
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická

Katedra telekomunikační techniky



Jednoduché zabezpečovací zařízení GSM

Září 2005

Vedoucí práce:

Petr Janata

Ing. Petr Hofman

Čestné prohlášení

Prohlašuji, že jsem zadanou bakalářskou práci zpracoval sám s přispěním vedoucího práce a používal jsem pouze literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé bakalářské práce nebo její části se souhlasem katedry.

Datum: 01. 09. 2005

.....

podpis studenta

Anotace

Cílem této bakalářské práce je navrhnout a realizovat spolehlivé zabezpečovací zařízení za použití technologie GSM, které využije nedávno spuštěné služby lokalizace mobilních stanic, Locator.

V Bakalářské práci se budu věnovat jednotlivým oblastem, kterými jsem prošel během návrhu aplikace. V úvodu práce se zabývám konstrukcí rozhraní k testování AT příkazů. Dále jsou popsány principy komunikace mezi mikropočítačem a mobilním telefonem. Hlavní část je věnována návrhu samotné aplikace Autoalarm.

Detailní pohled je věnován samotné komunikaci mezi volající a volanou mobilní stanicí. Zde jsou uvedeny různé způsoby průběhu navazování spojení a otestované AT příkazy.

Funkčnost celého programu spočívá ve správné reakci na vstupní čidla a datové zprávy z mobilního telefonu.

Summary

The main target of this work is to design and to implement unfailable safety appliance with use any GSM technology. This technology makes use of new service of localization mobil phones which is called Locator.

In my Bachelor's work I applied to individual fields which I had to know for realization of this work. In the introduction of this work I describe the construction boundary line for testing AT directions. It is followed by description of principles of communication between PC and mobile phone. The main part of my work is dedicated to application itself. The application is called Autoalarm.

I described in detail the communication between broadcasting and accepting station. Here are stated phases of connection of AT directions in singles.

The function of my program consists in relevant reaction of entering sensors and data messages from mobile phone.

Obsah

Obsah.....	4
Seznam obrázků	6
Seznam použitých zkratk.....	7
Úvod.....	8
1 Rozhraní mezi počítačem a mobilní stanicí.....	9
2 AT příkazy	11
2.1 Syntaxe AT příkazů	11
2.2 Příklady použití AT příkazů	11
3 Příklady použití AT příkazů během navazování spojení.....	15
4 Výběr vhodného jednočipu	17
4.1 Řady PIC16Fxx a PIC16Fxxx od firmy Microchip.....	17
5 Architektura mikročipu PIC16F628 firmy Microchip.....	19
5.1 Speciální registry	20
6 Sériové rozhraní.....	21
6.1 Asynchronní SCI (Serial Communication Interface).....	21
6.2 USART(Universal Synchronous Asynchronous Receiver Transmitter)	21
6.3 Asynchronní komunikace jednočipu PIC16F628	22
6.3.1 Vysílání dat	22
6.3.2 Kroky k nastavení asynchronního přenosu:.....	23
6.3.3 Příjem dat.....	23
6.3.4 Postup při asynchronním příjmu dat.....	24
7 Vývojový diagram.....	25
7.1 Popis částí programu.....	25
7.2 Další možná rozšíření softwarové části	25
7.3 Hlavní část programu.....	27
7.4 Podprogramy vývojového diagramu – Smyčka I.,II.....	28
7.5 Podprogramy vývojového diagramu – Alarm I.	29
7.6 Podprogramy vývojového diagramu – Alarm II.....	30
8 Algoritmy.....	31
8.1 Definování proměnných registrů a příznaků	31
8.2 Ukázkové podprogramy.....	33
9 SMS PDU formát	35
9.1 Příjem zprávy SMS v PDU formátu	36
9.2 Poslání zprávy SMS v PDU formátu	37
9.3 Příklad odeslání SMS pomocí AT příkazu	39
10 Blokové schéma zapojení.....	41
11 Zabezpečovací systémy	43

11.1	Imobilizér	43
11.2	Autoalarm	43
11.3	Dálkové ovládání	43
11.4	Snímače v automobilu	43
11.5	Signalizace	43
12	Lokalizace polohy mobilního telefonu.	44
12.1	Základní principy určování polohy telefonu	44
12.2	Princip lokalizace polohy mobilní stanice	44
12.3	Příklad použití služby Locator	45
13	Závěr	47
14	Použité zdroje	48
	Příloha A – Schémata zapojení Autoalarm a komunikace MT50-RS232.....	49
	Příloha B – DPS zapojení Autoalarmu.....	51
	Příloha C – Fotografie konstrukce Autoalarmu, komunikace MT50-RS232..	52
	Příloha D– Seznam součástek.....	53

Seznam obrázků

Obr. 1.1: Napěťové úrovně dle doporučení RS232	9
Obr. 1.2: Aplikační schéma převodníku RS232 na TTL	9
Obr. 1.3: Nastavení COM portu v programu Hyperterminal.....	10
Obr. 4.1: Konfigurační slovo mikropočítače PIC16F628.....	18
Obr. 5.1: Pouzdro DIP s 18 piny mikročipu PIC16F628.....	19
Obr. 5.2: Struktura paměti RAM mikročipu PIC16F628	20
Obr. 6.1: Blokové schéma TX registru.....	22
Obr. 6.2: Komunikace po seriové lince TX registru	23
Obr. 6.3: Blokové schéma RX registru.....	23
Obr. 6.4: Komunikace po seriové lince RX registru.....	24
Obr. 7.1: Vývojový diagram programu Autoalarm	27
Obr. 7.2: Vývojový diagram programu Autoalarm – Smyčka I., II.	28
Obr. 7.3: Vývojový diagram programu Autoalarm – Alarm I.....	29
Obr. 7.4: Vývojový diagram programu Autoalarm – Alarm II.	30
Obr. 9.1: Komunikační vrstvy SMSC.....	35
Obr. 9.2: Uspořádání dat SMS-DELIVER formátu.....	37
Obr. 9.3: Uspořádání dat SMS-SUBMIT formátu.....	39
Obr. 9.4: Okno Hyperterminálu – příkaz AT-CMGS	40
Obr. 9.5: 7-bitové kódování	40
Obr. 10.1: Blokové schéma zapojení	41
Obr. 10.2: Schéma optočlenu PC817	41
Obr. 10.3: Polohové čidlo	42
Obr. 12.1: Výpis PDU datagramu programem PDUSpy	46

Seznam použitých zkratk

BTS	(Base Transiever Station)	Základnová stanice.
DA	(Destination Adress)	Cílová adresa.
DCS	(Data Coding Scheme)	Nastavení kódování dat.
DSP		Deska plošného spoje.
GSM	(Global System for Mobile Communications)	Globální systém pro mobilní komunikaci.
ITU-T	(Internacional Telecommunication Union, Telecommunication sector)	Mezinárodní telekomunikační unie – oblast telekomunikací.
LSb	(Least Significant bit)	Nejnižší významový bit.
MO	(Mobile Oriented)	Volající mobil.
MR	(Message Reference)	Referenční číslo zprávy.
MS	(Mobile Station)	
MT	(Mobile Terminated)	Volaný mobil.
OA	(Originating Adress)	Telefonní číslo odesílatele.
PDU	(Protocol Data Unit)	Datový rámec SMS.
PID	(Protocol IDentifier)	Typ formátu dat.
RS232	(Recommended Standard number 232)	Doporučení standardu 232.
RxD	(Receive data)	Příjem dat.
SCA	(Service Center Adress)	Číslo střediska zpráv.
SCI	(Serial Comunication Interface)	Sériové rozhraní.
SCTS	(Service Center Time Stamp)	Parametr s datem a časem doručení SMS.
SIM	(Subscriber Identitiy Module)	Identifikační karta mobilního telefonu.
SMS	(Short message Service)	Krátká textová zpráva.
SMSC		SMS centrum.
TTL	(Transistor-transistor logic)	Tranzistorově – tranzistorová logika.
TSR	(Transmit Shift Register)	Rotační registr pro vysílání dat.
TxD	(Transmit data)	Vysílání dat.
UD	(User Data)	Uživatelská data.
UDL	(User Data Lenght)	Velikost uživatelských dat udávaná bytech.
USART	(Universal Synchronous Asynchronous Receiver Transmitter)	Univerzální synchronní, asynchronní příjem a vysílání dat.
VP	(Validity Period)	Čas platnosti SMS zprávy.

Úvod

Krádeže automobilů jsou v dnešním světě jeho obvyklou součástí. Výrobci vozidel se snaží zabezpečit své výrobky různými druhy autoalarmů, jenže mnohé z nich jsou nedostačující a krádeži nezabrání.

Masově rozšířené houkající a blikající alarmy mají celou řadu nevýhod. Aktivovaný alarm již nechává kolemjdoucí naprosto klidnými a pokud nejste zrovna v doslechové vzdálenosti od Vašeho vozu, nemáte možnost krádeži zabránit.

Mnohem výhodnější je tedy stav, kdy autoalarm nespustí ani klakson, ani světla, ale tichý alarm pomocí SMS (Short message Service). Operátoři mobilních telefonních sítí jsou schopni identifikovat polohu mobilního telefonu s přesností na 500 metrů.

V úvodní části bakalářské práce jsou popsány parametry komunikace mezi mobilním telefonem a PC. Jednočipové mikroprocesory a mnohé pomocné komunikační obvody pracují s úrovněmi TTL (Transistor-transistor logic) a pro připojení k PC je třeba je konvertovat pomocí převodníku úrovní. Je zde uvedeno aplikační schéma konvertoru RS232 (Recommended Standard number 232)/TTL a nastavení terminálu.

Kapitola 2 rozebírá syntaxi AT příkazů a jejich použití na mobilním telefonu Siemens MT50 podporující standard GSM 07.05. a GSM 07.07. (Global System for Mobile Communications).

Kapitola 3 uvádí AT příkazy testující navázání spojení mezi volající a volanou mobilní stanicí. Výsledky jsou základem pro realizaci algoritmu podprogramu Alarm1.

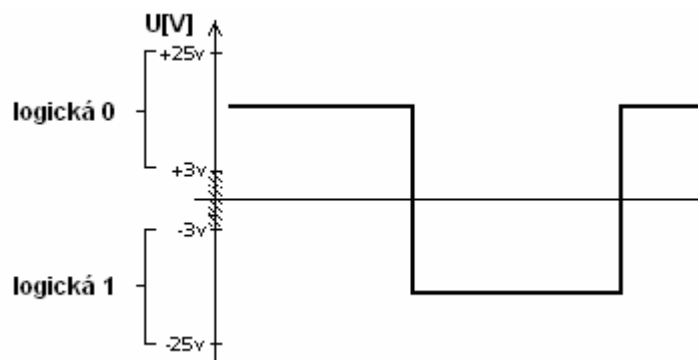
Pro návrh softwarové části je nutné popsat architekturu mikročipu a asynchronní přenos dat, což je popsáno v kapitole 4 až 6.

V kapitolách 7 až 8 se budu zabývat autoalarmem, který při krádeži vozidla zavolá a pošle SMS s udáním aktuální polohy vozu. Využiji novou službu „Locator“ mobilního operátora T-mobile. Služba využívá schopnosti lokalizovat MS (Mobile Station) v závislosti na hustotě pokrytí dané lokality BTS (Base Transceiver Station).

Následovat bude podrobný popis struktury PDU (Protocol Data Unit) datagramu s použitím AT příkazů. Závěrem mé práce jsou uvedeny základy lokalizace v GSM (Global System for Mobile Communications).

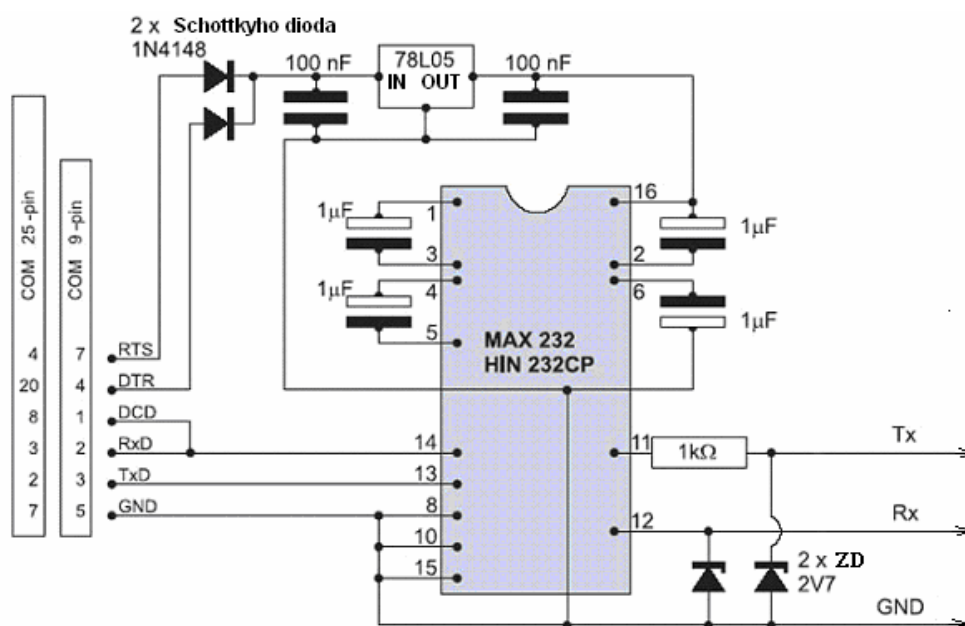
1 Rozhraní mezi počítačem a mobilní stanicí

Komunikace mezi PC a MS (Mobile Station) je asynchronní plně duplexní. Probíhá na sériovém rozhraní. Sériový port COM přenáší data dle doporučení RS232 (Recommended Standard number 232). Logické úrovně u RS232 jsou definovány -12V je logická 1 a +12V je logická 0 (viz Obr.1.1). MT pracuje s napětovými úrovněmi 3V. Je tedy nutné napětové úrovně konvertovat. Jedním z převodníků je integrovaný logický obvod od firmy Maxim MAX232.



Obr. 1.1: Napětové úrovně dle doporučení RS232

Převodník obsahuje dvě dvojice oddělovačů konvertujících napětové úrovně. Napětí pro RS232 se získává pomocí nábojové pumpy a výstupní napětí proto značně závisí na kvalitě použitých kondenzátorů, která u elektrolytických kondenzátorů časem značně klesá. Specialitou firmy MAXIM jsou obvody MAX203 a MAX233, které dokáží pracovat úplně bez potřeby vnějších kondenzátorů.

