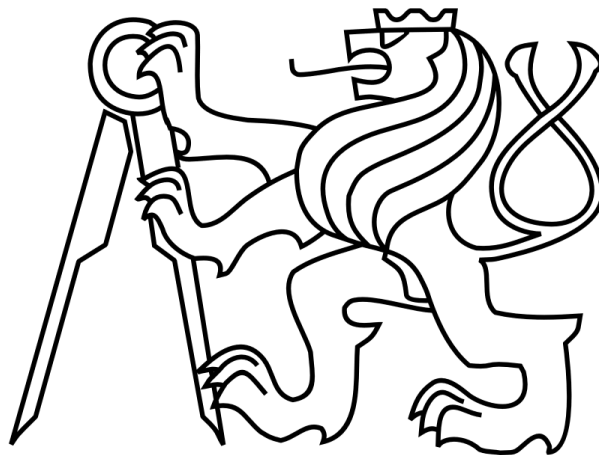


České vysoké učení technické v Praze

Fakulta elektrotechnická

katedra mikroelektroniky



Diplomová práce

Autoalarm s GSM/GPRS komunikací a GPS lokalizací

Autor: Bc. Filip Prokop

Vedoucí práce: Ing. Tomáš Vítek

2014

Čestné prohlášení

Prohlašuji, že jsem zadanou diplomovou prací na téma „Autoalarm s GSM/GPRS komunikací a GPS lokalizací“ zpracoval sám s přispěním vedoucího práce a používal jsem pouze literaturu uvedenou na konci práce. Souhlasím se zapůjčováním práce a jejím zveřejňováním.

V Praze dne

podpis

Poděkování

Děkuji vedoucímu mé diplomové práce, panu Ing. Tomáši Vítkovi za ochotu, cenné rady a připomínky, které vedly k úspěšnému vyhotovení. Dále děkuji za obětovaný čas a věcnou komunikaci. Děkuji své rodině za podporu a klidné prostředí při tvorbě práce.

Anotace

Tato práce pojednává o návrhu a realizaci autoalarmu s GSM/GPRS komunikací a GPS lokalizací. Základem návrhu je analýza trhu, podle které se snažím nabídnout zařízení, jež by splňovala kritéria využitelná v praxi. Cílovou vlastností autoalarmu je jeho ovladatelnost SMS příkazy a zaslání GPS dat na mobilní telefon uživatele nebo na FTP server, kde si data importuje PHP skript a zobrazí pozici na mapě.

Klíčová slova

autoalarm, GPS lokalizace, SMS příkazy

Abstract

This thesis deals with design and realization of the car alarm with GSM/GPRS communication and GPS localization. The analysis of market is the foundation of design according to I make efforts to offer the device that would carry out standards usable in practice. The target attribute is the controllability by SMS commands and sending GPS data to a user's mobile phone or to a FTP server, where it imports data to the PHP script and shows position on the map.

Index Terms

car alarm, GPS localization, SMS commands

Obsah

| | | |
|--------|---|----|
| 1. | Úvod | 1 |
| 2. | Teoretický rozbor | 2 |
| 2.1. | GSM..... | 2 |
| 2.1.1. | Architektura sítě GSM..... | 2 |
| 2.1.2. | Telefonní hovory a SMS zprávy..... | 4 |
| 2.1.3. | Datový přenos GPRS..... | 5 |
| 2.2. | Družicová navigace GPS | 7 |
| 2.2.1. | Struktura GPS | 7 |
| 2.2.2. | Vlastnosti signálu GPS | 8 |
| 2.2.3. | Struktura přijímačů GPS..... | 10 |
| 2.3. | Komunikační rozhraní | 13 |
| 2.3.1. | Serial Peripheral Interface (SPI)..... | 14 |
| 2.3.2. | Inter Integrated Circuit (I ² C) | 15 |
| 2.3.3. | RS-232 (UART) | 17 |
| 2.4. | Controller Area Network (CAN)..... | 20 |
| 2.4.1. | Fyzická vrstva | 21 |
| 2.4.2. | Linková vrstva | 23 |
| 3. | Analýza trhu | 28 |
| 3.1. | Konkurenční modely | 28 |
| 3.1.1. | Jablotron CA-1803B..... | 28 |
| 3.1.2. | Keetec T GSM..... | 29 |
| 3.1.3. | Spy 10 PKE | 30 |

| | | |
|--------|-------------------------------------|----|
| 3.2. | Předběžný návrh řešení | 31 |
| 3.3. | Výběr komponent | 31 |
| 3.3.1. | Mikrokontrolér | 31 |
| 3.3.2. | GSM a GPS modul..... | 32 |
| 4. | Návrh hardware | 34 |
| 4.1. | Napájení..... | 35 |
| 4.1.1. | Nabíjecí obvod..... | 35 |
| 4.1.2. | Snižující měnič napětí | 36 |
| 4.1.3. | Lineární stabilizátor napětí | 37 |
| 4.2. | GSM/GPS modul | 37 |
| 4.3. | Mikrokontrolér..... | 39 |
| 4.4. | Akcelerometr | 39 |
| 4.5. | Paměť EEPROM..... | 40 |
| 4.6. | Poplachové vstupy a výstupy..... | 40 |
| 4.6.1. | Poplachové vstupy..... | 41 |
| 4.6.2. | Připojení CAN sběrnice..... | 41 |
| 4.6.3. | Poplachové výstupy..... | 42 |
| 5. | Návrh Firmware..... | 43 |
| 5.1. | Inicializace mikrokontroléru..... | 43 |
| 5.1.1. | Generování hodinového signálu | 43 |
| 5.1.2. | Komunikační moduly | 45 |
| 5.1.3. | Analogově digitální převodník | 47 |
| 5.1.4. | Časovače a přerušení | 49 |
| 5.2. | Komunikace s perifériemi..... | 51 |

| | | |
|--------|--|----|
| 5.2.1. | Obsluha akcelerometru MMA7455 | 51 |
| 5.2.2. | Obsluha paměti EEPROM 24FC1025 | 53 |
| 5.2.3. | Obsluha GSM/GPS modulu SIM908 | 54 |
| 5.3. | Běh programu | 57 |
| 6. | Zhodnocení | 59 |
| 7. | Závěr | 61 |
| I. | Seznam literatury | 62 |
| II. | Seznam použitých zkratk | 64 |
| III. | Přílohy | 68 |
| | Seznam SMS příkazů pro autoalarm | 68 |
| | Schéma a DPS autoalarmu | 69 |
| | Vrchní vrstva DPS | 72 |
| | Napájecí vrstva DPS | 73 |
| | GND vrstva DPS | 74 |
| | Spodní vrstva DPS | 75 |
| | Fotografie autoalarmu | 76 |
| | Obsah příloženého CD | 79 |

1. Úvod

V současnosti se automobily bez alarmů neobejdou. I když statistika krádeží vozidel má sklon ke stagnaci, musíme se stále zabývat otázkou jak auto před nimi chránit. K důvodům sestupného trendu krádeží vozidel patří také tlak výrobců a jejich dodavatelů na sofistikovanější bezpečnostní systémy.

Dále se začaly rozmáhat alarmy, které nejsou vyrobeny od výrobce automobilu a instalují se dodatečně. Tyto alarmy mají v sobě zabudované komunikační systémy jako GSM či GPS a bezpečnost vozidel před krádeží opatřených těmito moduly rapidně vzrůstá. Proto mě zaujala tato tematika a přišlo mi vhodné se jí důkladněji zabývat.

Práce bude pojednávat o kompletním návrhu GSM/GPS autoalarmu, kde po zjištění vlastností již uvedených systémů na trhu, zahrnu jednotlivé funkce do alarmu tak, aby byl alarm unikátní a v budoucnu využitelný.

2. Teoretický rozbor

Kapitola bude pojednávat o systémech, protokolech a sběrnicích použitých ve vývoji autoalarmu. Zmíněny budou systém GPS a GSM se službou GPRS, u které se využívá připojení k internetu pomocí IP či PPP protokolu. V zařízení budou implementovány sběrnice typu UART (RS-232), I²C, CAN a SPI. Každá má svoje pole působnosti a různorodost využití s mnoha výhodami a nevýhodami, které budou demonstrovány.

2.1. GSM

V 70. letech 20. století se ukázalo, že analogové systémy nebudou stačit poptávce telefonních hovorů. Na řadu přišlo vytvoření nové standardizační skupiny GSM (Groupe Spécial Mobile), kterou zveřejnila roku 1982 Konference evropských správ pošt a telekomunikací CEPT. Název GSM je později interpretován jako Global System for Mobile Communications (Globální systém pro mobilní komunikaci). Systém vyřešil otázku fungování jedné mobilní stanice s jedním číslováním ve všech ostatních státech, které používají systém GSM, známý pod názvem roaming.[5]

Základní rádiová rozhraní u GSM jsou GSM 900 a GSM 1800. Frekvenční pásmo GSM 900 je 890 – 915 MHz pro uplink a 935 – 960 MHz pro downlink, u GSM 1800 se pohybujeme v hodnotách 1710 – 1785 MHz pro uplink a 1805 – 1880 MHz pro downlink.

GSM používá kombinovanou metodu FDMA/TDMA přístupu. Ve frekvenční oblasti má každý kanál přidělen 200 kHz pásmo a v časové oblasti máme k dispozici 8 „timeslotů“ na kanál. V kapitole jsou použity zdroje [5],[6] a [7].

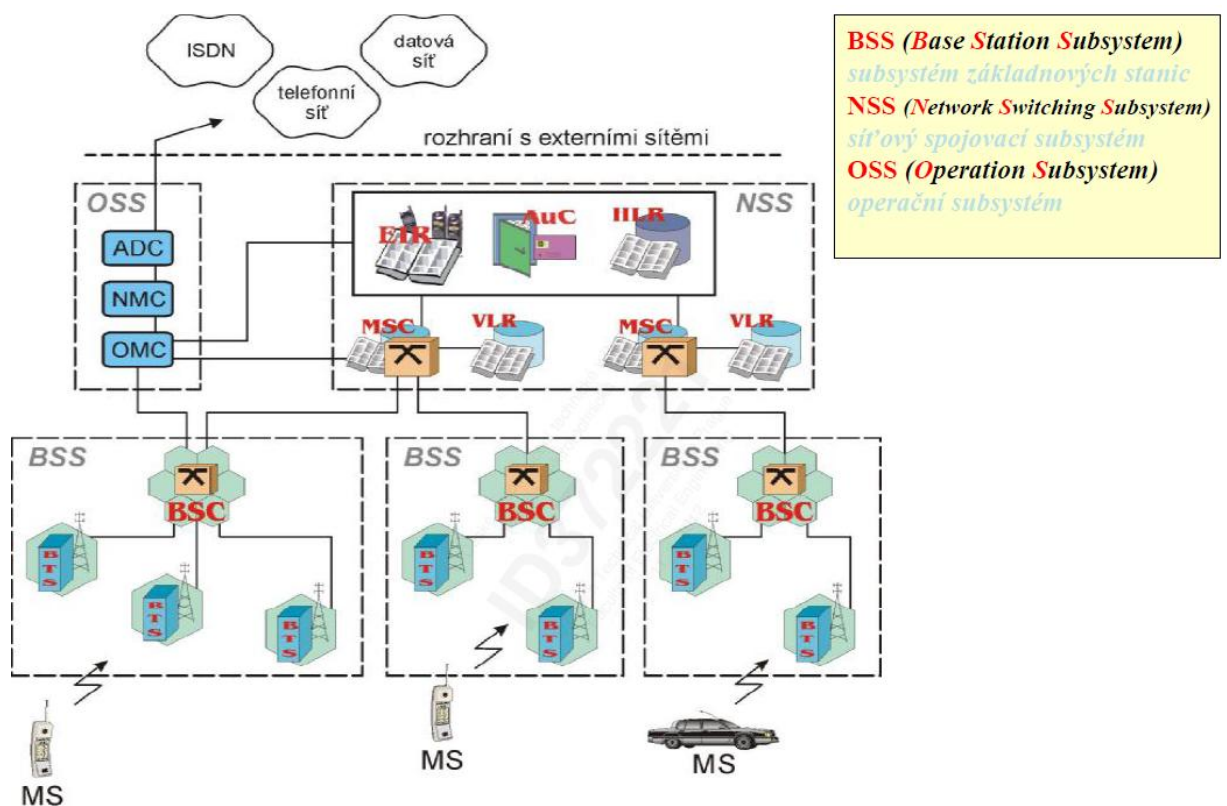
2.1.1. Architektura sítě GSM

Operátor využívající síť GSM spravuje alespoň jednu MSC (Mobile switching centre), pod kterou spravuje oblasti LA (Location area). Oblasti LA je přiřazena základnová řídicí jednotka (BSC), která spravuje určitý počet základnových stanic (BTS).[5]

Kapitola 2: Teoretický rozbor

Na obrázku 2-1 můžeme vidět strukturu sítě GSM rozdělenou na 3 základní bloky – subsystém základnových stanic (BSS), síťový spojovací subsystém (NSS) a operační subsystém (OSS).

V subsystému základnových stanic komunikují mobilní stanice se stanicemi BTS spadajícími pod BSC. Úkolem BSC je zjišťovat, zda má iniciovat přepínání mobilních stanic mezi jednotlivými BTS (známé pod názvem handover) a přidělování či uvolňování rádiových kanálů. Provoz subsystému vyžaduje informace o poloze mobilních zařízení, aby mohla BSC zajistit správné přiřazení BTS.



Obrázek 2-1: Struktura sítě GSM [5]

Síťový spojovací subsystém zahrnuje telefonní ústřednu MSC. Jejímí funkcemi jsou aktualizace polohy, ověřování, registrace a handover. Tyto funkce zajišťují zařízení tvořící tzv. identifikační databáze, které zahrnují domovský registr HLR, návštěvnický registr VLR, registr mobilních zařízení EIR a ověřovací centrum (AuC).

Domovský registr HLR obsahuje informace o zákaznících operátora. Jsou jimi identifikační údaje IMSI (mezinárodní identifikace zákazníka), MSISDN (ISDN číslo

Kapitola 2: Teoretický rozbor

zákazníka), lokalizační údaje LMSI (pomocný vyhledávací klíč) a MSRM (roamingové číslo zákazníka). Návštěvní registr VLR obsahuje údaje z HLR, navíc TMSI (dočasné identifikační číslo mobilní stanice) a LAI (číslo oblasti, ve které byla mobilní stanice zaregistrována). Ověřovací centrum (AuC) obsahuje kopii tajného klíče uloženého na SIM kartě a zajišťuje bezpečnost při rádiovém přenosu. Registr mobilních zařízení EIR obsahuje identifikační čísla mobilních stanic IMEI, které jsou rozděleny do 3 databází:

- White list – známá a správná čísla IMEI
- Black list – ukradené mobilní telefony
- Gray list – nejasná mobilní zařízení

Operační subsystém zajišťuje registraci a tarifkaci zákazníků služeb a provoz předešlých subsystémů (BSS, NSS). Operační systém dělíme na 3 části – ADC (administrativní centrum), NMS (síťové centrum) a OMC (operační centrum). ADC se podílí na správě a managementu účastníků sítě GSM, tedy aktivaci, tarifkaci a placení účtů. NMC monitoruje mobilní stanice, zajišťuje řízení toku v síti. OMC řeší úlohy údržby a provozu sítě, spravuje AuC a EIR.

2.1.2. Telefonní hovory a SMS zprávy

Obsluhu telefonního hovoru můžeme demonstrovat pomocí obrázku 2-2, kde:

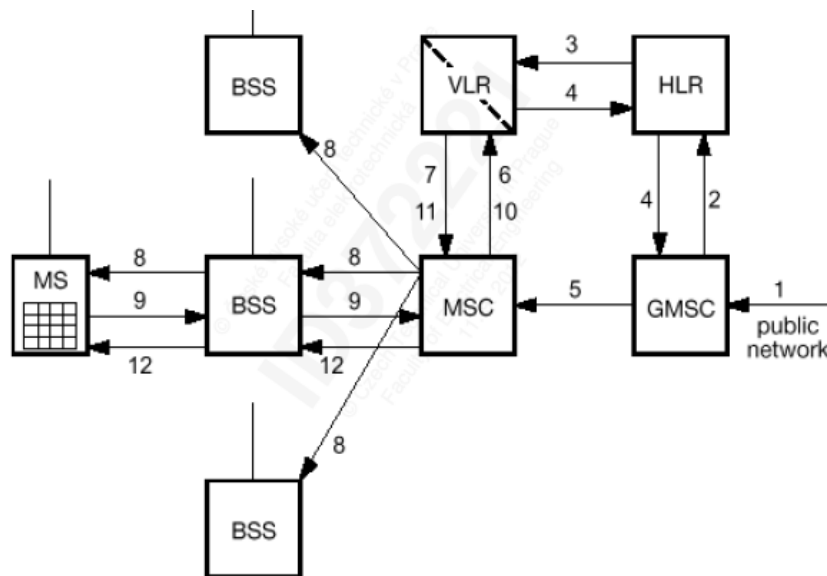
1. Příchozí hovor mobilního účastníka je směrován na bránu GMSC (MSC).
2. Podle čísla IMSI koncového účastníka se rozhodne pro daný HLR.
3. Dále se zavolá příslušný VLR o poskytnutí roamingového čísla MSRN.
4. Roamingové číslo se vrací do brány GMSC.
5. Hovor se poté přesměruje na lokální MSC,
6. která zjišťuje pozici a dostupnost mobilní stanice pomocí VLR
7. Pokud VLR potvrdí dostupnost koncového účastníka,
8. povolí se rádiové spojení a zasílá se požadavek na spojení do všech oblastí VLR.
9. Volaný účastník se ozve z aktuální BSS a
10. provedou se veškeré zabezpečovací a ověřovací procedury pomocí EIR a AuC.

Kapitola 2: Teoretický rozbor

11. Nakonec VLR potvrdí MSC,
12. že telefonní hovor může být proveden (12).

Funkce SMS (Short Message Service) se stala k údivu tvůrců celosvětovým fenoménem. Posílání krátkých zpráv na svá mobilní zařízení o délce maximálně 160 znaků využívá většina účastníků sítě GSM.

Odeslaná SMS zpráva je přijata SMS centrem (SMSC). Poté SMSC odešle žádost do HLR, která odpoví, zda je příjemce zprávy aktivní či neaktivní. Pokud je neaktivní, ponechá si SMSC nějaký čas zprávu u sebe, pak pokus opakuje. Pokud je aktivní pošle HLR potvrzení a SMSC se pokusí zprávu doručit.



Obrázek 2-2: Obsluha telefonního hovoru [5]

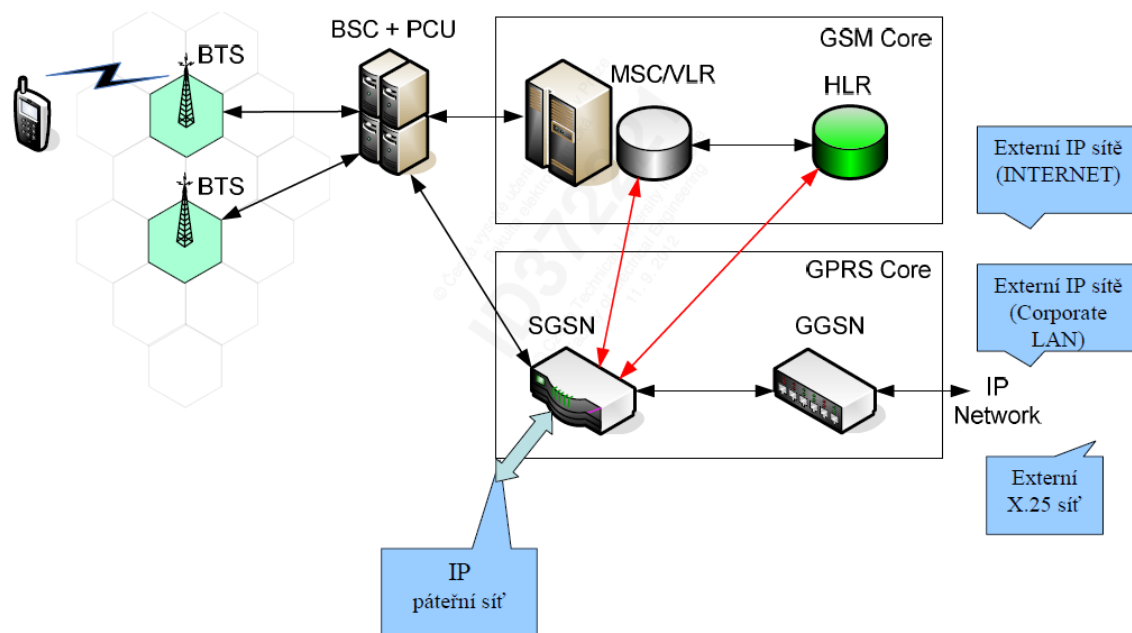
2.1.3. Datový přenos GPRS

GPRS (General Packet Radio Service) představuje rozšíření sítě GSM o přenos dat s přepojováním paketů. Rozšíření spočívá ve vybudování nové paralelní sítě, která používá pouze rádiové rozhraní GSM. Mobilním uživatelům umožňuje připojení k internetu a dalším sítím založených na protokolech IP nebo X.25. Pro rozšíření museli operátoři vyměnit modul v BTS a udělat upgrade softwaru. Dále se přidali do sítě zařízení SGSN

Kapitola 2: Teoretický rozbor

(Serving GPRS Support Node) a GGSN (Gateway GPRS Support Node). S ohledem na různá kódování používaných kanálů a přidělování až osmi „time slotů“ lze dosáhnout přenosové rychlosti od 9 kbit/s až k teoretickým 171,2 kbit/s.[6]

GPRS používá pro přenos a zabezpečení dat čtyři kódovací schémata, která se liší v objemu přenesených dat na úkor zabezpečení. Schémata označujeme CS-1 až CS-4 (Coding Scheme). CS-1 disponuje nejvyšší úrovní zabezpečení – jen 160 ze 456 bitů se používá pro přenos informace na rozdíl od CS-4, která umožňuje použít pro přenos dat 400 ze 456 bitů. V praxi se však setkáme většinou jen s CS-1 a CS-2.



