedené farebne. Vysvetlenie významu jednotlivých farieb:

- Červená farba označuje bezpečnosť – každý bit má 2 fázy, a to fázu 1 a fázu 2. Význam jednotlivých fáz sa rozumie (je chápaný) ako stav log. 1 (vysoká úroveň) a log. 0 (nízka úroveň). Hodnoty jednotlivých fáz musia byť rozdielne. Zavedením čítača stavu 0 - cnt_off a stavu 1 cnt_on, ktoré sa inkrementujú po každej detekcii v jednotlivých fázach bitu – musia byť rovnaké. V prípade opaku je potrebné ukončiť detekciu komunikačného protokolu RC6 a skočiť na začiatok – príjem Leadera, čítače sa automaticky vynulujú. Vhodný čas na kontrolu rovnosti oboch čítačov je počas čakania na okamih, pri ktorom očakávame príchod stredu fázy 1 nasledujúceho bitu.
- Modrá farba (charakterizuje fázu 1) – po uplynutí doby čakania na stred fázy 1 sa prevedie test stavu signálu, podľa ktorého sa inkrementuje príslušný čítač.
- Žltá farba (charakterizuje fázu 2) – prebieha analogicky ako fáza 1 s tým rozdielom, že sa v nej nastaví hodnota detekovaného bitu. Ak je úroveň signálu 0, musela byť úroveň signálu v predchádzajúcej fáze 1 => hodnota bitu = 1. Pri zistenej úrovni 1 to platí obrátene.
- Zelená farba – nastavuje hodnotu bitu na hodnotu 1. Keďže je na začiatku pole pre hodnoty bitov vynulované, tak nie je potrebné nastavenie na nulu.

Podľa tohto algoritmu je potrebné uskutočniť začiatok detekcie od štartovacieho bitu postupne všetkými bitmi vrátane stavu nečinnosti. Ak štartovací bit a jednotlivé mód bity nadobudnú čo len malú zmenu v dĺžke ich samotného trvania (nemajú predpísanú hodnotu), tak sa musí detekcia komunikačného protokolu RC6 ihneď ukončiť ako chybná a musí začať od začiatku.

Z uvedených vývojových diagramov vyplýva, že na detekciu protokolu RC6 je potrebný len programovateľný čítač s generovaním prerušenia pri dosiahnutí stavu 0.



Obr. 1.9: Obecná detekcia (identifikácia) jedného bitu komunikačného protokolu RC6

1.2. Vlastnosti programovateľného IR prijímača

Návrh programovateľného IR prijímača musí vyhovieť určitým požiadavkám nasledujúcimi vlastnosťami:

- príjem a identifikácia príkazov diaľkového ovládania
- programovateľnosť pomocou externého zariadenia cez sériové rozhranie

- výkon príkazov na základe naprogramovanej tabuľky
- schopnosť obsluhovať rôzne periférie – DI, DO, analógové VV, I2C, 1-Wire, atď.
- flexibilita pri zmene a vývoji nových periférií

Na základe týchto vlastností možno určiť požiadavky na HW a SW. Z pohľadu stránky HW sa použijú nasledovné komponenty:

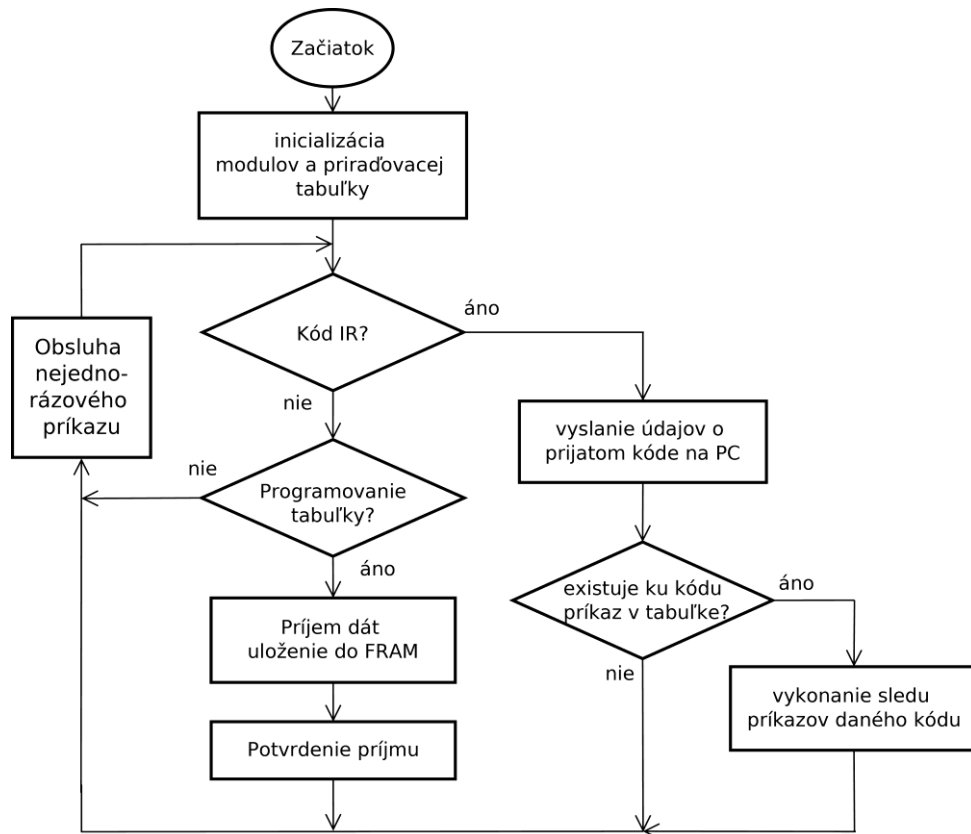
- programovateľný mikroprocesor dostatočne rýchly na detekciu IR kódu
- snímač kódu diaľkového ovládania
- schopnosť mikroprocesora komunikovať s okolím cez I2C, 1-Wire, sériové rozhranie, dostatočné množstvo digitálnych a analógových VV, možnosť viacerých sériových rozhraní (Master a Slave)
- konektor sériového rozhrania (RS232, USB)
- pamäť na uchovanie naprogramovanej tabuľky pre dekodovanie povelov
- zálohovaný časovací obvod pre možnosť výkonu príkazov viazaných na reálny čas
- zobrazovací prvok (displej) na vizualizáciu dát pre príkazy, ktoré to vyžadujú – napr. zobrazenie aktuálneho času, teploty, prijatého kódu, atď.

Z pohľadu stránky SW sa kladú požiadavky najmä na:

- spracovanie signálu z IR prijímača pre daný komunikačný protokol RC6
- komunikačný protokol pre komunikáciu s programovacím a diagnostickým zariadením
- obsluha I2C, 1-Wire a sériového rozhrania
- obsluha vizualizačného prvku – posielanie dát na displej
- maximálna flexibilita vzhľadom na požiadavky pripojenia periférií

Algoritmus práce programovateľného IR prijímača je uvedený na obr. 1.10.

Program behá v nekonečnej slučke, pričom testuje detekciu IR kódu alebo programovací telegram tabuľky. V prípade zaregistrovania IR kódu sa údaje o ňom (mód bity, trailer, adresa, kód) vyšlú na sériové rozhranie (USB) s PC a otestuje sa jeho príjem – ak k nemu existuje podľa naprogramovanej priradovacej tabuľky kód príkazu, vykoná sa sled príkazov pod daným kódom. Ak sa zistí príjem telegramu programovania priradovacej tabuľky, prijme sa celý sled programovacích telegramov, nová tabuľka sa uloží po riadkoch dĺžky 16 bytov do pamäte FRAM, potvrdí sa príjem tabuľky a znovu sa čaká na novú (ďalšiu) udalosť. Na návrh odborného vedúceho práce je pri komunikácii medzi PC a IR prijímačom použitý zjednodušený modbus protokol.



Obr. 1.10: Algoritmus práce programovateľného IR prijímača

1.2.1. Návrh HW na základe požadovaných vlastností

Na základe požadovaných vlastností HW sa rozhodlo pre nasledovné HW komponenty pre programovateľný IR prijímač:

- Mikroprocesor Cypress PSoC CY8C29466
- IR senzor typu TSOP 1738 (dal by sa použiť aj typ SFH 5110-38)
- Prevodník TTL na USB rozhranie, pretože bežné počítače majú RS232 len výnimočne
- Pamäť FRAM typ 24256 [6]
- Obvod reálneho času typ DS1307 [7]
- Na návrh odborného vedúceho sedemsegmentový modul displeja so štyrmi zobrazovacími ciframi, ktorý dodal k dispozícii odborný vedúci práce

1.2.2. Voľba SW na základe požadovaných vlastností

Vývojové prostredie sa nazýva PSoC Designer a má dve časti – Device Editor a Application Editor. Je dostupné zdarma na [8]. Podrobný popis práce s Designerom je v dokumente Designer's Guide to the Cypress PSoC. SW pozostáva z nasledovných častí:

- Device Editor – konfigurácia vnútornej štruktúry procesora
- Application Editor – programovanie v assembleri
- Assembler pre Cypress procesory – podrobný popis je v dokumente Assembly Language User Guide [9]
- Jazyk C je tiež k dispozícii na použitie, ale je potrebná licencia a je pomerne neefektívny – zaberá veľa pamäte a je potrebné dávať pozor na pretečenie zásobníka

- Po preložení programu vznikne súbor hex, ktorý sa do procesora napaľuje prostredníctvom miniprogramátora PSoC MiniProg a programu PSoC Programmer

1.2.3. Popis vlastností mikroprocesora Cypress

Vlastnosti procesora Cypress PSoC CY8C29466 sú podrobne popísané v dokumente PSoC Mixed-Signal Array [10]. Základné vlastnosti tohto procesora sú:

- 8-bitový procesor s rýchlosťou do 24 MHz
- Napájacie napätie v rozmedzí 3.0 – 5,25V
- Priemyselný rozsah teplôt -40 až +85°C
- možnosť nakonfigurovať až 12 analógových blokov
- ADC prevodníky do 14 bitov
- DAC prevodníky do 9 bitov
- programovateľné filtre a komparátory
- 16 digitálnych blokov
- 8 až 32 bitové časovače, čítače a PWM
- 4 plne duplexné UARTy
- 32 kB flash pamäť programu
- 8x256 bytová RAM
- konfigurovateľnosť pinov – pull up, pull down, high Z, strong, open drive módy
- 12 analógových vstupov
- Štyri 40mA analógové výstupy
- Konfigurovateľné prerušenia na všetkých VV
- I2C rozhranie
- 1-Wire rozhranie
- WatchDog
- možnosť zmeny vnútornej štruktúry procesora počas behu programu
- Kompilátor pre programovacie jazyky C a assembler, linker

Veľkou výhodou práce so Cypress procesormi je flexibilita a veľký výber digitálnych a analógových modulov. Po ich nakonfigurovaní do projektu v PSoC Designeri sa automaticky pridajú do projektu ich knižničné moduly vrátane funkcií prerušenia. Namiesto vytvárania celého podprogramu prerušenia stačia na príslušne označené miesto vlastné riadky zdrojového textu.

1.3. Konfiguračná aplikácia v PC

Program v PC bude mať nasledovné úlohy:

- Vytvorenie priradovacej tabuľky, kde sa priradí kód ku kódu diaľkového ovládania z pevne naprogramovaných príkazov s prípadnými parametrami
- Zobrazovanie údajov o prijatom kóde, ktoré vyšle IR prijímač po prijatí kódu – diagnostická funkcia
- Výber údajov z tabuľky, vytvorenie sledu telegramov za účelom naprogramovania IR prijímača a prevedenie vyslania telegramov
- Zabezpečenie bezchybnej sériovej komunikácie s IR prijímačom

Na základe týchto uvedených požiadaviek a taktiež aj odporúčenia odborného vedúceho práce bol zvolený programovací jazyk Python [11]. Je to skriptovací jazyk, ktorý je zdarma k dispozícii, na internete možno nájsť veľa príkladov na ľubovoľnú tému. Doporučuje sa nainštalovať EPD s dodatočnou inštaláciou modulu pyserial. Pre vytvorenie vrchnej masky je

vhodné použiť program wxglade [12], pomocou ktorého možno vytvárať rýchlo a efektívne masky s widgetmi, pričom sa dá následne previesť export pythonovského programu (skriptu). Tento program je vhodné v editore geany [13] dokončiť naprogramovaním udalostí (eventov).

2. Realizácia IR prijímača - popis hardwarovej časti

Na doporučenie odborného vedúceho práce bola použitá doska CNT od firmy Dynamag Group a.s., ktorá bola pôvodne vyvinutá pre úplne iné zariadenie a obsahuje len doporučené zapojenia jednotlivých elektronických komponentov. Využitím tejto dosky s malými modifikáciami (odstránenie niekoľkých odporov), doplnením schémy hlavne o periférie bola vytvorená schéma hardwaru. Táto schéma je rozmiestnená na štyroch výkresoch formátu A4 (viď príloha B).

2.1. Stručný popis hlavného riadiaceho prvku mikroprocesora CY8C29466

Hlavné jadro celej riadiacej aplikácie programovateľného IR prijímača tvorí mikroprocesor CY8C29466 (umiestnený na výkrese CNT_cpu – príloha B1). Výhodou použitia mikroprocesorov typu CYPRESS je vytvorenie vlastnej vnútornej konfigurácie použitím rôznych sériových rozhraní, digitálnych a analógových funkčných blokov (vrátane viacerých UARTov) a schopnosť komunikovať s okolím cez I2C a 1-Wire. Je schopný počas behu programu dynamicky meniť svoju štruktúru nakonfigurovaním rôznych funkčných blokov.