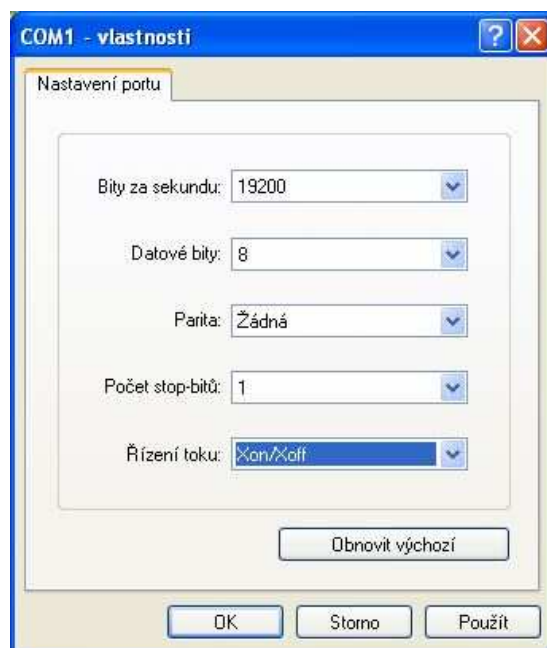


Obr. 1.2: Aplikační schéma převodníku RS232 na TTL

Aplikační schéma převodníku RS232 na TTL (na Obr.1.3) slouží ke komunikaci mezi PC a MT pomocí AT příkazů (tj. souboru příkazů sloužících ke spuštění jednotlivých funkcí mobilního přístroje). U každého výrobce se však soubor podporovaných příkazů může lišit. AT příkazy jsou ASCII znaky zasílané po lince TxD do telefonu. Odpověď telefonu přichází po lince RxD. Tyto příkazy mohou mít různý tvar. AT příkazů je velké množství a je možné je rozdělit do několika skupin:

- Standardní příkazy AT dle ITU-T (Internacional Telecommunication Union, Telecommunication sector) jsou definovány pro modemy
- AT příkazy GSM 07.07 - rozšíření pro mobilní komunikaci
- AT příkazy GSM 07.05 - navíc rozšířeny o příkazy pro obsluhu SMS
- AT příkazy GSM 07.07 – definovány pro GPRS
- AT příkazy GSM 11.14 – definovány pro SIM Toolkit
- Speciální AT příkazy

Tyto příkazy lze zasílat z PC pomocí softwarového terminálu. Lze vybrat z široké nabídky programů, jedním z nich je například Hyperterminál. Ten je dodáván spolu s příslušenstvím ve Windows XP. V terminálu je nutné nastavit: rychlost přenášených bitů/s, počet datových bitů, parita, počet stop bitů, typ řízení toku dat a číslo COM portu.



Obr 1.3: Nastavení COM portu v programu Hyperterminal

Rychlost je volena podle typu mobilního telefonu. U MT50 je rychlost 19200 bit/s.

2 AT příkazy

2.1 Syntaxe AT příkazů

Pro zjištění zda mobilní telefon příkaz podporuje se používá syntaxe:

- *AT+<příkaz>=?<CR>*.

Existují programy využívající této syntaxe k nalezení podporovaných AT příkazů. Jedním z nich je program AT Scan. Program posílá sekvenci znaků ve tvaru AT+xxxx. Na špatně zadané příkazy telefon odpoví zprávou ERROR. V opačném případě pošle zprávu OK.

- Pro zjištění nastavených hodnot z mobilu použijeme *AT+<příkaz>? <CR>* .
- Pro nastavení dat v mobilu, napíšeme *AT+<příkaz>= <parametr><CR>*.

Znak <CR> je potvrzení příkazu klávesou ENTER.

Při komunikaci z procesoru se potvrzení <CR> nahradí znakem 0Dh. Nejjednodušším AT příkazem je samotná dvojice znaků AT (AT příkaz se ukončuje Enterem). Odpovědí telefonu na správně zadaný a provedený AT příkaz je OK. V kapitole 2.2 je uveden seznam nejčastěji používaných příkazů s příklady odpovědí, které posílá mobilní telefon. Pro detailnější informace odkazují na firemní datasheet pro používaný mobil. MT50 podporuje stejné AT příkazy jako např. Siemens C35. Zde je většina příkazů uvedena podrobněji. U některých bylo třeba najít odezvu na daný příkaz testováním přes rozhraní PC-MT. Odpovědi jsou uvedeny bez řídících znaků, které telefon při odpovědi pošle, t.j. <CR> a <LF>. Mobilní telefon má standardně nastavené echo, tzn. přeposílání přijatého AT příkazu, což lze využít ke kontrole správnosti komunikace s MT. Toto echo je možné deaktivovat příkazem ATE0<CR>. Poslední AT příkaz lze opakovat zasláním A/<CR>, což je jediný příkaz nezačínající písmeny AT. Základní AT příkazy jsou uvedeny níže.

Telefon posílá tyto zprávy: **OK, RING, NO CARRIER, ERROR, NO DIALTONE, BUSY, CONNECT.**

2.2 Příklady použití AT příkazů

Řada otestovaných příkazů mobilním telefonem Siemens MT50.

- **AT** - Umožňuje otestovat komunikaci s MT.

AT <CR>

OK

- **ATA** - Vyzvednutí příchozího hovoru. Pokud na MT žádný hovor nepřichází tak MT odešle ERROR. Příchozí hovor MT zasláním zprávy RING.

ATA <CR>

- OK
- **ATH** - Zrušení probíhajícího hovoru (zavěšení).
ATH <CR>
OK
- **ATD** - Vytočení telefonního čísla.
ATD0602123456; <CR>
OK
- **AT+CPBR** - Vyčte z MT seznam. Vyčte z MT tel. čísla od místa 1 do místa 4. Samozřejmě lze čísla upravit podle potřeby. Před tímto příkazem je třeba definovat paměť, z které chceme číst viz +CPBS.
AT+CPBR=1,4 <CR>
+CPBR: 1,"732123123",129,"FRANTA"
+CPBR: 2," 732124563",129,"PETR"
+CPBR: 3," 732128123",129,"ALENA"
+CPBR: 4," 732129633",129,"MARTIN"
OK
- **AT+CPBS** - Definuje pozici telefonního seznamu v MT.
"SM" = paměť SIM-karty.
"ME" = paměť telefonu.
AT+CPBS="ME" <CR>
OK
- **AT+CPMS** - Definuje paměť pro práci s SMS. "SM" = paměť SIM karty
"ME" = paměť telefonu.
AT+CPMS="SM","SM" <CR>
OK
- **AT+CMGL** - Vypíše SMS zprávy. Číslo v příkazu udává typ čtených zpráv:
Pro SIEMENS:
0 - přijaté, nečtené zprávy (nastaveno defaultně)
1 - přijaté, přečtené zprávy
2 - uložené, neodeslané zprávy
3 - uložené, odeslané zprávy
4 - všechny zprávy
AT+CMGL=1 <CR>
+CMGL: 5,1,,26
0791246030500200040C912470233823740000505050806332
8008C769B365CD4263
v hexa kódu:

0D	0A	2B	43	4D	47	4C	3A	20	35	2C	31	2C	2C	32	36	0D
0A	30	37	39	31	32	34	36	30	33	30	35	30	30	32	30	30
30	34	30	43	39	31	32	34	37	30	32	33	33	38	32	33	37
34	30	30	30	30	35	30	35	30	35	30	38	30	36	33	33	32
38	30	30	38	43	37	36	39	42	33	36	35	43	44	34	32	36
33																

- **AT+CMGD** - Smazání příslušné SMS zprávy. Číslo v příkazu udává pozici zprávy, která má být smazána.

AT+CMGL=0

OK

v hexa kódu:

**41 54 2B 43 4D 47 4C 3D 30 0D
0D 0A 4F 4B 0D 0A**

- **AT+CMSS** - Odešle SMS zprávu uloženou v seznamu. Číslo v příkazu udává pozici zprávy v seznamu. Je potřeba nejdříve definovat paměť, se kterou se bude pracovat viz +CPMS. SMS musí být uložena i s telefonním číslem (paměť zpráv - odeslané). Druhý povel odešle uloženou SMS ze SIM na tel. číslo zadané za pozicí SMS. Telefonní číslo je sice součástí uložené SMS, ale pokud zadáme jiné jako parametr tohoto příkazu, pak má toto nové číslo přednost.

**AT+CMSS=1 nebo AT+CMSS=1,+420603123456<CR>
+CMSS: 200**

OK

v hexa kódu:

**41 54 2B 43 4D 53 53 3D 31 0D 0D 0A 2B 43 4D 53 53
3A 20 32 30 30 0D 0A 0D 0A 4F 4B 0D 0A**

- **AT+CMGS** - Odeslání SMS ve tvaru PDU. Po zadání tohoto příkazu je možné odeslat SMS v PDU formátu. Parametrem příkazu je délka PDU rámce.

**AT+CMGS=21 <CR>
>01000C91247023382374000009C12AF5196406A54D<CTRL-
Z>**

+CMGS: (mr) OK

mr=reference zprávy

- **AT+CLIP** - Zapne zobrazování telefonního čísla příchozího hovoru. Při příchozím hovoru to potom vypadá:

RING

+CLIP: "+420602123456",145,,,,,0

Při zadání čísla "0" v příkazu se zobrazování čísla příchozího hovoru vypne.

AT+CLIP=1<CR>

OK

v hexa kódu:

**41 54 2B 43 4C 49 50 3D 31 0D
0D 0A 4F 4B 0D 0A**

- **ATE** - Zapíná a vypíná echo z MT. Zapnuté echo znamená, že při zadání povelu do MT se vám před potvrzením vrátí i zadaný příkaz. Defaultně je echo zapnuto. Echo se zapne, pokud do příkazu zadáte jedničku.

příklad vypnutí echa:

ATE0<CR>

OK

v hexa kódu:

**41 54 45 30 0D
0D 0A 4F 4B 0D 0A**

- **AT+CCLK** - Stáhne aktuální čas zobrazovaný na displeji MT.

AT+CCLK?

+CCLK: "02/01/02,05:39:08"

OK

v hexa kódu:

**41 54 2B 43 43 4C 4B 3F 0D
0D 0A 2B 43 43 4C 4B 3A 20 22 30 32 2F 30 31 2F 30
32 2C 30 35 3A 33 39 3A 30 38 22 0D 0A 0D 0A 4F 4B
0D 0A**

- **AT+CLCC** - Přehled aktuálního volání. Nula nám říká, že se volá z MO (mobile oriented=volající mobil). A trojka indikuje právě vyzváněcí tón. Číslo 145 znamená, že je číslo v mezinárodním formátu tj. má 12 čísel s předvolbou +420.

příklad :

AT+CLCC

at+clcc<CR> (echo)

<CR><LF>+CLCC:

1,0,3,0,0,"+420732833247",145<CR><LF>

OK

v hexa kódu:

**41 54 2B 43 4C 43 43 0D
0D 0A 2B 43 4C 43 43 3A 20 31 2C 30 2C 33 2C 30 2C
30 2C 22 2B 34 32 30 37 33 32 38 33 33 32 34 37 22
2C 31 34 35 0D 0A 0D 0A 4F 4B 0D 0A**

- **AT+CEER** - Výpis chybového hlášení po ukončení spojení mezi MO (mobile oriented) a MT (mobile terminated). Po ukončení spojení lze pomocí tohoto příkazu identifikovat, jakým způsobem bylo spojení ukončeno.

příklad:

AT+CEER<CR>

<CR><LF>+CEER: 0,0<CR>

<CR><LF>OK<CR><LF>

3 Příklady použití AT příkazů během navazování spojení

Při ověřování, zda bylo volání přijato, jak probíhá a jakým způsobem bylo ukončeno, slouží příkazy AT+CLCC a AT+CEER.

Vytočím-li číslo na MO příkazem ATD (viz minulá kapitola), mohou nastat tyto případy:

- MT ukončil spojení zavěšením: MO vyšle zprávu BUSY
- MT zvedl a následně ukončil aktivní spojení (hovor) zavěšením: MO vyšle zprávu NO CARRIER.
- Po navázání spojení mezi MO a MT nedojde k hovoru (majitel neslyší či nezvedá mobil): MO nevyšle žádnou zprávu. Na straně MT se na displeji po minutě a půl objeví nepřijatý hovor. MO po uplynutí doby dvou minut ukončí spojení.
- Volaný mobil je vypnutý: Z MO se ozývá „volaný účastník je nedostupný“. To se se opakuje minutu a půl. MO nevyšle žádnou zprávu.

O průběhu volání nás informuje příkaz AT+CLCC (List current calls):

- vyzváněcí tón :

```
AT+CLCC<CR>
<CR><LF>+CLCC: 1,0,3,0,0,"+420732833247",145<CR><LF>
<CR><LF>OK<CR><LF>
```

v hexa kódu:

```
41 54 2B 43 4C 43 43 0D
0D 0A 2B 43 4C 43 43 3A 20 31 2C 30 2C 33 2C 30 2C 30 2C 22
2B 34 32 30 37 33 32 38 33 33 32 34 37 22 2C 31 34 35 0D 0A
0D 0A 4F 4B 0D 0A
```

- volba volaného čísla:

```
AT+CLCC<CR>
<CR><LF>+CLCC: 1,0,2,0,0,"+420732833247",145<CR><LF>
<CR><LF>OK<CR><LF>
```

v hexa kódu:

```
41 54 2B 43 4C 43 43 0D
0D 0A 2B 43 4C 43 43 3A 20 31 2C 30 2C 32 2C 30 2C 30 2C 22
2B 34 32 30 37 33 32 38 33 33 32 34 37 22 2C 31 34 35 0D 0A
0D 0A 4F 4B 0D 0A
```

- spojení mezi MO a MT je aktivní (hovor):

```
AT+CLCC<CR>
<CR><LF>+CLCC: 1,0,0,0,0,"+420732833247",145<CR><LF>
<CR><LF>OK<CR><LF>
```

v hexa kódu:

```
41 54 2B 43 4C 43 43 0D
```

0D 0A 2B 43 4C 43 43 3A 20 31 2C 30 2C 30 2C 30 2C 30 2C 22
2B 34 32 30 37 33 32 38 33 33 32 34 37 22 2C 31 34 35 0D 0A
0D 0A 4F 4B 0D 0A

- Po ukončení spojení (datové cesty) mezi MO a MT :

AT+CLCC<CR>
<CR><LF>OK<CR><LF>
v hexa kódu:
41 54 2B 43 4C 43 43 0D
0D 0A 4F 4B 0D 0A

Ukončení hovoru zavěšením na straně MT lze zjistit, protože MO vyšle zprávu NO CARRIER.

V případech kde jsme od MO neobdrželi žádnou zprávu je nutné použít příkaz AT+CEER (Cellular Extended Error Report):

- MT byl zavěšen:

AT+CEER<CR>
<CR><LF>+CEER: 8,17<CR><LF>
<CR><LF>OK<CR><LF>
v hexa kódu:
41 54 2B 43 45 45 52 0D
0D 0A 2B 43 45 45 52 3A 20 38 2C 31 37 0D 0A 0D 0A 4F 4B 0D
0A

- MT nebyl zvednut:

AT+CEER<CR>
<CR><LF>+CEER: 8,19<CR><LF>
<CR><LF>OK<CR><LF>
v hexa kódu:
41 54 2B 43 45 45 52 0D
0D 0A 2B 43 45 45 52 3A 20 38 2C 31 39 0D 0A 0D 0A 4F 4B 0D
0A

- MT byl vypnutý resp. nebyl signál:

AT+CEER<CR>
<CR><LF>+CEER: 8,16<CR><LF>
<CR><LF>OK<CR><LF>
v hexa kódu:
41 54 2B 43 45 45 52 0D
0D 0A 2B 43 45 45 52 3A 20 38 2C 31 36 0D 0A 0D 0A 4F 4B 0D
0A

- Mezi MO a MT probíhá volání (je navázáno spojení):

AT+CLCC<CR>
<CR><LF>OK<CR><LF>
v hexa kódu:
41 54 2B 43 4C 43 43 0D
0D 0A 4F 4B 0D 0A

4 Výběr vhodného jednočipu

Lze vybírat ze tří nejznámějších rodin mikročipů. Jsou to mikročipy PIC firmy Microchip. Dále mikropočítače Motorola - rodina HC12 a AVR od firmy Atmel.