Obrázek 2-3: Architektura GPRS v síti GSM [5]

Zařízení GGSN obsluhuje připojení do vnějších sítí, plní tedy funkci směrovače (routeru) pro všechny uživatele připojené v dané GSM síti. Dále monitoruje data pro vyúčtování za použité služby. GGSN přechovává data získaná z HLR a SGSN, aby mohl obsloužit příchozí a odchozí data – identifikace účastníka pomocí IMSI a lokalizace adresou uzlu SGSN, ve které je účastník registrován.

Směrování příchozích a odchozích dat zajišťuje zařízení SGSN (Serving GPRS Support Node). Dalšími úkoly zařízení jsou šifrování, ověřování, řízení mobility a sběr dat

Kapitola 2: Teoretický rozbor

pro vyúčtování. SGSN uchovává identifikační údaje – IMSI, PDP adresy a informace o poloze – přidružený VLR a adresy uzlů GGSN.

Připojení k GPRS se odehrává v několika krocích. Nejprve pošle uživatel žádost do SGSN. Při změně uzlu SGSN a následné identifikaci s Packet-TMSI se uskuteční spojení s původní SGSN (s další žádostí o údaje). V případě neznámé identity uživatele u obou SGSN je po uživateli vyžádán IMSI. Následně si SGSN ověří uživatele a přidělí mu nové TLLI (Temporary Logical Link Identity), které identifikuje uživatele GPRS.

Po úspěšném připojení k GPRS se musí pro následnou výměnu dat aktivovat PDP kontext. GGSN rozpozná uživatele a umožní mu komunikaci s vnější sítí nebo i více sítěmi. Nejprve si uživatel zažádá o aktivaci PDP kontextu, který obsahuje údaje o koncové síti jako jméno přístupového uzlu (APN) či IP adresu. Aktivuje se zabezpečení mezi koncovými účastníky a SGSN ověří žádost. Dále zkontroluje dostupnost požadované služby a pošle APN do GGSN. SGSN poté vytvoří TID (Tunnel Identity) a spojí se s GGSN. GGSN se spojí s vnější sítí a pošle žádost o IP adresu cizímu serveru, který jí následně pošle a GGSN jí přesměruje uživateli.

2.2. Družicová navigace GPS

GPS (Global Positioning System) vznikl roku 1973 sloučením pokusných systémů amerických vzdušných sil a námořnictva. Systém nám umožňuje určit pomocí několika družic polohu kdekoli na Zemi. Přesnost se pohybuje pro běžné uživatele od několika metrů až po jednotky centimetrů pro vojenské a komerční použití. Systém GPS se dělí na tři bloky – kosmický, řídicí a uživatelský. V kapitole jsou použity zdroje [5] a [8].

2.2.1. Struktura GPS

Kosmický podsystém představuje 21 družic + tři záložní družice, které obíhají Zemi v šesti drahách po čtyřech ve výšce 20 190 km a se sklonem k rovníku 55°. Jeden oběh kolem Země trvá 11 h 58 min a družice se pohybují rychlostí 11 300 km/h. Na Zemi je běžně pozorovatelných 4-8 družic, může jich být až dvanáct v jeden okamžik. Družice jsou vybaveny přijímačem, vysílačem, atomovými hodinami s cesiovým nebo rubidiovým

Kapitola 2: Teoretický rozbor

oscilátorem o frekvenci 10,23 MHz a přístrojem pro detekci atomového výbuchu. GPS systém používá několik nosných frekvencí (L1-L5) s pseudonáhodnými kódy, které jsou zpracovávány použitím CDMA (Code division multiple access).

Řídící podsystem se skládá z hlavní řídicí stanice (Master Control Station) a čtyř pozemních vysílačů, které monitorují nepřetržitě viditelné družice a posílají data do MCS. MCS vypočte údaje o oběžných drahách a korekci času, které jsou následně poslány družicím a ty je pak v navigační zprávě přeposílají přijímačům.

Uživatelský podsystem tvoří GPS přijímače, které získávají od kosmických družic údaje o své poloze, rychlosti a čase. Pro určení polohy postačuje signál ze třech satelitů, přibýváním viditelných satelitů přijímačem se přesnost zvyšuje. Autorizovaní uživatelé (např. armáda USA) mají přístupnou vyšší přesnost polohy (jednotky cm) a času (pod 10 s) systému pro pohyb vojáků v poli či navádění raketových systémů.

2.2.2. Vlastnosti signálu GPS

Nosné frekvence signálu jsou odvozeny od nominální frekvence oscilátoru. Následně jsou na ně modulovány navigační zprávy a pseudonáhodné kódy.

- L1 (1575,42 MHz) používá kód C/A pro civilní aplikace a kód P (L1M) pro vojenské využití
- L2 (1227,60 MHz) má od roku 2005 k dispozici civilní signál L2C a vojenský signál L2M
- L3 (1381,05 MHz) přenáší informace o balistických raketách, jaderném výbuchu a vyšších hodnotách energie v infračerveném spektru
- L4 (1841,40 MHz) informuje o ionosférickém zpoždění, které se negativně promítá ve výpočtu polohy, lze eliminovat použitím dvou kmitočtů
- L5 (1176,45 MHz) se využívá zatím jen pro vojenské účely (provoz od roku 2012), v plánu jsou civilní účely v rámci letecké navigace

Na nosné vlny jsou modulovány pseudonáhodné kódy $C(t)$ a $P(t)$ a data $D(t)$ pomocí modulace BPSK (Binary Phase Shift Keying), kde kódy a data nabývají hodnot +1 a -1.

Kapitola 2: Teoretický rozbor

Modulace pseudonáhodnými kódy umožňuje přenos rozprostřeným spektrem, které zvyšuje odolnost proti rušení a kódové oddělení družic, které mohou pracovat na stejném kmitočtu (CDMA).

Kód $C(t)$ známý jako C/A (Coarse Acquisition) nebo Goldův kód má ostré maximum autokorelační funkce, které umožní přesné měření a zároveň korelační funkce dvou různých kódů dosahuje signifikantně menších hodnot. Bitová rychlost kódu je 1,023 Mb/s. Kódem se můžou měřit vzdálenosti nebo se dá přepnout na přesnější kód $P(t)$ neboli kód P .

Kód P (Precision) umožňuje přesnější měření vzdálenosti díky své vyšší bitové rychlosti (10,23 Mb/s) a možnosti měření na dvou kmitočtech, které eliminuje vliv ionosférické refrakce. Kódem P určíme polohu s přesností minimálně 3 m. Možnost zneužití se vyřešilo překódováním na kód Y , který je dostupný pouze autorizovaným uživatelům, jejichž přístroje obsahují kód W . Kód W dekóduje kód Y na kód P . Překódování na kód Y je známé pod pojmem $A-S$ (Anti-Spoofing). Režim $A-S$ eliminuje možnost imitování nepřátelských družic.

K určení polohy přijímače potřebujeme souřadnice družice, které jsou vypočítány podle dráhy družice. Ta je posílá ve formě navigační zprávy, která se dělí na pět „subframů“ po 300 bitech:

1. obsahuje telemetrické slovo TLM (diagnostika), HOW (hand-over word) nesoucí identifikační údaje a časovou hodnotu TOW (time of week)
2. efemeridy družice
3. efemeridy družice
4. almanach, koeficienty ionosférického modelu
5. almanach pro nejdéle sloužících družic

Přijímač obdrží přesné efemeridy družice a zároveň měří dobu příchodu zprávy. Pro určení polohy v souřadnicovém systému Země potřebujeme data z nejméně čtyř družic. Řešíme soustavu čtyř rovnic o čtyřech neznámých – souřadnice x,y,z a čas t . Čas družic a

Kapitola 2: Teoretický rozbor

přijímačů není synchronizován, proto počítáme s rozdílem absolutního času družic a doby trvání přenosu zprávy. Soustavu vyřešíme pomocí rovnice:

$$(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2 = (c(t_{mi} - t_0))^2 ; i = 1,2,3,4$$

Efektivní hodnota chyby určení polohy se vyjádří součinem směrodatné odchylky chyby měření na dráze mezi družicí a přijímačem a členu PDOP (Positional Dilution of Precision), který reprezentuje polohu družic. Ten se dá zmenšit zahrnutím více družic do výpočtu, které jsou zároveň od sebe co nejvíce vzdáleny.

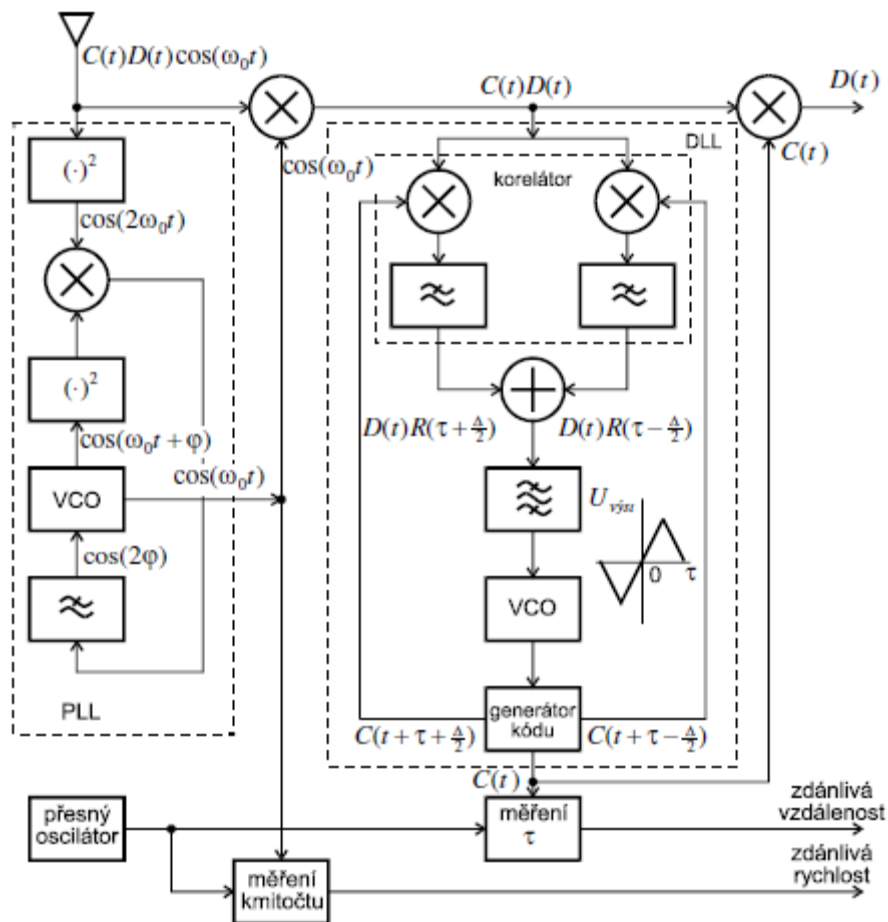
Měření a výpočty polohy jsou přesné i za použití jediného kmitočtu s civilním kódem C/A, proto se vláda USA rozhodla v zájmu bezpečnosti zkreslit přesnost měření. Metoda pod názvem Výběrový přístup SA (Selective Availability) záměrně zkresluje přesnost měření vzdálenosti změnou hodinového kmitočtu signálu družic a efemeridů, které se vysílají v navigační zprávě.

Autorizovaní uživatelé mají k dispozici speciální kryptografický kód, kterým potlačí nejen SA, ale i A-S a dokáží tedy zpracovat kód Y. Těmto uživatelům je zpřístupněna přesná navigační služba PPS (Precision Positioning Service), zatímco běžný uživatel si musí vystačit se standardní navigační službou SPS (Standart Positioning Service).

2.2.3. Struktura přijímačů GPS

Struktura přijímačů GPS se odvíjí v závislosti na jejich použití. Mezi nejsložitější patří letecké přijímače, u kterých se kladou nároky nejen na ovládání, ale i na odolnost vůči klimatu a mechanickému poškození. Přijímače pro běžné uživatele jsou většinou vybaveny následujícími bloky:

- anténa
- blok anténní elektroniky
- přijímač a počítač (RPU – Receiver and Processor Unit)
- řídicí a zobrazovací jednotka (CDU – Control and Display Unit)
- zavaděč dat (DLS – Data Loader System)



Obrázek 2-4: Typické schéma přijímače GPS [8]

Parametrem *antény*, který potřebujeme pro správnou funkci systému GPS, je hemisférická směrová charakteristika. Rozlišujeme dva typy antén:

- FRPA (Fixed Radiation Pattern Antenna) – pevná směrová charakteristika
- CRPA (Controlled Radiation Pattern Antenna) – řízená směrová charakteristika

Konstrukce *FRPA antén* jsou šroubovicové kuželové, šroubovicové válcové a mikropáskové ploché. *CRPA antény* jsou využívány v zarušeném prostředí, kde se předpokládá natočení minima směrové charakteristiky směrem ke zdroji rušení. Předpokládané potlačení rušení se pohybuje v rozmezí 30 – 40 dB.

Blok anténní elektroniky (AE) se v levných přijímačích vůbec nenachází nebo plní pouze funkci zesilovače. V opačném případě disponuje kompenzátorem ztrát ve vedení a filtrem rušení. V dalších případech chrání AE anténu proti přepětí v atmosféře, provádí testování přijímače a převádí signál na mezifrekvenční kmitočty.

Kapitola 2: Teoretický rozbor

Přijímač a počítač (RPU) plní funkci zpracování přijatého signálu. Vypočítává GPS souřadnice a obsluhuje sběrnice přijímače. K RPU jsou připojeny následující vstupní a výstupní zařízení:

- Číslicově analogový převodník
- Jiné navigační systémy
- Přístrojová brána
- Zařízení pro zavedení kryptografických klíčů
- Řídící a zobrazovací jednotka CDU
- Zavaděč dat

Číslicově analogový převodník plní svoji funkci hlavně v leteckých aplikacích, kde signalizuje odchylku od plánované trati. *Daty jiných navigačních systémů* se dosáhne vyšší spolehlivosti a přesnosti. V leteckých aplikacích se připojují inerciální systémy schopné překlenout krátké výpadky signálu nebo u pozemních vozidel můžeme připojit piezoelektrický gyroskop. Testování nebo diagnostika se provádí přes *přístrojovou bránu*, která používá RS-422 pro komunikaci s externími přístroji. Pro zavedení *kryptografických klíčů*, které potlačují AS a S-A se může použít klávesnice CDU nebo připojení externího zařízení přes sběrnici.

Řídící a zobrazovací jednotka (CDU) je součástí mnoha GPS zařízení. Používá se pro zobrazení informací o poloze či stavu zařízení. Důležité jsou prvky traťové navigace schopné určit příčnou odchylku od tratě. CDU se může používat i pro diagnostiku, jelikož lze vkládat přes klávesnici příkazy a potřebné údaje jsou následně zobrazeny např. na LCD displeji.

Pro údaje, které jsou potřeba vložit do přijímače, se může využít CDU, ale i speciální médium pod názvem *Zavaděč dat (DLS)*. Data traťových bodů, šifrovacích klíčů či speciálních senzorů jsou v DLS zálohovány a uspořádané pro další využití. Využívaným médiem pro tyto účely jsou např. bezkontaktní paměťové karty (NavData Card).

Kapitola 2: Teoretický rozbor

Získaná data se dají zpracovat ve třech možných konfiguracích navigačního přijímače – *několikakanálový*, *sekvenčně měřící* a *multiplexní navigační přijímač*.

Několikakanálový navigační přijímač patří z hlediska funkčních parametrů k ideálnímu řešení. Tvoří jej vysokofrekvenční jednotka a alespoň čtyři shodné měřící přijímače, které kontinuálně měří každou družici zvlášť. Přijímače mohou využívat k měření stejnou časovou základnu, proto odpadají problémy s manévrujícími uživateli, u kterých se neprojeví odchylka časového normálu uživatele.

Přijímač, který neprovádí měření současně, nazýváme *sekvenčně měřící* navigační přijímač. Měření probíhá postupně v krátkých časových úsecích pro každou viditelnou družici. Z důvodu dlouhé doby potřebné k příjmu dat se konstruují dva kanály. První kanál provádí sekvenční měření a druhý, který je také vybaven měřícím přijímačem zabezpečuje vyhledávání signálu, příjem a aktualizaci dat či předběžné měření družice vyhodnocující vhodnost přepojení do měřícího kanálu.

Aby se nám podařilo přijmout data od všech družic do jednoho kanálu, musíme zkrátit trvání měřícího cyklu na dobu kratší než je trvání jednoho bitu. Tento typ přijímače nazýváme *multiplexní*. Přijímač zpracovává měření podobným způsobem jako *několikakanálový* přijímač. Z důvodu menší energie signálu ($1/\text{počet družic}$) dochází k větší nepřesnosti a chybám měření.

2.3. Komunikační rozhraní

K obsluze periférií využívá mikrokontrolér komunikační rozhraní. Tyto zařízení plní většinou funkce, kterými mikrokontrolér nedisponuje a jejichž integrace do mikrokontrolérů nemá význam, je drahá (Lab-on-a-chip) nebo je úplně nemožná. Jsou to přesné analogové digitální a digitálně analogové převodníky, akcelerometry, senzory a aktuátory, větší paměti atd.

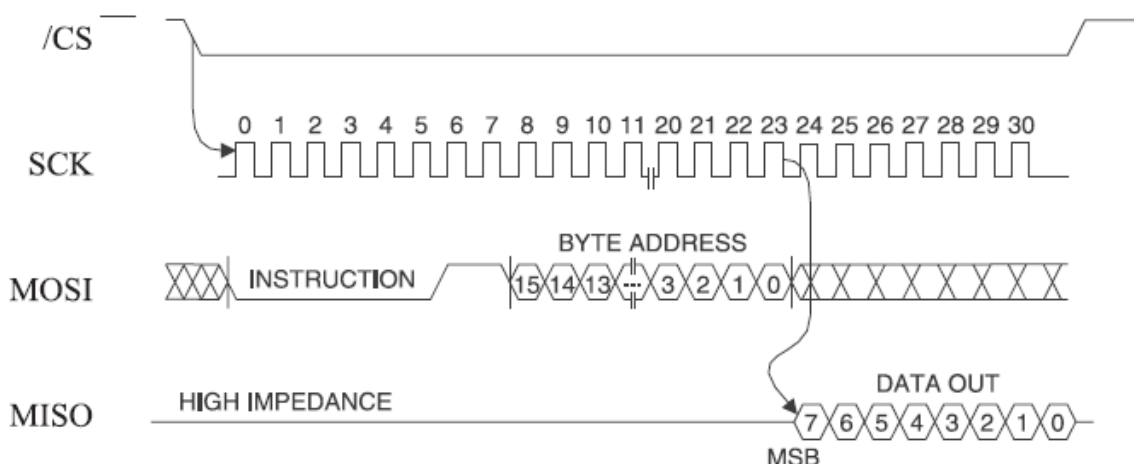
2.3.1. Serial Peripheral Interface (SPI)

Rozhraní SPI se používá pro synchronní sériovou komunikaci mezi mikrokontroléry a jejich perifériemi. Komunikace probíhá v plně duplexním režimu Master-Slave. Zařízení Master, které je reprezentováno většinou mikrokontrolérem generuje hodiny a vybírá Slave zařízení, s kterými se chystá komunikovat. Slave zařízení představují externí paměti, řadiče displejů nebo AD a DA převodníky.[9]

Sběrnice využívá ke komunikaci čtyři vodiče:

- **MOSI** (Master Out Slave In) – datový výstup
- **MISO** (Master In Slave Out) – datový vstup
- **SCK** (Serial Clock) – hodinový signál
- **SS** (Slave Select) nebo **CS** (Chip Select) – aktivace Slave zařízení

Signály MOSI, MISO a SCK jsou společné pro všechna zařízení a SS signál má každé Slave zařízení svůj. Neaktivní Slave zařízení mají jejich datové výstupy (MISO) v režimu vysoké impedance a jejich SS je v úrovni log. 1. Slave se aktivuje přepnutím signálu SS z log. 1 na log. 0 následně Master začne generovat hodinový signál a vysílat či přijímat data. Data jsou zpracovávána posuvnými registry, ve kterých se jeden datový bit detekuje při přechodu hodinového signálu z log. 0 na log. 1. Komunikace se ukončí přepnutím SS signálu z log. 0 do log. 1.



Obrázek 2-5: Čtení dat z paměti s rozhraním SPI [9]

Kapitola 2: Teoretický rozbor

SPI pracuje většinou v 8bitovém nebo 16bitovém režimu vysílání a přijímání dat a frekvence hodinového signálu se pohybuje v řádech kHz i MHz. SPI moduly integrované v mikrokontrolérech se dají nastavit do pracovních režimů žádaných uživatelem. Slave select nebývá hardwarově implementován do SPI modulu v režimu Master (řeší se softwarově), ale můžeme nastavit klidovou hodnotu hodinového signálu, jeho frekvenci nebo fázový posun vůči datovým signálům, a tím i okamžik zpracování jednotlivých bitů.

2.3.2. Inter Integrated Circuit (I²C)

Sběrnice I²C je na rozdíl od SPI typu Multimaster, což znamená, že každé zařízení může pracovat jako Master [9]. Každé připojené zařízení má jedinečnou 7bitovou nebo 10bitovou adresu, dokážeme tedy připojit až 1024 zařízení ke sběrnici. I²C disponuje funkcemi detekcí kolize na sběrnici a arbitrací schopnou předcházet iniciování komunikace dvěma či více Master stanicemi najednou. Používá se k obsluze periférií komunikujících s mikrokontrolérem a její dostatečná propracovanost umožňuje použití i v průmyslu.