2.2. Popis vstupno-výstupných portov

Ako vstupné porty sú použité detekcia komunikačného protokolu RC6, 1s prerušenie od obvodu reálneho času - DS1307 (flag_1s) a rozhranie 1-Wire (obojsmerná komunikácia), pričom naopak na výstupné porty sú nakonfigurované PWM, zopnutie a vypnutie svetla, aktivácia a deaktivácia alarmu. Na obr. 3.5 je znázornená konfigurácia portov s priradenými akciami.

2.3. Popis napájacích častí procesora, periférií a prevodníkov sériovej komunikácie

Prototyp zariadenia disponuje možnosťou voľby komunikácie pomocou sériového rozhrania USB a RS-232, ktoré je možné prepínať pomocou jumperov JMP101-104.

Napájacia časť zariadenia je vybavená stabilizátorom 7812 na napájanie periférií a modul spínaného napájacieho zdroja IZ1205SA na napájanie mikroprocesora a prevodníkov (MAX232 a FT232), ktorý na výstupe obsahuje LC filter typu dolná priepusť.

Na výkrese cnt_power (viď príloha B2) sa nachádzajú prevodníky MAX232 na prevod z rozhrania TTL na RS232 a FT232 z TTL na USB. Oba tieto prevodníky sú prevzaté, príp. upravované podľa doporučených zapojení od výrobcu.

2.4. Popis hodinového obvodu DS1307, pamäte FRAM a displeja

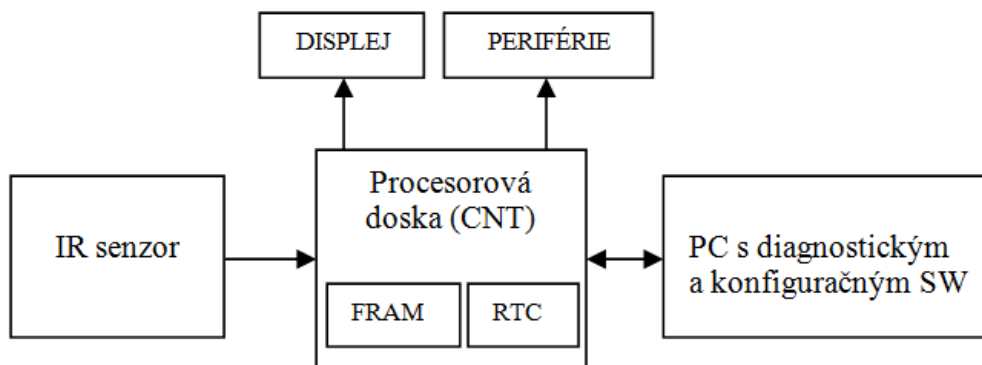
Výkres DS_CNTL801_RTC obsahuje obvod reálneho času DS1307, ktorý je riadený rozhraním I2C pomocou signálov SDA a SCL. Jeho taktovacia frekvencia je určená externým oscilátorom určeným kryštálom (32kHz). Výhodou tohto obvodu je aj záložný zdroj 3.3V –

v prípade výpadku elektrickej energie sa uchová reálny čas (na doske je použitá batéria CR2032) a možnosť prerušenia (SQW), ktoré je nastavené na 1s.

Rozhraním I2C je ovládaná pamäť FRAM, do ktorej sa zapisuje a číta obsah konfiguračnej tabuľky. Zobrazovací modul displeja bol dodaný odborným vedúcim ako samostatná jednotka. Jeho hlavným riadiacim prvkom je mikroprocesor CY8C27443 [14]. Displej je zapojený v multiplexnom režime (jednotlivé segmenty sú vnútorne navzájom pospájané).

Na spracovanie IR žiarenia slúži senzor TSOP1738, pričom jeho doporučené zapojenie je prevzaté od výrobcu [15].

Bloková schéma celej realizácie je znázornená na obr. 2.1.

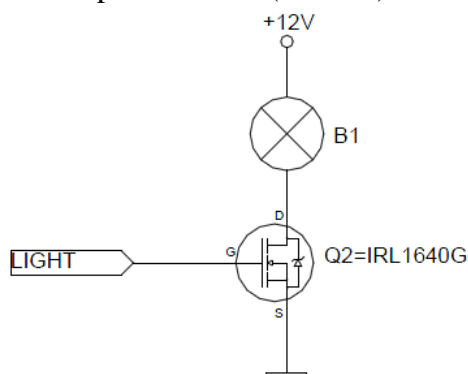


Obr. 2.1: Blokovaná schéma realizácie

2.5. Popis jednotlivých periférií

Na doporučenie odborného vedúceho boli všetky schémy periférií zostavené v programe gEDA [16].

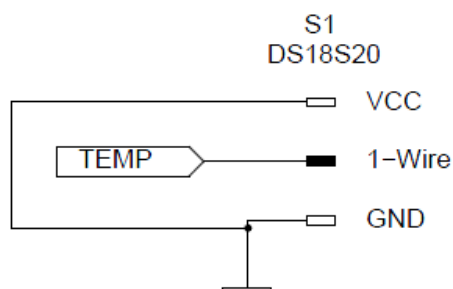
1. Zopnutie svetla (obr. 2.2)



Pomocou príkazov zopnutia a vypnutia svetla sa aktivuje výstup mikroprocesora LIGHT, na základe ktorého sa žiarovka rozsvieti (log. 1) alebo zhasne (log. 0). IRL - MOSFET tranzistor s log. vstupom.

Obr. 2.2: Schéma zapojenia zopnutia svetla

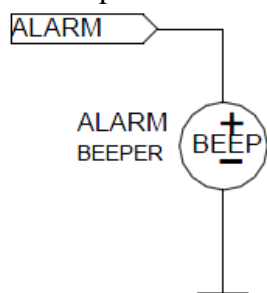
2. Teploměr obsluhovaný rozhraním 1-Wire (obr. 2.3)



Pomocou vstupno-výstupného portu (obojsmerného portu) mikroprocesora P0[3] a podľa naprogramovaných signálov je rozhraním 1-Wire ovládaný teplomer (senzor DS18S20) [17]. Rozhranie 1-Wire (jeden vodič) je súčasne aj napájacím, pričom je použitý mód vzájomného prepojenia VCC + GND.

Obr. 2.3: Zapojenie teplomera DS18S20

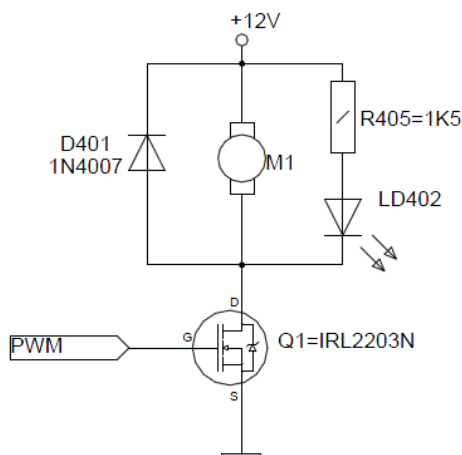
3. Zopnutie alarmu (obr. 2.4)



Alarm je ovládaný výstupom ALARM cez výstupný pin P0[0], pričom ako akustický prvok je použitý piezobzuzičiak.

Obr. 2.4: Schéma zopnutia alarmu

4. Riadenie otáčok ventilátora pomocou PWM (obr. 2.5)



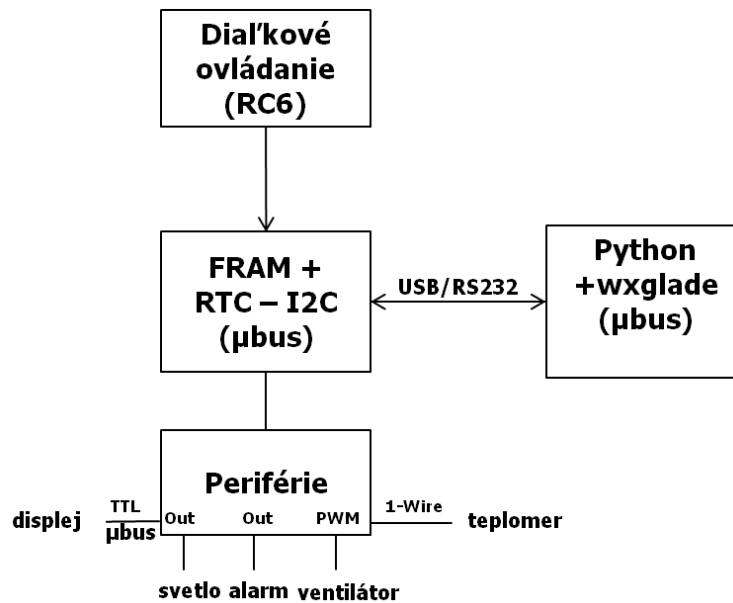
Ventilátor (M1) je spínaný na základe výstupného portu z mikroprocesora P0[4] naprogramovaním podprogramu PWM. Zvolením kroku 25 sa dajú zvyšovať alebo znižovať otáčky ventilátora, ktoré sú signalizované LED diódou LD402. Kvôli spínaniu indukčnej záťaže a možnému poškodeniu spínača (MOSFET) je antiparalelne k ventilátoru zapojená ochranná dióda.

Obr. 2.5: Schéma zapojenia ventilátora riadeného pomocou PWM

3. Realizácia IR prijímača - popis softwarovej časti

SW sa skladá z týchto nasledujúcich častí:

1. SW procesora CY8C29466 IR prijímača
2. SW konfiguračného programu (tabuľky) na PC vytvorenej programovacím jazykom Python



Obr. 3.1: Blokovaná schéma návrhu SW programovateľného IR prijímača

Obr. 3.1 zobrazuje vzájomné prepojenie návrhu programovateľného IR prijímača pomocou jednotlivých uvedených blokov s vyznačením jednotlivých rozhraní a komunikačnými protokolmi, ktorými je obsluhovaná (riadená) celá riadiaca aplikácia IR prijímača. Pod názvom mikrobus sa rozumie zjednodušený modbus protokol (viď kapitola 5).

3.1. Popis SW procesora CY8C29466 IR prijímača

V nasledujúcich podkapitolách sa bližšie rozoberie štruktúra mikroprocesora vrátane jeho globálnych parametrov a konfigurovateľných SW blokov v programe PSoC Designer.

3.1.1. Popis použitých vnútorných SW blokov procesora CY8C29466



Obr. 3.2: Nakonfigurované SW bloky programom PSoC Designer

Obr. 3.2 znázorňuje všetky vnútorné SW bloky, ktoré sú nakonfigurované (nižšie popísané) v mikroprocesore CY8C29466.

Counter16 - 16-bitový čítač použitý pri detekcii RC6 protokolu.

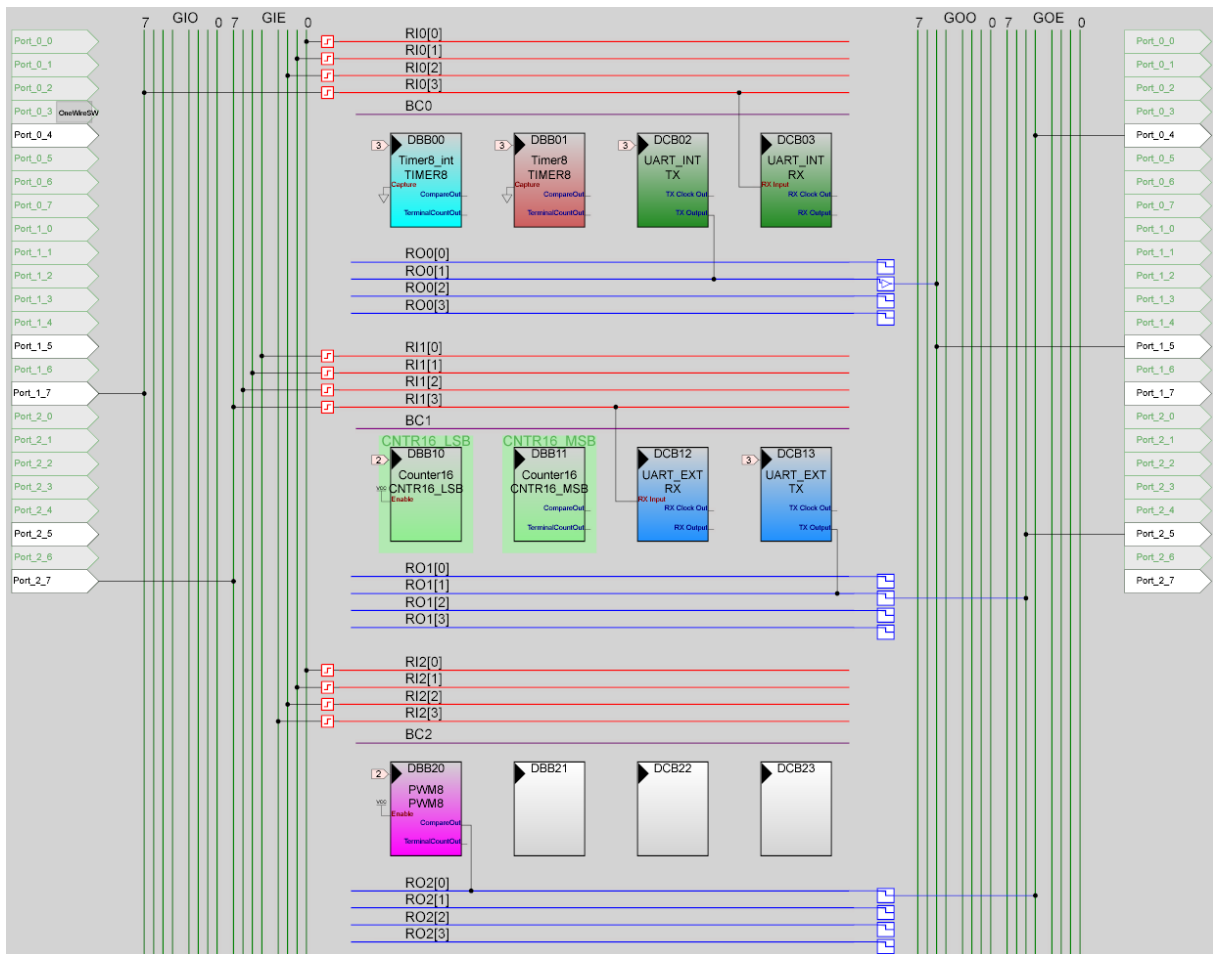
OneWireSW – modul s podprogramami pre obsluhu 1-Wire komunikácie.