Jednočipových mikropočítačů lze rozdělit podle typu přístupu ke konstrukci paměti. Řada rodiny PIC mikročipů má tzv. Harvardovu architekturu, s oddělenými sběrnicemi pro program a data. Naproti tomu mikropočítač HC12 má architekturu von Neumann, se společnou pamětí pro program i data. Nespornou výhodou mikropočítače HC12 je podpora BDM (Background Debug Mode). Protože jsem byl zvyklý na přátelské uživatelské IDE (Integrated Development Environment) od firmy Borland, programovacího jazyku C++, rozhodl jsem se pro firmu Microchip, která dodává vývojový software MPLAB.

4.1 Řady PIC16Fxx a PIC16Fxxx od firmy Microchip

Mikropočítače rodiny PIC16Cxx a PIC16Cxxx jsou vyrobeny technologií CMOS a jsou založeny na architektuře RISC (Redundance Instruction Set). Jsou to 8-mi bitové jednočipy. Programátor má k dispozici sadu 35 instrukcí.

- **PIC16F83** - Disponuje 512 slovy FLASH paměti určené pro program, 36 byte RAM paměti, 64 byte pro EEPROM a maximální frekvencí 10MHz.
- **PIC16F84** - Novější verze mikročipu PIC16C84. Vyšší maximální taktovací kmitočet, nižší spotřeba proudu, větší rozsah napájecího napětí. Novější typ paměti EEPROM. Drobné odlišnosti v konfiguračním slově – převrácená polarita bitu PWRTE. Velikost datové paměti je 68 byte.
- **PIC16F84A** - Rychlejší verze procesoru, drobná vylepšení elektrických vlastností oproti jeho předchůdci.
- **PIC16F627** - Patice DIP 18 pinů. Rozmístění pinů je kompatibilní jako u svého předchůdce. Více datových pinů - 15. Typ programové paměti FLASH s kapacitou 1024 slov. Větší programová paměť RAM s kapacitou 224 byte rozdělená do více bank. Větší paměť EEPROM, 128 bytů, více časovačů - 3, 2 analogové komparátory, programovatelné referenční napětí, modul CCP pro měření a generování časových impulsů, sériový port USART/SCI, nízkonapětíové programování, jiné adresy registrů pro práci s EEPROM, jiný formát konfiguračního slova.
- **PIC16F628** - Jako PIC16F627, větší paměť programu (2048 slov), podrobněji v kapitole 5.
- **PIC16F873** - Větší pouzdro s 28 piny, z nichž 24 je datových. Paměť programu má velikost 4096 slov, větší paměť RAM 192 byte, větší paměť EEPROM 128 byte. Disponuje třemi časovači. Má navíc modul CCP pro měření a generování časových impulsů, sériový port USART/SCI, sériový port MSSP (SPI, I2C), 5 10-bitových

A/D převodníků, přístup procesoru k programové paměti Flash, nízkonapěťové programování, jiné adresy registrů pro práci s EEPROM, jiný formát konfiguračního slova.

- **PIC16F874** – Kompatibilní s PIC16F873, více portů (33 datových pinů), 8 10-bitových A/D převodníků, paralelní komunikace PSP.
- **PIC16F876** - Zde je rozdíl jen ve větší kapacitě paměti programu (8192 byte), RAM (368 byte), EEPROM (256 byte).
- **PIC16F877** - Jako PIC16F873, více portů (33 pinů datových pinů), 8 10-bitových A/D převodníků, paralelní komunikace PSP, větší paměť programu (8192 byte), větší paměť RAM (368 byte), větší paměť EEPROM (256 byte).

Konfigurační slovo slouží k nastavení různých režimů procesoru. Jako příklad uvádím konfigurační slovo mikročipu PIC16F628 na obr. 4.1. Nastavuje se během nahrávání programu do paměti. Je zapsáno na adresu 2007h. Příkaz CONFIG se uvádí na začátku zdrojového kódu a slouží k definici parametrů konfiguračního slova.

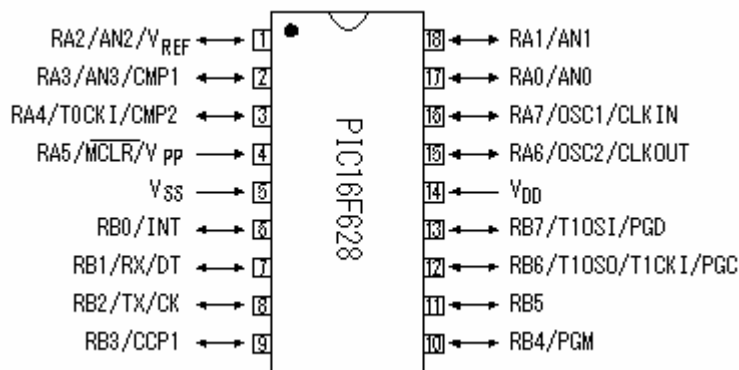
13	12	11	10	9	8	7	6	5	4	3	2	1	0
CP1	CP0	CP1	CP0	-	CPD	LVP	BODEN	MCLRE	FOSC2	PWRT	WDTE	FOSC1	FOSC0

Obr. 4.1: Konfigurační slovo mikropočítače PIC16F628

CP1,CP0 :	bity ochrany kódu
CPD :	bit ochrany dat
LVP :	povolení nízkého napětí při nahrávání kódu
BOD Reset:	povolení BOD Reset
MCLRE :	výběr pinu RA5/MCLR
PWRT :	pin pro povolení Power-up časovače
WDTE :	povolení WATCHDOGu
FOSC2, FOSC1, FOSC0 :	výběr typu oscilátoru

5 Architektura mikročipu PIC16F628 firmy Microchip

PIC16F628 má stejný počet pinů jako PIC16F84A. Paměť ROM je rozdělena do více bank. Soubor instrukcí je totožný, má více datových pinů, větší programovou paměť. Více je vidět v tabulce porovnání základních vlastností tab. 5.1.



Obr. 5.1: Pouzdro DIP s 18 piny mikročipu PIC16F628

Zkratky jednotlivých pinů:

RA0-7 : Input/Output port A RB0-7 : Input/Output port B AN0-3 : Analog input port RX : USART Asynchronous Receive TX : USART Asynchronous Transmit DT : Synchronous Data CK : Synchronous Clock CCP1 : Capture In/Compare Out/PWM Out OSC1/CLKIN : Oscillator In/External Clock In OSC2/CLKOUT : Oscillator Out/Clock Out MCLR : Master Clear (Active low Reset)	Vpp : Programming voltage input T0CKI : Clock input to Timer0 T1OSO : Timer1 oscillator output T1OSI :Timer1 oscillator input PGD : Serial programming data PGC : Serial programming clock PGM : Low voltage programinng input INT : External interrupt VDD : Positive supply for logic and I/O pins Vss : Ground reference for logic and I/O pins
--	---

Porovnání vlastností mikropočipů v řadě PIC16

	PIC16F84A	PIC16F628A	PIC16F873
Paměť rprogramu	1024 slov	2048 slov	4096 slov
Speciální registry	68 bytů od 0Ch	224 bytů od 20h	192 bytů od 20h
I/O port	13 pinů RA : 5 RB : 8	16 pinů RA : 8 RB : 8 RA5 pouze výstupní	22 pinů RA : 6 RB : 8 RC : 8
Časovač	1	3	3
CCP	-	1	2
Komparátor	-	2	-
A/D	-	-	1
USART	-	1	1
SSP	-	-	1
vestavěný OSC	-	4MHz	-

Tab.5.1: Porovnání vlastností mikropočipů v řadě PIC16, nejbližší typu PIC16F628A

5.1 Speciální registry

Paměť RAM je rozdělena do čtyř bank. Každá má 128byťů. 32 bytů je vyhrazeno speciálním registrům. Počínaje 33. bytem lze volně zapisovat do registrů u každé banky. Podrobněji je vše vidět na obr.5.2. Žlutě jsou zobrazeny registry společné všem bankám. Šedou barvou jsou vyznačeny blíže nedefinované registry. Světle modře jsou zobrazeny společné registry pro banky 0 a 2. A růžově společné registry pro banky 1 a 3.

Bank0		Bank1		Bank2		Bank3	
INDF	00h	←	80h	←	100h	←	180h
TMR0	01h	OPTION REG	81h	←Bank0	101h	←Bank1	181h
PCL	02h	←	82h	←	102h	←	182h
STATUS	03h	←	83h	←	103h	←	183h
FSR	04h	←	84h	←	104h	←	184h
PORTA	05h	TRISA	85h	00h	105h	00h	185h
PORTB	06h	TRISB	86h	←Bank0	106h	←Bank1	186h
00h	07h	00h	87h	00h	107h	00h	187h
00h	08h	00h	88h	00h	108h	00h	188h
00h	09h	00h	89h	00h	109h	00h	189h
PCLATH	0Ah	←	8Ah	←	10Ah	←	18Ah
INTCON	0Bh	←	8Bh	←	10Bh	←	18Bh
PIR1	0Ch	PIE1	8Ch	00h	10Ch	00h	18Ch
00h	0Dh	00h	8Dh	00h	10Dh	00h	18Dh
TMR1L	0Eh	PCON	8Eh	00h	10Eh	00h	18Eh
TMR1H	0Fh	00h	8Fh	00h	10Fh	00h	18Fh
T1CON	10h	00h	90h	00h	110h	00h	110h
TMR2	11h	00h	91h		111h		111h
T2CON	12h	PR2	92h		112h		112h
00h	13h	00h	93h		113h		113h
00h	14h	00h	94h		114h		114h
CCPR1L	15h	00h	95h		115h		115h
CCPR1H	16h	00h	96h		116h		116h
CCP1CON	17h	00h	97h		117h		117h
RСТА	18h	TXSTA	98h		118h		118h
TXREG	19h	SPBRG	99h		119h		119h
RСREG	1Ah	EEDATA	9Ah	GPR (48 Bytes)	11Ah	00h	11Ah
00h	1Bh	EEADR	9Bh		11Bh		11Bh
00h	1Ch	EECON1	9Ch		11Ch		11Ch
00h	1Dh	EECON2	9Dh		11Dh		11Dh
00h	1Eh	00h	9Eh		11Eh		11Eh
CMCON	1Fh	VRCON	9Fh		11Fh		11Fh
General Purpose Register (80 Bytes)	20h	General Purpose Register (80 Bytes)	A0h		120h		120h
	6Fh		EFh	00h	14Fh		14Fh
(16 Bytes)	70h	←	F0h	←	150h	←	150h
	7Fh		FFh		16Fh		16Fh
					170h		170h
					17Fh		17Fh
							1EFh
							1F0h
							1FFh

Common registers on all banks

Common registers on bank 0 and bank 2

Common registers on bank 1 and bank 3

Unimplemented

Peculiar registers to each bank

Obr. 5.2: Struktura paměti RAM mikročipu PIC16F628

6 Sériové rozhraní

Komunikace mikrokontroléru s jinými systémy (jinými mikrokontroléry, počítači, inteligentními obvody apod.) je realizována řízeným tokem dat. Pro omezení počtu použitých I/O vývodů (pinů) je nejvíce využívána sériová komunikace.

6.1 Asynchronní SCI (Serial Communication Interface)

Základní tvar při asynchronním přenosu dat začíná startovací sekvencí. Po něm následuje osm nebo sedm datových bitů, vysílaných od LSB po MSB. Jako poslední je vysílán stop bit. Tento stop bit musí být dlouhý nejméně jako datový bit. Přenos dalšího slova je opět zahájen start bitem. O asynchronní přenos jde, protože stop bit má definovanou pouze minimální délku tak, že může být i mnohem delší. Přenos každého slova je synchronizován a tedy i zahájen startovací sekvencí (start bitem).

Délka start bitu a datového bitu se určí z požadované rychlosti následovně:

Doba trvání bitu (s) = 1 / Baud rate

Baud je jednotka pro měření rychlosti přenosu dat mezi koncovými zařízeními. Baud Rate udává počet změn signálu za sekundu. Počet změn se pak vyjadřuje v baudech. Nesmíme zaměňovat s Bps (bit per second = bit za sekundu) s pojmem baud.

6.2 USART(Universal Synchronous Asynchronous Receiver Transmitter)

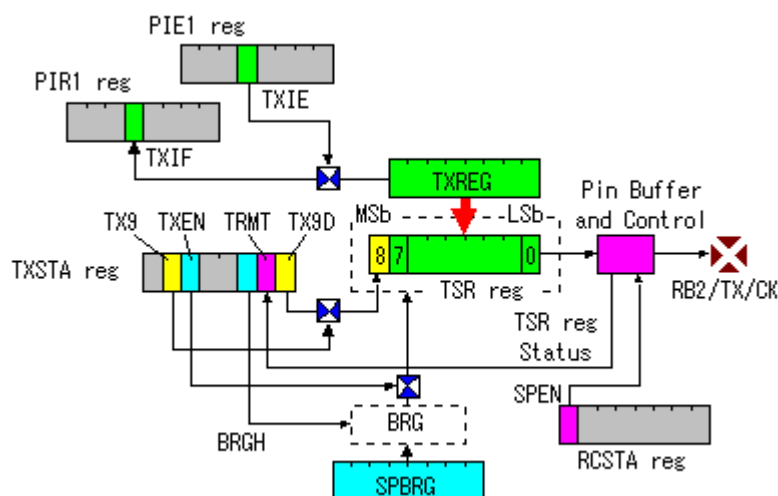
USART se používá v těchto základních režimech:

- asynchronní (v duplexním modu)
- synchronní – master (poloduplexní)
- synchronní – slave (poloduplexní)

Při komunikaci s počítačem PC nebo obvody USART může být použit formát s 5-8 datovými bity, s paritním bitem (sudá nebo lichá parita – data s paritním bitem musí mít sudý nebo lichý počet jedniček) a 1; 1,5 nebo 2 stop bity. U multi-procesorové komunikace se používá 9 datových bitů.

6.3 Asynchronní komunikace jednočipu PIC16F628

6.3.1 Vysílání dat



Obr 6.1: Blokové schéma TX registru.

Při asynchronním přenosu dat pomocí USART musí být nastaven bit SPEN registru RCSTA na 1. To způsobí, že pin RB2 pracuje jako TX. Když se nahrávají data do TXREG, přesunou se bity do TSR (Transmit Shift Register) vše je řízeno hardwarově a je posláno z TX portu. Přesun dat je proveden nastavením TXEN bitu TXSTA registru na 1, LSb (Least Significant bit) je poslán jako první.