2.3.2.1. Fyzická vrstva

Rozhraní je reprezentováno dvěma signály:

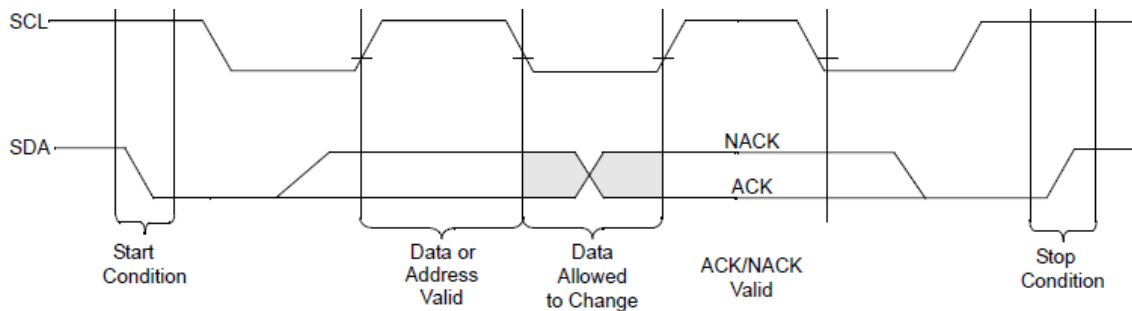
- SDA (Seriól Data) – datový signál
- SCL (Seriól Clock) – hodinový signál

Jelikož jsou oba vodiče typu otevřený kolektor, tak jsou k nim připojeny „pull-up“ rezistory o hodnotě 2 – 10 k Ω , které definují klidovou hodnotu v log. 1. Maximální délku vodičů limituje jejich nejvyšší přípustná kapacita 400 pF. Rozhraní operuje v mnoha rychlostních módech - od 100 kHz (Standart-mode) až po 5 MHz (Ultra Fast-mode). Hodinový signál je generován vždy Master stanicí.

Zařízení, které chce vysílat (Master), zahájí komunikaci Start podmínkou, kdy přepne SDA a následně SCL vodič do log. 0. Začne posílat data, která mohou měnit svou logickou úroveň pouze při SCL v log. 0. Při přepnutí SCL z log. 0 do log. 1 je datový bit zpracován. Po odeslání bytu je vyslán požadavek na Acknowledgement (ACK) nebo Not-

Kapitola 2: Teoretický rozbor

Acknowledgement (NACK), při kterém vysílající stanice uvolní SDA signál (log. 1) a přijímající stanice přepne SDA signál do log. 0 v případě úspěšného přijetí bytu. Pokud zůstane signál v log. 1, signalizuje to chybu (při zápisu) nebo požadavek na ukončení komunikace (při čtení). Následuje Stop podmínka, při které se uvolní nejprve SCL signál, poté SDA signál (log. 1) a uvolní se celá sběrnice pro komunikaci mezi ostatními stanicemi.



Obrázek 2-6: Průběh komunikace v I²C [9]

Speciální jev u I²C je Opakovaná Start podmínka (Repeated Start condition), která během přenosu znovu iniciuje komunikaci z důvodu např. požadavku na čtení z paměti, kdy v předchozí sekvenci se zapisoval požadavek na čtení z konkrétní adresy a v následující sekvenci bude zařízení data číst. Identifikátor zápisu/čtení se nachází na 8. bitu adresovacího bytu (7bitová adresa), log. 0 představuje zápis, log. 1 čtení.

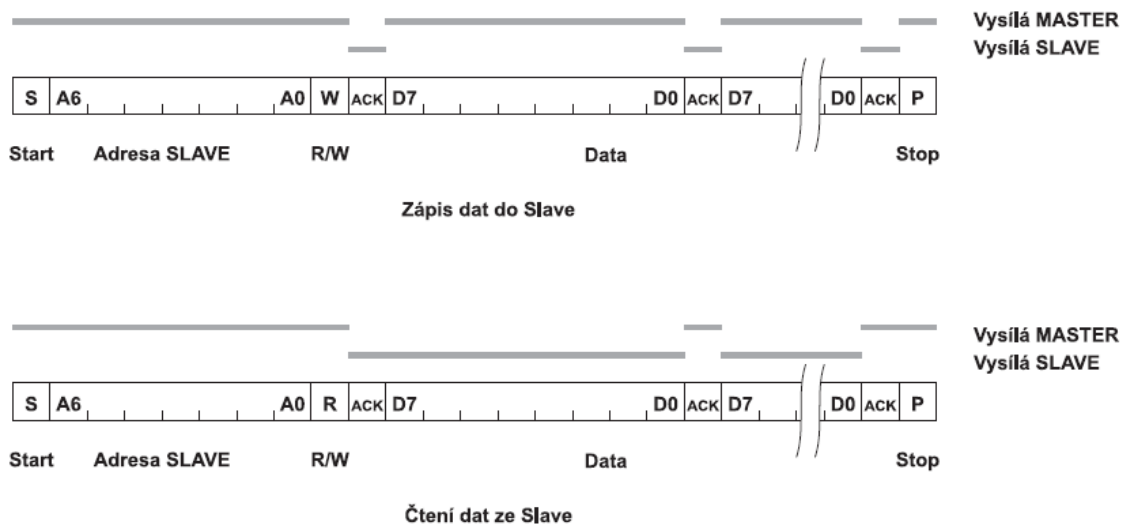
2.3.2.2. Linková vrstva

Rozhraní používá ve většině případů 7bitovou adresu, formát rámce při této délce adresy vidíte na obrázku 2-7. Datová sekvence, jak již bylo předesláno, začíná Start podmínkou následována adresou, Read/Write bitem a ACKem stanice, která je adresována. Následuje zápis/čtení bytů. Každý byte je potvrzen ACKem (nebo odmítnut NACKem) přijímající stanicí až do generování Stop podmínky, která uvolní komunikaci na sběrnici.

I²C využívá k arbitraci metodu s detekcí kolize. Jelikož sběrnice používá otevřené kolektory ke zjištění úrovně na sběrnici, může dojít ke kolizi, když jedna stanice vysílá log. 0 a druhá log. 1. Stanice mohou zahájit vysílání pouze v případě, že se sběrnice nachází v klidovém stavu. Zařízení, které vysílá, musí porovnávat vysílané bity se skutečnou úrovní

Kapitola 2: Teoretický rozbor

sběrnice (SDA). Pokud zařízení zjistí rozdíl, znamená to, že vysílají dvě a více stanic současně a musí vysílání okamžitě ukončit a uvolnit sběrnici.



Obrázek 2-7: Rámce rozhraní I²C[9]

Každá stanice v I²C má přidělenou jedinečnou adresu. Vysílající stanice generuje Start podmínku s následnou adresou do všech zařízení připojených ke sběrnici. Cílová stanice poté potvrdí přijetí ACK bitem, ostatní zůstávají v klidovém stavu (odmítnou NACK bitem). U periférií se nepoužívají k nastavení adresy všechny bity, ale mají některé továrně přednastaveny, zbylé definujeme připojením požadované logické úrovně k pinům, jež představují jednotlivé adresové bity. Sběrnice si vyhrazuje několik adres pro zvláštní účely. Adresa 0000000 se používá k vysílání do všech stanic (general call) připojených ke sběrnici, dále jsou rezervovány adresy 0000011, 000001XX a 11111XX.

2.3.3. RS-232 (UART)

Modul univerzálně asynchronního přijímače/vysílače (Universal Asynchronous Receiver/Transmitter - UART) se používá ke zpracování dat komunikačního protokolu RS-232, RS-485 (Recommended Standart), LIN a IrDa v napěťové logice TTL (a podobných logik s nižším napájecím napětím). Jelikož v autoalarmu budu využívat RS-232, bude tato kapitola pojednávat výhradně o tomto komunikačním protokolu s využitím UART modulu.

Kapitola 2: Teoretický rozbor

Komunikace je umožněna pouze mezi dvěma stanicemi. Sběrnice pracuje v plně duplexním režimu. Rychlost přenosu se vyjadřuje v baudech za sekundu (Bd/s), což vyjadřuje modulační rychlost, která v případě RS-232 se rovná bitové rychlosti ($1 \text{ Bd/s} = 1 \text{ bit/s}$). Hodnoty základní rychlosti se odvíjí od násobku 150 s mocninou od nuly (150 Bd/s) do sedmi (19200 Bd/s) při základu dva. Vyšší rychlosti jsou násobky nejvyšší základní rychlosti. Můžeme dosáhnout rychlostí 115200 Bd/s a vyšších v závislosti na délce vodičů a jejich kapacitě, která by neměla přesáhnout 2500 pF .

Koncové zařízení označujeme zkratkou DTE (Data Terminal Equipment), které převážně vysílá data a řídí komunikaci (PC, mikrokontrolér) a DCE (Data Communication Equipment) představující převážně vstupní zařízení, které bývá zastoupeno modemem nebo komunikačním modulem.

Signály používané v komunikaci po RS-232 jsou:

- TxD (Transmit Data) – vysílaná data
- RxD (Receive Data) – přijímaná data
- RTS (Request To Send)
- CTS (Clear To Send)
- DTR (Data Terminal Ready)
- DCD (Data Carrier Detect)
- DSR (Data Set Ready)
- RI (Ring Indicator)

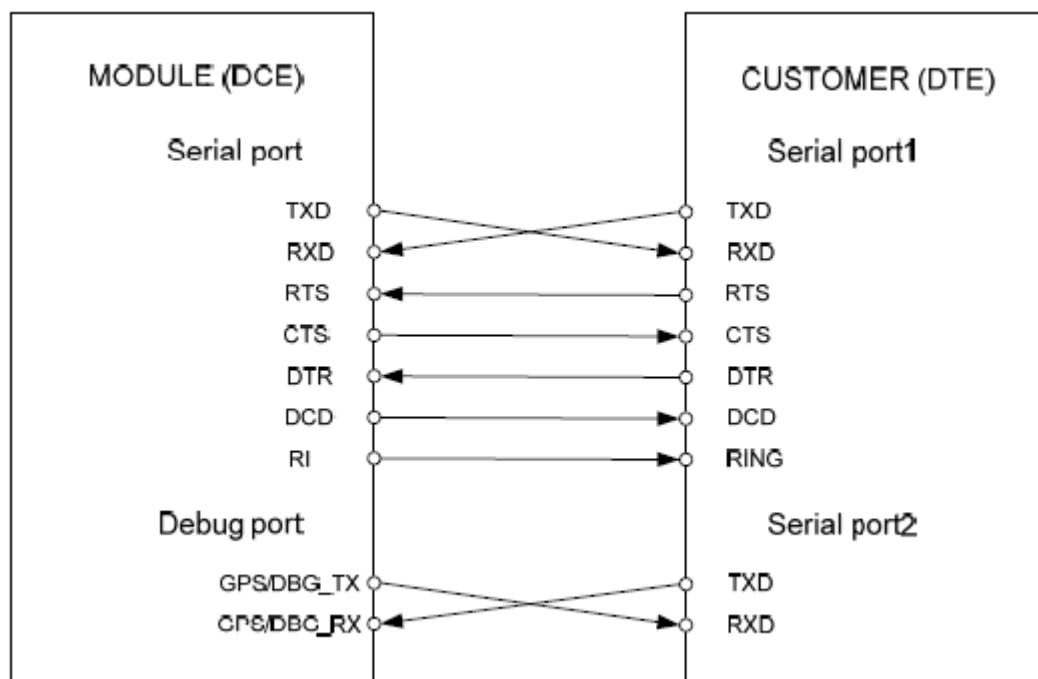
Signál RTS je nastaven do log. 1 zařízením DTE, které je připraveno vysílat data do DCE. Stanice DCE setrvává v přijímacím módu a musí potvrdit připravenost nastavením signálu CTS do log. 1. Signál DCD je nastaven do log. 1 DCE stanicí, pokud zrovna přijímá data od vzdálené DCE stanice. Signál DTR je nastaven do log. 1 stanicí DTE, když je připravena vysílat nebo přijímat data. Signál DSR indikuje, že je stanice připojena ke komunikační lince. Pokud chce stanice DCE zahájit komunikaci, nastaví RI signál do log. 1 [11].

Kapitola 2: Teoretický rozbor

Komunikace po RS-232 probíhá následovně:

1. DTE stanice monitoruje RI signál
2. DCE stanice nastaví RI do log. 1
3. DTE zachytí požadavek a akceptuje ho nastavením DTR do log. 1
4. Po zachycení signálu DTR, odpoví stanice DCE nastavením DSR do log. 1
5. DCE po zachycení externího signálu nastaví signál DCD do log. 1.
6. Přenos dat může začít signály TxD a RxD, jsou řízeny signály RTS a CTS
7. Na konci přenosu jsou signály DTR, DSR a DCD nastaveny do log. 0.

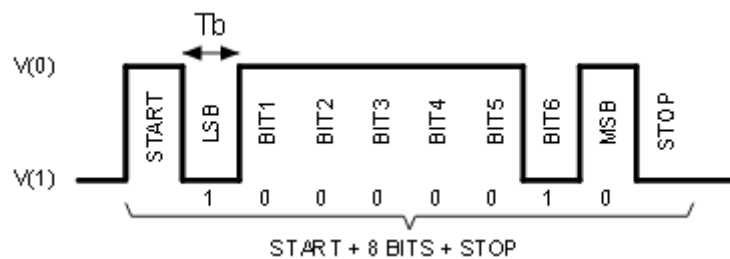
Pro mnoho aplikací se používají pouze signály RxD, TxD, CTS a RTS nebo jen RxD a TxD. Tyto signály jsou i většinou hardwarově implementovány v UART modulech mikrokontrolérů (minimum tedy TxD a RxD), a tím i zjednodušená implementace komunikace po RS-232. Pokud chceme obsluhovat zbylé signály v RS-232, musíme je zpracovávat softwarově mikrokontrolérem.



Obrázek 2-8 Zapojení modulu SIM908 s využitím RS-232 [19]

Kapitola 2: Teoretický rozbor

Datová sekvence začíná Start bitem, který je následován osmi nebo devíti bitovou sekvencí, která je vysílána od LSB (Least Significant Bit) k MSB (Most significant bit). Za datovými bity můžeme poslat pro kontrolu sudou nebo lichou paritu a sekvenci uzavřeme Stop bitem. Stanice DCE a DTE je potřeba konfigurovat na stejný režim datové sekvence a stejnou rychlost v případě, že stanice nedisponují funkcí „autobauding“, která automaticky rozpozná rychlost přijímaných dat.



Obrázek 2-9: Datové sekvence v RS-232 [12]

V UART modulech mikrokontrolérů můžeme nastavit vlastnosti datové sekvence, klidovou úroveň datových vodičů, počet Stop bitů (1-2) a potřebnou rychlost vysílání kompatibilní s možnostmi stanice DCE.

2.4. Controller Area Network (CAN)

Pro komunikaci mezi zařízeními v automobilu byla navržena firmou Bosch sériová sběrnice CAN. Pomocí této sběrnice můžeme zjistit informace o stavu motoru, světel, dveří či oken. Vysoká rychlost a lehká implementace jí přenesly i do dalších odvětví průmyslu. Sběrnice dosahuje rychlosti až 1 Mb/s a zařízení spolu komunikují v režimu multimaster. Komunikace probíhá po dvou vodičích (CAN_H a CAN_L) zakončených na obou koncích sběrnice přizpůsobovacím vedením o impedanci 120 Ω .

Základní vlastností sběrnice je vysílání dat všech připojených zařízení bez ohledu na poptávku na vysílaná data. Všechna zařízení si jsou navzájem rovnocenná (peer to peer komunikace). Data jsou přenášena v tzv. rámcích, které se dělí na několik segmentů. Rámce vysílaných dat neobsahují adresy, nýbrž jen identifikátory, které informují, o jaká data se

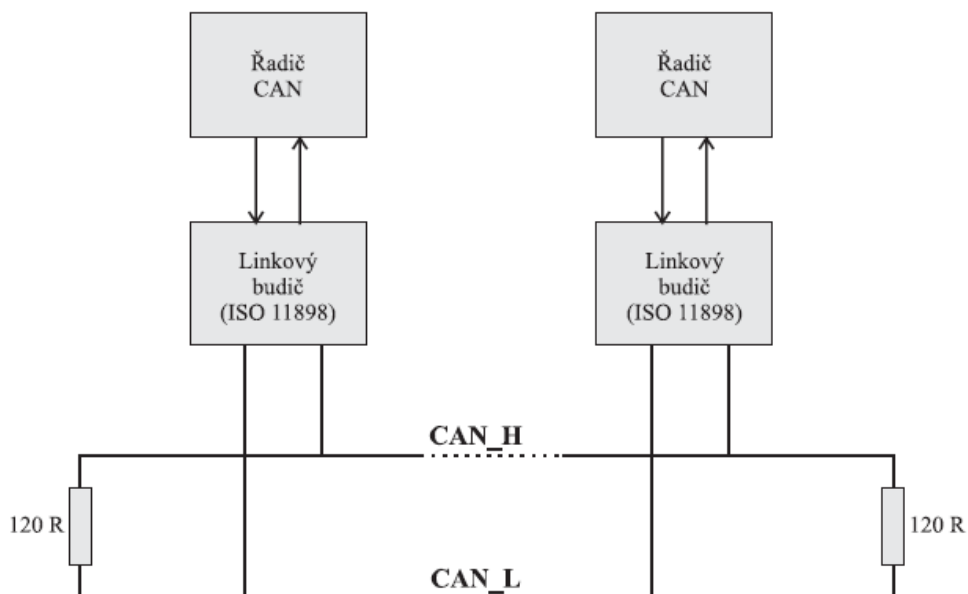
Kapitola 2: Teoretický rozbor

jedná. Každý vysílaný rámeček filtrují všechny přijímače a rozhodují se, zda daný rámeček vymazat nebo ho akceptovat a dále zpracovat. Detekuje-li jakékoli zařízení chybu při příjmu dat, je vyslán chybový rámeček a přenos se musí opakovat.

Specifikace pro sběrnici CAN zahrnuje norma ISO 11898. V současné době rozlišujeme dvě základní specifikace. CAN 2.0A využívá ke komunikaci standardní 11bitový identifikátor, zatímco CAN 2.0B komunikuje s rozšířeným 29bitovým identifikátorem rámeček. Existují další specifikace CAN 1.0, CAN 1.1 a CAN 1.2, které jsou sice starší, ale zato zpětně kompatibilní s novými standardy.[9], [13], [14] a [15]

2.4.1.Fyzická vrstva

Sběrnici CAN realizují dva datové vodiče, na kterých jsou vysílána stejná data, jen se liší napěťové úrovně. Rozlišujeme dvě logické hodnoty: dominantní a recesivní úroveň. Definice logické úrovně na sběrnici se odvíjí od tzv. „drátového součinu“, kde jsou paralelně připojeny tranzistory s otevřeným kolektorem a „pull-up“ rezistorem (v praxi se nepoužívá). Vysílá-li alespoň jedno zařízení dominantní úroveň, zatímco ostatní recesivní, je na sběrnici dominantní úroveň. Pokud všechna zařízení vysílají recesivní úroveň, je na sběrnici recesivní úroveň.



Obrázek 2-10: Propojení CAN sběrnice se stanicemi [9]

Kapitola 2: Teoretický rozbor

Pro zpracování CAN signálů jsou používány speciální linkové budiče, které jsou připojeny mezi sběrnici a CAN řadič. Linkový budič zpracovává data vodičů CAN_H a CAN_L a s řadičem komunikuje pomocí vysílacího (TX) a přijímacího (RX) vodiče.

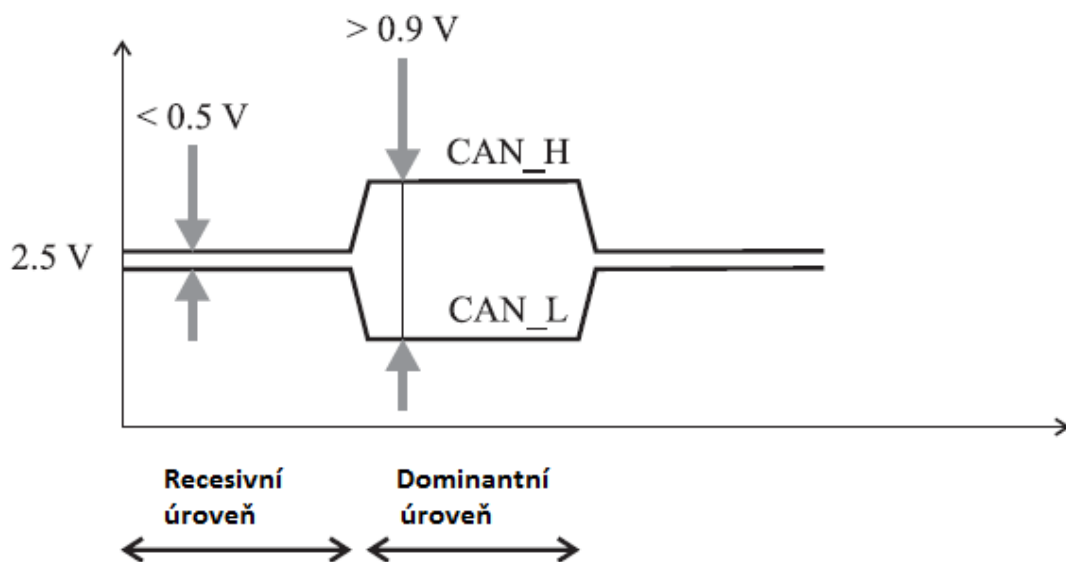
Recesivní úroveň definujeme vzorcem:

$$U_{CAN_H} \leq U_{CAN_L} + 0,5 V$$

Dominantní úroveň musí splňovat kritéria daná vzorcem:

$$U_{CAN_H} \geq U_{CAN_L} + 0,9 V$$

Pro vodiče CAN_H a CAN_L definujeme podle normy ISO klidové napětí 2,5 V.