PWM8 - modul s podprogramami pre obsluhu jednobytovej šírko-pulznej modulácie.

UART_EXT, Timer8 – moduly pre obsluhu sériovej komunikácie s rozhraním na PC.

UART_INT, Timer8_int – moduly pre obsluhu sériovej komunikácie s rozhraním na displej

K týmto blokom boli PSoC Designerom automaticky priradené obslužné podprogramy do knižničných zdrojov (Library Source) SW. Ich súčasťou sú aj podprogramy na obsluhu prerušenia napr. counter16int.asm, kde stačí na označené miesto dopísať svoj program obsluhy prerušenia. Pripojenie SW blokov na vnútorné zbernice a piny procesora je na Obr. 3.3.



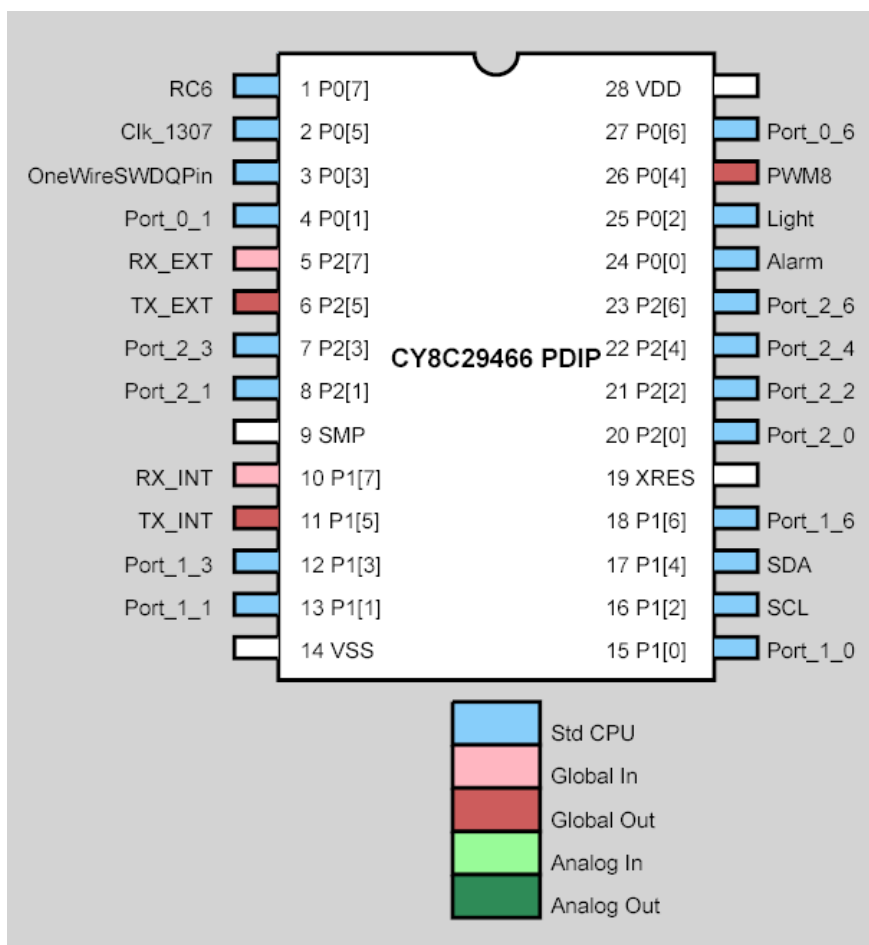
Obr. 3.3: Vnútroštruktúra pripojenia SW blokov na zbernica a piny procesora

Čísla pri ľavých horných častiach blokov udávajú taktovaciu frekvenciu pre jednotlivé SW bloky nastavovanú v Global Resources PSoC Designera – obr. 3.4. Číslo 2 znamená pripojenie VC2 k blokom čítača a PWM, z čoho sa dá následne určiť vnútorná taktovacia frekvencia týchto spomenutých blokov – $24/3/8 = 1 \text{ MHz}$ (perióda $T = 1 \mu\text{s}$).

Global Resources	Value
Power Setting [Vcc / SysClk freq]	5.0V / 24MHz
CPU_Clock	SysClk/1
32K_Select	External
PLL_Mode	Disable
Sleep_Timer	512_Hz
VC1= SysClk/N	3
VC2= VC1/N	8
VC3 Source	SysClk/1
VC3 Divider	156
SysClk Source	Internal
SysClk*2 Disable	No
Analog Power	SC On/Ref High
Ref Mux	(Vdd/2)+/(Vdd/2)
AGndBypass	Disable
Op-Amp Bias	Low
A_Buff_Power	Low
SwitchModePump	OFF
Trip Voltage [LVD (SMP)]	4.81V (5.00V)
LVDThrottleBack	Disable
Watchdog Enable	Disable

Obr. 3.4: Nastavenie globálnych parametrov PSoC Designera jednotlivých SW blokov

Celkové obsadenie pinov procesora je znázornené na obr. 3.5.



Obr. 3.5: Konfigurácia pinov procesora

Popis jednotlivých pinov:

- RC6 – príjem detekovaného signálu prostredníctvom komunikačného protokolu RC6
- Clk_1307 – výstupný signál z RTC DS1307 – vyvolanie prerušenia 1s
- OneWireSWDQPin – signál 1-Wire – vyvolanie teploty
- RX_EXT - prijímač sériového signálu na vonkajšie rozhranie
- TX_EXT – vysielateľ sériového signálu na vonkajšie rozhranie
- RX_INT – prijímač sériového signálu na vnútorné rozhranie
- TX_INT – vysielateľ sériového signálu na vnútorné rozhranie
- PWM8 – šírko-pulzná modulácia
- Light – zopnutie svetla
- Alarm – zopnutie alarmu
- SDA, SCL – piny určené pre komunikačnú zbernicu I2C
- Ostatné piny sú neobsadené (nevyužité)

3.1.2. Popis SW modulov procesora CY8C29466

Na vykonávanie programu procesora CY8C29466 sú použité nasledovné moduly:

- modul cnt_main.asm
- modul RC6.asm
- modul ir_receiver.asm
- modul cnt_sl_idle.asm
- modul cnt_sl_unicast.asm
- modbus_config.inc

Zoznam prevzatých modulov, ktoré budú vysvetlené v prílohe C:

- modul boot.asm
- modul cnt_fram.asm
- modul cnt_master.asm
- modul cnt_slave.asm
- modul cnt_ds1307.asm
- modul ds18s20.asm – 1-Wire
- modul I2C_sw.asm – I2C
- I2C_sw.inc
- modul math.asm
- modul utils_delay.asm

3.1.3. Modul cnt_main.asm

Hlavný program, v ktorom prebehne inicializácia:

- použitých nakonfigurovaných SW blokov
- komunikačnej adresy
- príjmu RC6 kódu
- otestovanie behu obvodu RTC DS1307 – nastavenie flagu flag_1s, ktorý sa nastaví v prerušení *PSoCGPIPOINT.asm* pri príchode nábežnej hrany na pin Clk_1307 procesora (CY8C29466) a následný skok do hlavnej slučky v module *cnt_slave.asm*.

3.1.4. Modul RC6.asm

Tento modul obsahuje podprogramy na spoľahlivú detekciu komunikačného protokolu RC6, ktoré pozostávajú z:

- *init_RC6.asm* – v tomto podprograme prebehne inicializácia pre príjem RC6 protokolu. Okrem nastavenia premenných sa registre 16-bitového čítača (counter16) naplnia hodnotami 0xFF, pričom je čítač pripravený na spustenie (je vynulovaný).
- *receive_RC6.asm* – tento podprogram realizuje príjem RC6 kódu zachyteného IR snímačom podľa vývojových diagramov (obr. 1.6 - obr.1.10). Pomocou makra TEST_BIT_RC6 sa dá rýchlo prestaviť port a pin, ktorým sa prijíma RC6 protokol.

Na začiatku sa meria pomocou 16-bitového čítača (counter16) dĺžka Leadera. Ak nie je dostatočne veľká (2666 μ s), detekcia sa ukončí a príjem prebehne od začiatku. Tento stav funguje ako filter prípadných poruchových impulzov z IR snímača – vylepšenie oproti protokolu RC5. Po úspešne vykonanej detekcii Leadera nastáva detekcia jednotlivých bitov (štart bit, mód bity, adresné a kontrolné bity + stav nečinnosti). Pri nábežnej hrane štartovacieho bitu sa naštartuje counter16 a po jeho dobehu (nastavenie timerFlag v prerušení Counter16INT) sa otestuje stav signálu RC6, znovu odštartuje counter16 a celý dej sa opakuje. Každý z bitov má 2 fázy, v jednej musí byť v stave ON, v druhej v stave OFF – podľa toho sa inkrementujú premenné cnt_on a cnt_off. Keď je zistený stav pri oboch fázach jedného bitu, porovnajú sa premenné cnt_on a cnt_off. Ak nie sú rovnaké, detekcia sa ukončí a príjem začne odznova. Tým sa dosahuje vysoká spoľahlivosť detekcie RC6 protokolu. Nakoniec sa testuje stav nečinnosti na príjme signálu tak, že counter16 sa spustí na dobu 2660 μ s a čaká sa na príchod stavu OFF signálu. Ak toto nastane, detekcia je prerušená a začne sa znova. Ak toto nenastane a je detekovaný timerFlag v stave 1 (counter16 dobehol), detekcia bola úspešná a zistený IR kód je ďalej spracovaný.

- *cakaj_koniec_doby.asm* - pomocný podprogram – čakanie na príchod timerFlag, ktorý sa nastaví po dobehu counter16
- *konverzia_imp.asm* - vypočíta skutočný počet impulzov v danom meranom časovom intervale ako rozdiel 0xFF a aktuálneho stavu čítača pri meraní dĺžky trvania Leadera
- *Set_Cnt16_222us.asm*
- *Set_Cnt16_444us.asm*
- *Set_Cnt16_666us.asm*
- *Set_Cnt16_888us.asm*
- *Set_Cnt16_950us.asm*
- *Set_Cnt16_2650us.asm* - nastavenie podmienok pre meranie danej doby pomocou čítača counter16 a jeho spustenie.

3.1.5. Modul ir_receiver.asm

V tomto module sa nachádzajú nasledovné podprogramy, ktoré zabezpečujú chod IR prijímača:

- *prevod_tab_kod.asm* - postupne prechádza až 15 riadkov konfiguračne tabuľky vo FRAM (štartovacie adresy 0x10 až 0xFF po kroku 0x10) a porovnáva 1.byte riadku s identifikovaným kódom z IR prijímača. Ak sa tieto byty zhodujú, 2.byte v aktuálnom riadku tabuľky je potom tabuľkový kód, ktorý sa následne vykoná. Ak ide o príkaz s parametrom, 3.byte je tento parameter.

- *akcie.asm* - tu sa vykonajú príkazy tabuľkového kódu popísané v 4.4. Na začiatku sa porovnáva tabuľkový kód postupne s kódmi jednotlivých akcií a v prípade zhody sa skočí na vykonanie akcie.

Tieto môžu byť

- jednorazové – napr. zapni svetlo
- s určitou dobou trvania – napr. zapni svetlo na určitú dobu danú parametrom v sek.
- trvalé – napr. zobrazuj reálny čas – do doby kým nepríde iný príkaz
- *data_displej.asm* - vyslanie dvoch hexa dát na displej
- *diagnostika_PC.asm* - vyslanie detekovaných diagnostických dát na PC
- *kod_RC6.asm* - vyslanie detekovaného RC6 kódu na displej, potom pauza 0.5 sekundy
- *display_init.asm* - vyslanie príkazu inicializácie displeja – zobrazí znaky '- - - -'
- *realny_cas.asm* - načítanie reálneho času z obvodu DS1307 a vyslanie na displej. Desatinná bodka bliká striedavo v sekundovom takte.
- *teplota.asm* - načítanie teploty z obvodu DS18S20 a vyslanie na displej
- *pwm_pridaj.asm* - zvýšenie šírky pulzu o 25 a aktivácia novej hodnoty PWM
- *pwm_uber.asm* - zníženie šírky pulzu o 25 a aktivácia novej hodnoty PWM
- *zapni_svetlo.asm* - nastavenie výstupného pinu pre ovládanie zopnutia svetla
- *vypni_svetlo.asm* - zmazanie výstupného pinu pre ovládanie zopnutia svetla
- *zapni_alarm.asm* - nastavenie výstupného pinu pre ovládanie zopnutia alarmu. Výstup pre piezobzučiac sa spína striedavo v sekundových intervaloch
- *vypni_alarm.asm* - zmazanie výstupného pinu pre ovládanie zopnutia alarmu

3.1.6. Modul *cnt_sl_ide.asm*

Veľmi dôležitá časť programu – hlavnej slučky. Je vyvolávaný v module *cnt_slave.asm* v podprograme *ModbusSlave.asm*, kde sa testuje príjem znakov z PC. Tým je možné súčasne komunikovať s PC a napr. zobrazovať reálny čas. Ak v podprograme *receive_RC6.asm* je prijatý kód z diaľkového ovládania, pomocou podprogramu *diagnostika_PC.asm* sa vyšle zistený kód na rozhranie s PC, pomocou podprogramu *kod_RC6.asm* na displej, ktorý ho zobrazí, pomocou naprogramovanej tabuľky v FRAM a podprogramu *prevod_tab_kod.asm* sa prevedie zistený kód na tabuľkový kód, ktorý je vykonaný v podprograme *akcie.asm*.