Když je obsah TXREG poslán do TRS, TXIF bit registru PIR přejde na logickou 1 a provede se přerušení. Ale bit TX1E registru PIE1 musí být v logické jedna. Toto přerušení znamená, že obsah TXREG byl poslán do TSR a je možné převzít data z TXREG. Data mohou být průběžně posílána, jestliže dojde k přerušení a data jsou v TXREG. TXIF bit nemůže být nastavován softwarově. TXIF je smazán (nastaven do log 0), jestliže jsou data v TXREG.

TRMT bit TXSRA registru je nastaven na 1, jakmile jsou data TSR registru poslána. Tento bit nepodporuje přerušení. Proto k ověření, zda je TSR registr prázdný, je třeba TRM bit ověřovat opakovaně. Protože TSR registr není umístěn v operační paměti, nemůžete číst a zapisovat obsah tohoto registru přímo. Rychlost přenosu dat je řízena BRG (Baud Rate Generator).

Parita není podporována hardwarem, ale je podporována softwarově (je brána jako devátý datový bit). V tomto případě je devátý bit jako parita přenášena nastavením parity bitu obsazeným v TX9D TXSTA registru.

6.3.2 Kroky k nastavení asynchronního přenosu:

Inicializace SPBRG registru pro vhodnou přenosovou rychlost. Jestliže je potřeba vysoká přenosová rychlost, nastavuje se BRGH na logickou jedna.

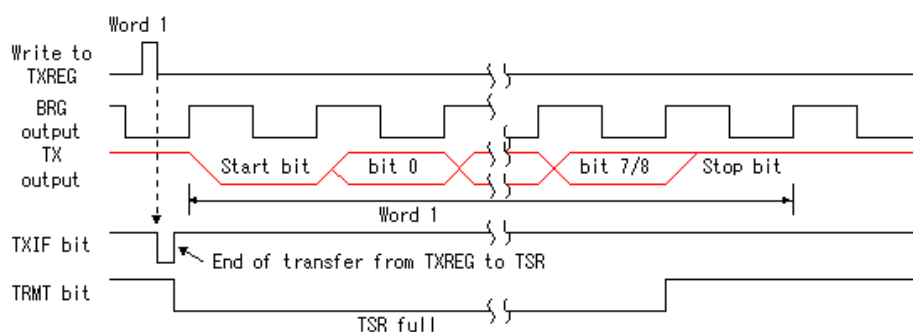
Nastavení asynchronního seriového portu se provádí nastavením bitu SYNC registru TXTA a nastavením bitu SPEN registru RCSTA do logické jedna.

V případě potřeby přerušení nastavte TXIE bit na 1 registru PIE1.

V případě potřeby devátého datového bitu je třeba nastavit TX9 registru TXSTA do logické jedna.

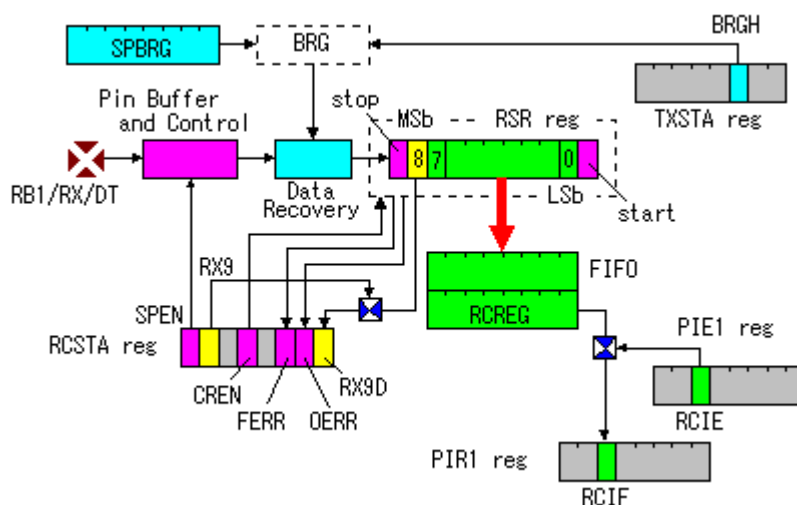
Povolení přenosu se provádí nastavením TXEN bitu TXSTA registru do logické jedna. V tomto bodě, TXREG registr a TXIF bit PIR1 registru nastaven na logickou 1 jsou podmínkami, aby bylo možné uskutečnit přenos dat.

Nyní může začít přenos dat zapsáním do TXREG.



Obr. 6.2: Komunikace po seriové lince TX registru

6.3.3 Příjem dat



Obr 6.3: Blokové schéma RX registru.

K nastavení asynchronního přenosu dat je nutné, aby bit SPEN registru RCSTA byl nastaven v logické 1. To způsobí, že RB1 se chová jako RX.

Data jsou přijímána do RSR registru pomocí rotačních logických obvodů. Data v RX jsou třikrát vzorkována a posuzována podle High a Low úrovní. Přenosovou rychlostí definovanou v SPRG registru a BRHG bitu v TXSTA registru. Jakmile je detekován stop bit, je obsah RSR registru přesunut do RCREG. RCIF bit PIR1 registru je nastaven a provede se přerušení. V případě že chceme vyvolat přerušení, je potřeba nastavit bit RCIE v PIE1 registru. RCREG je založen na principu FIFO paměti (first in first out). Pojme 2x 8bitů. Má funkci vyrovnání zpoždění při softwarovém zpracování a příjmu dat. RCIF je hardwarově nastaven v případě, že je celý RCREG přečten. V případě naplnění RCREG a dokončení přenosu z RSR registru dojde k nastavení OERR bitu v RCSTA registru a nastane tzv. opakující se chyba. Data, která byla načtena do RSR registru, jsou teď ztracena. Příjem dat je tedy ukončen. K obnovení příjmu je nutné smazat příznak bitu CREN registru RCSTA a znova nastavit. Nyní je příznak chyby OERR smazán (nastaven na log 0).

6.3.4 Postup při asynchronním příjmu dat

Nastavení rychlosti v Bd pomocí SPBRG registru. V případě potřeby vysoké rychlosti se nastavuje BRGH bit registru TXSTA.

Povolení asynchronního portu smazáním bitu SYNC registru TXSTA a nastavením bitu SPEN registru RCSTA do log. jedničky.

Nastavením bitu RCIE registru PIE1 povolíme přerušení.

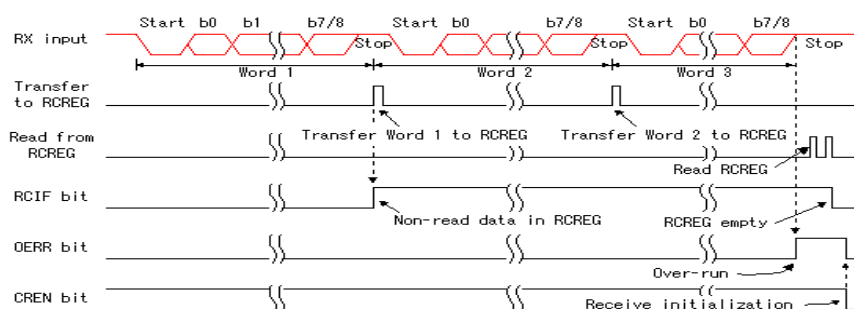
V případě příjmu 9-tého datového bitu nastavíme RX9 bit registru RCSTA.

Povolení příjmu provedeme nastavením bitu CREN registru RCSTA do logické hodnoty „1“.

V případě příjmu dat se nastaví příznak bitu RCIF registru PIR1. Vyvolá se přerušení v případě, že jsme jej vyžádali nastavením RCIE příznaku.

Přečteme 8 bitů z registru RCREG.

Pokud nastane chyba, smažeme chybu smazáním a obnovením bitu CREN registru RCSTA na log 1.



Obr. 6.4: Komunikace po seriové lince RX registru.

7 Vývojový diagram

7.1 Popis částí programu

Program začne nastavením systémových registrů a čeká tak dlouho, dokud se nepřipojí MS. Připojení je signalizováno rozsvícením LED (dioda D2 viz příloha A). Příkazem AT+CLIP se testuje zadání čtyřmístného PINu mobilního telefonu. Nebyl-li PIN zadán, pošle se čtyřmístné číslo příkazem AT+CLIP=xxxx. Program přechází do deaktivovaného režimu, LED nesvítí v krátkých intervalech, bliká. Ve Smyčce I se testuje připojení mobilu vysláním AT a zda nepřišla SMS jednou za 1s.

Po přepnutí tajného vypínače do polohy ON přejde za čtyři minuty program do aktivního stavu. Řidič má tedy dostatek času, aby vystoupil z vozu. Program se zacyklí ve Smyčce II, kde se testuje připojení mobilu a příjem nové SMS. Zde je možné vyřadit startér či zapalování zasláním SMS ve správném tvaru. Zde má program funkci imobilizéru. Odpojení startéru resp. zapalování se provede vypnutím relé 1 resp. relé 2. V případě otevření dveří resp. indikaci polohového čidla, přejde alarm po třiceti sekundách do části Alarm1. Třicet sekund je pro řidiče, který po vstupu program vyřadí z aktivovaného stavu přepnutím tajného vypínače, do stavu OFF. Pokud se tak nestane spustí se Alarm1.

V této části se zavolá aktuální majitel. Program rozpozná čtyři etapy volání. Volbu telefonního čísla, prozváněcí tón, hovor a ukončení navázání cesty mezi MO a MT. Majitel vozu může tak slyšet, co se děje v jeho automobilu, pokud přijme hovor od alarmu. Program v tomto případě přejde do aktivního stavu. Může zde tedy vyřadit auto z provozu zasláním SMS. Může nastat situace, kdy majitel není v dosahu svého mobilu nebo je v metru mimo signál. V těchto případech je změněn aktuální majitel na dalšího volaného v pořadí a je znovu voláno. Alarm cyklicky volá na první tři telefoní čísla na SIM kartě. Na začátku programu se testuje není-li nastaven pin RA4 na log. 1. Je-li nastaven a program přejde do stavu alarmu, spustí se podprogram Alarm2. Čtvrtý pin portu RA se nastaví pomocí JP4 (viz příloha A). Alarm2 slouží k určení aktuální polohy vozu, v případě krádeže. Každé čtyři minuty je posílána SMS majiteli vozu s informací o aktuálním času a místě.

7.2 Další možná rozšíření softwarové části

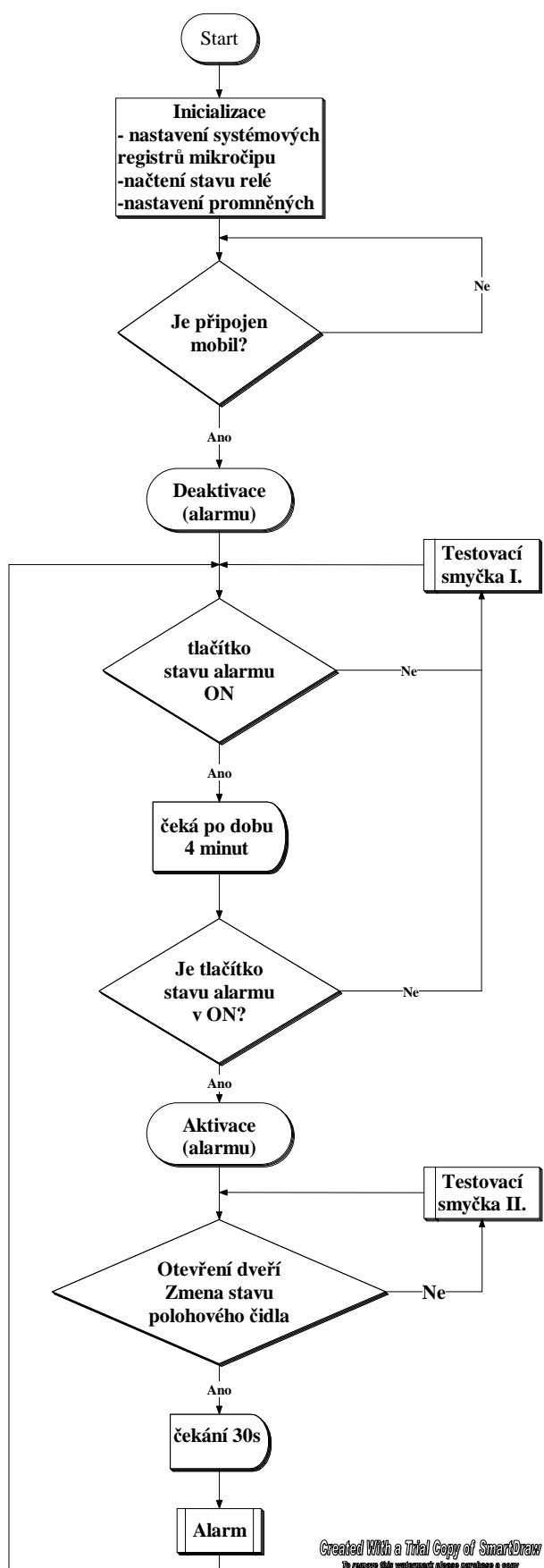
- Zadávaní čtyřmístného PINu do telefonu lze zakázat AT příkazem.
- Část programu Smyčka I je možné rozšířit o nastavení počtu majitelů zasláním SMS. Tzn. v deaktivovaném stavu se budou nastavovat uživatelská data a v aktivovaném stavu se bude nastavovat stav relé (zastavení kradeného vozu).

- Pro více variant provedení alarmu je možné přepínat mezi algoritmi zasláním kódované SMS (v části programu deaktivace, při zacyklení ve Smyčce I).
- Aktuálního majitele lze změnit prozvoněním alarmu v deaktivovaném stavu.

Operátor T-mobile spustil v březnu tohoto roku službu „Kde je..“. Tato služba umožňuje zasláním SMS ze SIM karty Universal, zjistit polohu MS s přesností danou hustotou pokrytí dané lokality, BTS. Nabízejí se dvě varianty použití této služby. První z nich je vyžádání polohy pomocí AT příkazu CMSS s parametrem, v kterém je číslo pozice na SMS kartě telefonu. Druhou možností je vyžádání polohy zasláním SMS majitelem znajícím LPIN.