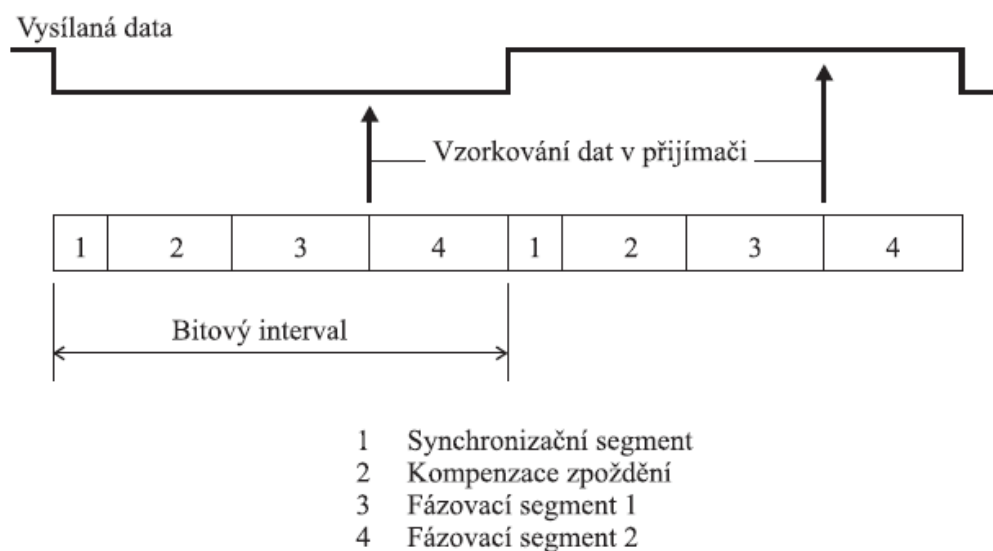
Obrázek 2-11: Dominantní a recesivní úroveň sběrnice CAN [9]

Sběrnice používá k přenosu dat kód NRZ (Non Return To Zero – bez návratu k nule). Jelikož se používají pouze datové vodiče, musí si každá stanice generovat hodinový signál sama a synchronizovat jej pomocí datového signálu vysílající stanice. Z důvodů nespolehlivé synchronizace hodin přijímače je v modulaci zahrnuta metoda vkládání bitů, která po 5bitové sekvenci stejné hodnoty vloží jeden bit opačné hodnoty.

Pro určení frekvence hodinového signálu dělíme každý bit na čtyři segmenty. Data se vysílají na začátku 1. segmentu a na přijímací straně se vzorkují na konci 3. segmentu.

Kapitola 2: Teoretický rozbor

Pokud se změní datový signál mimo konec 4. segmentu, opakuje se vzorkování na dalším konci 3. segmentu a postupně se doladuje frekvence hodinového signálu.



Obrázek 2-12: Rozdělení bitu pro určení frekvence hodinového signálu [9]

Přenosová rychlost se odvíjí od vzdálenosti jednotlivých zařízení. Maximální rychlosti 1 Mb/s jsme schopni dosáhnout při vzdálenosti menší než 40 m, naopak při vzdálenost větší než 5200 m nemůžeme přesáhnout 10 kb/s.

2.4.2.Linková vrstva

Linková vrstva zajišťuje adresaci, arbitraci, zabezpečení a reaguje na chybové stavy. Linkovou vrstvu dělíme na dvě podvrstvy:

- MAC (Medium Access Control) – reprezentuje jádro CAN protokolu. Zajišťuje přístup k médiu, potvrzování, detekci chyb, kanálové kódování a potvrzování správně přijatých rámců.
- LLC (Logical Link Control) – filtruje přijaté zprávy potvrzené MAC vrstvou, ohlašuje přetížení stanice a zajišťuje zotavení

Kapitola 2: Teoretický rozbor

Komunikaci po CAN sběrnici realizujeme pomocí rámců. Dělíme je podle funkcí na čtyři typy:

- Datový rámeček (Data Frame)
- Žádost o data (Remote Frame)
- Indikátor přetížení (Overload Frame)
- Chybový rámeček (Error Frame)

2.4.2.1. Datový rámeček

V závislosti na specifikaci sběrnice dělíme rámečky na standardní s 11bitovým identifikátorem nebo rozšířené s 29bitovým identifikátorem. Datové rámečky dělíme na sedm polí, které mají následující vlastnosti:

1. Komunikaci zahajujeme Start bitem (SOF – Start Of Frame), který je v dominantní úrovni.
2. Arbitrační pole obsahuje identifikátor zprávy (11 nebo 29 bitů) a RTR bit (Remote Transmission Request), který musí mít v datovém rámečku dominantní úroveň a v žádosti o data recesivní úroveň
3. V řídicím poli se nachází IDE bit, r0 a r1 bity a čtyři bity DLC (Data Length Code). IDE bit rozlišuje délku zprávy mezi standardní rámeček, v kterém je IDE v dominantní úrovni a rozšířený rámeček s recesivní úrovní. Bity r0 a r1 (r1 se vyskytuje pouze u rozšířeného rámečku) jsou rezervovány pro budoucí využití a bity DLC definují počet datových bytů.
4. Datové pole obsahující 0 až 8 bytů, které jsou vysílány od MSB.
5. CRC pole je dlouhé 15 bitů + 1 zakončovací bit (CRC Delimiter) a slouží k zabezpečení komunikace a minimalizaci chyb v přenosu. Zabezpečovací CRC kód se počítá z polynomu $x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + 1$ a zahrnuje všechny předcházející bity. CRC delimiter setrvává v recesivní úrovni.
6. Potvrzovací pole tvoří ACK a ACK Delimiter (zakončovací) bity. Bity se vysílají v recesivní úrovni a přijímací stanice potvrdí správnost přijatých dat vysláním ACK bitu v dominantní úrovni. Ke správnému potvrzení musí být ACK dominantní bit

Kapitola 2: Teoretický rozbor

obklopen recesivními CRC Delimiter a ACK delimiter bity. Vysílací zařízení zjistí, že nejméně jedno zařízení správně přijalo rámeček.

7. Konec rámce (EOF – End Of Frame) obsahuje sedm bitů v recesivní úrovni.

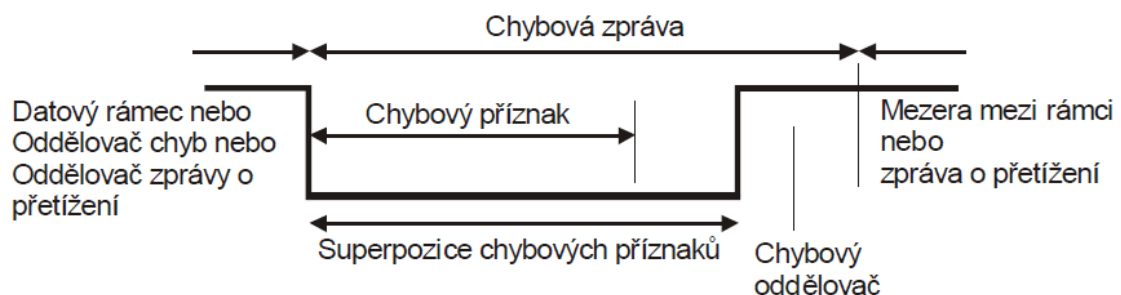
Mezi rámci se musí vytvořit interval (Intermission) o minimální délce tří bitů v recesivní úrovni.

2.4.2.2. Rámeček žádosti o data

Pokud má stanice na sběrnici zájem o určitá data, vysílá rámeček žádosti o data. Rámeček žádosti o data obsahuje všechna pole obsažená v datovém rámci až na datové pole. RTR bit se nachází v recesivní úrovni. Identifikátor rámce se shoduje s identifikátorem žádaného datového rámce. Žádaný datový rámeček má vždy vyšší prioritu než rámeček žádosti se stejným identifikátorem.

2.4.2.3. Chybový rámeček

Při zjištění chyby vysílá dotyčná stanice chybový rámeček. Nejprve se vysílá chybový příznak o délce šesti bitů, které definují buď aktivní chybový rámeček (6bitový interval v dominantní úrovni) nebo pasivní chybový rámeček (interval v recesivní úrovni). Aktivní chybový rámeček vyvolá chybu i u ostatních stanic, které reagují také vysláním chybového rámce. Pasivní chybový rámeček vysílá stanice při detekci více než 126 chyb, proto je také vyslán v recesivní úrovni, aby neovlivňoval komunikaci na sběrnici.



Obrázek 2-13: Chybový rámeček [14]

Kapitola 2: Teoretický rozbor

Po odeslání chybového příznaku se čeká 0 až 6 bitů na dosažení recesivní úrovně sběrnice (v případě aktivního rámce na odeslání příznaku ostatních stanic) a poté se zakončuje sekvence 8bitovým intervalem v recesivní úrovni (Error delimiter).

2.4.2.4. Rámec přetížení

Pokud není zařízení schopno přijímat rámce, odešle rámec přetížení. Rámec má stejnou strukturu jako aktivní chybový rámec, ale může se odeslat pouze po ukončeném vysílání datového nebo chybového rámce.

2.4.2.5. Přístup k médiu

Přístupová metoda, kterou využívá sběrnice CAN, se označuje CSMA/CD+AMP (Carrier Sense Multiple Access with Collision Detection and Arbitration on Message Priority). Princip CSMA spočívá v čekání vysílající stanice určitý časový interval, aby mohla proběhnout případná komunikace na sběrnici, teprve poté může stanice vysílat data. Funkce CD nám umožní zjistit kolizi s další vysílací stanicí. Stanice chybu indikuje, pokud vysílá recesivní úroveň a další zařízení dominantní úroveň, poté musí okamžitě ukončit vysílání dat a okamžitě přejít do přijímacího režimu. Funkce AMP upřednostní zprávy s přednastavenou vyšší prioritou identifikátoru rámce.[17]

2.4.2.6. Zabezpečení dat, detekce chyb a chybové stavy

Sběrnici CAN lze považovat za velmi spolehlivou k přenosu dat. K zabezpečení se používají následující funkce:

- CRC zabezpečení – pokud se neshoduje odeslaný a přijímaný CRC kód, vysílá se chybový rámec
- Monitoring – vysílač porovnává vysílané a zpětně přijaté bity, pokud se neshodují, přestane vysílat v případě dominantního stavu sběrnice při vysílání recesivním stavu. V ostatních případech vysílá chybový rámec
- Vkládání bitů – při vysílání v kódování NRZ se po pěti stejných bitech vkládá bit opačné hodnoty, při příjmu je automaticky tento bit ignorován

Kapitola 2: Teoretický rozbor

- Kontrola formátu zprávy – bity, které mají předem definovanou hodnotu (CRC a ACK delimiter, konec rámce) a jakákoli stanice detekuje jejich opačnou hodnotu, vysílá chybový rámeček
- Potvrzení příjmu zprávy – stanice potvrdí příjem rámce dominantní úrovní ACK bitu, v opačném případě se vysílá chybový rámeček a vysílání se opakuje

V každém zařízení jsou implementovány chybové čítače TEC (Transmit Error Count) a REC (Receive Error Count), které se inkrementují detekcí chyby při příjmu (REC) nebo při vysílání (TEC). Zároveň se tyto čítače dekrementují při správném příjmu či vysílání. Podle hodnot čítačů se jednotlivé stanice nacházejí v následujících chybových stavech:

- Aktivní – každý z čítačů musí mít hodnotu menší než 127, při detekci chyby se vysílá aktivní chybový rámeček, přerušuje se vysílání na sběrnici a inkrementuje se hodnota čítače TEC nebo REC v závislosti na situaci (0,1 nebo 8 jednotek)
- Pasivní – jeden z čítačů má hodnotu vyšší než 126, při detekci chyby se generuje pasivní chybový rámeček, který nenarušuje komunikaci na sběrnici, při úspěšné komunikaci se čítače dekrementují a zařízení se vrací do aktivního stavu
- Odpojen (Bus-off) – čítač TEC má vyšší hodnotu než 255, dotyčné zařízení je odpojeno a návrat do předchozích stavů je možný pouze programově nebo resetem řadiče

3. Analýza trhu

Před samotným návrhem je důležité si stanovit, co má být výsledkem práce, jaké by bylo budoucí uplatnění, a které komponenty se dají v dnešní době sehnat. Současně musíme zjistit, jaké podobné výrobky se na trhu nachází a zvolit takové funkce, s kterými by alarm byl odlišný od ostatních. A nejen odlišný, ale aby měl lepší parametry, uživatelský přístup či nižší cenu.

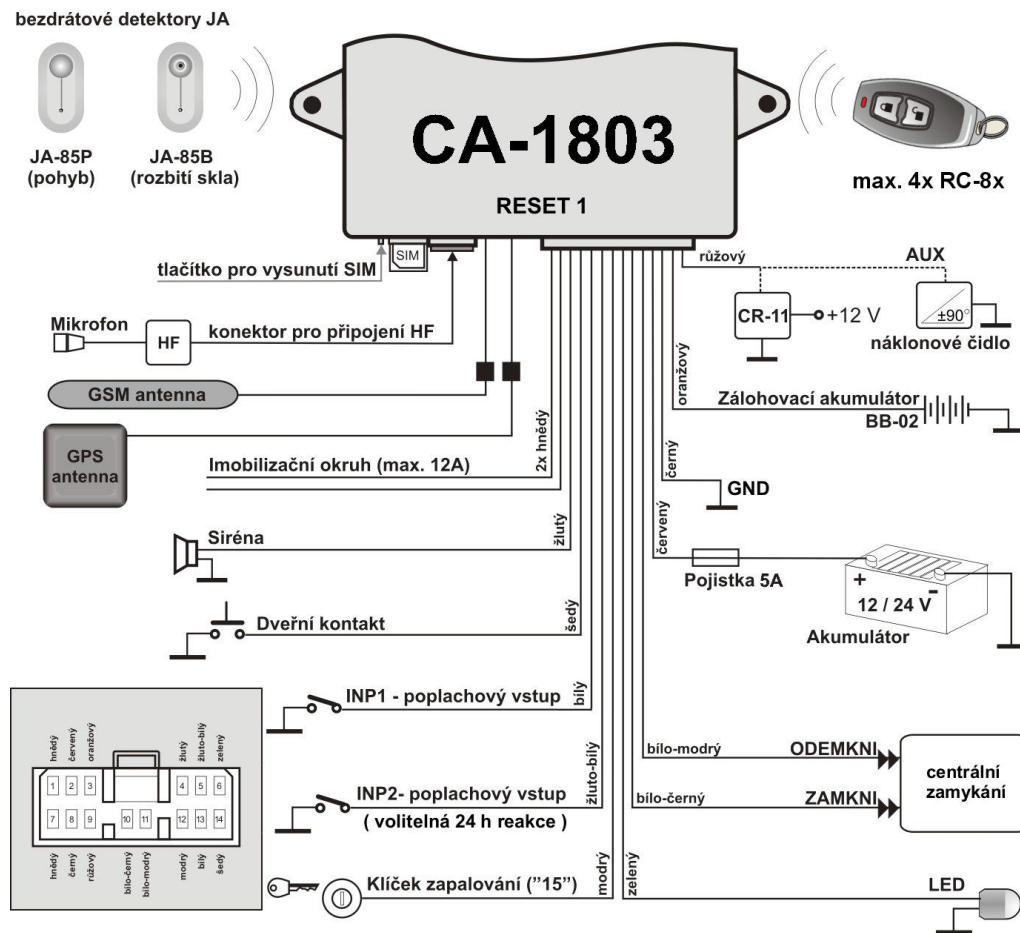
3.1. Konkurenční modely

Na českém trhu působí mnoho výrobců autoalarmů, z nichž nejvýznamnější je firma Jablotron, která ke svým produktům nabízí i služby servisu a instalace zařízení. Dále za zmínku stojí zahraniční firmy Keetec, Mistral a Spy.

3.1.1. Jablotron CA-1803B

Jedná se o nejvybavenější alarm od Jablotronu. Kombinuje v sobě funkce zabezpečení automobilu, GSM komunikaci, monitoring provozu přes GPS a imobilizér. Dá se nastavit pomocí SMS zpráv i prozvoněním. Alarm při plném využití dokáže vyhodnotit otevření dveří a kufříku, klíček v zapalování a zapnutí spotřebiče v autě (poklesem napětí).[1]

Na obr. 3-1 můžete vidět zapojení autoalarmu CA-1803BT ke zbývajícím komponentám a vozidlu. Součástí autoalarmu je GSM a GPS anténa, mikrofon, siréna, signalizační LED, záložní akumulátor, náklonové čidlo a vodiče pro propojení alarmu s centrálním zamykáním (verze s originálním nebo dodaným dálkovým ovládačem), akumulátorem, zapalováním, dveřními kontakty a extra poplachovými vstupy.



Obrázek 3-1: Připojení vstupů a výstupů autoalarm CA-1803BT [1]

Alarm disponuje solidní výbavou s možností si nastavit pomocí SMS jednotlivé funkce nebo se přihlásit na webovou aplikaci, kde můžeme intuitivně měnit vlastnosti přes internet. Zajímavostí jsou bezdrátové detektory pohybu a rozbitého skla. Alarmu ale chybí připojení na CAN sběrnici, které lze řešit dokoupením přídatného modulu. Firma dále nabízí službu střežení, s kterou firemní operátoři okamžitě informují při mimořádné události. Klidový odběr autoalarmu je podle výrobce max. 20 mA a uváděný napájecí rozsah je 8 až 32 V.

3.1.2. Keetec T GSM

Slovenska firma Keetec vyrobila autoalarm, který obsahuje přímé propojení na CAN sběrnici a GSM komunikací. Bohužel chybí lokalizace GPS [2]. Alarm nabízí

Kapitola 3: Analýza trhu

podobné funkce jako CT-1803BT, ale už ne v množství jako u CT-1803BT. Nenalezneme například náklonové čidlo nebo bezkontaktní senzor pohybu.

Kromě napojení na CAN sběrnici nabízí alarm také vstupy ze zapalování, kontaktů dveří a kapoty nebo výstupy pro odemykání a zamykání, sirénu či immobilizér. Autoalarm je schopen poslat varování až na 4 telefonní čísla. Udávaný klidový odběr je 30 mA a rozsah napájecího napětí 8 až 16 V.

3.1.3. Spy 10 PKE

Dalším zástupcem autoalarmů je Spy 10 PKE od firmy Spy. Systém obsahuje podobné vlastnosti, jako předešlé alarmy. Najdeme u něj náklonové čidlo, možnost připojení na různá místa v autě a vlastní ovladače pro odemykání a zamykání auta [4]. Alarm disponuje funkcí PKE (Passive keyless entry), která snímá polohu uživatele (resp. dálkového ovládače) a podle toho můžeme nastavit vlastnosti alarmu. Alarm tedy dokáže auto při vzdálení majitele (ovládače) automaticky zamknout.



Obrázek 3-2: Autoalarm Spy 10 PKE od firmy Spy

V technických specifikacích nalezneme klidový odběr proudu do 30 mA a rozsah napětí od 9 do 18 V. Tento systém ovšem nedisponuje komunikačním systémem GSM ani

polohovým systémem GPS. Připojením na CAN sběrnici rovněž nedisponuje. Výhodou je bezesporu velmi nízká cena.

3.2. Předběžný návrh řešení

Cílem této kapitoly je se zamyslet nad možným řešením v závislosti na již uvedených modelech na trhu. Nejlepším autoalarmem z uvedených byl bezpochyby CA-1803BT. Disponuje systémy GSM i GPS, několika senzory a množstvím vstupů a výstupů. Autoalarm T GSM sice nemá tolik funkcí (nemá GPS), ale lze ho připojit na CAN sběrnici. U Spy 10 PKE můžeme vyzdvihnout akorát nízkou cenu, která odpovídá výbavě (bez GSM i GPS). Podle parametrů těchto alarmů jsem volil další postup, který by měl skončit vytvořením autoalarmu s výbornými parametry a zároveň by měl šanci se uchytit, jako konkurence schopný výrobek.

V mé práci proto najdete postup návrhu autoalarmu, který v sobě integruje:

- komunikační systémy GSM, GPRS a GPS
- automobilovou sběrnici CAN, z které bude vyčítat informace
- akcelerometr, schopný zaznamenat otřesy či pohyb vozidla
- vstupy/výstupy z/do vozidla pro zjištění stavu dveří, klíče v zapalování, atd.

3.3. Výběr komponent

Pro aplikaci musíme vybrat vhodné komponenty. Mezi hlavní komponenty patří mikrokontrolér, GSM a GPS modul, napájecí obvody, převodníky signálu a konektory. Mezi hlavní aspekty výběru jsem řadil funkčnost, dostupnost, cenu, spotřebu a kompatibilitu.

3.3.1. Mikrokontrolér

Jádrem autoalarmu je mikrokontrolér, který zastává funkci řídicí jednotky. Vykonává instrukce podle uživatelského programu, reaguje na signály z okolí a jedná podle

Kapitola 3: Analýza trhu

nich. Moje kritéria pro výběr mikrokontroléru byla kompatibilita se sběrnými perifériemi, výrobce, dostupnost a cena. Výkon je u dnešních mikrokontrolérů dostatečný.