3.1.7. Modul *modbus_config.inc*

V tomto module sú uložené konštanty pre kódy akcií, kódy príkazov a parametrov komunikačného protokolu mikrobús a niektoré chybové kódy.

3.2. SW konfiguračného programu

Konfiguračný SW v PC obsahuje nasledovné moduly:

- *tabulka_kodov_IR_snimac.py* – maska (GUI) vytvorená v programe wxglade a doplnená o obsluhu udalostí (eventov) v jazyku python (viď obr. 4.1)
- *ir_cnt.py* – zjednodušená verzia programov pre obsluhu komunikácie s IR prijímačom
- *mbus_crc.py* – program na výpočet kontrolnej sumy protokolu MODBUS

Konfiguračný SW v PC plní nasledovné úlohy:

- vytvorenie konfiguračnej tabuľky prevodu kód tlačidla → tabuľkový kód príkazu
- zápis tabuľky FRAM procesora IR prijímača
- vyčítanie tabuľky z FRAM procesora IR prijímača
- vymazanie tabuľky v FRAM procesora IR prijímača
- načítanie a nastavenie času v obvode DS1307 IR prijímača
- výpis diagnostických bytov prijatého RC6 protokolu

3.2.1. Popis modulov konfiguračného programu

Modul *tabulka_kodov_IR_snimac.py* obsahuje nasledovné podprogramy:

- *zapis_tabulka* – priradí k akcii jej kód z pevnej tabuľky a pripravené hodnoty kódov zapíše do aktuálneho riadku tabuľky.
- *citaj_FRAM* – načíta aktuálnu konfiguračnú tabuľku z FRAM IR prijímača a zapíše do masky.
- *zapis_FRAM* - konfiguračnú tabuľku v FRAM IR prijímača prepíše aktuálnymi hodnotami tabuľky v maske
- *zmaz_FRAM* - aktuálnu konfiguračnú tabuľku v FRAM IR prijímača prepíše hodnotami 0xFF
- *NacitajCas* – načíta aktuálny čas z obvodu DS1307 IR prijímača a zapíše ho do masky.
- *ZapisCas* – nastavený čas z masky zapíše do obvodu DS1307 IR prijímača
- *devConnect* – prehľadá komunikačné porty a pokúsi sa na nich zakomunikovať s IR prijímačom – zostane pripojený na porte, kde zistil odozvu
- *zobr_dat* – príjem a zobrazenie diagnostických dát – zvláštne vlákno (thread) programu, cez ktoré sú prijímané dáta od IR prijímača

Modul *ir_cnt.py* obsahuje podprogramy na vyslanie telegramov v rámci akcií v predchádzajúcom module.

4. Užívateľský návod

4.1. Vytvorenie a zápis konfiguračnej tabuľky

The screenshot shows a Windows-style application window titled "Programovateľný IR prijímac". The interface is divided into several sections:

- File menu:** Located at the top left.
- Configuration Table:** A table with 15 rows, each representing a button configuration. The columns are: "cislo riadku tab.", "aktualny kod tlacidla", "1.kod tlacidla", "1.kod akcie", and "1.param.akcie".
- Control Panel:** On the left side, there are buttons for "zapis priradenie", "citaj FRAM", "zapis FRAM", and "zmaz FRAM".
- Time and Date:** Fields for "datum" (23.04.2011) and "cas" (17:58:23) with buttons "citaj cas" and "nastav cas".
- Results:** Fields for "vysledok citania" (OK) and "vysledok nastavenia".
- Diagnosis:** Fields for "diagnostika uvod" (0x95), "diagnostika trailer" (0x80), "diagnostika adresa" (0x04), and "diagnostika kod" (0x02).

cislo riadku tab.	aktualny kod tlacidla	1.kod tlacidla	1.kod akcie	1.param.akcie
1	0x02	0x01	0x01	0
	aktualna akcia	2.kod tlacidla	2.kod akcie	2.param.akcie
		0x02	0x02	0
	aktualny parameter	3.kod tlacidla	3.kod akcie	3.param.akcie
	0	0x03	0x03	0
		4.kod tlacidla	4.kod akcie	4.param.akcie
		0x06	0x04	0
		5.kod tlacidla	5.kod akcie	5.param.akcie
		0x5a	0x05	0
		6.kod tlacidla	6.kod akcie	6.param.akcie
		0x5b	0x06	0
		7.kod tlacidla	7.kod akcie	7.param.akcie
		0x58	0x07	0
		8.kod tlacidla	8.kod akcie	8.param.akcie
		0x59	0x08	0
		9.kod tlacidla	9.kod akcie	9.param.akcie
		0x5c	0x0a	15
		10.kod tlacidla	10.kod akcie	10.param.akcie
		0x0c	0x09	0
		11.kod tlacidla	11.kod akcie	11.param.akcie
		0xff	0xff	0
		12.kod tlacidla	12.kod akcie	12.param.akcie
		0xff	0xff	0
		13.kod tlacidla	13.kod akcie	13.param.akcie
		0xff	0xff	0
		14.kod tlacidla	14.kod akcie	14.param.akcie
		0xff	0xff	0
		15.kod tlacidla	15.kod akcie	15.param.akcie
		0xff	0xff	0

Obr. 4.1: Konfiguračná tabuľka vytvorená jazykom Python a jej maska (GUI) programom wxglade

Táto tabuľka pre diaľkové ovládanie priradí k ľubovoľnému tlačidlu diaľkového ovládania ľubovoľnú akciu z pevne naprogramovanej tabuľky príkazov. Užívateľ nemusí poznať kódy jednotlivých tlačidiel, môže si ich priradiť podľa vlastnej potreby na príslušné tlačidlo – môže ich "naučiť" a navzájom konfigurovať.

Postup:

1. naštartuje sa program *tabulka_kodov_IR_snimac.py* s pripojeným IR prijímačom k PC pomocou USB kábla – objaví sa maska ako na obr. 4.1 – v pravej časti sa nachádza 15 riadkov tabuľky so stĺpcami 'kod tlačidla', 'kod akcie' a 'param. akcie'
2. editovateľné okno číslo riadku tabuľky je automaticky nastavené na 1. Stlačením žiadaného tlačidla na diaľkovom ovládaní sa vyplní okno 'aktualny kod tlačidla'. V roletovom okne 'aktualna akcia' navolíme typ akcie, ktorú chceme vykonať pri stlačení tohto tlačidla. Ak zvolená akcia vyžaduje parameter, v okne 'aktualny parameter' ručne naeditujeme jeho hodnotu
3. stlačením tlačidla 'zapis priradenie' sa automaticky zapíšu predvolene údaje do prvého riadku tabuľky s tým, že do okna 'kod akcie' sa zapíše kód priradený pevne k danej akcii. Automaticky sa inkrementuje číslo riadku.
4. takto postupujeme dovtedy, pokiaľ nie je vyplnených všetkých 15 riadkov tabuľky, nepoužitie riadky by mali mať kódy tlačidla a akcie rovné 0xFF
5. po stlačení tlačidla 'zapis FRAM' sa obsah riadkov z masky zosníma, prenesie a zapíše do FRAM IR prijímača, stav prenosu je indikovaný widgetom gauge-vodorovným modrým pásom – prenos trvá cca 2-3 sekundy.
6. pri malých zmenách v konfigurácii je dobré vyčítať aktuálnu konfiguračnú tabuľku z IR prijímača stlačením tlačidla 'citaj FRAM' a následne urobiť požadované zmeny
7. pri stlačení tlačidla 'zmaz FRAM' sa celý obsah FRAM prepíše hodnotami 0xFF

4.2. Načítanie a nastavenie času v obvode DS1307 IR prijímača

V ľavej strednej časti masky sa nachádzajú widgety pre načítanie a nastavenie času v obvode reálneho času DS1307 IR prijímača. Stlačením tlačidla *citaj cas* sa v oknách *datum a cas* objavia hodnoty aktuálneho dátumu a času vyčítané z RTC DS1307. Je možné ich ručne editovať a upravené potom stlačením tlačidla 'nastav cas' zapísať do obvodu DS1307. Opakovaným vyčítaním času sa dá overiť jeho zápis. Výsledky operácií zápisu a čítania sú vypisované v oknách výsledok čítania a výsledok nastavenia.

4.3. Výpis diagnostických bytov prijatého RC6 protokolu

V ľavej dolnej časti masky sa nachádzajú okná pre diagnostické bity, ktoré zobrazujú 4 byty vyslané na PC po USB linke v momente zaregistrovania RC6 kódu. Z nich je možné usudzovať problémy pri prijímaní RC6 kódu.

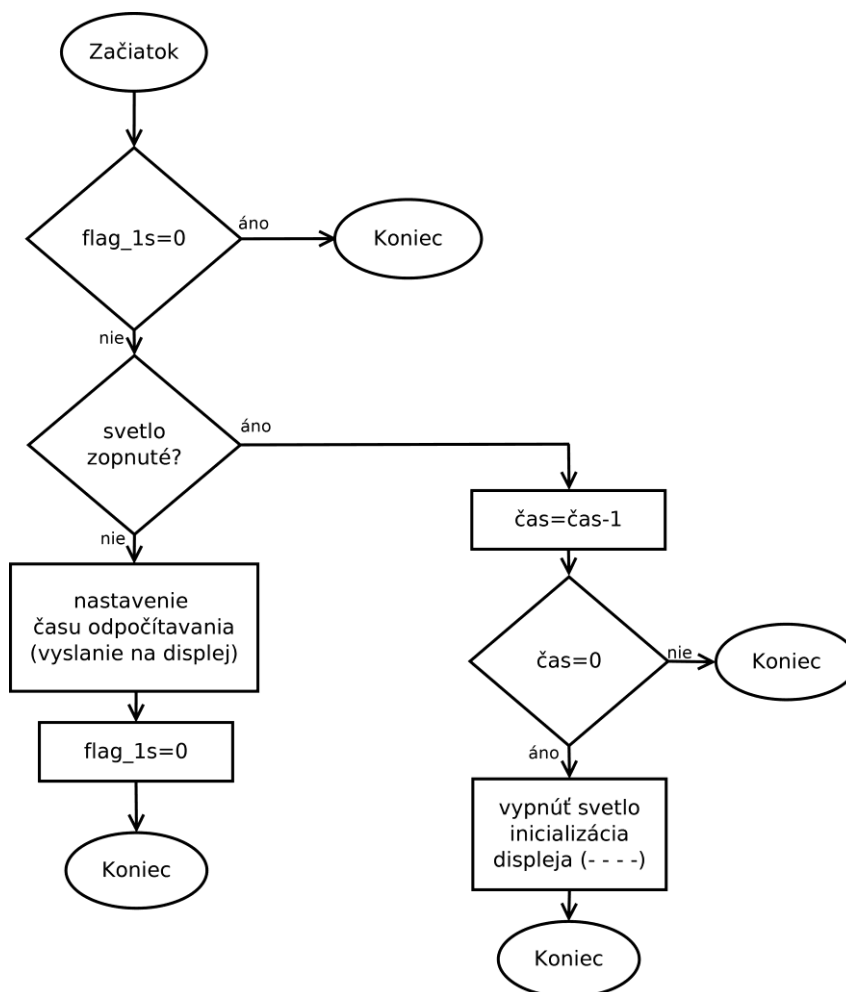
4.4. Popis povolených príkazov pri vytváraní konfiguračnej tabuľky

1. Jednorázové príkazy:

- zapni svetlo – zopne svetlo, ktoré ostane svietiť až do príkazu vypnutia svetla
- vypni svetlo – vypne svetlo, pričom bol predtým vyvolaný príkaz zopnutia svetla
- zapni alarm – zopne alarm v 1 s intervale
- vypni alarm – vypne alarm
- pridaj PWM – zvyšujú sa otáčky ventilátora
- uber PWM – znižujú sa otáčky ventilátora
- zobraz aktuálnu teplotu – zobrazenie aktuálnej teploty jednorázovým príkazom
- inicializácia displeja – na displeji sa zobrazia znaky "----"

2. Príkazy trvajúce obmedzenú dobu:

- zopni svetlo na dobu danú parametrom – znázornené na obr. 4.2

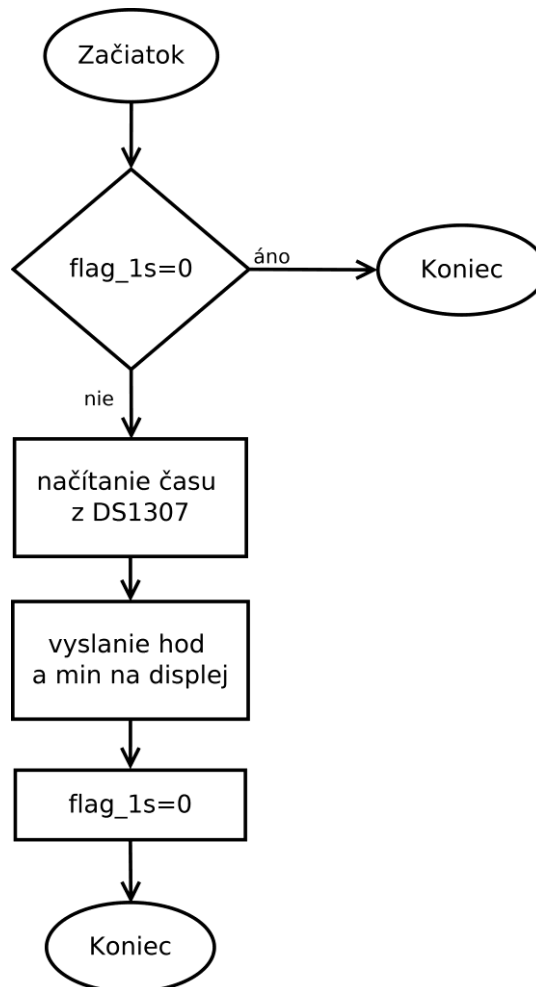


Obr. 4.2: Algoritmus zopnutia svetla na obmedzenú dobu

Pri spustení príkazu s nastaví čas doby zopnutia svetla hodnotou parametru príkazu a zapne sa svetlo. Na displeji sa zobrazuje zvyšný čas trvania zopnutia po sekundách a v momente dopočítania na nulu sa vypne svetlo a na displeji sa zobrazí stav inicializácie (----). Algoritmus príkazu je zobrazený na obr. 4.2.