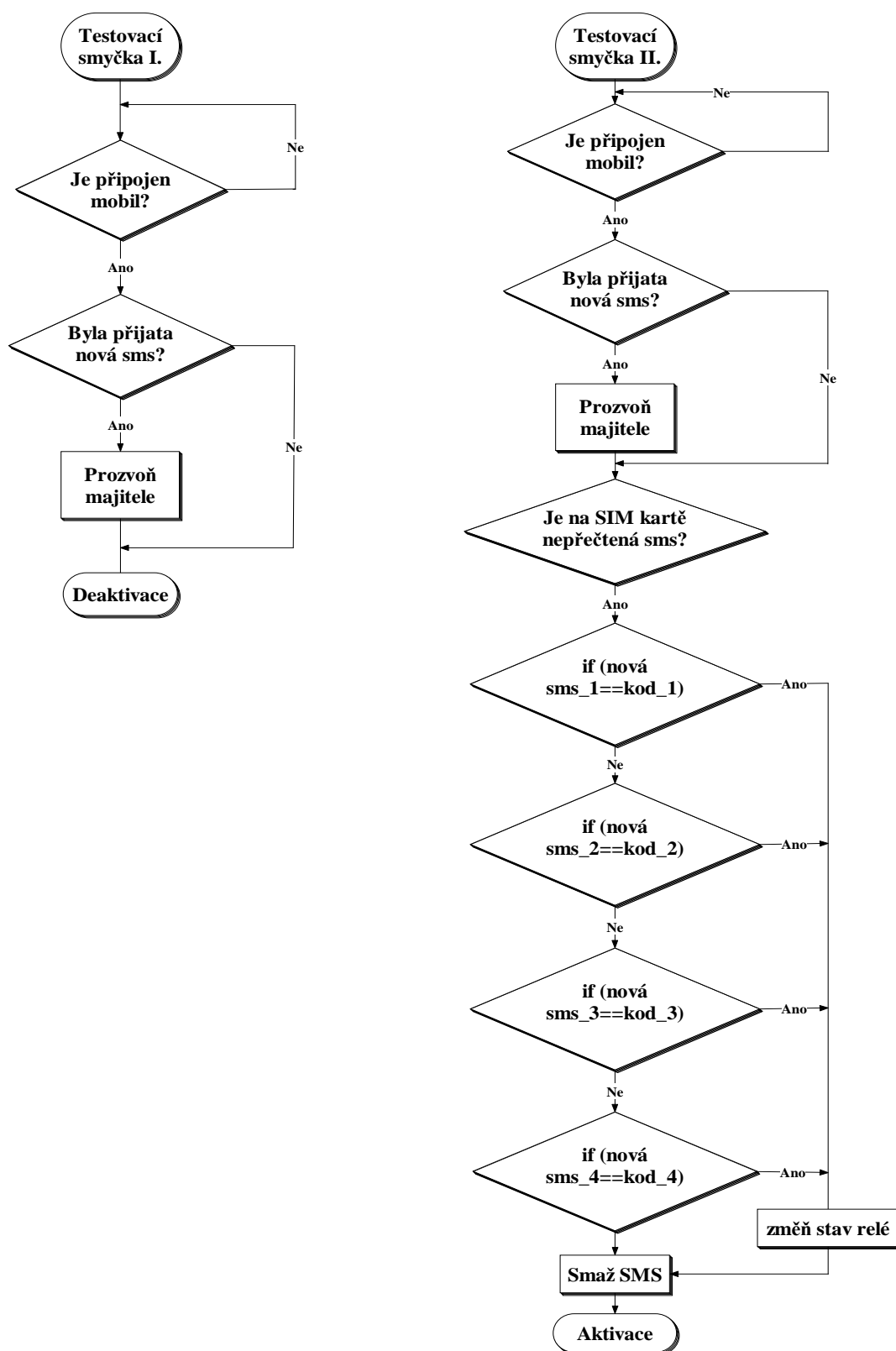
Pro softwarovou variabilitu dle přání koncového zákazníka, je možné využít vlastností novější řady PIC16F877. Stejně tak by bylo přehlednější, využít předností programovacího jazyka C resp. C++. Existuje řada překladačů překládajících prog. jazyk C do strojového formátu HEX. Source Boost IDE je vývojové prostředí, které podporuje překladače: BoostC, C2C, C2C++ a P2C. Překladače lze integrovat do vývojového prostředí MPLAB. Pouze BoostC je však uvolněn zdarma. Omezení této demo licence je v optimalizaci překladu zdrojového kódu. U ostatních překladačů je třeba zakoupit licenci. Web: <http://www.picant.com/c2c/c.html>

7.3 Hlavní část programu



Obr. 7.1: Vývojový diagram programu Autoalarm

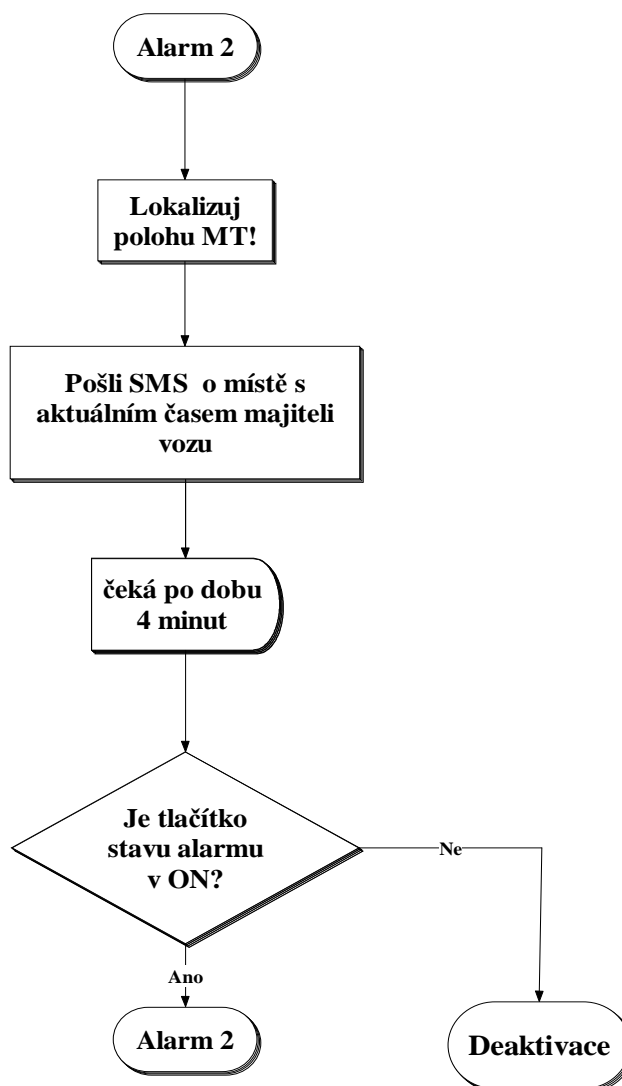
7.4 Podprogramy vývojového diagramu – Smyčka I.,II.



Obr. 7.2: Vývojový diagram programu Autoalarm – Smyčka I., II.



7.6 Podprogramy vývojového diagramu – Alarm II.



Created With a Trial Copy of SmartDraw
To remove this watermark please purchase a copy

Obr. 7.4: Vývojový diagram programu Autoalarm – Alarm II.

8 Algoritmy

8.1 Definování proměnných registrů a příznaků

pocet	equ	20H	;registr používaný na počet cyklů při blikání diodou
cislo	equ	21H	;do tohoto registru se ukládá kolikáté číslo na SIM
			;kartě
flags	equ	22H	;zde se ukládá stav relé.Tento registr se poté zapíše
			;do EEPROM
flags1	equ	23H	
obeh	equ	24H	
tmp	equ	25H	;pomocné registry
tmp_1	equ	26H	
casuj0	equ	27H	
casuj1	equ	28H	
casuj2	equ	29H	
pamet_1	equ	30H	
pamet_2	equ	31H	
pamet_3	equ	32H	
pamet_4	equ	33H	
pamet_5	equ	34H	
pamet_6	equ	35H	
pamet_7	equ	36H	
pamet_8	equ	37H	
pamet_9	equ	38H	
pamet_10	equ	39H	
pamet_11	equ	3AH	
pamet_12	equ	3BH	
pamet_13	equ	4CH	
pamet_14	equ	4DH	
pamet_15	equ	4EH	
pamet_16	equ	4FH	
pamet_17	equ	50H	
pamet_18	equ	51H	
pamet_19	equ	52H	
pamet_20	equ	53H	
pamet_21	equ	54H	
pamet_22	equ	55H	
pamet_23	equ	56H	
pamet_24	equ	57H	
pamet_25	equ	58H	
pamet_26	equ	59H	
pamet_27	equ	5AH	
pamet_28	equ	5BH	
pamet_29	equ	5CH	
pamet_30	equ	5DH	
pamet_31	equ	5EH	
pamet_32	equ	5FH	
TMPW	equ	3CH	;pro dočasné uložení STATUS a W registru v
			;případě, že nastane přerušení
TMPS	equ	3DH	
flags2	equ	3EH	
pocet_4min	equ	3FH	;počet cyklů
stav	equ	40H	;registr pro uložení příznaku stavu, ve kterém se
			;nachází program. Deaktivace resp. Aktivace
eeeprom_adr	equ	41H	
eeeprom_byte	equ	42H	
flags3	equ	43H	;pro příznky obdržení zpravy z mobilu
clcc	equ	44H	;zde se nastaví příznky podle typu odpovědi na AT
			;příkaz AT+CLCC
znak	equ	45H	
znak_ak	equ	46H	;registr pro uložení znaku při přerušení

chyba	equ	47H	;pro nastavení příznaku chyby, OERR nebo FERR
pin	equ	48H	;pro příznak zadání PIN do mobilu
ceer	equ	49H	;zde se nastaví příznaky podle typu odpovědi na AT ;příkaz AT+CEER
cislo_ak	equ	4AH	
pom	equ	4BH	
pom2	equ	60H	
pom1	equ	61H	
;			
deaktiv	priznaky stavu equ	00H	;pokud je tajný spínač v poloze OFF nastaví se do ;log 1
aktivni	equ	01H	;pokud je tajný spínač v poloze ON nastaví se do log ;1
;			
poplach_volan	equ	03H	;příznak, že už byl poplach zavolán na MT
;			
spojeni_ok	priznaky flags1 equ	01H	;tento příznak se nastaví do log 1 pokud MS odpoví ;na příkaz AT odpovědí OK
aktivace_ok	equ	02H	;tento příznak se nastaví do log 1 pokud MS odpoví ;na příkaz AT+CPIN=7159 OK
konec_prikazu	equ	04H	;tento příznak se nastaví do log 1 pokud MS odpoví ;na AT příkaz OK nebo ERROR
sms_ok	equ	07H	;příznak platného přijetí SMS
;			
bylo_volano	priznaky flags3 equ	02H	;nastaví se do log 1 pokud bylo voláno a hovor byl ;přijat
busy	equ	03H	;nastaví se pokud MS poslala zprávu BUSY
nocarr	equ	04H	;nastaví se pokud MS poslala zprávu NO CARRIER
;			
re1	priznaky flags equ	0H	;podle tohoto bitu je rele K1 buď zapnuto nebo ;vypnuto
re2	equ	01H	;podle tohoto bitu je rele K2 buď zapnuto nebo ;vypnuto
;			
priznaky prikazu AT+CLCC			
clcc_prozvaneci	equ	01H	;MO(mobile oriented) prozvání MT(mobile ;terminated)
clcc_hovorovy	equ	02H	;MT přijal hovor od MO
clcc_volba	equ	03H	;MO posílá tel číslo na ústřednu
clcc_konec_spojeni	equ	04H	;hned po příkazu AT+CLCC následuje OK, tzn.
;			
jakykoliv	priznak prisel znak equ	00H	
;			
oerr	priznaky promenne chyba equ	00H	
ferr	equ	01H	
;			
pin_zadan	priznak zadani pinu equ	00H	
;			
ceer_spojeno	priznaky prikazu at+ceer equ	00H	;navázána cesta mezi mobily +CEER: 0,0
ceer_vypnuto	equ	01H	;mobil uživatele je vypnutý +CEER: 8,16
ceer_zaveseno	equ	02H	;uživatel zavěsil +CEER: 8,17
ceer_obsazeno	equ	03H	;linka byla obsazena pouze u pevné linky v případě ;odpověď BUSY tj. ceer_zaveseno +CEER: 8,27
mobilu je poslána			


```

ceer_nezvedl    equ 04H                                ; v případě ze majitel telefon neslysel cili nezvedl
                                                         ;+CEER: 8,19

#define         RELE1          PORTA,0
#define         RELE2          PORTA,1
#define         TEST          PORTA,2
#define         LED            PORTA,3
#define         OK1            PORTB,0    ;signalizace otevření dveří LOW=otevřeny-dioda svítí
#define         TL             PORTB,3    ;
#define         OK2            PORTB,5    ;tajný přepínač mezi aktivním a deaktivovaným
                                                         ;stavem LOW=ON-dioda svítí tzn. aktivní režim
#define         OK3            PORTB,6    ;ladění programu

```

8.2 Ukázkové podprogramy

Přerušení jsem použil hlavně pro detekci připojení a příjem odpovědí z MS. Jinak by při příjmu zpráv, které nenásledují bezprostředně za příkazem došlo k naplnění WATCHDOG. To by způsobilo restart microchipu. A znovu by se nastavily registry na počáteční hodnoty, což by změnilo funkci alarmu.

```

;int. od sériové linky pouze pro test připojení mobilu
;-----uloz data-----
        movwf    TMPW          ; uloží W
        swapf    STATUS,W      ; prohodí nibble STATUS a uloží do W
        bcf      STATUS,RP0    ; nastavení banky 0
        movwf    TMPS         ; uložit STATUS
;-----
        BCF      STATUS,RP0    ;přijme znak z mobilu
        BTFSS    RCSTA,OERR
        GOTO     $+3
        BCF      RCSTA,CREN
        BSF      RCSTA,CREN
        MOVF     RCREG,W
;*****
;      obsluha přerušení
;*****
IntRet swapf    TMPS,W          ; původní registr STATUS do W
        movwf    STATUS        ; navrácení registru STATUS
        swapf    TMPW,F        ; příprava původního registru W
        swapf    TMPW,W        ; návrat původního registru W
        retfie

```

Procedura ukládá přicházející znaky do paměťových registrů. V případě kdy jsem potřeboval zjistit, jakou odpověď pošle MS.