Pro komunikaci s perifériemi je nutná kompatibilita mikrokontroléru se sběrnými SPI a I2C a CAN. Tři jednotky UART využiju v komunikaci s GSM modulem, GPS modulem a s počítačem. V mé praxi jsem se setkal s mikrokontroléry od společností Atmel a Microchip, ale i s mikrokontroléry s jádrem 8051 nebo ARM (ADUC od Analog Devices), které jsem však pro nulovou podporu ze strany výrobce zavrhl a jelikož vlastním programovací a debugovací zařízení pro mikrokontroléry od firmy Microchip, tak jsem zavrhl mikrokontroléry Atmel a dále jsem vybíral jen od Microchipu.

Vyhýbal jsem se mikrokontrolérům s pouzdrem BGA, které bych složitě pájel. Důležitý je dostatečný počet pinů, kde jsem volil mezi 80 až 100 piny. Pro mou aplikaci jsem našel vhodné mikrokontroléry v řadách PIC24 a dsPIC33.

3.3.2. GSM a GPS modul

Pro návrh autoalarmu potřebuju komponenty, které budou zasílat SMS s udávanou pozicí nebo posílat data přes internet o stavu zařízení či pozici. Proto jsem se zaměřil na GSM a GPS moduly, které dokážou tyto funkce vykonávat a zároveň komunikovat s mikrokontrolérem, který informace dále zpracuje.

Na trhu existují moduly, které jsou integrovány v jednom pouzdře nebo jsou oddělené. Zvolil jsem první možnost z důvodu úspory místa na desce a jednoho komunikačního rozhraní s mikrokontrolérem a začal zjišťovat technické parametry jednotlivých modulů. Pro návrh byla důležitá funkce SMS, GPRS a eventuálně zpracovat hovor pro možný odposlech. Nalezl jsem pouze 2 kombinované GSM/GPS moduly (nepočítám nižší řady modulů) a jelikož se cenově velmi lišily při podobných vlastnostech, tak jsem vybral levnější SIM908 od firmy SIMCOM.

| Moduly | Přesnost [m] | UART | I ² C | Cena [Kč] |
|-----------------|--------------|------|------------------|-----------|
| TELIT GM862-GPS | <2,5 | 2 | 0 | 3179 |
| SIMCOM SIM908 | 2,5 | 2 | 1 | 620 |

Tabulka 3-1: Vybrané GSM/GPS moduly

Kapitola 3: Analýza trhu

SIM908 umožňuje používat plnohodnotně služby GSM, GPRS a GPS, disponuje přístupovými rozhraními UART a I²C a dalšími funkcemi, které by mohli zčásti nahradit řídicí jednotku – rozhraní pro LCD displej, GPIO piny, PWM a AD převodníkem.

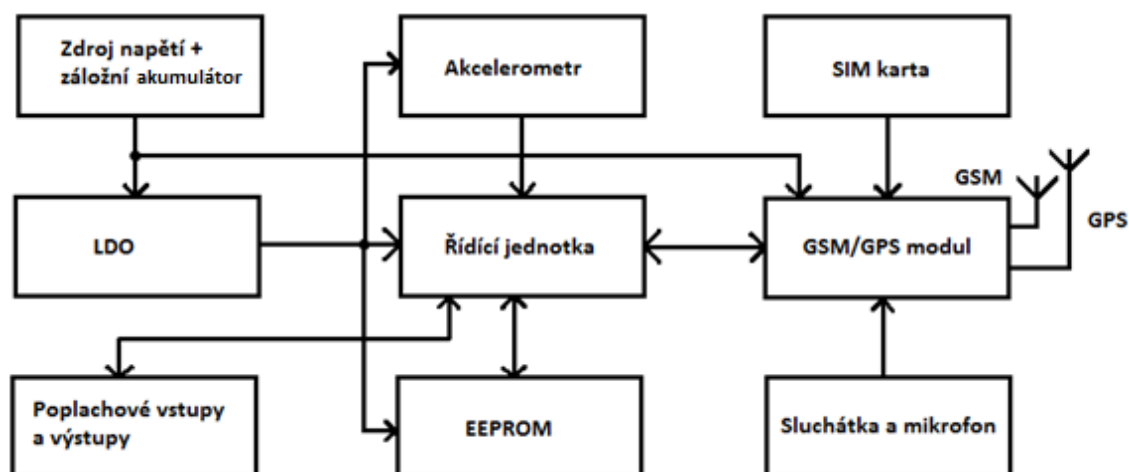


Obrázek 3-3: SIM908 od firmy SIMCOM

4. Návrh hardware

Technické řešení této práce spočívá v realizaci autoalarmu, který bude schopen přijímat SMS zprávy s žádostí o aktuální pozici vozu, trasování či nastavení funkcí zařízení. Dále se předpokládá rychlá reakce při narušení bezpečnosti vozu a následné oznámení mimořádné události majiteli vozu. Majitel si poté vyžádá GPS souřadnice vozu (pokud mu nebudou okamžitě poslány), které mohou být využity policií při hledání vozidla. K dopadení pachatele lze využít i případný odposlech či odpojení imobilizačního okruhu. Tato kapitola bude pojednávat o hardwarové části technického řešení.

Samotnému návrhu desky plošných spojů v autoalarmu předchází schéma, které rozdělíme do funkčních bloků a popíšeme v jednotlivých kapitolách. Ovšem předtím než jsem se začal věnovat jednotlivým blokům, musel jsem si zjistit rozměry součástek a konektorů, které jsem chtěl v autoalarmu použít a podle toho navrhnout i rozměr desky plošných spojů vyhovující velikostem dostupných krabiček. Proto jsem zvolil rozměr DPS 123 x 74 mm, která se vejde do krabičky KM-40 o rozměrech 131,5 x 82 x 35 mm.



Obrázek 4-1: Blokové schéma autoalarmu

4.1. Napájení

Vstupní napětí autoalarmu je 12 V, tudíž se připojí k akumulátoru automobilu. Na vstup jsem vložil ochranné prvky – varistor (18 V) vratnou proudovou pojistku PTC (2 A) a ochrannou diodu proti přepólování (2 A). Vstupním napětím jsou napájeny dva integrované obvody LT3652 a LT3430. Schéma naleznete v příloze.

4.1.1. Nabíjecí obvod

LT3652 slouží k nabíjení záložní Li-Ion akumulátor (7,2 V). Vstupní napětí může dosáhnout 40 V. Obvod umí hlídat teplotu akumulátoru pomocí termistoru NTC. Nabíjecí napětí V_{BAT} jsem zvolil 8,38 V, protože nabíjecí napětí by nemělo překročit 8,4 V. Voleného napětí docílíme výpočtem děliče napětí na V_{FB} pinu, který snímá napětí na baterii. Napětí na V_{FB} pinu po dosažení 3,3 V ukončí nabíjení a ve výsledném vzorci se promítá i rezistor ve vstupu pinu, který kompenzuje chybu proudového pracovního bodu. Hodnoty rezistorů vypočítáme podle následujících vzorců:

$$R_{FB1} = (U_{BAT} \times 2,5 \times 10^5) \div 3,3 = 635 \text{ k}\Omega$$

$$R_{FB2} = (R_{FB1} \times 2,5 \times 10^5) \div (R_{FB1} - (2,5 \times 10^5)) = 412 \text{ k}\Omega$$

Výsledné hodnoty jsou 635 k Ω pro R_{FB1} a 412 k Ω pro R_{FB2} . V konečném návrhu nebyl proud z děliče postačující, proto jsem R_{FB1} volil 77 k Ω a 50 k Ω pro R_{FB2} .

Nabíjecí proud jsem volil 100 mA, jelikož v této aplikaci není potřeba rychlého nabíjení a zároveň se vyhnu proudovým špičkám, které by mohly ohrozit funkčnost zařízení. Rezistor R_{SENSE} nastavuje velikost nabíjecího proudu podle vzorce:

$$R_{SENSE} = 0,1 \div I_{CHG} = 1 \Omega$$

Obvod pracuje ve 4 stavech, které signalizuje pomocí LED diod na pinech CHRG (zelená led) a FAULT (červená LED). ON stav připojí pin na GND a rozsvítí LED, v OFF stavu setrvává pin v režimu vysoké impedance. Obrázek 4-2 ukazuje pracovní režimy obvodu. Ochranu proti podbití akumulátoru řeším v následující kapitole.

| STATUS PINS STATE | | CHARGER STATUS |
|-------------------|-------|--|
| CHRG | FAULT | |
| OFF | OFF | Not Charging — Standby or Shutdown Mode |
| OFF | ON | Bad Battery Fault (Precondition Timeout / EOC Failure) |
| ON | OFF | Normal Charging at C/10 or Greater |
| ON | ON | NTC Fault (Pause) |

Obrázek 4-2: Pracovní režimy obvodu LT3652

4.1.2. Snižující měnič napětí

LT3430 plní funkci snižujícího měniče napětí (Step-down converter). Vstupní napětí obvodu je 12 V, ale může být až 60 V. Obvod pracuje na spínací frekvenci 100 kHz je schopen dodat proud o velikosti 3 A. Účinnost obvodu dosahuje účinnosti 92 % při vstupním napětí 12 V a výstupním proudem 1-2 A.

Výstupní napětí jsem volil 4,1 V pro GSM/GPS modul (dále pro CAN buffer a LDO), které jsem nastavil pomocí děliče napětí na FB pinu, jehož hodnoty rezistorů jsem spočítal podle vzorce:

$$R_1 = \frac{R_2(U_{OUT} - 1,22)}{1,22} \quad (\Omega)$$

R_2 jsem volil 10 k Ω a výsledná rezistence R_1 vyšla 23,6 k Ω . Nejbližší hodnota rezistoru R_1 na trhu byla 23,2 k Ω , což snížilo výsledné napětí na 4,07 V. Z doporučeného rozsahu hodnot výstupního induktoru (10-100 μ H) jsem zvolil 82 μ H induktor s maximálním proudem 2,4 A. Výstupní hodnotu kondenzátoru jsem volil doporučených 100 μ F.

Na pin SHDN jsem připojil dělič napětí, který při poklesu napájení pod 6 V vypne spínání obvodu a zároveň zdroj napětí pro celou DPS, aby nedošlo při odpojení hlavního zdroje napětí k podbití Li-Ion akumulátoru (7,4 V), což by ho mohlo zničit.

Kapitola 4: Návrh hardware

V návrhu DPS jsem navrhoval dostatečnou šířku výkonových cest (zároveň vzdálených od signálových cest) o šířce minimálně 1 mm při tloušťce spoje 35 μm s výjimkou nejbližšího okolí padů.

4.1.3. Lineární stabilizátor napětí

Pro potřeby řídicí jednotky, akcelerometru a EEPROM paměti potřebujeme snížit napětí na 3,3 V. Tento úkol splní lineární nízkoubytkový stabilizátor napětí. Důležité parametry stabilizátoru jsou výstupní proud a minimální pokles napětí, jelikož budeme snižovat jen ze 4,1 V na 3,3 V. Vybraný obvod MCP1825T disponuje fixním výstupním napětím 3,3 V, dostačujícím výstupním proudem 500 mA a maximálním vstupním napětím 6 V.

Z důvodu ochrany obvodu jsem zapojil mezi výstup a vstup diodu, která zabrání při vypnutí autoalarmu ke snížení vstupního napětí pod setrvávající výstupní napětí (vlivem nabitých kondenzátorů), což by mohlo vést ke zničení obvodu.

4.2. GSM/GPS modul

Modul SIM908 zajišťuje kombinaci funkcí GSM (s GPRS) a GPS. Modul komunikuje s mikrokontrolérem a vykonává jeho příkazy. V případě autoalarmu to bude zjištění pozice vozidla, informování majitele SMS zprávou, odeslání navigačních dat na internet přes GPRS (GPS tracker), či možný odposlech kabiny automobilu.

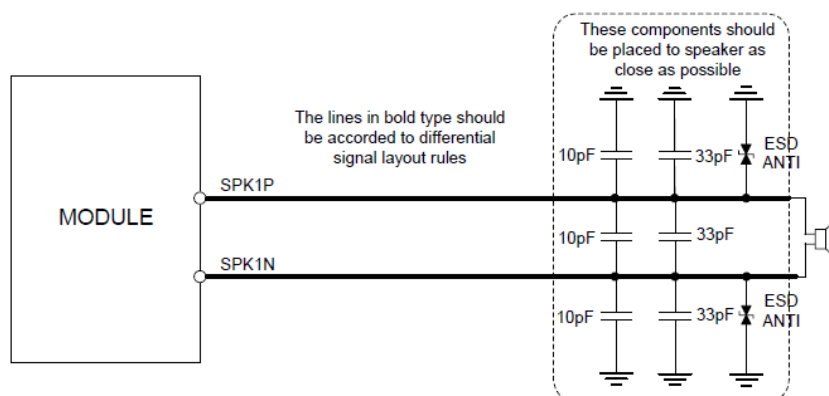
Modul je napájen napětím 4,1 V a je potřeba zajistit minimální šířku cesty napájení 2 mm při tloušťce spoje 35 μm . Kvůli proudovým špičkám dosahujících 2 A je vhodné co nejbližší k napájení modulu připojit kondenzátor s nízkou hodnotou ESR, jelikož při poklesu pod 3,1 V by se vypnul modul. V mém případě jsem připojil kondenzátor 1 mF. Měřil jsem napětí při připojování do sítě a uskutečnění telefonního hovoru a nezjistil jsem sebemenší pokles.

Kapitola 4: Návrh hardware

Modul se zapíná a vypíná sepnutím tranzistoru na pinu PWRKEY po nejméně 1 s. K zajištění chodu hodin reálného času i po vypnutí modulu jsem připojil kondenzátor 1 mF na VRTC pin.

K programování modulu se používá sériové rozhraní na bázi RS-232. Jelikož je maximální povolené napětí na pinech digitálních signálů 3,1 V, musel jsem z důvodu 3,3 V logiky řídicí jednotky oddělit piny děličem napětí, který snížil napětí na 2,7 V. K dispozici je druhá sériová linka (pouze signály RXD a TXD) pro periodické přijímání GPS zpráv.

K modulu jsem připojil dva konektory „minijack“ pro sluchátka a mikrofon. Autoalarm se poté bude dát využít jako odposlouchávací zařízení v automobilu nebo i jako případný komunikátor. Připojení audio vstupu i výstupu je totožné a zapojoval jsem je na základě obrázku 4-3.



Obrázek 4-3: Připojení audio výstupu modulu [19]

Připojení SIM karty jsem realizoval také podle katalogového listu SIM908 [19]. Použil jsem pouzdro s detekcí SIM karty, které umožní ověřit mikrokontrolérem přítomnost SIM karty.

K zjištění stavů modulu poslouží piny NETLIGHT, STATUS a programovatelné GPIO piny. STATUS pin indikuje zapnutí modulu. NETLIGHT pin signalizuje registraci v síti (blikání LED po 3 s), neúspěšnou registraci (blikání po 800 ms) a GPRS komunikaci (blikání po 300 ms). Na jednom GPIO jsem připojil LED pro debugovací účely.

Kapitola 4: Návrh hardware

GSM a GPS anténu připojuji pomocí SMA konektorů, které jsem umístil co nejbližší patřičným vstupům. K ochraně vysokofrekvenčních signálů před možným rušením a k impedančnímu přizpůsobení vstupů jsem podél signálových cest rovnoměrně a ve stejné vzdálenosti umístil na DPS viasy, které jsou připojené k GND.

4.3. Mikrokontrolér

Funkci řídicí jednotky celého autoalarmu zastupuje mikrokontrolér z řady dsPIC33E. Mikrokontrolér komunikuje s GSM/GPS modulem, zpracovává informace, které následně vyhodnocuje podle programu. Přijímaná data zapisuje do EEPROM paměti, ze které může následně číst – uplatní se, pokud chceme zaznamenat celou trasu automobilu, kterou si následně můžeme zobrazit v aplikaci na internetu (Google Maps). Akcelerometr, připojený k mikrokontroléru, může informovat mikrokontrolér o nežádoucím pohybu či otřesech vozidla, bez indikace na poplachových vstupech.

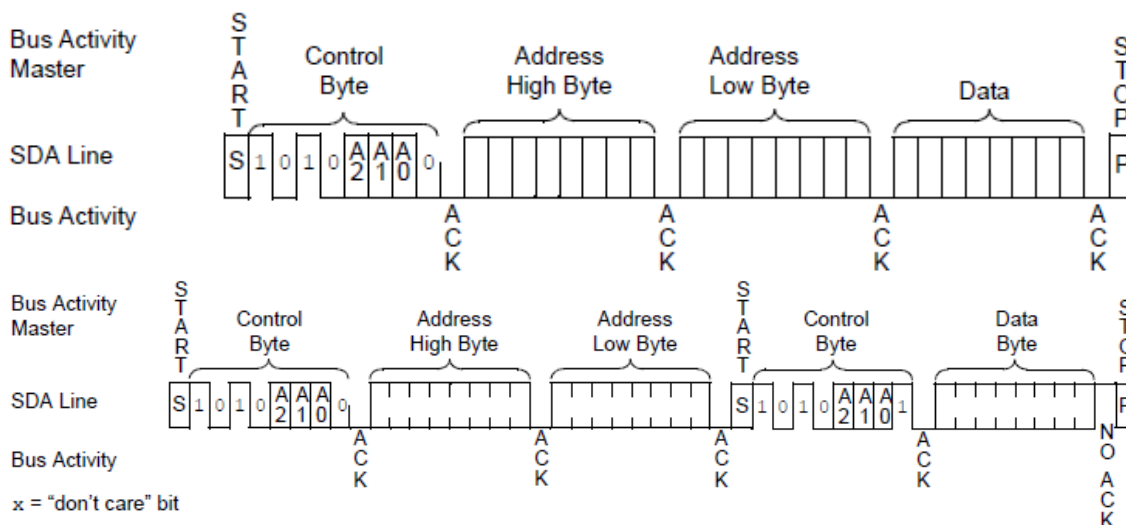
Mikrokontrolér má několik napájecích vstupů 3,3 V, které jsou napájeny z lineárního stabilizátoru napětí (LDO). Ke každému jsem připojil co nejbližší (pod mikrokontrolérem z druhé strany DPS) kondenzátor 100 nF pro vyhlazení případných záskmitů napájecího napětí. Zdroj hodinového signálu zajišťuje externí krystalový rezonátor o kmitočtu 20 MHz. Analogově digitálním převodníkem snímám přes dělič napětí úroveň napětí na autobaterii a záložním Li-Ion akumulátoru. Programování a debugování mikrokontroléru zajišťuje zařízení PICkit 3.

4.4. Akcelerometr

Akcelerometr MMA7455L upozorňuje mikrokontrolér o pohybu a otřesech vozidla. Komunikace probíhá po SPI nebo I²C rozhraní s dvěma indikátorovými piny událostí. Akcelerometr má nastavitelnou citlivost – dostupné jsou módy $\pm 2g$ (64 LSB/g), $\pm 4g$ a $\pm 8g$ v osách x, y a z. Zrychlení můžeme číst v 8 nebo 10bitových sekvencích. Zrychlení můžeme vyhodnocovat buď pulzně (otřes vozidla) nebo úrovní (rozjíždění a brzdění vozidla či jeho náklon). Napájení je 3,3 V.

4.5. Paměť EEPROM

Paměť EEPROM 24FC1024 od firmy Microchip poskytuje 1024 kbit přepisovatelné, elektricky mazatelné paměti. Přenos dat probíhá po sériové I²C sběrnici o maximální frekvenci hodinového signálu 1 MHz. Na oba vodiče sběrnice (SDA a SCL) jsou připojeny „pull-up“ rezistory o velikosti 3,3 kΩ k definování klidové hodnoty na sběrnici v log. 1. Výrobce deklaruje životnost dat více než 200 let a více než 1 milion zapisovacích cyklů na paměťovou buňku.



Obrázek 4-4: Nahoře - zápis bytu do paměti, dole - čtení bytu z paměti

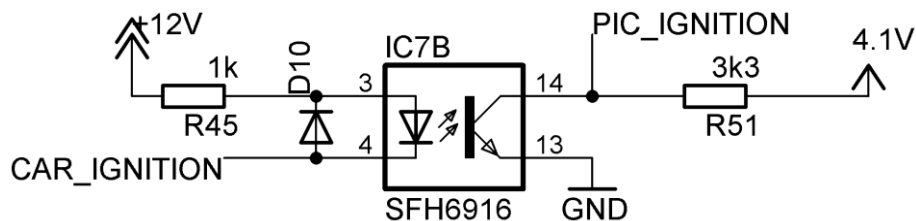
Paměť je napájena 3,3 V. Její adresové piny jsou připojeny k log. 0, což definuje adresu 1010000. Jelikož nebudu potřebovat ve své aplikaci ochranu proti zápisu, připojím Write-Protect pin trvale k log. 0. Do paměti lze zapisovat nebo z ní číst po jednotlivých bytech nebo bytových sekvencích. Přesun dat se odehrává, dokud nějaká stanice nevyšle NACK nebo neukončí spojení Stop podmínkou. Adresa bytů se automaticky inkrementuje.

4.6. Poplachové vstupy a výstupy

Tyto signály budou k dispozici pro koncového uživatele. Každý signál si uživatel zapojí na příslušné místo ve vozidle, aby mohl plnit autoalarm svoji funkci – tzn. zjišťování stavu určitých částí vozidla s patřičnou reakcí autoalarmu nebo signalizování uživatelských nastavení.

4.6.1. Poplachové vstupy

Uživatel připojuje tyto vstupy na místa ve vozidle, která chce kontrolovat. Jedná se především o kontakty dveří, světel a zapalování (detekce klíče). Z důvodu bezpečnosti autoalarmu proti vnějším vlivům jsem oddělil jednotlivé vstupy optočleny, které eliminují nebezpečné napět'ové překmity a náboje na vstupech.