3. Trvalé príkazy:

- zobrazenie aktuálneho času – znázornené na obr. 4.3



Obr. 4.3: Algoritmus riadenia zobrazenia aktuálneho času

Pri tomto príkaze je každú sekundu načítaná hodnota aktuálneho času z obvodu DS1307 a minúty so sekundami sú poslané na displej (obr. 4.3).

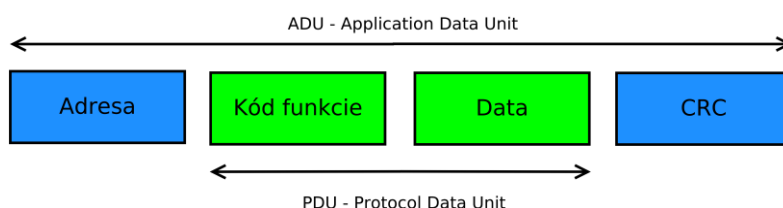
5. Popis komunikačného protokolu Modbus (μbus)

Rozlišujú sa dva základné typy komunikácie, a to:

- RTU – (Remote Terminal Unit) – binárny formát
- ASCII – znakový formát

Zariadenie programovateľného IR prijímača využíva typ komunikácie RTU (μbus) v rámci komunikácie na obsluhu FRAM + RTC (I2C) a Python + wxglade (konfiguračná tabuľka) a ďalej aj na komunikáciu s perifériou displej. Pod pojmom μbus sa rozumie zjednotený Modbus protokol, ktorý nie je na všetkých úrovniach kompatibilný s Modbus protokolom. Kompatibilný s Modbus protokolom je iba na nižšej úrovni.

Formát rámca protokolu je znázornený na obr. 5.1.



Obr. 5.1: Formát rámca protokolu Modbus (mikrobus) (prevzaté a upravené z [18])

Z obr. 5.1 je vidieť, že protokol obsahuje štyri časti, pričom adresa obsahuje 2 byty – 1. byte pre druh zariadenia a 2. byte pre číslo v skupine. Kódu funkcie prislúcha 1 byte a CRC (kontrolná suma) obsahuje 2 byty. V rámci tohto protokolu sa vysielajú telegramy medzi:

- doskou CNT a displejom (jednosmerná komunikácia smerujúca z CNT na displej), ktorá sa na displej zapíše vo forme dvoch hexa čísel a zobrazením znaku (-) na všetkých štyroch zobrazovačoch – inicializácia displeja.
- doskou CNT a PC (obojsmerná komunikácia) – vyslanie telegramu umožňuje zápis a čítanie FRAM, čítanie a zápis reálneho času (RTC).

Uvedené zápisy a čítania (FRAM a RTC) sa dajú zobrazit' na konfiguračnom SW na PC.

5.1. Popis použitých telegramov

5.1.1. Rozhranie IR prijímač ↔ PC:

- *test prítomnosti zariadenia*

PC: DEV_GROUP, DEV_ID, CMD_TEST, CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)

DEV_ID – číslo v skupine zariadení

CMD_TEST – kód príkazu

CRC1, CRC2 – byty kontrolnej sumy

IR prijímač: DEV_GROUP, DEV_ID, STATUS_ERROR, ERR_NONE, CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)
DEV_ID – číslo v skupine zariadení
STATUS_ERROR – kód stavu chyby
ERR_NONE – bez chyby
CRC1, CRC2 – byty kontrolnej sumy

- *zápis do FRAM IR prijímača*

PC: DEV_GROUP, DEV_ID, CMD_WRITE_FRAM, PAGE_NR, ADDR_MSB, ADDR_LSB, D1...D16, CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)
DEV_ID – číslo v skupine zariadení
CMD_READ_FRAM – kód príkazu
PAGE_NR – číslo stránky pamäte
ADDR_MSB – vrchný byte adresy čítania dát 16 bytov
ADDR_LSB - spodný byte adresy čítania dát 16 bytov
D1...D16 – 16 bytov dát na zápis do FRAM
CRC1, CRC2 – byty kontrolnej sumy

IR prijímač: DEV_GROUP, DEV_ID, STATUS_ERROR, ERR_NONE, CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)
DEV_ID – číslo v skupine zariadení
STATUS_ERROR – kód stavu chyby
ERR_NONE – bez chyby
CRC1, CRC2 – byty kontrolnej sumy

- *čítanie z FRAM IR prijímača*

PC: DEV_GROUP, DEV_ID, CMD_READ_FRAM, PAGE_NR, ADDR_MSB, ADDR_LSB, CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)
DEV_ID – číslo v skupine zariadení
CMD_READ_FRAM – kód príkazu
PAGE_NR – číslo stránky pamäte
ADDR_MSB – vrchný byte adresy čítania dát 16 bytov
ADDR_LSB - spodný byte adresy čítania dát 16 bytov
CRC1, CRC2 – byty kontrolnej sumy

IR prijímač: DEV_GROUP, DEV_ID, CMD_READ_FRAM, PAGE_NR, ADDR_MSB, ADDR_LSB, D1...D16, CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)

DEV_ID – číslo v skupine zariadení
CMD_READ_FRAM – kód príkazu
PAGE_NR – číslo stránky pamäte
ADDR_MSB – vrchný byte adresy čítania dát 16 bytov
ADDR_LSB - spodný byte adresy čítania dát 16 bytov
D1...D16 – 16 bytov dát z FRAM
CRC1, CRC2 – byty kontrolnej sumy

- *načítanie času z obvodu DS1307*

PC: DEV_GROUP, DEV_ID, CMD_READ_TIME, CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)
DEV_ID – číslo v skupine zariadení
CMD_READ_TIME – kód príkazu
CRC1, CRC2 – byty kontrolnej sumy

IR prijímač: DEV_GROUP, DEV_ID, CMD_READ_TIME, RTC_MEM1 ...
RTC_MEM8, CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)
DEV_ID – číslo v skupine zariadení
CMD_READ_TIME – kód príkazu
RTC_MEM1 ... RTC_MEM8 – vyčítané dáta 8 bytov
CRC1, CRC2 – byty kontrolnej sumy

- *zápis času z obvodu DS1307*

PC: DEV_GROUP, DEV_ID, CMD_SET_TIME, RTC_MEM1 ... RTC_MEM8,
CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)
DEV_ID – číslo v skupine zariadení
CMD_SET_TIME – kód príkazu
RTC_MEM1 ... RTC_MEM8 – data času na zápis
CRC1, CRC2 – byty kontrolnej sumy

IR prijímač: DEV_GROUP, DEV_ID,STATUS_ERROR, ERR_NONE,
CRC1,CRC2

Popis položiek telegramu:

DEV_GROUP – typ skupiny zariadenia (IR prijímač)
DEV_ID – číslo v skupine zariadení
STATUS_ERROR – kód stavu chyby
ERR_NONE – bez chyby
CRC1, CRC2 – byty kontrolnej sumy

- *vyslanie diagnostických bytov (4 byty)*

IR prijímač: úvod, trailer, ir_adresa, ir_kod

úvod – bity pred trailerom

trailer – trailer bit – striedavo sa mení

ir_adresa – adresa vysielajúceho diaľkového ovládania

ir_kod – prijatý kód z diaľkového ovládania

5.1.2. Rozhranie IR prijímač → displej:

- *inicializácia displeja*

IR prijímač: GRP_DISP, GRPNR_DISP, CMD_DISP_INIT, CRC1, CRC2

Popis položiek telegramu:

GRP_DISP – typ skupiny displeja

GRPNR_DISP – číslo v skupine displejov

CMD_DISP_INIT – kód príkazu

CRC1, CRC2 – byty kontrolnej sumy

- *zobrazenie 2 hexa znakov*

IR prijímač: GRP_DISP, GRPNR_DISP, CMD_DISP_HEX, HEX1, HEX2, CBDP, CRC1, CRC2

Popis položiek telegramu:

GRP_DISP – typ skupiny displeja

GRPNR_DISP – číslo v skupine displejov

CMD_DISP_HEX – kód príkazu

HEX1 – hexa číslo 1

HEX2 – hexa číslo 2

CBDP – riadiaci byte desatinnej bodky, horné 4 bity – flag zobrazenia
dolné 4 bity – pozícia zobrazenia desatinnej bodky

CRC1, CRC2 – byty kontrolnej sumy

6. Experimentálne skúšky zariadenia

So zariadením, ktorého obrázok je v prílohe D, sa previedli nasledovné skúšobné testy:

- Príjem RC6 kódu z dvoch rôznych diaľkových ovládačov (DVD a TV ovládačov)
- Zobrazenie prijatých kódov na displeji
- Vytvorenie konfiguračnej tabuľky na PC aplikácii so všetkými druhmi akcií.
- Zápis konfiguračnej tabuľky do FRAM IR prijímača (trvá asi 3 sekundy)
- Vyčítanie konfiguračnej tabuľky z FRAM IR prijímača (trvá asi 3 sekundy)
- Zmazanie konfiguračnej tabuľky vo FRAM IR prijímača (trvá asi 3 sekundy)
- Vykonanie všetkých druhov príkazov pomocou diaľkového ovládača.
- Dlhodobá prevádzka zariadenia bez poruchy (napr. “zamrznutie“)

Skúšky prebehli bez problémov s tým, že pri presune konfiguračnej tabuľky sa doporučuje, aby súčasne nebol zobrazovaný čas, pretože môže dôjsť z časových dôvodov k prerušeniu komunikácie s PC.

7. Záver

Táto bakalárska práca sa zaoberá návrhom riešenia pre HW a SW programovateľného IR prijímača. Ako riadiaci procesor bol navrhnutý Cypress CY8C29466. SW v IR prijímači detekuje IR kód komunikačného protokolu RC6 od firmy PHILIPS. Konfiguračný SW na PC je vytvorený pomocou programovacieho jazyka Python. Boli vykonané nasledovné práce:

- Analýza IR diaľkových ovládačov a nárokov na ich implementáciu pre detekciu mikropočítačom. Ako najvýhodnejším sa ukázal protokol RC6, ktorý je vylepšeným nástupcom protokolu RC5.
- V JSA (assembler) jazyka Cypress bol vytvorený program na detekciu komunikačného protokolu RC6 podľa navrhnutých algoritmov riešení (vývojových diagramov). Počas celého testovania nebol zistený ani jeden prípad registrácie iného kódu RC6 ako očakávaného
- Boli preštudované možnosti prepojenia navrhovaného zariadenia s konfiguračným PC. Bolo vybrané sériové rozhranie USB a aplikácia na strane PC bola urobená v jazyku python.
- V JSA (assembler) jazyka Cypress bol vytvorený program na komunikáciu s PC za účelom prenosu konfiguračnej tabuľky do FRAM IR prijímača
- Bol vytvorený PC program v programovacom jazyku Python na vytvorenie konfiguračnej tabuľky pre priradenie kódu príkazu ku kódu diaľkového ovládania
- Bol vytvorený PC program v programovacom jazyku Python na zosnímanie obsahu konfiguračnej tabuľky a jeho vyslanie na mikroprocesor pomocou mikrobusu - zjednodušeného modbus protokolu
- Boli urobené návrh a realizácia HW zapojenia programovateľného IR prijímača
- Naprogramovanie komunikácie medzi IR prijímačom a 7-segmentovým displejom so štyrmi ciframi – vizualizácia dejov v IR prijímači
- Naprogramovanie vysielania dát z prijatého IR kódu na PC pre diagnostické účely
- Vytvorenie a naprogramovanie sady inštrukcií s ohľadom na periférie
- Skúšky všetkých požadovaných vlastností preukázali plnú funkčnosť a spoľahlivosť chodu (prevádzky) zariadenia

Zadanie bakalárskej práce bolo splnené v celom rozsahu.

Hlavným motívom vytvorenia programovateľného IR prijímača je uľahčenie života ľuďom s obmedzenou pohyblivosťou. Pomocou diaľkového ovládania, na ktoré sú už zvyknutí z obsluhy televízora, môžu vyvolať aj rôzne iné funkcie ako zapnutie a vypnutie svetla, aktivácia alarmu alebo ovládanie otáčok ventilátora.