Uložil jsem znaky, které přišly a následně poslal zpět se zobrazením pomocí terminálu (Hyperterminal, Telix). U většiny příkazů existuje více odpovědí. Bez použití přerušení by bylo nutné napsat funkci pro každou opověď znovu. Zde nastavím jen příznaky podle hodnot uložených v registru paměť_1 až paměť_32.

```

cti_znaky
        call     null_ram      ;vymaž RAM
                                                         ;počet znaků, které chceme uložit do
                                                         ;paměťových registrů

        movlf    20H, ciso_ak
        movlf    0FFH, znak_ak

                                                         ;pomocné proměnné pro nastavení doby
                                                         ;testování jestli nepřijde nějaký znak
                                                         ;po sériové lince

        movlf    0FFH, pom
        movlf    05H, pom1
        movlf    096H, pom2

znova_1

```

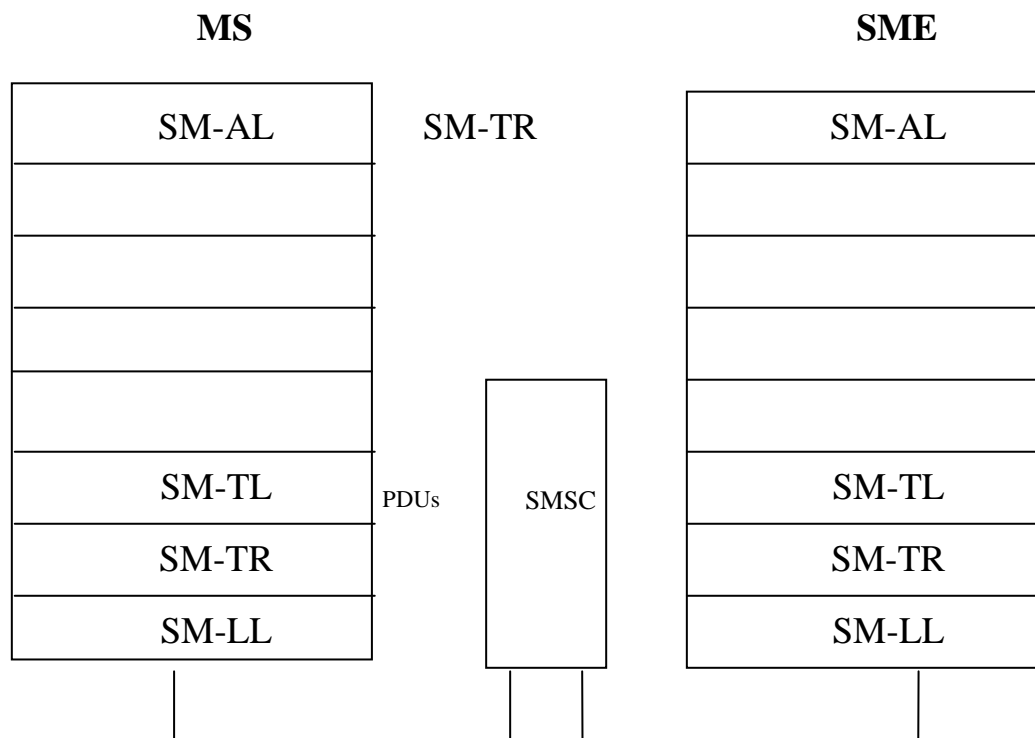
```
znova2    nop
pokr_24    nop
           ;vymazání WATCHDOGu
           clrwdt
           ;macro, které po každém zavolání zmenší obsah předaného registru o
           ;jedna
           ;jestliže je registr nulový, nastane skok na zadané návěští
           z_s_na pom, pokr_25
           movlf 0FFH, znak_ak           ;vymaž znak_ak
           ;povolí přerušení po dobu 9 instrukcí, pak jej zas zakáže
           ;v případě že nastane přerušení tj. přijde znak po sériové lince do RX.
           ;Uloží se do proměnné znak_ak
           test_znak
           ; by-li nastaven příznak jakykoliv registru znak uloží se příchozí znak
           ;v reg. znak_ak
           ; jestli ne, pokračuje se v cyklu skokem na návěští pokr_26
           btfsz znak, jakykoliv
           goto pokr_26
           ;toto macro uloží aktuální znak podle pořadí do prom. reg. Pamet_1 až
           ;pamet_32
           uloz_znak znak_ak,cislo_ak
           ;zmenší o jedna cislo_ak je-li nula skoč
           decfsz cislo_ak
           goto pokr_24
           goto pokr_25
pokr_26    goto pokr_24
pokr_25    movlf 0FFH, znak_ak
           movlf 0FFH, pom
           ;zmenší o jedna pom2 je-li nula skoč
           decfsz pom2
           goto znova2
           ;zmenší o jedna pom1 je-li nula skoč
           decfsz pom1
           goto znova_1
return
```

Ukázka alarmové části programu alarm_c2. Podprogram zašle SMS s udáním polohy a aktuálním časem. Nová SMS je zasílána každých pět minut.

```
alarm_c2
           call lokace
           movlf 0FEH, obeh
pet_min    call cekame_1
           zmen_pin LED
           decfsz obeh
           goto pet_min
           ;je-li OK1 v Low jsou dveře otevřeny .V
           ;případě, že ne, pokračuje se skokem na
           ;navěští alarm_c2
           otevreny_dvere OK1, test_stavu
goto alarm_c2
```

9 SMS PDU formát

Je soubor pravidel pro komunikaci pomocí krátkých textových zpráv. PDU (Protocol Data Unit) datagram obsahuje mimo textových zpráv, informace pro směrování, způsob kódování a dekodování, informace o času.



Obr. 9.1: Komunikační vrstvy SMSC

Přenos dat probíhá na nejvyšší aplikační vrstvě SM-AL (Short Message Application Layer) a je řízená přenosovou vrstvou SM-TL (Short Message Transport Unit). Pro komunikaci s uživatelem (MMI) jsou definovány AT+C příkazy. Je to rozšíření sady příkazu dle doporučení GSM-07.07. Pro realizaci komunikace mezi dvěma přístroji lze použít terminálu (např. TeliX či Hyperteminál).

Rozdělení podle druhu komunikace pomocí SM-TL přenosové cesty:

PDU typ	Směr	Funkce
SMS-DELIVER	SMSC => mobilní telefon	doručí krátkou zprávu
SMS-DELIVER REPORT	Mobilní telefon => SMSC	doručí příčinu selhání (pokud je to nutné)
SMS-SUBMIT	Mobilní telefon => SMSC	doručí krátkou zprávu
SMS-SUBMIT REPORT	SMSC => mobilní telefon	doručí příčinu selhání (pokud je to nutné)
SMS-STATUS REPORT	SMSC => mobilní telefon	doručí hlášení o stavu
SMS-COMMAND	Mobilní telefon => SMSC	doručí příkaz

Podrobně se budeme zabývat SMS-DELIVER a SMS-SUBMIT datagramem. Popisující způsob doručení uživatelských dat zprávy a její přidružené informace mezi zařízeními (MS a SMSC).

9.1 Příjem zprávy SMS v PDU formátu

Středisko zpráv **SCA** (Service Center Address) - Délka telefonního čísla SCA tzn. počet oktetů nutných pro číslo střediska zpráv plus 1 oktet na typ čísla.

Délka	Typ čísla		BCD číslice
	1.	2.	0-8 oktetů
	oktet	oktet	
Příklad:	07	91	246030500200

BCD kód ... 2 číslice představují jeden oktet, přičemž pořadí číslic je v každém oktetu změněn. Pokud je počet číslic ve středisku zpráv lichý, posledním znakem bude znak 'F'.

Type adress z SCA. Mezinárodní formát telefonního čísla (např. +420 603 052000) je 91 a národní formát(např. 603 052000) je 81.

Telefonní číslo SCA. Telefoní číslo je v BCD kódu. Pokud je telefonní číslo tvořeno lichým počtem číslic, pak se poslední číslice doplní znakem "F". Po převodu bude SCA T-mobilu „+420 603 052000“a Oskara „+420608005681“

PDU type (SMS-DELIVER) TP-MTI, TP-MMS, TP-SRI, TP-UDHI a TP-RP

MTI (Message Type Indicator) – hodnota určující druh zprávy obsažené v datagramu PDU.

MMS (More Messges to Send) – hodnota určující, zda je v SMSC více zpráv k zaslání. Takový případ nastává, zejména je-li MS delší dobu vypnuta .

SRI (Status Report Indication) – je-li hodnota nastavena, bude poslán report jako potvrzení příjmu, po doručení SMS. Nastavováno v SMSC.

UDHI (User Data Header Indication) – přítomnost hlavičky na začátku textové části SMS.

RP (Replay-Path) – příjemce může bezplatně odpovědět.

TP-OA (Originating Address) Délka telefonního čísla odesílatele. Je to počet čísel (ne oktetu!) telefonního čísla např. 420 732 123456 je 12 čísel (0Ch)

TP-OA Formát telefonního čísla odesílatele. Mezinárodní formát telefonního čísla je 91 a národní formát je 81.

TP-OA Telefonní číslo odesílatele " +420 732 123456 ". Oktet telefonního čísla je v opačném pořadí. Pokud je telefonní číslo tvořeno lichým počtem číslic, pak se poslední číslice doplní znakem "F".

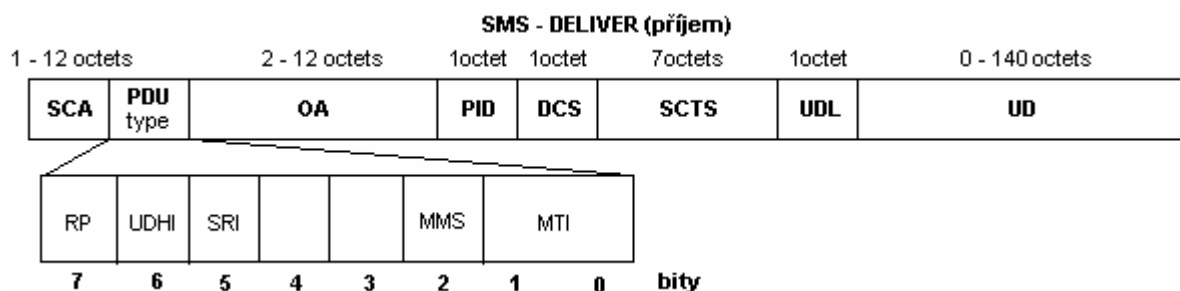
TP-PID (Protocol Identifier) Informace podle které se SMSC rozhoduje, ve kterém formátu nebo pomocí jakého protokolu má být zpráva doručena.

TP-DCS Nastavení kódování TP-UD

TP-SCTS (Service Center Time Stamp) Parametr s datem a časem doručení zprávy. Oktet časové zprávy je v opačném pořadí.

TP-UDL (User Data Length) Počet znaků zprávy (HEXa) v TP-UD před jejím zakódováním. Zpráva bude kódována podle nastavení v TP-DCS.

SCA	PDU	TP-OA	PID,DCS	SCTS, UDL
07 91 24 60 30 50 02 00	04	0C 91 24 70 23 21 43 65	00 00	30 70 92 51 61 95 80



Obr. 9.2: Uspořádání dat SMS-DELIVER formátu

9.2 Poslání zprávy SMS v PDU formátu

Délka informace SCA Pokud je délka 00 bude použito telefonní číslo SCA uložené v telefonu. Jinak je zde uvedena délka následujícího SCA.

Formát telefonního čísla SCA. Mezinárodní formát telefonního čísla je 91 a národní formát je 81.

Telefonní číslo SCA. Oktet telefonního čísla je v opačném pořadí. Pokud je telefonní číslo tvořeno lichým počtem číslic, pak se poslední číslice doplní znakem "F". Před převodem bude SCA T-mobilu " +420 603 052000 ".

PDU type (SMS-SUBMIT) TP-MTI, TP-RD, TP-VPF, TP-SRR, TP-UDHI a TP-RP

RD(Riject Duplicate) – říká, aby SMSC nepřijala datagram se stejným MR, DA od stejného původce jako předešlá zpráva

VPF(Validity Period Format) – informuje je-li přítomna VP(Validity Period) a jejím formátu.

SRR(Status Report Request) - Status report je vyžadován.

Parametry MTI, UDHI a RP jsou zmíněny v popisu SMS-DELIVER datagramu.

TP-MR (Message Reference) Referenční číslo poslané zprávy z telefonu do SMSC (01 - 255) přiřadí telefon.

TP-DA (Destination Adress) Délka telefonního čísla příjemce. Je to počet čísel telefonního čísla např. 420 603 123456 je 12 čísel (0Ch)

TP-DA Formát telefonního čísla příjemce. Mezinárodní formát telefonního čísla je 91 a národní formát je 81.

TP-DA Telefonní číslo příjemce „+420 603 123456“. Oktet telefonního čísla je v opačném pořadí. Pokud je telefonní číslo tvořeno lichým počtem číslic, pak se poslední číslice doplní znakem ,F‘.

TP-PID (Protocol IDentifier) Informace podle které se SMSC rozhoduje, ve kterém formátu nebo pomocí jakého protokolu má být zpráva doručena.

TP-DCS (Data Coding Scheme) Informace o kodování zprávy. Bude použito 7-bitové kódování a zpráva je určena k uložení do telefonu.

Bity:	7-4	3	2-0
	Coding group	0	X X X
	7-4 bity ... 0;	3 bit...0	2-0 bit ... 0 =00H
	7-4 bity ... 1;	3 bit...0	2-1 bit...1;0 bit...0 =F6H

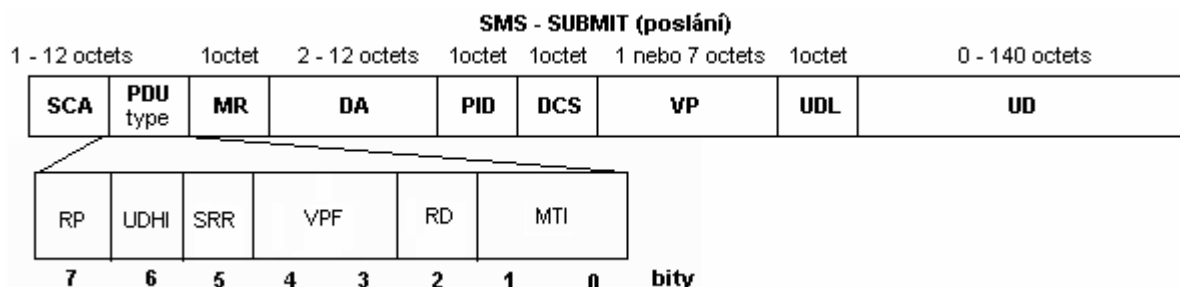
TP-VP (Validity Period) Čas platnosti SMS zprávy. „AA“ znamená 4 dny. Používá se délka 1 octet nebo 7 octet.

Bity:	0-7					
Year	Moth	Day	Hour	Minute	Second	TimeZone

TP-UDL Počet znaků zprávy (HEXa) v TP-UD před jejím zakódováním. Zpráva bude kódována podle nastavení v TP-DCS.

TP-UD Položka obsahuje zakódovanou zprávu „AUTOALARM“ s maximální délkou 140 bytů (nezakódovaná 160 znaků).

SCA	PDU	TP-DA	PID, DCS	VP	UDL, UD
00 07 91 24 60 30 50 02 00	11 00	0C 91 24 70 23 21 43 65	00 00	AA	09 C12AF5196406A54D



Obr. 9.3: Uspořádání dat SMS-SUBMIT formátu

9.3 Příklad odeslání SMS pomocí AT příkazu

00 21 00 0C 91 247023382374 0000 09 C12AF5196406A54D

kde posledních 8 znaků představuje SMS text: „AUTOALARM“

SMS je dlouhá 9 znaků, 7 bitovým kódováním dostaneme osm znaků v hexa kódu.

$(8-1)+1=8\text{byťů}$

8 bytů	1byt
AUTOALAR	M

Abychom dostali celkový počet znaků PDU datagramu, přičteme v našem případě 13 znaků. Dostaneme 21 znaků, které vložíme za '='. Dvě nuly znamenají, že se číslo střediska zpráv načítá z mobilního telefonu. Proto následující PDU č. střediska zpráv neobsahuje.

příkaz: (pozn. CTRL-Z je v hexa 1AH)

AT+CMGS=21

>0021000C91247023382374000009C12AF5196406A54D<CTRL-Z>

+CMGS: 24

OK

Na okně terminálu (Obr.9.4) je vidět výpis chybového hlášení příkazu vlivem špatně zadaného PDU.