Obrázek 4-5: Připojení poplachového vstupu k mikrokontroléru

Vstup optočlenu je napájen 12 V přes 1 k Ω rezistor, který omezuje proud diodou optočlenu. Dále jsem připojil paralelně v závěrném směru k optočlenu ochranou diodu, která eliminuje riziko průrazu diody optočlenu záporným napětím. Kontakt, který se nachází v log. 0 sepne výstupní tranzistor. Výstupní tranzistor je napájen ze snižujícího měniče přes rezistor 3,3 k Ω . V klidovém stavu je na vstupu mikrokontroléru log. 1 v sepnutém stavu log. 0.

4.6.2. Připojení CAN sběrnice

Automobilová sběrnice CAN se připojuje přes speciální integrovaný obvod (buffer, transceiver) k mikrokontroléru. Integrovaný obvod MCP2551 transformuje signály sběrnice CANH a CANL na vysílací (TXD) a přijímací (RXD) signál, které umí zpracovat CAN jednotka integrována v mikrokontroléru.

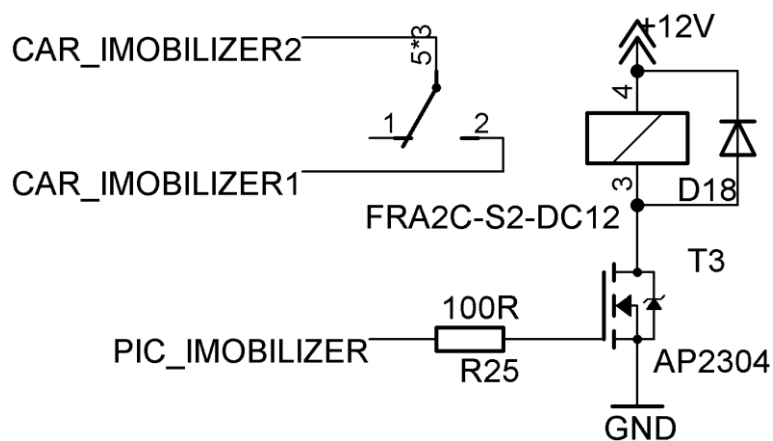
Obvod je napájen 4,1 V a chrání mikrokontrolér před nebezpečnými vlivy z vnější části sběrnice (EMI, ESD). Z důvodu ochrany ostatních uzlů ve vozidle budu CAN rozhraní chtít použít pouze ve čtecím (read-only) režimu.

4.6.3. Poplachové výstupy

V autoalarmu jsou dva signalizační poplachové výstupy. Prvním je LED dioda, která bude ve vozidle podle blikání signalizovat momentální stav autoalarmu a druhým je poplašná siréna, která se spustí okamžitě po zjištění vloupání nežádoucí osoby do uživatelova vozidla.

Uživatel může napojit imobilizační okruh vozidla na autoalarm. Po zjištění poplachu se automaticky nebo SMS zprávou odpojí tento okruh a vozidlo bude nepojízdné, dokud ho uživatel opět nevrátí do původního stavu (SMS zprávou). Imobilizační okruh se v autoalarmu napojuje na relé, které je spínané mikrokontrolérem. Uváděný maximální proud relé 40 A by měl být dostatečný pro zachování funkčnosti vozidla, jelikož u současných výrobců se pohybuje maximální proud mezi 15 a 25 A.

Z důvodu velkých proudů, které vyzařují EMI rušení do okolí jsem ohraničil relé viasy připojenými k GND na DPS, abych alespoň částečně omezil vyzařování do okolí, které by mohlo narušit chod systémů GSM a GPS.



Obrázek 4-6: Připojení imobilizačního okruhu k autoalarmu

5. Návrh Firmware

Po zdárném vyhotovení a osazení DPS začíná práce na vývoji programu do mikrokontroléru. Jedná se o obsluhu periférií, sběr dat a řízení celkového chodu autoalarmu. Mikrokontrolér programuju v jazyce C. Je možné ho programovat i v jazyce assembler kvůli zrychlení programu, ale vývoj by byl razantně pomalejší a výsledný efekt v této aplikaci nulový, z důvodu dostatečné rychlosti procesoru a výkonných překladačů. V jazyce C má výrobce Microchip vyhotovené knihovny, které mi ulehčily práci a daly se jednoduše implementovat, i když je bylo nutné občas poupravit pro můj návrh.

Vývoj mého programu se odehrával ve třech hlavních krocích:

- Inicializace modulů v mikrokontroléru
- Komunikace s perifériemi
- Celkový chod programu (run-time)

Práci na vývoji programu jsem si ulehčoval tlačítka a signalizačními LED, díky kterým jsem věděl, v jaké části programu se zrovna vykonávají příkazy a mohl potvrdit či vyvrátit vhodnost implementace nejnovější části kódu pro další aplikaci.

5.1. Inicializace mikrokontroléru

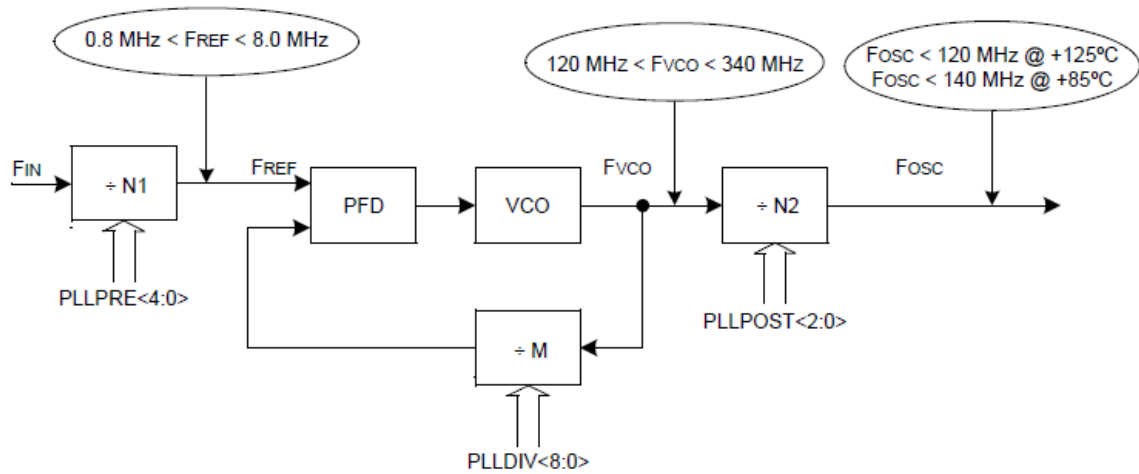
Jednotlivé komunikační moduly musí být uzpůsobeny možnostem periferních obvodů. Nejinak tomu je u systémových modulů mikrokontroléru, které ovlivňují běh a rychlost vykonávání instrukcí. Většina nastavení mikrokontroléru spočívá v zápisu příslušných hodnot do příslušných registrů, které jsou uvedeny v katalogových listech výrobce.

5.1.1. Generování hodinového signálu

Každý mikrokontrolér vykonává instrukce na základě hodinového signálu, který definuje taktovací frekvenci a rychlost. Jako zdroj kmitočtu jsem použil krystalový rezonátor o frekvenci 20 MHz. Tato frekvence spadá do HS (High-speed) módu

Kapitola 5: Návrh Firmware

mikrokontroléru. Ke zvýšení frekvence hodinového signálu jsem využil dostupného fázového závěsu PLL.



Obrázek 5-1: Generování hodinového signálu pomocí fázového závěsu PLL

Maximální frekvence, kterou se mohou vykonávat instrukce v teplotním rozsahu od -40 do +125 °C, je 60 MHz (MIPS – Million Instruction Per Second) a došel jsem k ní podle vzorce:

$$\begin{aligned} F_{cy} &= \frac{F_{osc}}{2} = F_{IN} \times \frac{PLLDIV + 2}{(PLLPRE + 2) \times 2(PLLPOST + 1)} \div 2 \\ &= 20 \times \frac{238 + 2}{(18 + 2) \times 2(0 + 1)} \div 2 = 60 \text{ MHz} \end{aligned}$$

Příslušné hodnoty jsem zapsal do registrů PLLDIV, PLLPRE a PLLPOST. Registr DOZE, který ještě dále dělí výsledný kmitočet, jsem nastavil na 0. V registru OSCCON jsem nastavil HS mód s fázovým závěsem PLL a čekal, dokud se fázový závěs nenaladil na nastavenou frekvenci. Nastavení na HS mód s PLL, jsem nastavil v prvotní inicializaci procesoru pomocí direktivy „pragma config“, která zajistí procesoru inicializační zdroj hodinového signálu, dokud nedoběhne program k uživatelskému nastavení hodinového signálu.

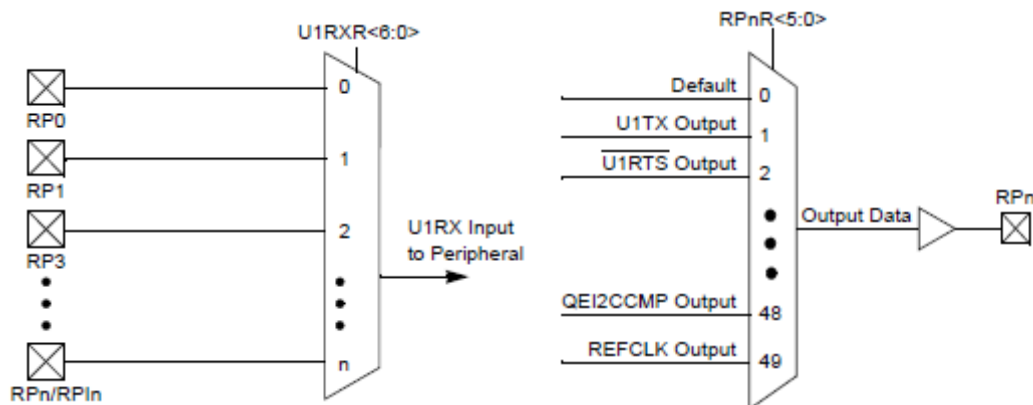
5.1.2. Komunikační moduly

V autoalarmu jsou implementovány komunikační sběrnice SPI, I²C a RS-232. Inicializace modulů zodpovědných za komunikaci na těchto sběrnících probíhá v nastavení registrů a přípravou na zdárnou komunikaci s perifériemi autoalarmu.

Mikrokontrolér disponuje na každém pinu několika funkcemi, které mají přednastavenou prioritu. Nejvyšší prioritu má většinou analogově digitální funkce, která pomocí registru ANSELx (x = A-G) přepíná mezi analogovým (analogově digitální převodník) nebo digitálním využitím. Defaultně je přednastaven analogový režim, tudíž pro všechny piny kromě pinů pro analogově digitální převodník nastavují digitální režim.

Dalšími registry, které musí být vhodně nastaveny, jelikož mají vyšší prioritu než používané registry digitálních modulů, jsou TRISx (x = A-G) registry. Pro každý pin do tohoto registru zapisují buď log. 1 pro vstup nebo log. 0 pro výstup.

Moduly se připojují k pinům pomocí speciální tabulky v [20]. Piny se definují před inicializací modulů. Vstupní a výstupní piny mají svůj extra přístup k mapování, ve kterém se vstupní označují jako RPIx (x = 0-127) a výstupní (zároveň mohou být i vstupní) jako RPx. Výstupní pin, ke kterému chceme připojit signál z modulu, má příslušný registr, do kterého se zapíše definovaná hodnota signálu z [20] představující pin. Opačný přístup je u mapování vstupních pinů, které se připojují zápisem definované hodnoty pinu do registru pro daný signál.



Obrázek 5-2: Vlevo - mapování vstupních pinů, vpravo - mapování výstupních pinů

5.1.2.1. SPI modul

Hlavní registry zodpovědné za fungování SPI jsou SPIxCON1, SPIxCON2 a SPIxSTAT (x =1-4). Ve SPIxCON1 registru aktivuji Master režim a SPI piny, nastavuji frekvenci hodinového signálu (SCK) na 625 kHz, klidový stav sběrnice v log. 0, délku sekvence 16bitů, fázování hodinového signálu a dat 180°. SPIxCON2 je zodpovědný za speciální Framed mód, který jsem nevyužil. SPIxSTA registr aktivuje celý SPI modul, indikuje přerušení a stav vysílajícího a přijímajícího bufferu.

Kmitočet hodinového signálu vyjádřím vzorcem:

$$F_{SCK} = \frac{F_{CY}}{\text{Primary Prescaler} \times \text{Secondary Prescaler}} = \frac{60}{16 \times 6} = 625 \text{ kHz}$$

5.1.2.2. I²C modul

Komunikace I²C se nastavuje v registrech I2CxCON, I2CxSTAT, I2CxBRG a I2CxADD (x = 1-2). Jelikož modul bude pracovat v režimu Master jako jediné zařízení na sběrnici, nebudeme potřebovat adresový registr I2CxADD, který definuje I²C adresu mikrokontroléru. V registr I2CxBRG definujeme frekvenci hodinového signálu SCL, kterou budu chtít 400 kHz, a jehož hodnotu spočítáme podle vzorce:

$$I2C_{BRG} = \left[\left(\frac{1}{F_{SCL}} - PGD \right) \times F_{CY} \right] - 1 = \left[\left(\frac{1}{4 \times 10^5} - 25 \times 10^{-10} \right) \times 6 \times 10^7 \right] - 1 \cong 149$$

PGD (Pulse Gobbler Delay) má typickou hodnotu 250 ns.

V registru I2CxCON aktivuji celý modul, definuji 7bitovou adresu, povolím Start, Stop a Restart podmínku, aktivuji ACK bit. Registr I2CxSTAT je k dispozici pouze ke čtení, kde si můžeme zjistit momentální stav na sběrnici (Start, Stop a Restart podmínku + ACK) a stav přijímajícího a vysílajícího bufferu.

5.1.2.3. UART modul

Modul se používá ke komunikaci na bázi RS-232 protokolu. V modulu jsou implementovány pouze RXD, TXD, CTS a RTS signály. Pokud chceme použít zbylé signály RS-232 ke komunikaci, musíme je implementovat softwarově v rámci vstupních/výstupních pinů a jejich portů.

UART modul nastavujeme pomocí registrů U_xMODE , U_xSTA a U_xBRG ($x = 1-4$). S modulem bude mikrokontrolér komunikovat rychlostí 115 200 Bd. Vzorec pro výpočet hodnoty, která se zapíše do registru a definuje rychlost komunikace „baud rate“ je:

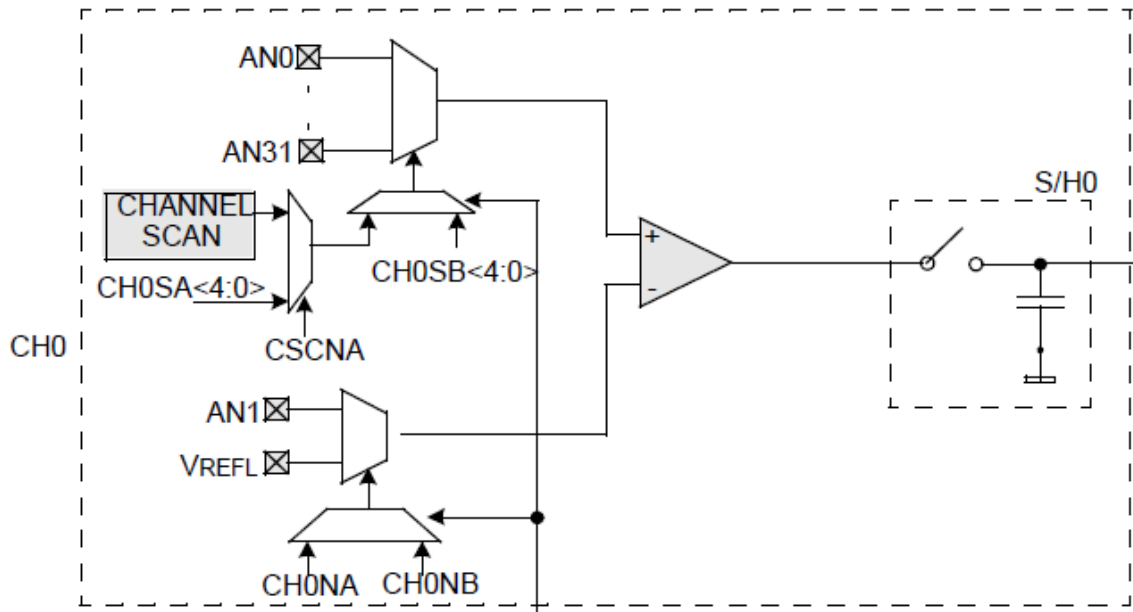
$$U_xBRG = \frac{F_{cy}}{4 \times Baud\ Rate} - 1 = \frac{6 \times 10^7}{4 \times 115200} - 1 \cong 129$$

V registru U_xMODE zapnu UART modul, vypnu IrDA mód, aktivuji všechny hardwarově implementované piny, definuji klidový stav RXD signálu v log. 1 (nutno přidat „pull-up“ rezistor), 8bitovou datovou sekvenci bez parity a s jedním Stop bitem. V registru U_xSTAT definuji klidovou hodnotu signálu TXD do log. 1., aktivuji vysílání, nastavím příznak přerušení při příjmu dat a deaktivuji adresový mód.

5.1.3. Analogově digitální převodník

V autoalarmu potřebuji měřit úroveň napětí autobaterie a záložního akumulátoru, aby byl uživatel informován při hrozícím vybití, odpojení jednoho nebo druhého akumulátoru a mohl podle toho jednat. Při odpojení autobaterie začne automaticky napájet autoalarm záložní Li-Ion akumulátor. Pokud klesne napětí pod předem definovanou hodnotu, generuje se přerušení a je informován uživatel SMS zprávou.

Mikrokontrolér disponuje dvěma moduly analogově digitálních převodníků. Každý modul se dělí na čtyři kanály, ke kterým lze připojit až 32 vstupů, které se vzorkují. V každém kanále se můžou multiplexovat dva vstupy, které se ale složitě vyhodnocují. Proto jsem zvolil možnost zprovoznění obou modulů, každý pro jednu baterii i z důvodu použitelnosti jednoho kanálu v 12bitovém vzorkování (čtyři kanály se mohou využívat pouze v 10bitovém vzorkování).



Obrázek 5-3: Vzorkovací kanál modulu analogově digitálního převodníku [20]

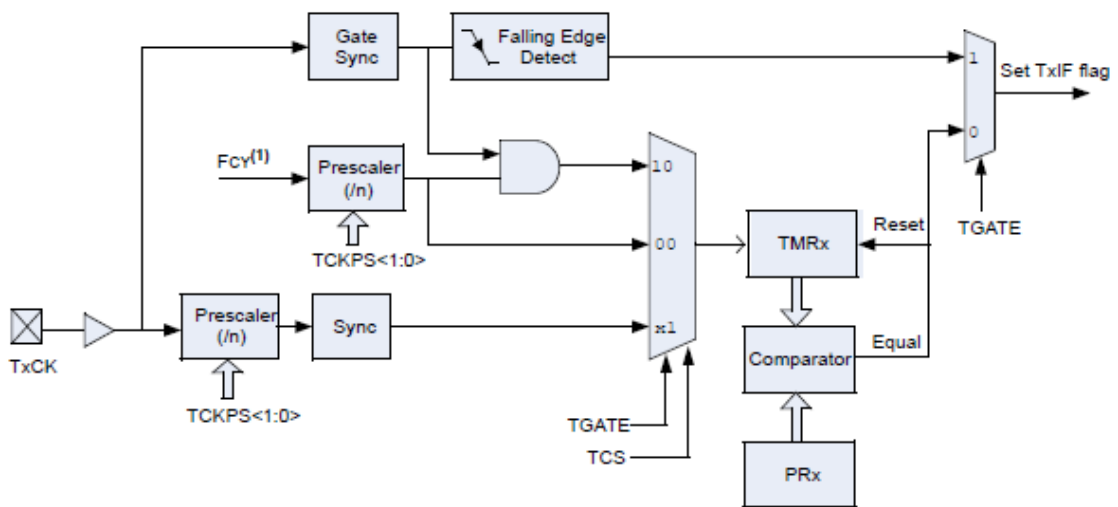
Analogově digitální převodník nastavuji pomocí registrů ADxCON1, ADxCON2, ADxCON3, ADxCHS0, AD1CSSH a ADxCSSL (x = 1-2). Již zmíněné registry ANSELx nastavím pro vzorkované vstupy do analogového módu a příslušné bity TRISx registrů do čtecího módu. ADxCSSL aktivuje příslušné analogové vstupy od AN0 do AN15 a AD1CSSH AN16 až AN31. Aktivoval jsem AN22 prvního modulu pro autobaterii. Jelikož pro druhý modul můžeme připojit pouze 16 vstupů (neexistuje AD2CSSH - v rozsahu vstupů AN16-AN31), aktivoval jsem AN9 pro záložní akumulátor. V registru ADxCHS0 připojím příslušný vzorkovaný signál ke kladnému vstupu komparátoru a GND k zápornému vstupu komparátoru.

V prvním z hlavních řídicích registrů modulu ADxCON1 aktivuji celý modul, 12bitové vzorkování, bezznaménkový výstup hodnot od 0 do 1 a automatické vzorkování a konverzi dat. V registru ADxCON2 nastavuji maximální vzorkovanou hodnotu na napájecí napětí (3,3 V) a minimální připojím k GND, dále aktivuji pouze první kanál CH0 a první kanál multiplexoru CH0SA a CH0NA. Generovat přerušování bude stačit každý 16. vzorek k porovnání hodnot. Nastavení časování má za úkol registr ADxCON3, ve kterém připojím k modulu hlavní zdroj hodinového signálu (60 MHz), který dál dělím 60 na pracovní

frekvenci převodníku 1 MHz. Automatické vzorkování se bude spouštět každých 30 cyklů této pracovní frekvence a spolu s generováním přerušení každých 16 vzorků vychází perioda přerušení na 0,48 ms. Samotné vyhodnocení napětí programem spustím po inkrementování proměnné na hodnotu 2083, což mi realizuje vyhodnocení každou 1 s.