Použitá literatura

- [1] BERGMANS, S Philips RC-6 Protocol [online]. SB-Projects, Holandsko, 2001 – [cit. 28.12.2010]. Dostupné na [www: <http://www.sbprojects.com/knowledge/ir/rc6.htm>](http://www.sbprojects.com/knowledge/ir/rc6.htm)
- [2] BERGMANS, S Philips RC-5 Protocol [online]. SB-Projects, Holandsko, 2001 – [cit. 28.12.2010]. Dostupné na [www: <http://www.sbprojects.com/knowledge/ir/rc5.htm>](http://www.sbprojects.com/knowledge/ir/rc5.htm)
- [3] CYPRESS SEMICONDUCTOR. Algorithm - RC5 Codec AN2091 [online] 2010. [cit. 30.11.2010]. Dostupné na [www: <http://www.cypress.com/?docID=27892>](http://www.cypress.com/?docID=27892)
- [4] SKOPAL, M MMIA: IR dekóder/kóder, komunikace [online]. UREL FEEC VUT BRNO. [cit. 15.11.2010]. Dostupné na [www: <http://www.urel.feec.vutbr.cz/MIA/2009/skopal/index.html>](http://www.urel.feec.vutbr.cz/MIA/2009/skopal/index.html)
- [5] KOLBA, Z. Implementace protokolu RC-5 na platforme procesoru ATMEL AVR, 2010. [cit. 21.11.2010]. Diplomová práce. Kód: 25B24C75-1A4F-41AF-8791-FFA23F480D2A
- [6] RAMTRON. FM24C256. Full Data Sheet. [online] 2002. [cit. 22.5.2011]. Dostupné na [www: <http://www.avrcard.com/Documents/datasheets/fm24c256.pdf>](http://www.avrcard.com/Documents/datasheets/fm24c256.pdf)
- [7] MAXIM. DS1307. 64x8, Serial, I2C Real-Time Clock. [online] 2008. [cit. 22.5.2011]. Dostupné na [www: <http://datasheets.maxim-ic.com/en/ds/DS1307.pdf>](http://datasheets.maxim-ic.com/en/ds/DS1307.pdf)
- [8] CYPRESS SEMICONDUCTOR. PSoC Designer. [online] 2010. [cit. 28.12.2010]. Dostupné na [www: <http://www.cypress.com/?id=2522>](http://www.cypress.com/?id=2522)
- [9] CYPRESS SEMICONDUCTOR. PSoC Designer: Assembly Language User Guide. [online] 2003. [cit. 28.12.2010]. Dostupné na [www: <http://www.cypress.com/?docID=2723>](http://www.cypress.com/?docID=2723)
- [10] CYPRESS SEMICONDUCTOR. CY8C29466/CY8C29566/CY8C29666/CY8C29866 Full Data Sheet. [online] 2010. [cit. 28.12.2010]. Dostupné na [www: <http://www.cypress.com/?docID=24691>](http://www.cypress.com/?docID=24691)
- [11] PYTHON PROGRAMMING LANGUAGE. Python v2.7.1 Documentation. [online] 1990-2010. [cit. 28.12.2010]. Dostupné na [www: <http://docs.python.org/>](http://docs.python.org/)
- [12] wxGlade. A GUI builder for widgets Documentation. [online] 2009. [cit. 28.12.2010]. Dostupné na [www: <http://wxglade.sourceforge.net/index.php#documentation>](http://wxglade.sourceforge.net/index.php#documentation)
- [13] GEANY: Home page. Geany v0.20 Manual. [online] 2010. [cit. 28.12.2010]. Dostupné na [www: <http://www.geany.org/manual/dev/index.html>](http://www.geany.org/manual/dev/index.html)
- [14] CYPRESS SEMICONDUCTOR. CY8C27143/CY8C27243/CY8C27443/CY8C27543/CY8C27643 Full Data Sheet. [online] 2011. [cit. 22.5.2011]. Dostupné na [www: <http://www.cypress.com/?docID=27426>](http://www.cypress.com/?docID=27426)
- [15] VISHAY TELEFUNKEN. TSOP1730/TSOP1736/TSOP1738/TSOP1756 Full Data Sheet. [online] 2001. [cit. 28.12.2010]. Dostupné na [www: <http://www.datasheetcatalog.org/datasheets/208/301092_DS.pdf>](http://www.datasheetcatalog.org/datasheets/208/301092_DS.pdf)

- [16] gEDA Tool Suite, documentation. [online] 2011. [cit. 22.5.2011]. Dostupné na [www: <http://geda.seul.org/wiki/geda:documentation>](http://geda.seul.org/wiki/geda:documentation)
- [17] MAXIM. DS18S20. High-Precision 1-Wire Digital Thermometer. [online] 2010. [cit. 22.5.2011]. Dostupné na [www: <http://datasheets.maxim-ic.com/en/ds/DS18S20.pdf>](http://datasheets.maxim-ic.com/en/ds/DS18S20.pdf)
- [18] RONEŠOVÁ, A. Přehled protokolu MODBUS [online] 2005. [cit. 22.5.2011]. Dostupné z [www: <http://home.zcu.cz/~ronesova/bastl/files/modbus.pdf>](http://home.zcu.cz/~ronesova/bastl/files/modbus.pdf)

Zoznam symbolov, veličín a skratiek

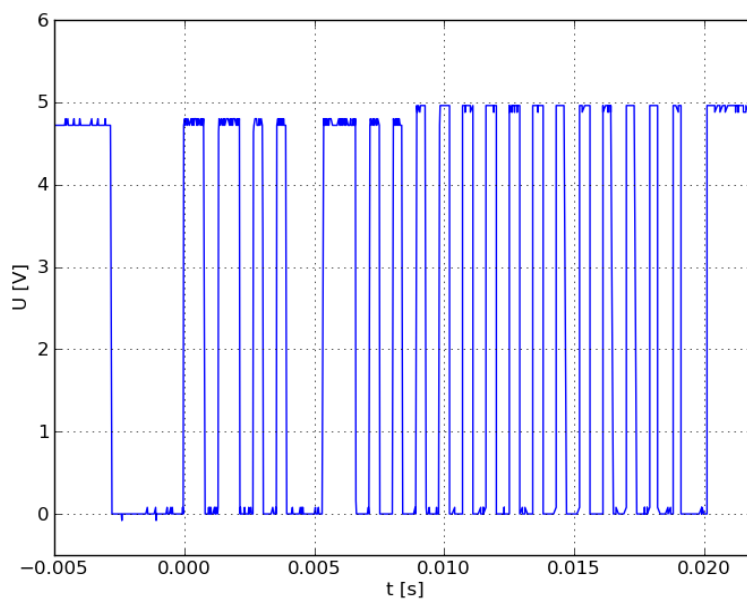
μbus	mikrobus
ADC	Analog to Digital Converter
ASCII	American Standard Code for Information Interchange
BCD	Binary Code Decimal
CD	Compact Disc
CNT	označenie pre riadiacu dosku (CONTROL)
CRC	Cyclic Redundancy Check
DAC	Digital to Analog Converter
DI	Digital Input
DO	Digital Output
DS	Dallas (Produkty od firmy Dallas)
DVD	Digital Video Disc or Digital Versatile Disc
FRAM	Ferroelectric Random Access Memory
GND	Ground – uzemnenie (zem)
GUI	Graphical User Interface
High Z	Vysoká impedancia [Ω]
HW	Hardware
I2C	Inter-Integrated Circuit
INT	Interrupt (prerušenie)
IR	Infrared
LED	Light-emitting diode
log.	logický
PC	Personal Computer
OFF	vypnutý
ON	zapnutý
PWM	Pulse-width modulation
Pull down	rezistor pripojený k zemi – stav nízka impedancia
Pull up	rezistor pripojený ku kladnému napájaciemu napätiu – stav vysoká impedancia
RAM	Random Access Memory
RC	Rivest Cipher
RS232	Recommended Standard 232
RTC	Real Time Clock
SMS	Short Message Service
Strong	Definovanie úrovne pinu Cypress procesoru (silný)
SW	Software
t	časová perióda (444 μs)
TTL	Transistor-transistor logic
TV	televízny, televízor
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
VCC	kladná polarita napájacieho napätia [V]
VCR	Video Cassette Recording
VV	Vstupy/Výstupy

Zoznam príloh

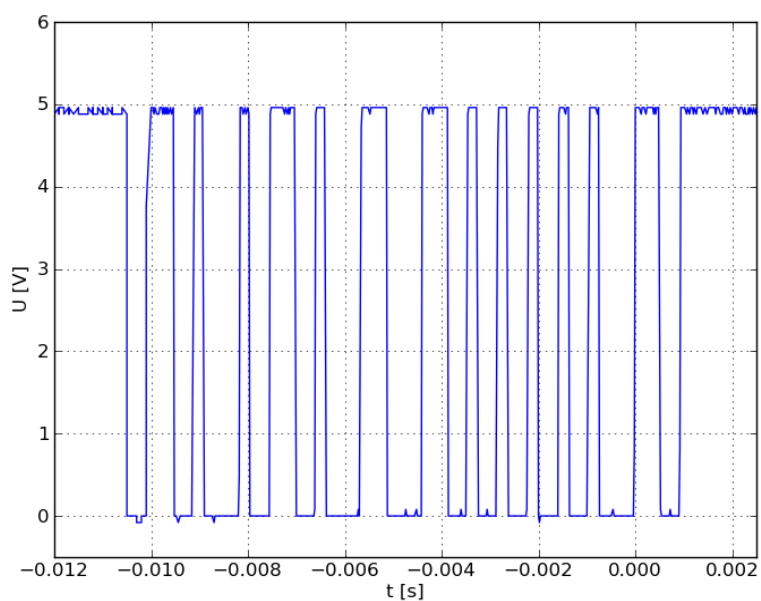
A	Priebehy zosnímaných komunikačných protokolov rôznych diaľkových ovládačov	40
A1	Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom PHILIPS TV pri stlačení tlačidla 1	40
A2	Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom T-HOME pri stlačení tlačidla 1	40
A3	Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom Panasonic TV pri stlačení tlačidla 1	41
A4	Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom HUMAX pri stlačení tlačidla 1	41
A5	Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom SILVERCREST pri stlačení tlačidla 1	42
A6	Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom Pioneer DVD pri stlačení tlačidla 1	42
A7	Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom Pioneer RECEIVER pri stlačení tlačidla 1	43
B	Výkresová dokumentácia	44
B1	Doska CNT s hlavným riadiacim procesorom celej aplikácie programovateľného IR prijímača	45
B2	Doska CNT – napájanie, konektory a prevodníky TTL/USB, TTL/RS232 ...	46
B3	Doska CNT s rozložením obvodu reálneho času DS1307 a štyroch pamätí FRAM 24256 riadenými cez komunikačnú zbernicu I2C	47
B4	Rozmiestnenie napájacieho zdroja periférií s jednotlivými perifériami	48
C	Popis prevzatých modulov (dodaných školiteľom)	49
D	Fotodokumentácia realizovaného zariadenia	51
E	Zoznam príloh na priloženom CD nosiči	52

A Priebehy zosnímaných komunikačných protokolov rôznych diaľkových ovládačov

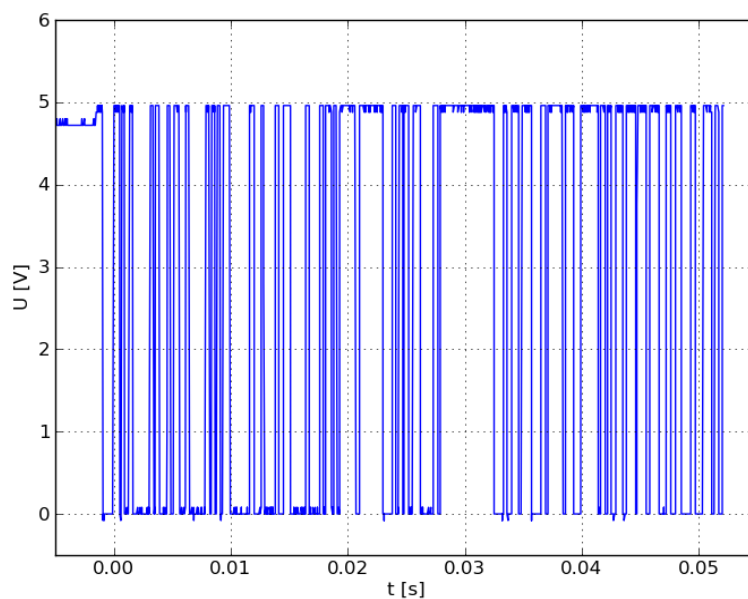
A1 Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom PHILIPS TV pri stlačení tlačidla 1



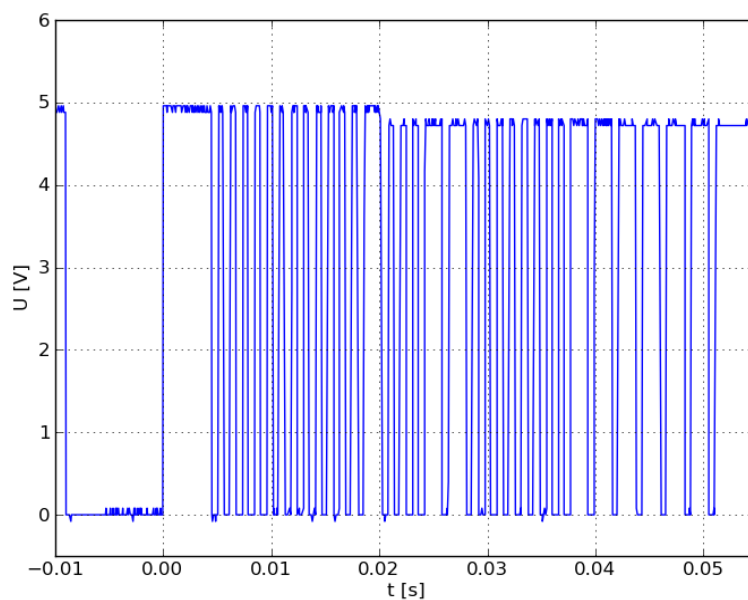
A2 Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom T-HOME pri stlačení tlačidla 1



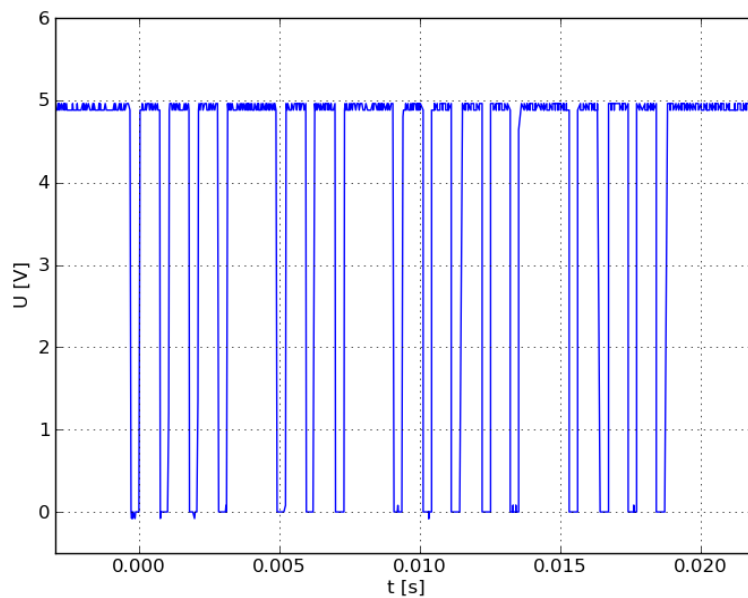
A3 Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom Panasonic TV pri stlačení tlačidla 1