AT+CMGW=21 - uložíme zprávu na SIM

>0021000C91247023382374000009C12AF5196406A54D<CTRL-Z>

+CMGW: 2

OK

Nastavení čísla střediska zpráv:

AT+CSCA=" +420603052000" - středisko zpráv - T-MOBILE

OK

AT+CSCA="+420608005681" - středisko zpráv - OSKAR

OK

```

19200 - Hyperterminál
Soubor Úpravy Zobrazit Zavolat Přenos Nápověda

AT+CMGS=21
> _→
+CMS ERROR: 304
AT+CMGS=21
> _01000C91247023382374000009C12AF5196406A54D→
+CMS ERROR: 304
AT+CMGS=21
> 01000C91247023382374000009C12AF5196406A54D→
+CMS ERROR: 500
AT+CMGS=21
> 0021000C91247023382374000009C12AF5196406A54D→
+CMGS: 201

OK

```

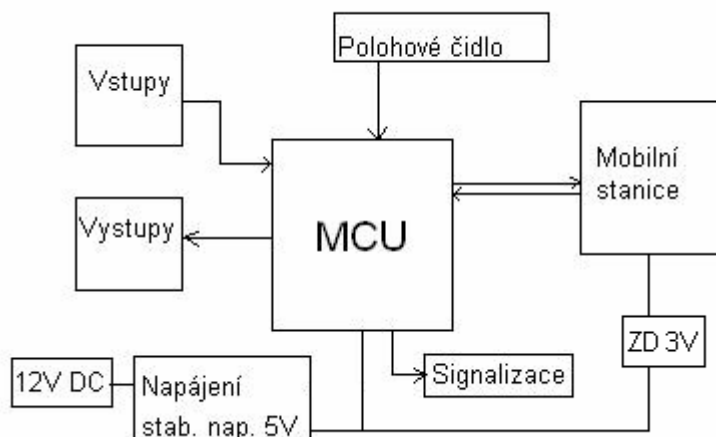
Obr. 9.4: Okno Hyperterminálu – příkaz AT-CMGS

7bitové kódování

				b7	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1
				b5	0	1	0	2	0	1	0	1
b4	b3	b2	b1		0	1	2	3	4	5	6	7
0	0	0	0	0	@		SP	0		P		p
0	0	0	1	1			!	1	A	Q	A	q
0	0	1	0	2	\$			2	B	R	B	r
0	0	1	1	3				3	C	S	C	s
0	1	0	0	4				4	D	T	D	t
0	1	0	1	5			%	5	E	U	E	u
0	1	1	0	6			&	6	F	V	F	v
0	1	1	1	7			'	7	G	W	G	w
1	0	0	0	8			(8	H	X	H	x
1	0	0	1	9)	9	I	Y	I	y
1	0	1	0	10	LF		*	:	J	Z	J	z
1	0	1	1	11			+	;	K	Ä	K	ä
1	1	0	0	12			,	<	L	Ö	L	ö
1	1	0	1	13	CR		-	=	M		M	
1	1	1	0	14		ß	.	>	N	Ü	N	ü
1	1	1	1	15			/	?	O		O	

Obr. 9.5: 7-bitové kódování

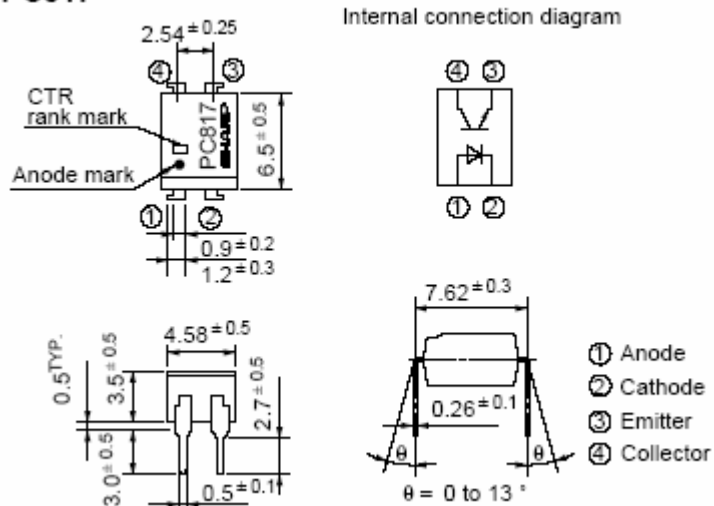
10 Blokové schéma zapojení



Obr. 10.1: Blokové schéma zapojení

Hlavní částí zabezpečovacího zařízení je mikropočítačová jednotka PIC16F628 a mobilní telefon Siemens MT50. PIC je taktován pomocí externího krystalu na frekvenci 4MHz. Řídící software reaguje na změnu stavu (napěťové úrovně), na jednotlivých vstupech. Změna stavu polohového čidla je indikována zkratem resp. odpojením pinu od napětí 5V. Výstupem je sepnutí či rozepnutí relátka. Komunikace MS s MCU probíhá na základě AT příkazů.

PC817



Obr. 10.2: Schéma optočlenu PC817

Galvanické oddělení vstupních obvodů je uskutečнено pomocí optočlenů OK1-OK4 – PC817, vysoká napěťová izolace mezi vstupem a výstupem (Viso=5000V).

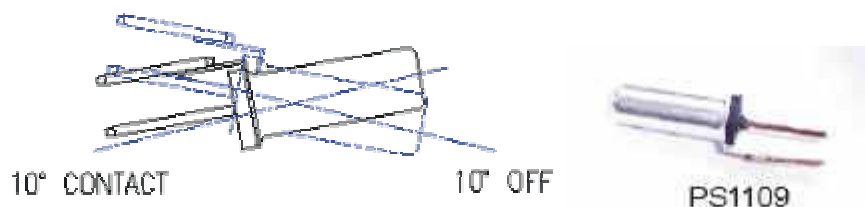
Maximální proud $I_F=50\text{mA}$. Proudová špička $I_{FM}=1\text{A}$. Napětí ve zpětném směru $U_r=6\text{V}$. Ztrátový výkon $P=70\text{mW}$. $V_{CEO}=35\text{V}$, $V_{ECO}=6\text{V}$, kolektorový ztrátový výkon $P_c=150\text{mW}$, $I_c=50\text{mA}$

Vstupní obvody jsou tvořeny vstupními odpory, LED signalizačními a optočlenem. Vstupní hodnoty odporů jsou $1\text{k}\Omega$ ($1\text{k}8$ Ohmu), což odpovídá vstupnímu nap. 5V . Pro změnu vst. napětí je třeba zvolit jiné hodnoty odporů. Tyto hodnoty jsou dány maximálním proudovým zatížením optočlenu ($I_F = 50\text{mA}$) a též proudovým omezením použité indikační LED diody. Rezistory jsou v provedení 204 – subminiaturní ($0,25\text{W}$). K připojení jednotlivých vstupů slouží svorkovnice ARK500/2.

Releové výstupy jsou tedy dokonale galvanicky odděleny od obvodu alarmu a umožňují spínat $12\text{V}/1\text{A}$. Byly voleny s ohledem na omezené místo na DSP. Pro spínání řádově 10A je třeba připojit reléový obvod (obsahující autorelé) řízený výstupem relé na DSP.

Polohové čidlo má citlivost na náklon ± 10 stupňů. Pracuje na mechanickém principu (kulička spíná kontakty). Zatížení $12\text{V} / 1\text{mA}$, odpor v sepnutém stavu $<20\text{Ohm}$, izolační odpor $> 10\text{M}\Omega$, životnost cca $10\,000$ cyklů.

Pro napájení MCU byl použit napěťový stabilizátor 78S05 s výstupem 5V .



Obr. 10.3: Polohové čidlo

11 Zabezpečovací systémy

11.1 Imobilizér

Imobilizér je pasivní zabezpečovací zařízení, které ve voze pouze pomocí sady kontaktů rozpojuje vybrané elektronické okruhy. Technickým parametrem je počet a maximální proudová zátěž rozpojených okruhů. Obvyklé jsou tři okruhy a možná zátěž od 10 až 30 A, či různé kombinace těchto hodnot. To umožňuje přerušit obvod např. napájení palivového čerpadla.

11.2 Autoalarm

Narozdíl od imobilizéru je aktivním zabezpečovacím zařízením. To znamená, že pomocí různých snímačů monitoruje stav vozidla. Hlášení snímačů jsou vyhodnocována v centrálním počítači. Dále může iniciovat činnost vysílajícího radiomajáku pro umožnění dohledávání vozu pomocí pozemní sítě monitorováním těchto signálů či pomocí satelitů na oběžné dráze.

11.3 Dálkové ovládání

Autoalarmy jsou většinou ovládány dálkovým ovládáním a to buď s infračerveným signálem nebo signálem radiovým. Nevýhodou infračerveného je nutnost směřovat paprsek signálu a nemožnost vyloučení zaslepení paprsku slunečním světlem. Nutností u obou signálů je použití tzv. „plovoucího kódu“. Ten zabezpečuje, že signál jednou ovladačem vyslaný, je již pro další použití neplatný. Příjímač testuje, zda nový ovládací signál byl změněn definovaným matematickým algoritmem.

11.4 Snímače v automobilu

Úlohou těchto snímačů neinformovat o narušení jakéhokoliv vstupu do vozu. Prostorová čidla vyplňují vnitřek vozidla a poznávají pohybující se předmět. Běžně se používají ultrazvuková čidla se signálem 40 kHz nebo mikrovlnná čidla s radiovým signálem až 10 GHz.

11.5 Signalizace

Řídící jednotka autoalarmu zpracovává údaje od vstupních čidel a ovladačů. Vyvolaný poplach je signalizován blikáním všech směrových světel, další poplachovou signalizací je výstup na výkonnou akustickou sirénu.

12 Lokalizace polohy mobilního telefonu.

GSM Positioning je určování polohy mobilní stanice s využitím infrastruktury GSM. Jde o určitou obdobou GPS. Zde je však poloha určována ze vzdálenosti od tří základových stanic. Tuto vzdálenost musí mobilní stanice znát, neboť vzhledem ke konečné rychlosti šíření radiových vln je nutno vysílat data s odpovídajícím předstihem, aby byla na základové stanici přijata přesně v synchronizaci. Při datovém toku 270 kb/s je doba trvání jednoho bitu cca 3,7 μ s, během kterých radiové vlny urazí zhruba 1,1 km. Toto je současně hlavním limitem maximálního poloměru buňky cca 35 km, které synchronizační informace urazí od základové stanice. Současné udržování spojení mobilního telefonu až se šesti základovými stanicemi je rovněž prováděno z důvodu vyhodnocování kvality signálu okolních buněk kvůli předávání.

12.1 Základní principy určování polohy telefonu

Celulární síť pro mobilní komunikaci se skládá z jednotlivých buněk. Mobilní telefon přitom nemá pevně určenou polohu jako klasický telefon, a tak si musí aktivně zjišťovat v jaké buňce se nachází. Mobil vždy komunikuje uvnitř nějaké buňky s jejím vysílačem BTS.

V praxi se používají tři metody:

- Síť zaznamenává přechod aparátu z jedné buňky do druhé.
- Síť při žádosti o hovor zkontroluje všechny buňky.
- Kompromis mezi předchozími řešeními - více buněk je sdruženo do skupiny a telefon hlásí pouze změnu skupiny buněk.

Nejúspornější je druhá metoda, kdy centrum sítě ví, které mobilní telefony se nacházejí v které buňce.

Jedna taková buňka může pokrývat území o rozloze asi třiceti kilometrů. V hustě osídlených místech je počet buněk daleko vyšší.

12.2 Princip lokalizace polohy mobilní stanice

Telefon při komunikaci s BTS, která probíhá ve stanovených intervalech, zároveň kontroluje časovou prodlevu komunikace. Tím si systém zjistí přesnou vzdálenost od vysílající stanice pomocí funkce Time Advance. TA je údaj, který zohledňuje zpoždění signálu při jeho cestě mezi mobilní a základnovou stanicí. V současné době lze považovat za asi nejpresnější způsob určení polohy mobilní stanice triangulaci. Technicky se jedná o zaměření mobilní stanice pomocí více okolních BTS, pokud možno navíc s využitím údajů o TA. V principu jde o nalezení průsečíku oblouků kružnic, které určují místo, kam svým signálem zasahují tři nejsilnější základny v okolí hledaného telefonu. Metoda je poměrně přesná a v závislosti na konkrétním řešení může dosahovat přesnosti na stovky metrů. Odesílání údajů o okolních základnových

stanicích totiž musí podporovat SIM karta mobilního telefonu nebo přímo mobilní telefon (vybaven speciálním chipem).

Metropolitní lokalizace dokáže určit polohu s přesností až na několik desítek metrů. Pro tak přesnou lokalizaci je potřeba najít triangulační kvotu mobilní stanice. Ta sice nevyplývá ze samotné funkce GSM, ale může být doplněna jako dodatkový software. Při takzvaném refresh request mobilní telefon udává svoji polohu v buňce a současně vyměřuje sílu signálu vůči ostatním buňkám. Pokud tedy program zpracuje data získaná ve stejný okamžik z několika BTS (minimálně tří), lze s velkou přesností vypočítat aktuální polohu mobilního telefonu.

12.3 Příklad použití služby Locator

Locator je vnitrostátní služba mobilního operátora T-mobile. Je založena na výše uvedeném principu určení lokace pomocí tří vysílačů určujících polohu MS. Majitelé SIM karet typu Profi SIM nebo Universal SIM spustí službu nejjednodušeji přes SIM tolkit menu Kde je... Nejprve je třeba založit login a heslo LPIN na T-Zones.cz.

U MT50 je možné použitím AT-příkazu AT+CKPD="x" procházet menu mobilního telefonu. Do menu se dostaneme dosazením za 'x' znak ']'. Šipka klavesnici dolů, pak znamená znak 'V'. Poté obdržíme z SMSC SMS. Ta lze vyčíst příkazem AT+CMGL=0. Barevně je vyznačena samotná zpráva. Osahuje text uvedený na obr. 12.1.

```
AT+CMGL=0
```

```
+CMGL: 11,0,,128
07912460305002002404B9757241F150809130900080812B9A0C769BC970B3998C76038190ECB0
DD9D06B5CB73FA1B049587D16116681C9ED37520A83C8C0E83623116C81DD697EDA0B7D8FE26D7
7520655AEF4E839AE539FDDD4286D5651708042AC668AA59EC548B8900CE1A4C253BD944A0554B
0683B541B0990E86D3D572A058CE05C3B96035
```

```
OK
```

PDU datagram obsahuje uživatelská data vypsaná pomocí programu PDU Spy. Tato data je potřeba uložit a přidat k nim PDU SMS-SUBMIT část a odeslat na mobilní číslo majitele vozu.

Nabízejí se dvě řešení lokalizace pomocí GSM.

- Využití SIM tolkit funkcí mobilního telefonu a AT příkazů, ovládajících tlačítka mobilního telefonu.