5.1.4. Časovače a přerušení

Procedury programu, které potřebuju vykonat méněkrát než každou periodu hlavní smyčky programu, obsloužím pomocí časovačů. To umožní rozdělit sériové vykonávání programu do několika paralelních na sobě nezávislých bloků se specifickými funkcemi. Díky tomu dokáží využít téměř maximální výkon mikrokontroléru bez čekacích smyček využívaných např. k signalizaci LED diodami (mikrokontrolér dále může zrychlit pomocí DMA-Direct Memory Access modulů, ale kvůli složitější implementaci na úkor už tak vysokého výkonu to není potřeba).



Note 1: Fcy is the instruction cycle clock.

Obrázek 5-4: Blok časovače mikrokontroléru [20]

Časovače se inkrementují dokud „nepřetečou“ v předem definované hodnotě, která vyvolá rutinu přerušení, ve které nastavím příznak přerušení či inkrementuji proměnnou. Po návratu z přerušení skočí program díky příznaku na proceduru, která se má vykonat. Jakmile časovač „přeteče“, začíná inkrementovat zase od nuly. V mikrokontroléru naleznou

Kapitola 5: Návrh Firmware

devět 16bitových časovačů, z nichž se osm může spojit do dvojic za vzniku 32bitových časovačů.

Nastavení časovačů probíhá pomocí registrů TxCON, PRx a TMRx ($x = 1-9$). V registru TxCON aktivuji celý modul, nastavím potřebnou „předděličku“ hodinového signálu, zvolím 32bitový nebo 16bitový mód a hlavní zdroj hodinového signálu (60 MHz). Z registru TMRx si můžu přečíst aktuální hodnotu časovače nebo do něj zapsat hodnotu, kterou bude následně inkrementovat. Do registru PRx, zapíšu hodnotu, po které časovač „přeteče“ a vyvolá přerušení.

Přerušení generuji, pokud potřebuji neodkladně obsloužit dotyčnou proceduru programu. Příkladem může být zvonění telefonu, kdy mi RI (Ring Indicator) signál, vyvolá rutinu přerušení nastavením příznaku přerušení, po kterém obsloužím hovor a následně se vrátím na místo, kde program skončil. Přerušení můžu nastavit do sedmi priorit přerušení, ve kterých platí, že pouze přerušení s vyšší prioritou může vyvolat přerušení v přerušení s nižší prioritou. Dále se může vyvolat tzv. „non-maskable“ (uživatelsky nenastavitelné) přerušení, které hlásí vážnou systémovou chybu, a následně musím vyvolat reset mikrokontroléru

Přerušení události, kterou chci vyvolat, nastavím registrem IECx ($x = 0-9$), ve kterém zapnu přerušení dotyčné události, nastavím potřebnou prioritu přerušení v registru IPCx ($x = 0-35$) a globálně povolím všechny přerušení v registru INTCON2. Následně napíšu speciální funkci – rutinu přerušení, na kterou program po vyvolání přerušení skočí, a ve které nulují příznak přerušení v registru IFSx ($x = 0-8$), aby se přerušení nevyvolávalo do nekonečna. Dále si nastavím pokud možno pouze uživatelský příznak události s několika příkazy, abych se v přerušení nacházel co nejkratší dobu z důvodu obsluhy dalších rutin přerušení, a vrátím se zpět na hlavní program, ve kterém pomocí nastaveného příznaku dokončím obsloužení dotyčné události (pokud je potřeba).

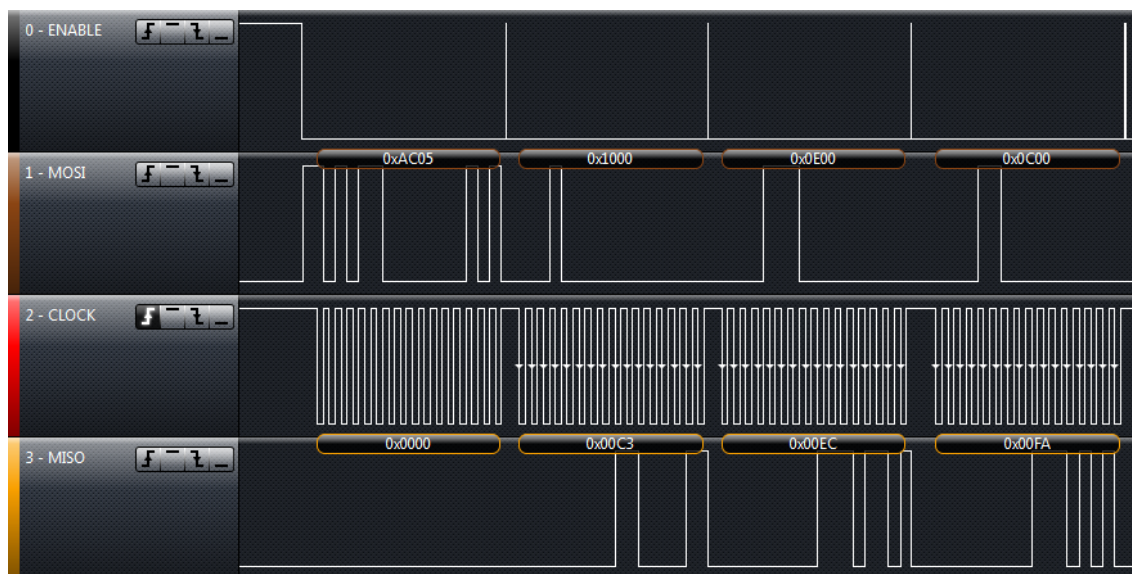
5.2. Komunikace s perifériemi

Aby mohla komunikace mezi mikrokontrolérem a ostatními zařízeními proběhnout, musí být nastaveny komunikační registry v mikrokontroléru, jak jsem popsal v předchozích kapitolách. Na straně periférie si musíme vyčíst z katalogových listů výrobce princip komunikace, což obnáší, za jakých podmínek můžeme do periférie zapisovat a číst, dále případné nastavení registrů, které umožní přístup k cílové informaci.

5.2.1. Obsluha akcelerometru MMA7455

Komunikace s akcelerometrem po sběrnici SPI probíhá v 16bitových sekvencích oddělených signálem Slave Select, ve kterých první bit nastaví čtení (log. 0) nebo zápis (log. 1), dalších 6bitů přísluší registru, s kterým chceme komunikovat a posledních 8bitů představují data, která chceme po signálu MOSI zapisovat nebo ze signálu MISO číst.

Na obrázku 5-5 můžeme vidět, názornou komunikační sekvenci, ve které nejprve zápisem do Mode Control registru aktivujeme akcelerometr do měřícího módu s rozlišením 64bitů/g a následně ze všech tří os akcelerometru vyčítám postupně zrychlení.

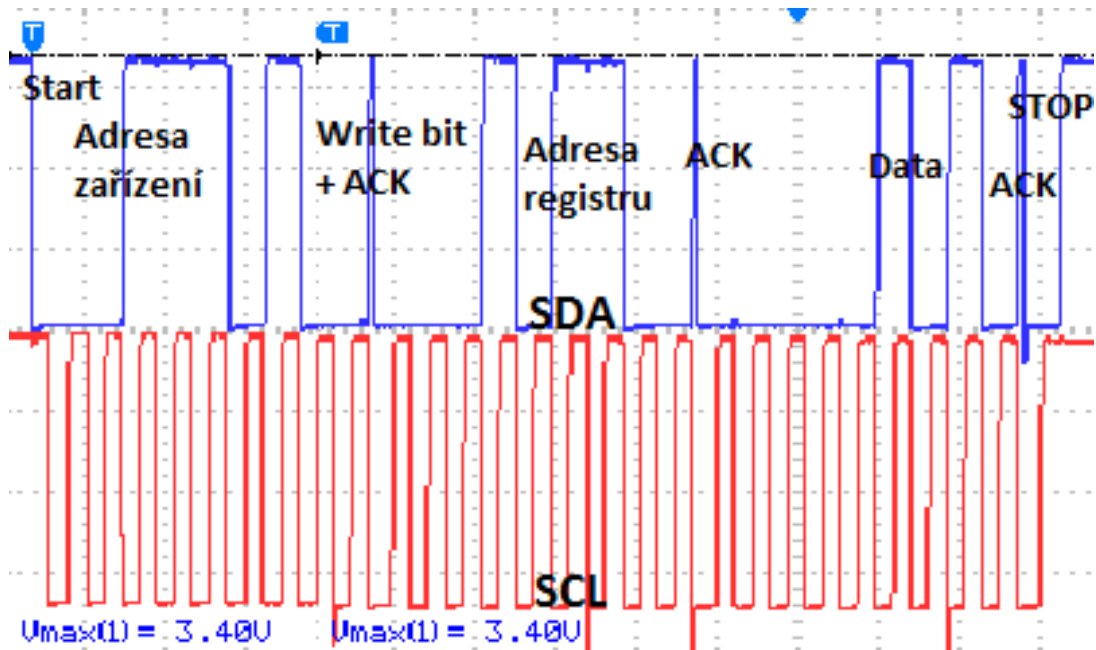


Obrázek 5-5: Komunikace s akcelerometrem po sběrnici SPI

Bohužel jsem nedokázal mikrokontrolérem vyčíst správná data, proto jsem byl přinucen připojit akcelerometr na I²C sběrnici, kde už komunikace probíhala v pořádku.

Kapitola 5: Návrh Firmware

Komunikaci po I²C sběrnici můžeme vidět na obrázku 5-6. Po generování Start podmínky vysílám adresu zařízení s Write bitem a čekám na ACK od akcelerometru. Následně posílám adresu registru, do kterého chci zapisovat (0x16), čekám na ACK a poté zapisuji stejná data, jako u SPI pro aktivaci akcelerometru. Můžete si všimnout zákmitu na ACK, který značí uvolnění signálu mikrokontrolérem a následné shození do log. 0 senzorem.



Obrázek 5-6: Komunikace s akcelerometrem po sběrnici I²C

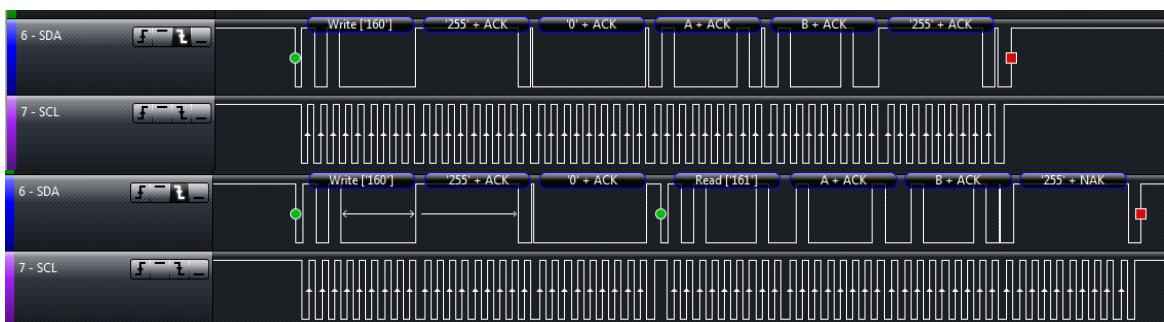
Abych akcelerometr mohl použít v aplikaci, kdy chci generovat přerušení po překročení nastaveného zrychlení, musím kalibrovat osy akcelerometru. Nejprve si vyčtu ze všech os zrychlení v měřicím módu s citlivostí 64bitů/g. Následně hodnotám přiřadím opačná znaménka a vynásobím je dvěma, jelikož offset registry, do kterých hodnoty zapisuji, mají 2x větší citlivost. Procedura se musí ještě jednou zopakovat, jelikož po první kalibraci, vykazují osy zrychlení v průměru 0,1 g. Po druhé kalibraci jsou již všechny osy na 0 g a můžu pokračovat v dalším nastavení.

Následně nastavím práh zrychlení pro rovnoměrné zrychlení vozidla (0,2 g) a pro pulzní zrychlení (2 g), kterým můžu indikovat např. náraz do vozidla. Jakmile nastavím hodnoty prahu, uvedu akcelerometr do pulzního módu, který vyhodnocuje jak rovnoměrné, tak i pulzní zrychlení. Po překročení prahu se generuje přerušení, které dokáži vyhodnotit

pro obě události zvlášť. Aby mohl akcelerometr nadále generovat přerušení, musí se v příslušném registru vynulovat příznak přerušení. Toho dokáží využít např. v případě, že mi někdo ujíždí s autem a nepotřebuji každou chvíli informovat prostřednictvím SMS, že je auto v pohybu.

5.2.2. Obsluha paměti EEPROM 24FC1025

K ukládání důležitých dat, jako jsou telefonní čísla uživatelů, systémové heslo nebo GPS pozice budu používat paměť EEPROM. Každou informaci ukládám na dvě místa v paměti a při zpětném čtení porovnávám. Pokud se data nebudou shodovat, bude nastavena defaultní informace pro konkrétní položku a informován uživatel. S pamětí komunikuji pomocí I²C sběrnice.



Obrázek 5-7: Nahoře - zápis do paměti EEPROM, dole - čtení z paměti EEPROM

Paměť je velká 128 kB a k adresování využívám 17 adresových signálů A0 až A16. Zápis do paměti znázorněný na obrázku 5-7 probíhá následovně. Po Start podmínce následuje adresa zařízení definována prvními čtyřmi bity továrně následována, adresovým signálem A16 a dvěma adresovými bity definovanými připojením logické úrovně na konkrétní piny paměti. Následuje Write bit a čekání na ACK, poté se zapisují dva byty, která představují adresové bity A15 – A8 (MSB) a A7 – A0 (LSB). Pak už můžu do paměti zapisovat data.

Paměť podporuje sekvenční zápis a čtení (do 128 bytů), což nám umožní zapsat adresu pro první byte sekvence a poté se pro další byty adresa automaticky inkrementuje. Na obrázku 5-7 je vidět zápis řetězce „AB“. Poslední byte je ukončovací znak řetězců \0, ve skutečnosti je pouze uvolněna sběrnice a při zpětném čtení není na tento byte brán zřetel.

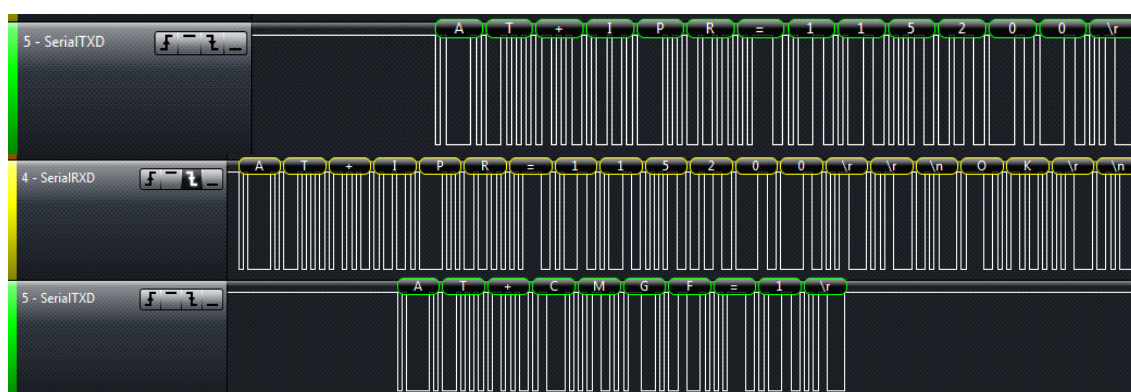
Kapitola 5: Návrh Firmware

Poté mikrokontrolér generuje Stop podmínku. Pro opětovný zápis nebo čtení z daného místa je nutné počkat 5 ms, jinak daná komunikace selže.

Čtení z paměti probíhá podobně. Zapišeme adresu zařízení a dat, poté se generuje Restart podmínka, po které následuje adresa zařízení s Read bitem. Paměť začíná posílat data, dokud mikrokontrolér neodešle požadavek na ukončení přenosu NACK následovaný Stop podmínkou.

5.2.3. Obsluha GSM/GPS modulu SIM908

Modul využívá ke komunikaci tzv. sériovou linku vycházející z rozhraní RS-232. V konečném návrhu jsem využil pouze signály TXD, RXD a RING. Mimo sériové rozhraní jsem používal signály PWRKEY k zapnutí a vypnutí modulu a signály, které detekují zapnutí modulu a jeho přihlášení do sítě (POWER a NETWORK status). Po sériové lince komunikují prostřednictvím AT příkazů. Jednotlivé AT příkazy mají různou dobu odezvy. Po přijetí odpovědi porovnávám řetězec s předpokládanou odpovědí. Pokud se liší, signalizuje mikrokontrolér chybu. Inicializace modulu SIM908 začíná zapnutím modulu startovacím pulzem PWRKEY. Následně odesílám příkazy pro nastavení baud rate, nastavení SMS do textového módu a vložení PIN kódu, po kterém se modul přihlásí do GSM sítě.



Obrázek 5-8: Komunikace s modulem SIM908, nahoře - příkaz, uprostřed - odpověď, dole - další příkaz

Na obrázku 5-8 jsem znázornil nastavení baud rate pro modul a následně ještě uvedl příkaz pro nastavení formátu SMS do textového módu. Každý AT příkaz musí být zakončen znakem `\r`. SIM908 používá AT signály standardizované v GSM síti, má ale i speciální

Kapitola 5: Návrh Firmware

instrukční sadu AT příkazů pro ovládání GPS, vlastního analogově digitálního převodníku nebo vstupních a výstupních pinů GPIO [21].

Přijetí příchozího hovoru k funkci odposlechu realizují příkazem ATA. Modul může sám vytočit uživatele příkazem ATDx; x je telefonní číslo uživatele. Příchozí hovor detekují RING signálem v log. 0 déle než 120 ms (RING má klidovou hodnotu v log. 1), protože 120 ms pulz detekuje příchozí SMS. SMS přečtu příkazem AT+CMGR=1,0 pro jedinou SMS v paměti. SMS po přečtení mažu z paměti modulu, abych mohl přijmout stejným příkazem novou SMS, jelikož první číslo v AT příkazu pro čtení SMS ukazuje na místo v paměti modulu.

Inicializace systému GPS probíhá zapnutím GPS části modulu, studeným (Cold) resetem a poté teplým (Warm) resetem. Chvilu se musí čekat, dokud modul „neuvidí“ dostatek satelitů a poté můžeme vyčítat GPS informace. Máme na výběr z několika typů zpráv ze standardu GPS, každá se liší, ve výsledku jsem používal tři GPS zprávy, pro zjištění polohy, rychlosti (v uzlech) a směru, dále pro zjištění polohy (s identifikátory souřadnic S,N,W,E) a pro zjištění rychlosti v kilometrech a směru pohybu.

Bohužel žádná zpráva neobsahuje GPS pozici v desetinných stupních, kterou podporuje Google Maps, proto jsem musel vytvořit speciální funkci pro převedení textového řetězce na celé číslo (vyhnul jsem se reálnému číslu), přepočítat na desetinné stupně a převést zpět do textového řetězce.

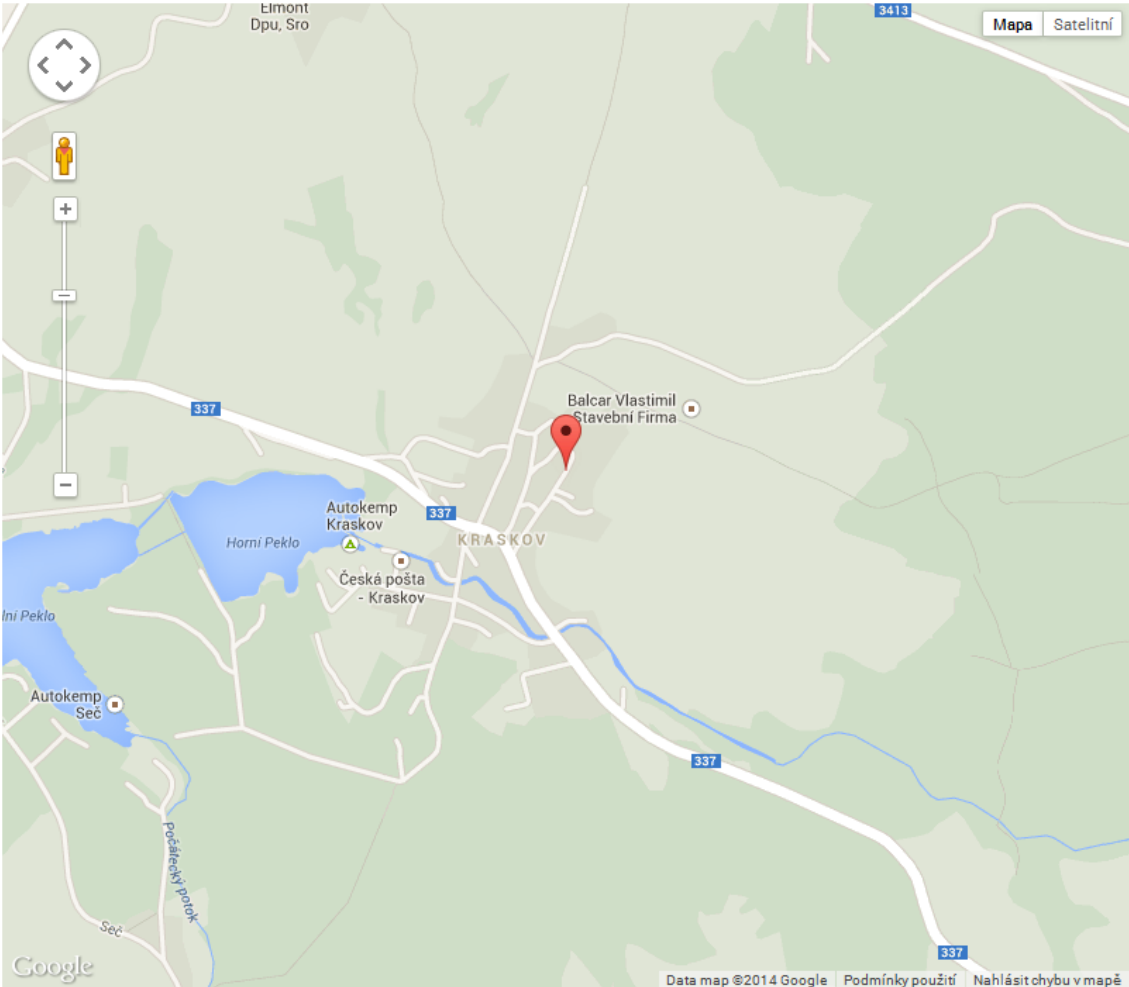
Abych dostal GPS data na vzdálenou adresu s Google Maps, potřebuji inicializovat GPRS systém a nastavit FTP údaje, kam se budou na vzdáleném serveru data ukládat. PHP skript [22] v pevných časových intervalech obnovuje webovou adresu s vloženou mapou a zjišťuje novou pozici z textového souboru, do kterého modul nahrává GPS data.