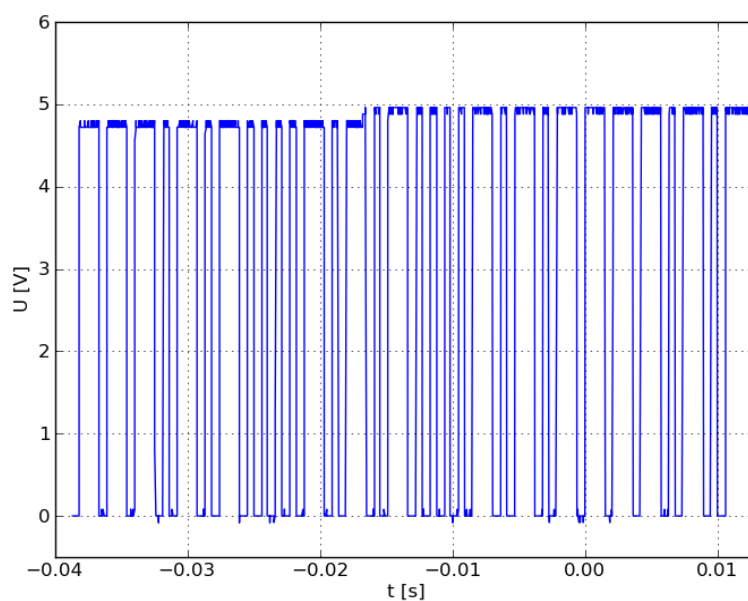
A4 Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom HUMAX pri stlačení tlačidla 1



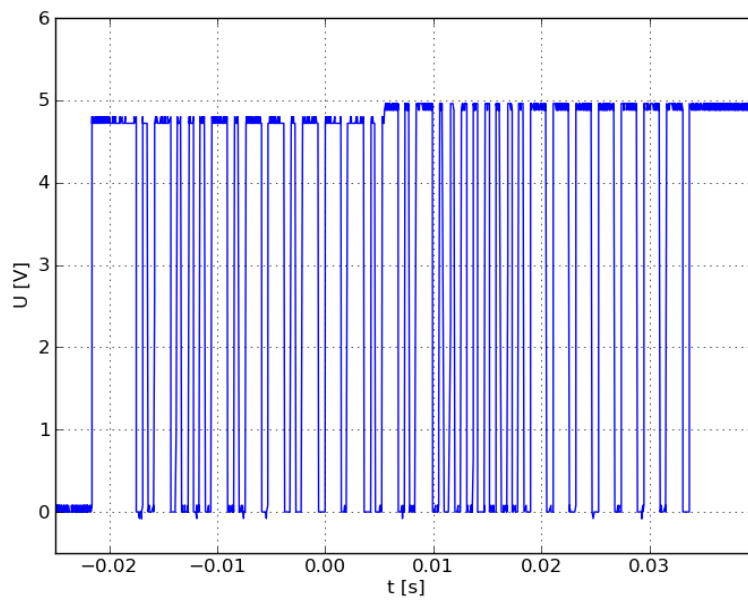
A5 Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom SILVERCREST pri stlačení tlačidla 1



A6 Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom Pioneer DVD pri stlačení tlačidla 1

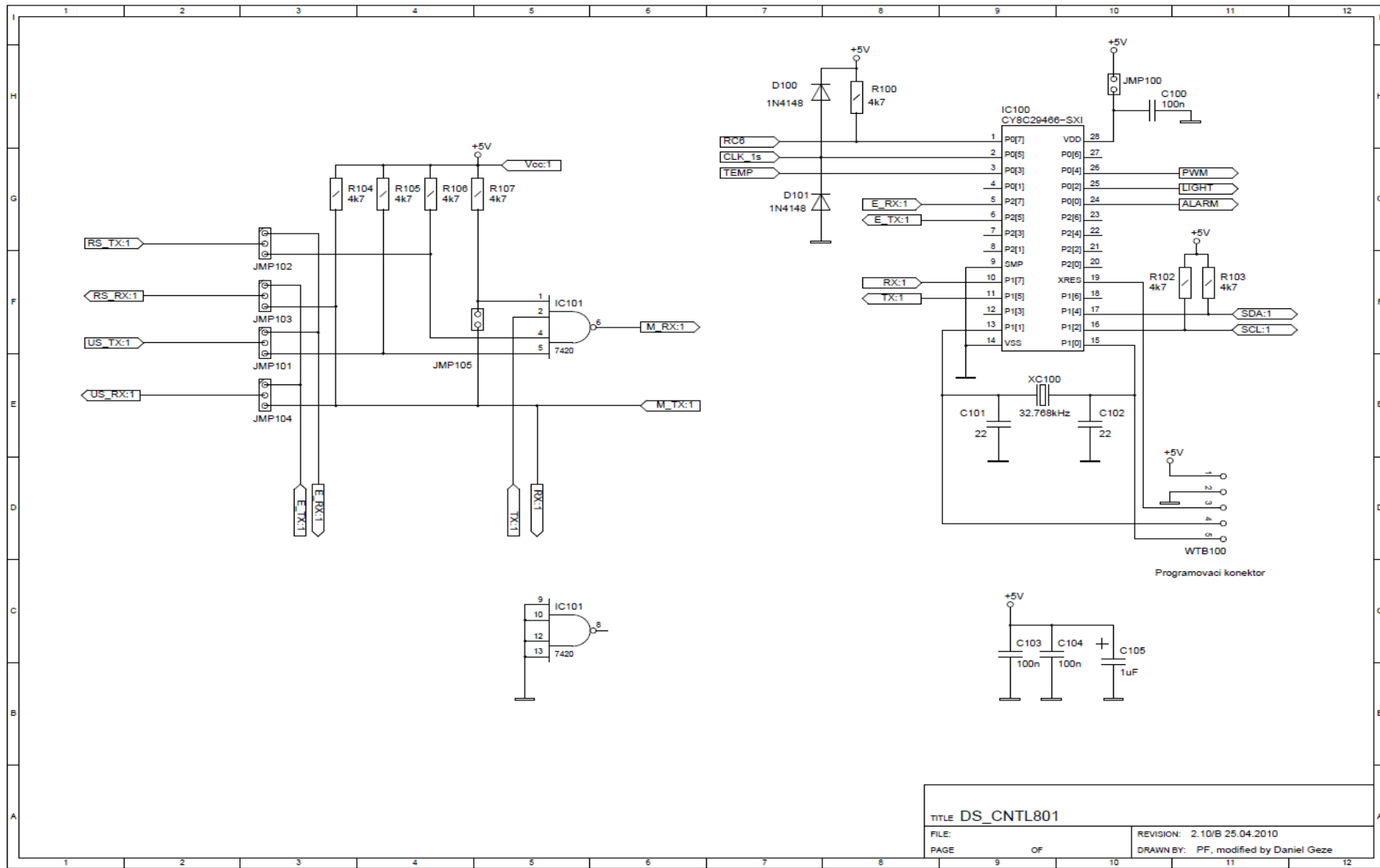


A7 Zobrazený priebeh signálu vyvolaný diaľkovým ovládačom Pioneer RECEIVER pri stlačení tlačidla 1



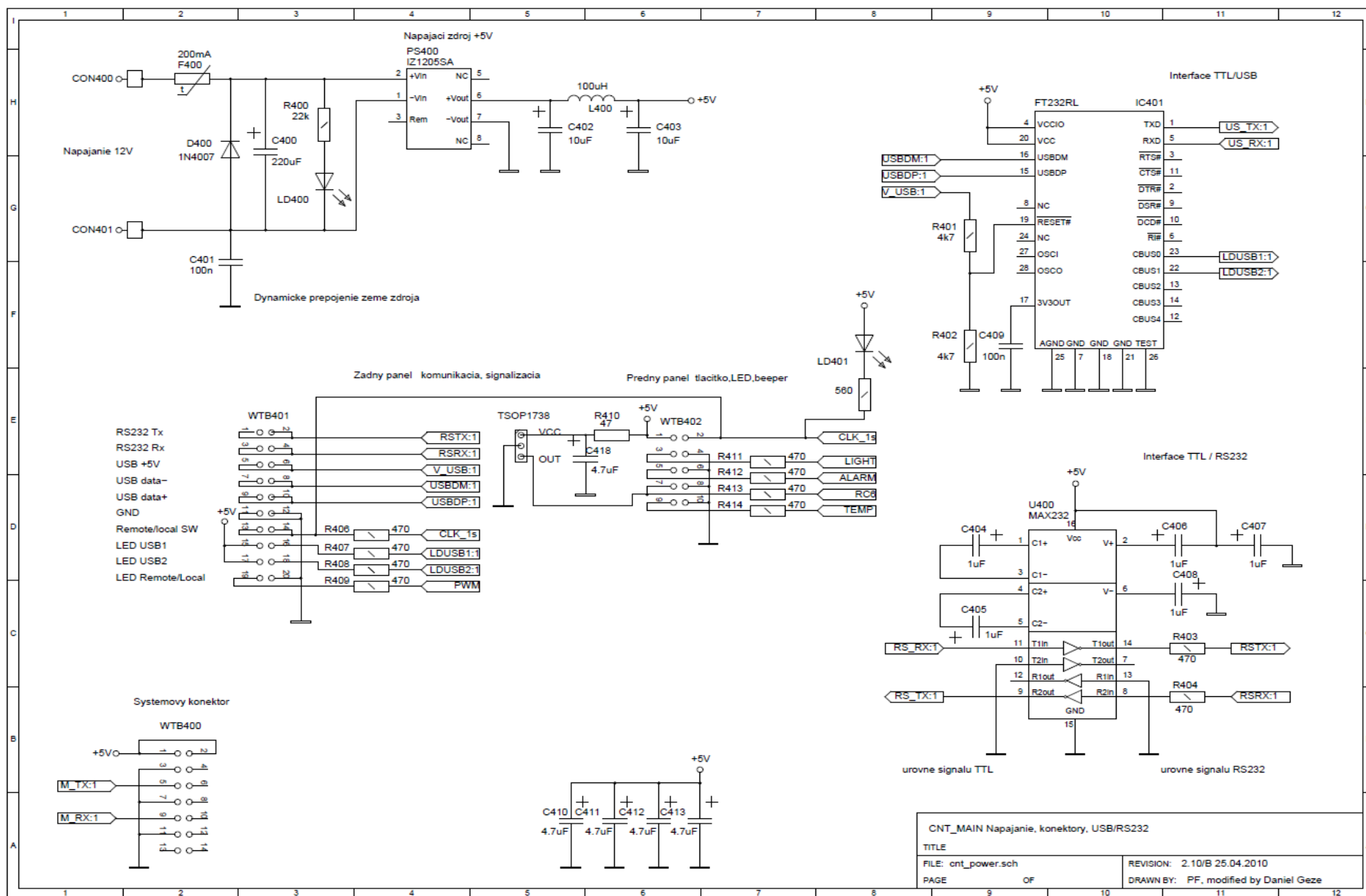
B Výkresová dokumentácia

B1 Doska CNT s hlavným riadiacim procesorom celej aplikácie programovateľného IR prijímača

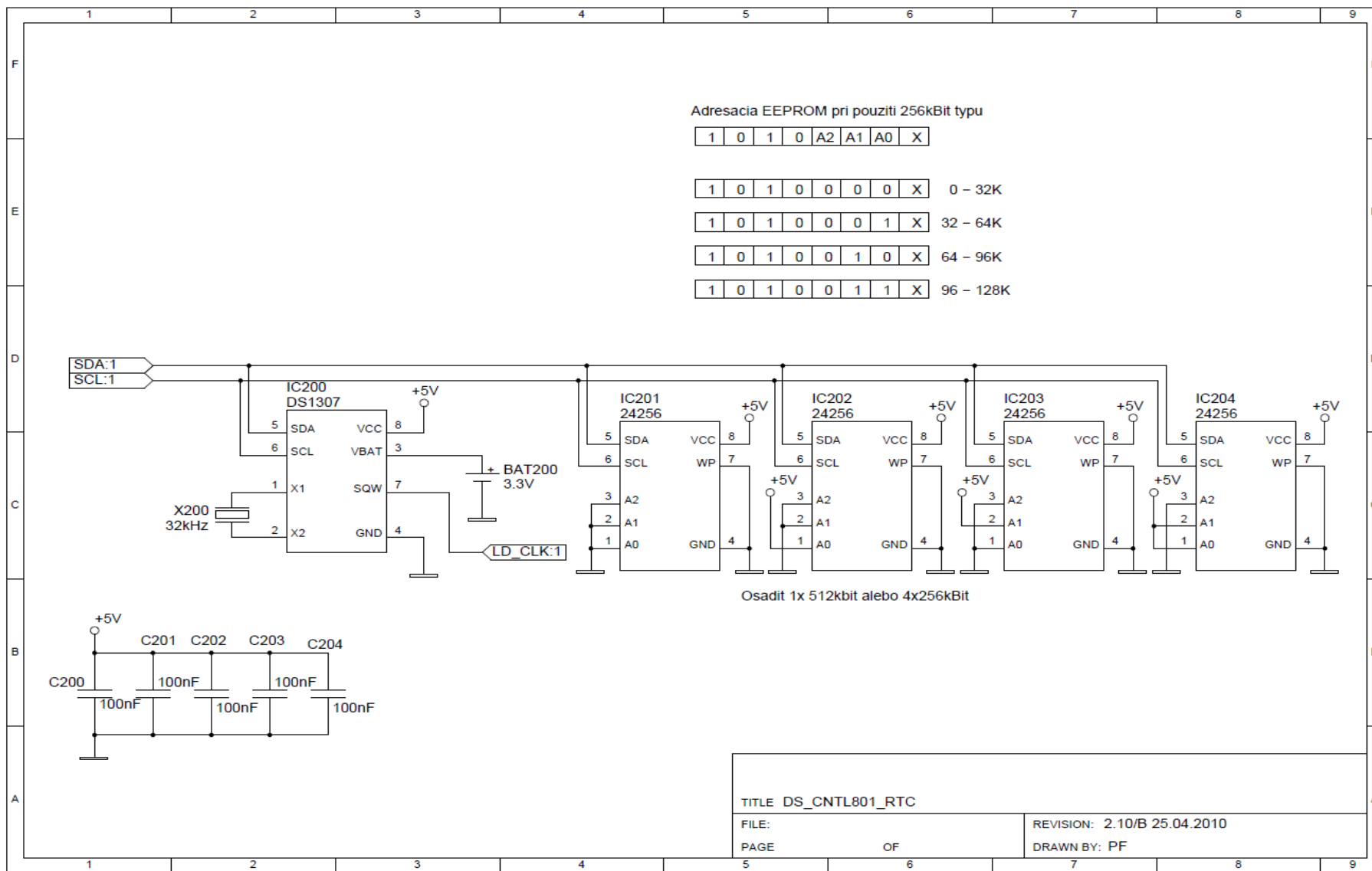


TITLE DS_CNTL801
 FILE: _____ REVISION: 2.10/B 25.04.2010
 PAGE _____ OF _____ DRAWN BY: PF, modified by Daniel Geze