- Nebo pomocí AT příkazů uvedených v kap. 9.3. vyčtením PDU části nesoucí informaci o poloze a aktuálním čase z příchozí SMS. Následným uložením do EEPROM a přeposláním na MT uživatele. Toto řešení přesahuje možnosti PIC16F628 s velikostí EEPROM 128 byte, protože délka uživatelské části PDU je 128 byte, což představuje celou kapacitu EEPROM, která je využívána i pro jiné funkce alarmu.

Volil jsem tedy první variantu. Druhé řešení vyžaduje větší EEPROM pro načtení celého PDU datagramu. Tyto parametry splňuje novější typ mikropočítače PIC16F877.



Obr. 12.1.: Výpis PDU datagramu programem PDUSpy

13 Závěr

Úkolem této bakalářské práce byla realizace zabezpečovacího systému s použitím GSM technologie. Hlavní důraz byl kladen na praktičnost. Výsledkem je, že tento alarm je schopen rozpoznat, kdy majitel vozu opravdu zaregistruje krádež nebo jiné narušení vozidla. Software rozpozná kdy je volaný mobilní telefon vypnutý, přijal hovor či byl v průběhu vyzvánění zavěšen. Druhou požadovanou vlastností Autoalarmu, bylo určení polohy kradeného vozu.

V této práci je uvedena konstrukce zapojení a popis komunikace. Dále zde uvádím otestované AT příkazy. Standard pro AT příkazy GSM 07.05 a GSM 07.07 podporují mobilní telefony Siemens C10, C35, C45, M50, C55. Navržený software je tedy kompatibilní i pro starší telefony od firmy Siemens.

Je zde nastíněna architektura použitého jednočipu, tzn. popis struktury speciálních registrů v paměti ROM (Read Only Memory). Dále popisují asynchronní řízení toku dat podle doporučení výrobce.

Vývojový diagram slouží k lepší orientaci ve zdrojovém kódu. Je zde podrobně rozepsána struktura PDU (Protocol Data Unit) datagramu a použita na praktických příkladech.

Principy lokalizace za použití GSM jsou úvodem pro návrh alarmové části řídicího programu. V kapitole 12.3 byly uvedeny dva způsoby použití služby Locator. Realizována byla varianta využívající SIM toolkit funkcí mobilního telefonu. Druhá vyžaduje novější mikropočítač z řady PIC16Fxxx s větší velikostí EEPROM. Např. PIC16F877, který by zároveň umožnil zvýšit počet vstupů pro připojení dalších senzorů.

Témata jsou hierarchicky koncipována způsobem, který usnadňuje návrh zabezpečovacího zařízení pro obdobné účely.

Dalším využitím, může být zabezpečení rodinných domků či chat. Automatické řízení technologie GSM (Global System for Mobile Communications) mikropočítačem najde široké uplatnění v dálkovém ovládání spotřebičů. Příkladem může být např. bojler, který je dálkově ovládaný SMS (Short message Service) či pouhým prozvoněním.

14 Použité zdroje

- [1] VACEK, Václav. *Učebnice programování PIC*. BEN – technická literatura, Praha 2000, 1. vydání. ISBN 80-86056-87-2
- [2] HRABÁČEK, Jiří. *Komunikace mikrokontroléru s okolím - 1.díl*. BEN, Praha 1999, 1.vydání. ISBN 80-86056-42-2
- [3] ROZEHNAL, Zdeněk. *Mikrokontroléry Motorola HC11*. BEN, Praha 2001. 1.Vydání. ISBN 80-86056-77-5
- [4] KREJČÍŘÍK, Alexandr. *SMS GSM pagery a alarmy - princip, použití, návody, příklady*. ISBN 80-7300-082-2
- [5] LÁNÍČEK, Robert. *Elektronika, obvody – součástky - děje*. BEN, Praha 1998. 1.Vydání. ISBN 80-86056-25-2
- [6] KREJČÍŘÍK, Alexandr. *Napájecí zdroje I*. BEN, Praha 1997. 2. Vydání. ISBN 80- 86056-02-3
- [7] CADSOFT EAGLE, *Manual, tutorial Eagle 4.15*. Dostupné z FTP:
< <ftp://ftp.cadsoft.de/pub/program/4.15>>.
- [8] Nebojsa Matic. *Mikroelektronika, PIC microcontrollers, for beginners too*.
Dostupné
z URL:<http://www.mikroelektronika.co.yu/english/product/books/PICbook/0_Uvod.htm>.
- [9] *SMS and the PDU format*. Dostupné z URL:
<<http://www.dreamfabric.com/sms/>>
- [10] *Microchip Microcontrollers PIC16F62X*. Dostupné
z URL:< <http://www.microchip.com>>
- [11] Microchip. *FLASH-Based 8-Bit CMOS Microcontrollers PIC16F62X*.
Dostupné
z URL<<http://ww1.microchip.com/downloads/en/DeviceDoc/40300c.pdf> >
- [12] HANKOVEC, David . DH servis <<http://www.dhservis.cz/>>
- [13] AMNEZ. Bramo <<http://bramo.pcsvet.net/>>
- [14] SNÁŠEL, Jaroslav. Článek: *Jak určit polohu mobilního telefonu*
Na webovém portálu MobilMania <<http://www.mobilmania.cz>>
- [15] Webové stránky Electronic Circuit Engineering. Dostupné z URL:
< http://www.interq.or.jp/japan/se-inoue/e_menu.htm >

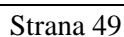
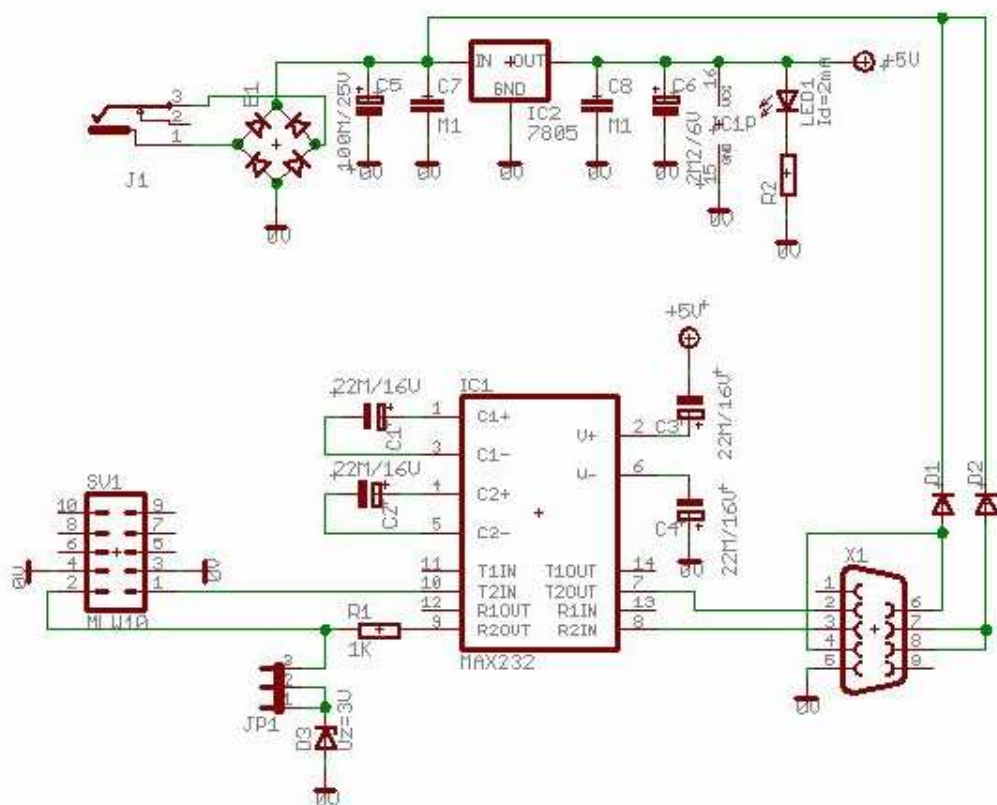
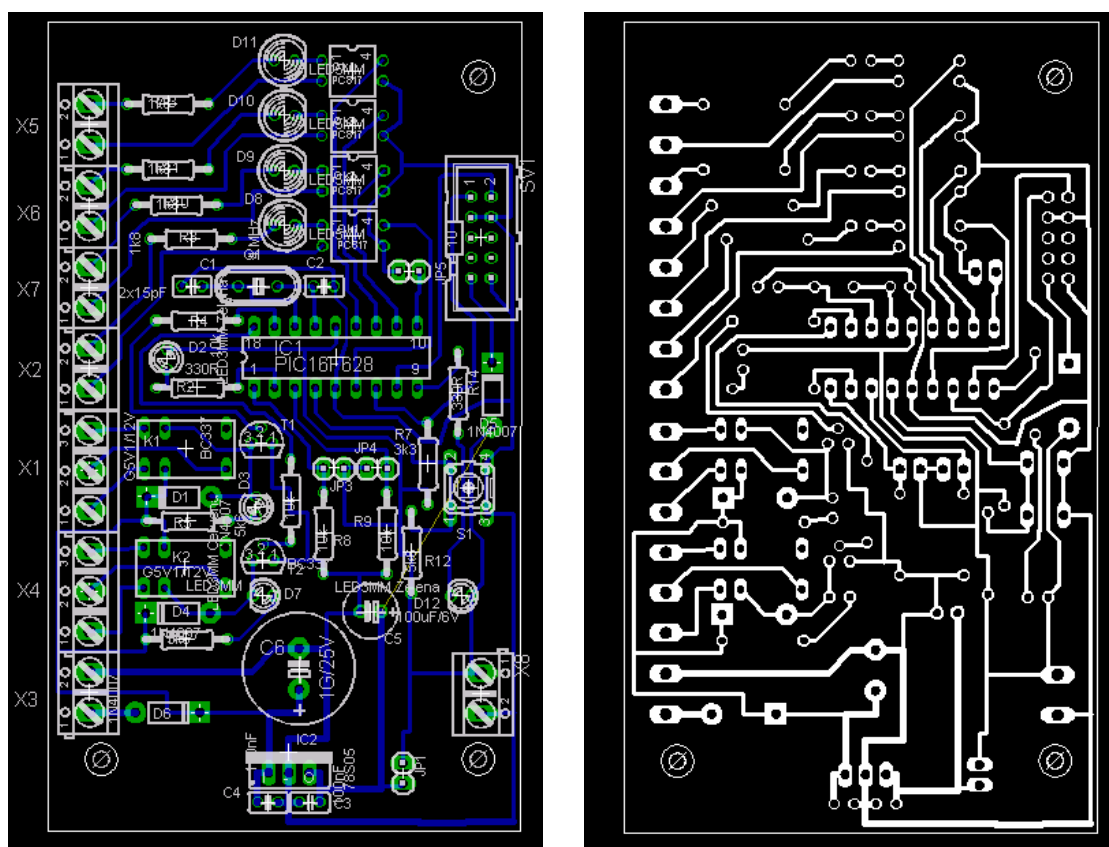


Schéma zapojení pro komunikaci mezi PC-MT pomocí RS232 rozhraní

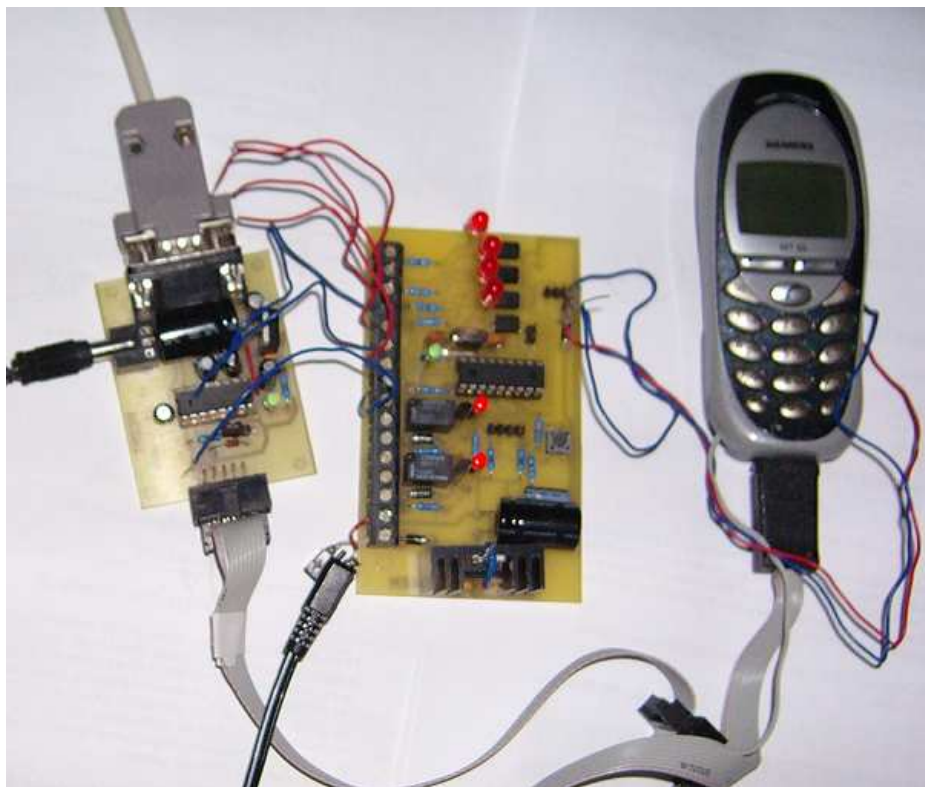


Příloha B – DPS zapojení Autoalarmu



Příloha C – Fotografie konstrukce Autoalarmu, komunikace MT50-RS232 a konstrukce záložního zdroje

Alarm, RS232-TTL, MT50



Konstrukce záložního zdroje 3,6V s MC34063 DC/DC měničem



Obvody využívající IO obvod MC34063 lze použít na pomalé dobíjení NiMH článků i jako záložní zdroj pro napájení mikropočítače či mobilního telefonu.

Příloha D– Seznam součástek

Autoalarm:

R1,R4,R8,R9	rezistor miniaturní 10k	D1,D4, D5,D6	dioda 1N4007
R2, R14	rezistor miniaturní 330R	D2,D12	LED zelená 3mm
R3,R10,R11,R13	rezistor miniaturní 1k8	D8-D11 D3,D7	LED červená 3mm
R5,R6,R12	rezistor miniaturní 5k6	T1,T2	tranzistor NPN univerzální BC337
R7	rezistor miniaturní 3k3	IC1	mikroprocesor PIC 16F628 (16F628A)
Q1	krystal 4MHz/HC49U-V	IC2	Napěťový stabilizátor 78S05
S1	mikrotlačítko OMRON	K1,K2	relé G5V1/12V (GM electronic)
OK1-OK4	Optočlen PC 817	X1,X4	svorkovnice ARK500/3
C1,C2	keramický 15pF	X2,X3, X5,X6, X7	svorkovnice ARK500/2
C3,C4	keramický 100nF	JP1, JP3,JP4,J P5	lámací konektory na jumpery
C5	elektrolyt 100uF/6V	Polohové čidlo	Citlivost +-10 stupňů
C6	elektrolyt 1000uF/25V	Patice	precizní 18 pin