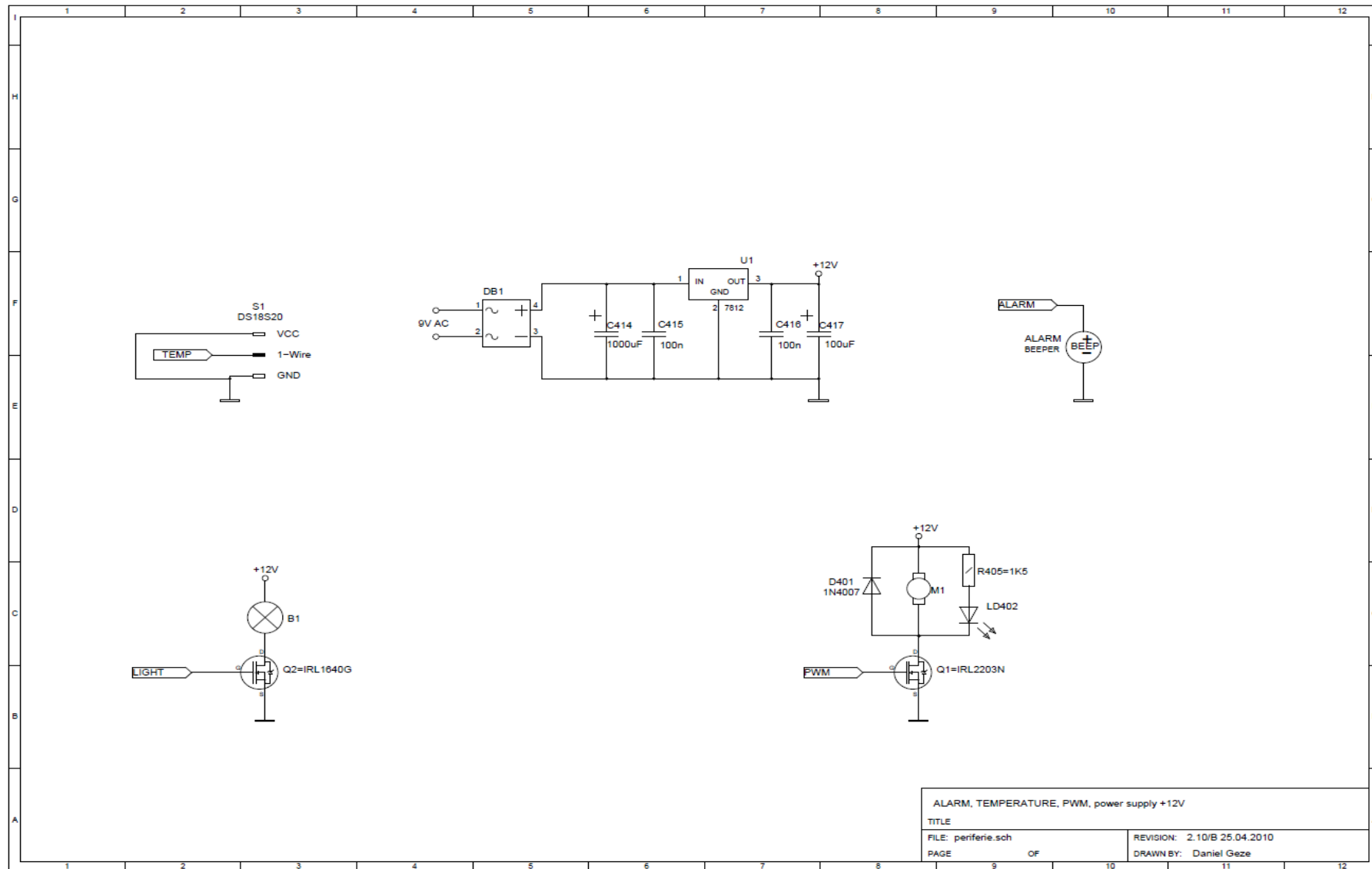
B2 Doska CNT – napájanie, konektory a prevodníky TTL/USB, TTL/RS232



B3 Doska CNT s rozložením obvodu reálneho času DS1307 a štyroch pamätí FRAM 24256 riadenými cez komunikačnú zbernicu I2C



B4 Rozmiestnenie napájacieho zdroja periférií s jednotlivými perifériami



C Popis prevzatých modulov (dodaných školiteľom)

- **modul cnt_ds1307.asm**

podprogramy:

- rtc_get_time - vyčítanie 8 bytov hodinového obvodu a uloženie do poľa rtc_mem
- rtc_set_time - nastavenie hodinového obvodu

- **modul cnt_fram.asm**

podprogramy:

- Read_FRAM - vyčítanie bloku dát (16 byte) z FRAM FM24C04 – I2C
- Write_Fram - zápis bloku dát (16 byte) do FRAM FM24C04 – I2C

- **modul cnt_master.asm**

podprogramy:

- send_message - vyslanie telegramu na displej
- WaitForTxInt - čakanie na odoslanie znaku do seriového rozhrania INT
- calcCRC – výpočet kontrolnej sumy telegramu Modbus

- **modul cnt_slave.asm**

podprogramy:

- ModbusSlave – hlavná slučka – čakanie na príjem znakov od PC a RC6 protokolu s vykonávaním akcií v podprograme StateIdle
- WaitForTx - čakanie na odoslanie znaku do sériového rozhrania EXT

- **modul cnt_sl_unicast.asm**

podprogramy:

- StateUnicast – vykonanie príkazov od mastera PC

- **modul ds18s20.asm**

podprogramy:

- Sensor_GetTemp - načítanie teploty z obvodu DS18S20.
- Sensor_GetID - načítanie ID senzora (8 bytov)
- Count_OneWireSW_CRC8 – výpočet kontrolnej sumy 1-Wire
- Match_ROM - výber zariadenia 1-Wire
- Delay750ms - oneskorenie

- **modul i2c_sw.asm**

podprogramy:

- I2C_Init, I2C_Start, I2C_Stop, I2C_Send, I2C_Recv – podprogramy pre obsluhu I2C rozhrania

- **math.asm**

podprogramy:

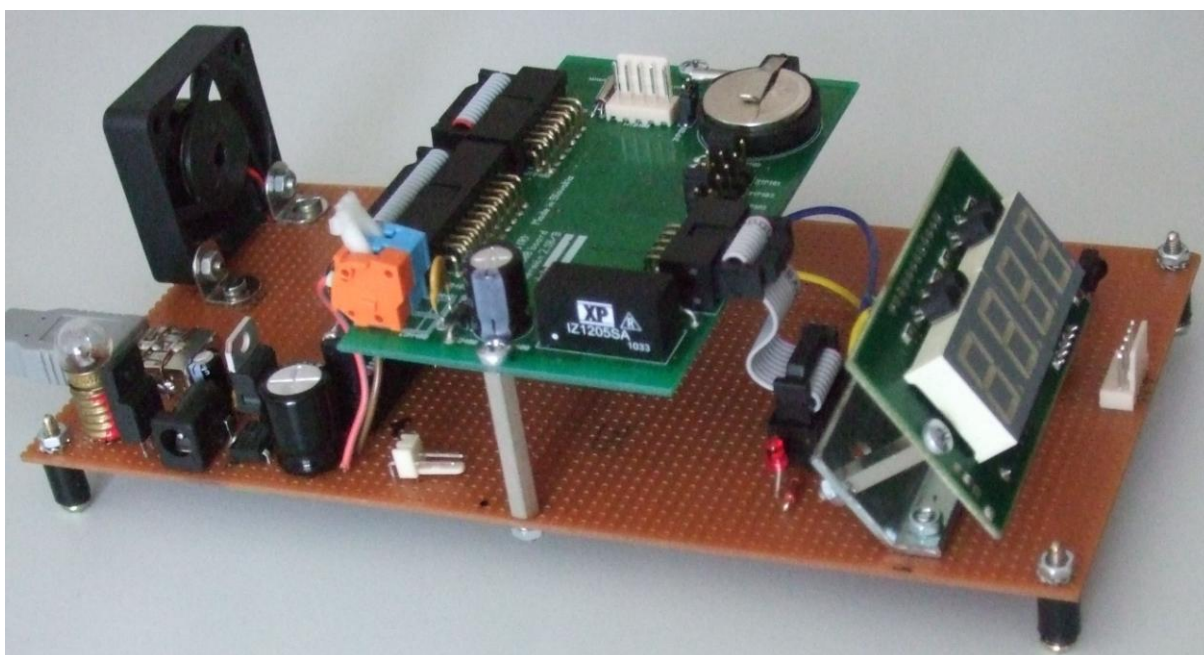
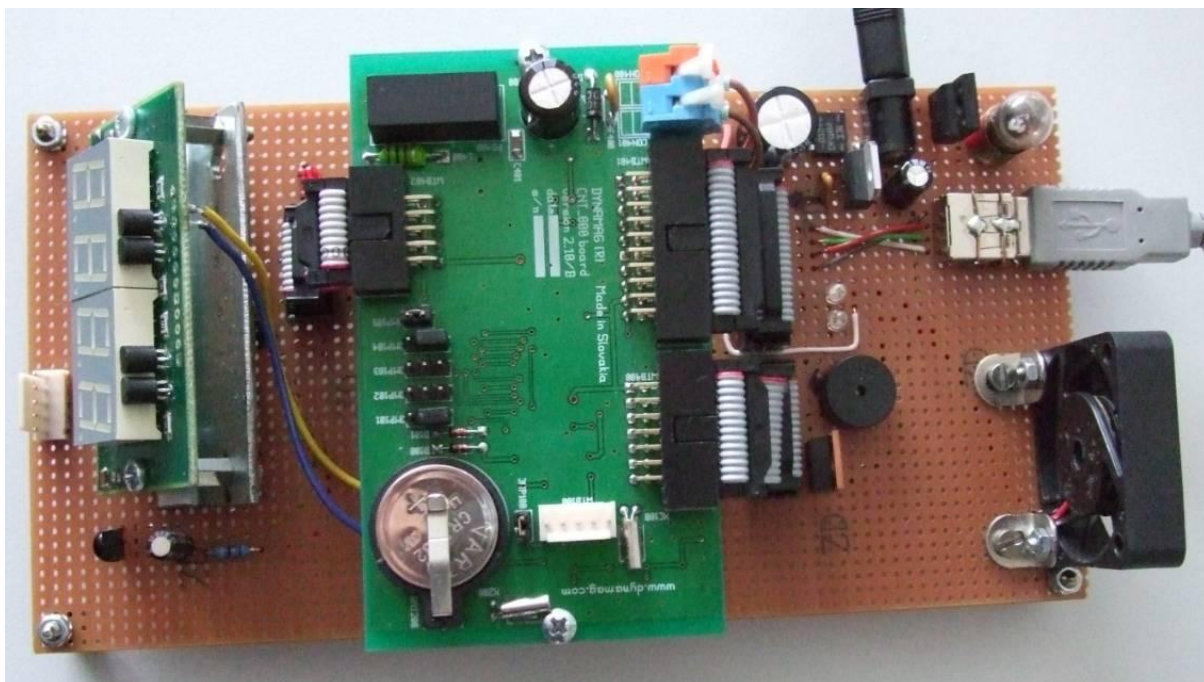
- SUMA, SUBT, DIV, MULT – matematické operácie
- BINBCD – prevod z binárneho kódu do BCD kódu

- **utils_delay.asm**

podprogramy:

- rôzne podprogramy pre oneskorenie

D Fotodokumentácia realizovaného zariadenia



E Zoznam príloh na priloženom CD nosiči

Elektronická verzia bakalárskej práce vo formáte PDF v adresári /thesis

Vytvorené zdrojové súbory v PSoC Designer ver. 4.4 v adresári /Source_Cypress

Vytvorené zdrojové súbory v Geany 0.18.1.1 v adresári /Source_Python

Fotodokumentácia zariadenia v adresári /photo

Katalógové listy vybraných elektronických komponentov v adresári /datasheets