Kapitola 5: Návrh Firmware

Firefox <http://sim908.borec.cz/> +

SIM908 GPS position Filip Prokop

| Time | Satellites | Speed OTG | Course |
|---------------------|------------|-----------|------------|
| 2014 Apr 17 - 15:04 | 10 | 0.000000 | 358.351562 |



Obrázek 5-9: Webová stránka zobrazující aktuální polohu vozidla na Google Maps

Pro posílání dat potřebuji nejprve poslat modulu informaci, o který typ datové komunikace se bude jednat. Mimo GPRS mám na výběr ještě CSD přenos, který ale nebudu využívat. Dále musím nastavit přístupový bod APN (bránu), ke kterému se budu připojovat. Poté už můžu zapnout GPRS spojení. Po 20 s se ujistím, že spojení proběhlo úspěšně a začnu nahrávat do modulu adresu FTP serveru, uživatelské jméno, heslo, jméno souboru, do kterého bude data ukládána a cesta k tomuto souboru na FTP serveru.

Kapitola 5: Návrh Firmware

Pokud vše proběhne úspěšně (v opačném případě se mikrokontrolér snaží poslat data znovu), zahájím FTP spojení a pošlu modulu informaci o délce vysílaného řetězce, ten mi žádost potvrdí a začne vysílat data do požadované délky. Poté spojení ukončím.

Na FTP serveru mám uloženou PHP stránku, která v pravidelných intervalech vyčítá informace z uloženého souboru. V uloženém souboru najde PHP skript informaci o poloze vozidla, jeho rychlosti a směru pohybu. Dále se vyčítá informace o počtu využitých satelitů, díky které si můžeme představit přesnost zobrazené pozice.

5.3. Běh programu

Mikrokontrolér vykonává instrukce podle napsaného programu v jazyce C, který kompilátor XC16 překládá do jazyka symbolických adres mikrokontroléru. Do běhu programu vstupují přerušovací rutiny časovačů, ve kterých se cyklicky kontrolují příznaky přerušení akcelerometru, stavy vstupních signálů od dveří nebo světel vozidla a vyhodnocuje se signál RING, který informuje o příchozím hovoru nebo SMS zprávě. Přerušovací rutinou také ukládám příchozí data z GSM/GPS modulu.

Prvním krokem po spuštění mikrokontroléru je inicializace, neboli nastavení počátečních podmínek. Nejprve proběhne konfigurace oscilátoru a čekání dokud se „nezamkne“ fázový závěs. Poté začnu konfigurovat jednotlivé piny, které budou mít z většiny digitální charakter (kromě pinů analogově digitálního převodníku), přiřadím směr komunikace (vstup nebo výstup) a „namapuju“ ke konkrétnímu bloku mikrokontroléru. Inicializace pokračuje nastavením jednotlivých bloků mikrokontroléru, jakými jsou komunikační sběrnice, časovače, analogově digitální převodník a přerušení.

Kapitola 5: Návrh Firmware

```
void __attribute__((interrupt,no_auto_psv)) _U1RXInterrupt( void )
{
    Rxdata0 [rxBuffer] = U1RXREG; // Data from hardware to software buffer
    rxBuffer++;

    if (rxBuffer == Array_Size) rxBuffer = false; // Check of length array
    IFS0bits.U1RXIF = false; // Clear the Interrupt flag
}

void __attribute__((interrupt,no_auto_psv)) _T1Interrupt( void )
{
    if(!RING)
    {
        if (PIC_PWD_STAT) // Is module power up?
        {
            SMSflag++;
            if(SMSflag>100)
            {
                SMSflag = false;
                CallFlag= true;
            }
        }
    }
    IFS0bits.T1IF = false; // Clear the Interrupt flag
}
```

Obrázek 5-10: Přerušovací rutiny RXD signálu GSM/GPS modulu a časovače pro kontrolu RING signálu

Po inicializaci hardwarové části následuje softwarová část, ve které si z paměti EEPROM uloží do paměti mikrokontroléru systémové údaje jako povolená uživatelská telefonní čísla nebo heslo pro změnu některých údajů. Následuje alokace paměťových míst (polí), do kterých budu ukládat GPS data, SMS, FTP údaje a příchozí/odchozí data z/do hardwarových bufferů.

V tuto chvíli je autoalarm schopen plnit své funkce. Oprávněná telefonní čísla mohou autoalarmu posílat příkazy prostřednictvím SMS. Každá SMS zpráva je podrobena řetězcovému algoritmu, který z ní vyčte příkaz a telefonní číslo, které následně ověří, jestli má povolení k ovládní autoalarmu. Autoalarm příkazy vyhodnotí, přiřadí jim příznaky, vykoná potřebné procedury a případně pošle zpět chtěné informace. Pro změnu telefonního čísla, musí zadat uživatel systémové heslo, které může také sám měnit. Autoalarmu je navržen pro autorizaci až třech telefonních čísel. Seznam příkazů, které autoalarm zpracuje, uvádím v příloze.

6. Zhodnocení

Podářilo se mi vyhotovit autoalarm podle zadání. Autoalarm pracuje v následujících krocích. Po připojení autoalarmu k napájení se spustí inicializace mikrokontroléru s perifériemi. Následuje čtení z EEPROM a kalibrace akcelerometru. Po vsunutí klíče do zapalování se sepne imobilizační okruh. Příkazem ZAPNOUT uvedeme autoalarm do pohotovostního režimu. Pokud autoalarm zjistí vložení klíčku do zapalování, otevření dveří, pohyb automobilu nebo náraz do něj, okamžitě informuje uživatele SMS zprávou a spustí sirénu. Příkazem VYPNOUT se deaktivuje ochrana automobilu.

Příkazem GPSINFO dostanu zpět SMS zprávu o GPS pozici, rychlosti a směru pohybu. Příkazem INTERNET se připojím k internetu a následně příkazem SLEDUJ můžu sledovat na mapě vložené na webu aktuální polohu automobilu. Příkazem NAHRAJ uloží autoalarm pouze jednu pozici automobilu. Pokud napíše cizí číslo autoalarmu, ignoruje ho. Pro autorizaci nového čísla zašlu příkaz ZMENAx ($x = 1-3$). Platí i pro neautorizovaná čísla. Autoalarm si vyžádá systémové heslo. Po potvrzení hesla zašlu telefonní číslo.

Systémové heslo lze změnit příkazem HESLO. Příkazy FTPUSER, FTPUSER, FTPSERVER a GPRSAPN nastavím údaje pro připojení na internet a FTP server, ze kterého budu číst GPS data pomocí PHP skriptu na webu. S autoalarmem se mohu kdykoliv spojit telefonicky zavoláním na jeho telefonní číslo. Pomocí mikrofonu můžu odposlouchávat dění v automobilu.

Příkazem IMOBOFF zablokují imobilizační okruh na dvě hodiny nebo dokud nepošlu příkaz IMOBON. Po zaslání příkazu STATUS dostanu SMS se stavem na vstupech autoalarmu.

Čtení zpráv pomocí CAN sběrnice se mi nepodařilo realizovat. Chyběla mi potřebná dokumentace k identifikování zpráv na sběrnici. Proto jsem ani neinicializoval CAN modul v mikrokontroléru. Autoalarm je však připraven na tento druh komunikace.

Kapitola 6: Zhodnocení

Klidová spotřeba autoalarmu je 100 mA/12 V. V současné době bych doporučil při přechovávání vozidla na bezpečném místě autoalarm vypínat úplně, dokud se nepodaří snížit spotřebu. Při běžné kapacitě autobaterií kolem 50 Ah/12 V, by měl autoalarm být schopen klidového provozu o délce 20 dní bez nastartování automobilu. Kapacita použitého záložního akumulátoru typu Li-Ion by měla postačit na 12 hodin klidového provozu.

Spotřeba se dá snížit vypnutím systémů GPS a GPRS. Dalšího snížení spotřeby dosáhnou snížením taktovací frekvence mikrokontroléru a jeho uvedením do režimu nízké spotřeby (Sleep mode). Pro opětovné probuzení mikrokontroléru využiju přerušení na vstupních signálech autoalarmu a signálech z modulu RXD a RING. Snížením proudu děličů napětí přispějí také k celkovému snížení spotřeby autoalarmu.

Cenu autoalarmu jsem kalkuloval pro výrobu 100 ks. Započítal jsem pouze náklady na součástky a komponenty, náklady spojené s vývojem a případným marketingem jsem nekalkuloval. Výsledná cena se pohybuje okolo 1900 Kč/ks.

7.Závěr

Autoalarm s GSM/GPRS komunikací a GPS lokalizací byl navržen a zprovozněn dle zadání. Autoalarm budu používat v praxi ve svém vozidle a je možné jeho budoucí komerční využití. Funkce autoalarmu jsou následující:

- Zpracování a vykonání SMS příkazů.
- Zjištění GPS pozice auta, jeho rychlosti a směru pohybu a zaslání těchto dat v SMS.
- Nahrání GPS dat na vzdálený FTP server.
- Zobrazení pozice, rychlosti a směru pohybu na Google Maps téměř v reálném čase.
- Detekce otevření dveří, rozsvícení světel a dalších míst, která chceme sledovat.
- Spuštění sirény v případě nežádoucí události a vyrozumění majitele SMS zprávou.
- Detekce pohybu vozidla nebo nárazu do vozidla.
- Odposlech automobilu.
- Možnost řídit imobilizační okruh.
- Autorizace až třech telefonních čísel.
- Záložní akumulátor pro případ chybějící nebo vybité autobaterie.

Jelikož se mi nepodařilo rozluštit data na CAN sběrnici, z důvodu nenalezení potřebné dokumentace, vynechal jsem její implementaci v konečném návrhu a odsunul do doby, než naleznu potřebné údaje.

V budoucím návrhu se chci zabývat lepším využitím GPS dat. K mapě o okamžité poloze bych chtěl doplnit mapu s celou trasou od započetí sledování, možnost podívat se na předchozí trasy a případně vykreslovat grafy o rychlosti, ujeté vzdálenosti nebo nadmořské výšce v závislosti na čase.

I. Seznam literatury

- [1] Základní uživatelský manuál, Jablotron CA-1803BT
<http://www.jablotron.com/DownloadHandler.aspx?method=GetFileDownload&fileID=2186&DontParse=true>
- [2] Uživatelská příručka, Keetec T GSM,
http://www.levnealarmy.cz/get_file.php?did=1898&lang=cz&download=0
- [3] Produktový list, Jablotron CA-1803BT
<http://www.jablotron.com/cz/katalog-produktu/autotechnika/zabezpeceni-vozidel/gsm-gps-autoalarmy/ca-1803bt.aspx>
- [4] Uživatelská příručka, Spy 10 PKE
http://www.levnealarmy.cz/get_file.php?did=1258&lang=cz&download=0
- [5] Chod J., Přednášky předmětu Mobilní komunikační sítě (A2M32MKS)
- [6] Balcar T., Datové přenosy GPRS, EDGE - rozšíření GSM
- [7] Gunnar, H., GSM Networks. Boston: Artech House, 1999
- [8] Hrdina Z., Pánek P., Vejražka F., Rádiové určování polohy, <http://radio.feld.cvut.cz/>
- [9] Dudáček K., Sériová rozhraní SPI, Microwire, I²C a CAN, K.D. 2002
- [10] Philips Semiconductors, The I²C-bus specification, Version 2.1, 01/2000
- [11] Dallas Semiconductor, Fundamentals of RS-232 Serial Communications
- [12] Playing with STM32F407 test board – Serial communication – RS232
<http://www.fmf.uni-lj.si/~ponikvar/STM32F407%20project/SerialCommunication-RS232.pdf>
- [13] Bosch, CAN Specification, Version 2.0
- [14] Novák J., CAN bus a jeho aplikace ve vozidlech
- [15] Microchip, AN1249, ECAN Operation with DMA on dsPIC33F and PIC24H Devices
- [16] Microchip, dsPIC/PIC24H Family Reference Manual

Kapitola I: Seznam literatury

- [17] Corrigan S. (Texas Instruments), Introduction to the Controller Area Network (CAN), SLOA101A, 07/2008
- [18] Introduction to the Controller Area Network (CAN)
- [19] SIMCOM, SIM908_Hardware Design_V1.00
- [20] Microchip, dsPIC33EPXXX (GP/MC/MU)806/810/814 and PIC24EPXXX(GP/GU)810/814
- [21] SIM908_AT Command Manual_V1.01.
- [22] Geolocation Tracker (GPRS + GPS) with SIM908 over Arduino and Raspberry Pi
<http://www.cooking-hacks.com/documentation/tutorials/geolocation-tracker-gprs-gps-geoposition-sim908-arduino-raspberry-pi>

II. Seznam použitých zkratk

ACK - Acknowledgement

ADC - Administrative Centre

APN - Access Point Name

ARM - Advanced RISC Machine

AuC - Authentication Center

A-S - Anti-Spoofing

BPSK - Binary Phase Shift Keying

BSC - Base Station Controller

BSS - Base Station Subsystem

BTS - Base Transceiver Station

CAN - Controller Area Network

CDMA - Code Division Multiple Access

CDU - Control and Display Unit

CS - Chip Select, Coding Scheme

CRC - Cyclic Redundancy Check

CRPA - Controlled Radiation Pattern Antenna

CSMA/CD+AMP - Carrier Sense Multiple Access with Collision Detection and Arbitration on Message Priority

CTS - Clear To Send

C/A - Coarse Acquisition

DCD - Data Carrier Detect

DCE - Data Communication Equipment

DLS - Data Loader System

DPS - Deska Plošných Spojů

DSR - Data Set Ready

EEPROM - Electrically Erasable Programmable Read-Only Memory

EIR - Equipment identity register

EMI - Electromagnetic interference

Kapitola II: Seznam použitých zkratk

EOF - End Of Frame

ESD - Electrostatic discharge

FB - Feedback

FDMA - Frequency Division Multiple Access

FRPA - Fixed Radiation Pattern Antenna

FTP - File Transfer Protocol

GGSN - Gateway GPRS Support Node

GMSC - Gateway Mobile Services Switching Center

GND - Ground

GPRS - General Packet Radio Service

GPS - Global Positioning System

GSM - Groupe Spécial Mobile

HLR - Home Location Register

HOW - Hand-Over Word

HS - High-speed

IDE - Integrated Development Environment

IMEI - International Mobile Station Equipment Identity

IMSI - International Mobile Subscriber Identity

IP - Internet Protocol

IrDA - The Infrared Data Association

ISO - International Organization for Standardization

LA - Location Area

LAI - Location Area Identifier

LED - Light-Emitting Diode

LIN - Local Interconnect Network

LLC - Logical Link Control

LMSI - Local Mobile Station Identity

LSB - Least Significant Bit

MAC - Medium Access Control

MCS - Master Control Station

Kapitola II: Seznam použitých zkratk

MIPS - Million Instruction Per Second

MISO - Master In Slave Out

MOSI - Master Out Slave In

MSISDN - Mobile Subscriber Integrated Services Digital Network Number

MSB - Most Significant Bit

MSC - Mobile Switching Center

MSRN - Mobile Station Roaming Number

NACK - Not Acknowledgement

NMS - Network Management Subsystem

NRZ - Non Return to Zero

NSS - Network switching subsystem

NTC - Negative Temperature Coefficient

OMC - Operations and Maintenance Center

OSS - Operation Support Subsystem

PDOP - Positional Dilution of Precision

PHP - Hypertext Preprocessor

PKE - Passive Keyless Entry

PLL - Phase Lock Loop

PPP - Point-to-Point Protocol

PPS - Precision Positioning Service

PTC - Positive Temperature Coefficient

REC - Receive Error Count

RI - Ring Indicator

RPU - Receiver and Processor Unit

RTR - Remote Transmission Request

RTS - Request To Send

RS - Recommended Standard

RxD - Receive Data

SA - Selective Availability

SCK - Serial Clock

Kapitola II: Seznam použitých zkratek

SCL - Seriál Clock

SDA - Seriál Data

SGSN - Serving GPRS Support Node

SIM - Subscriber Identity Module

SMA - SubMiniature version A

SMS - Short Message Service

SPI - Serial Peripheral Interface

SPS - Standart Positioning Service

SS - Slave Select

TDMA - Time Division Multiple Access

TEC - Transmit Error Count

TID - Tunnel Identity

TLM - Telemetry Word

TOW - Time Of Week

TTL - Transistor Transistor Logic

TxD - Transmit Data

UART - Universal Asynchronous Receiver Transmitter

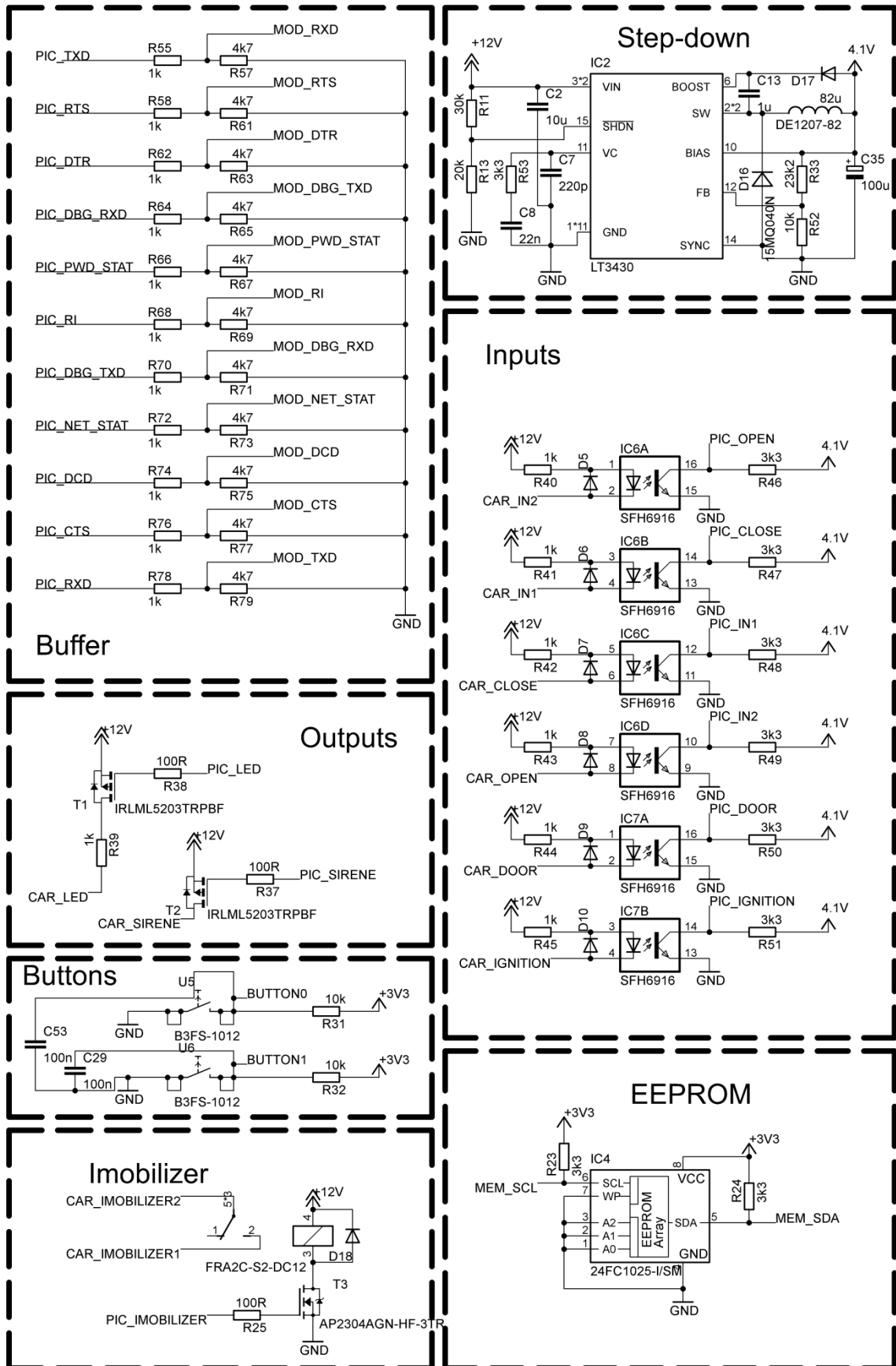
VLR - Visitor Location Register

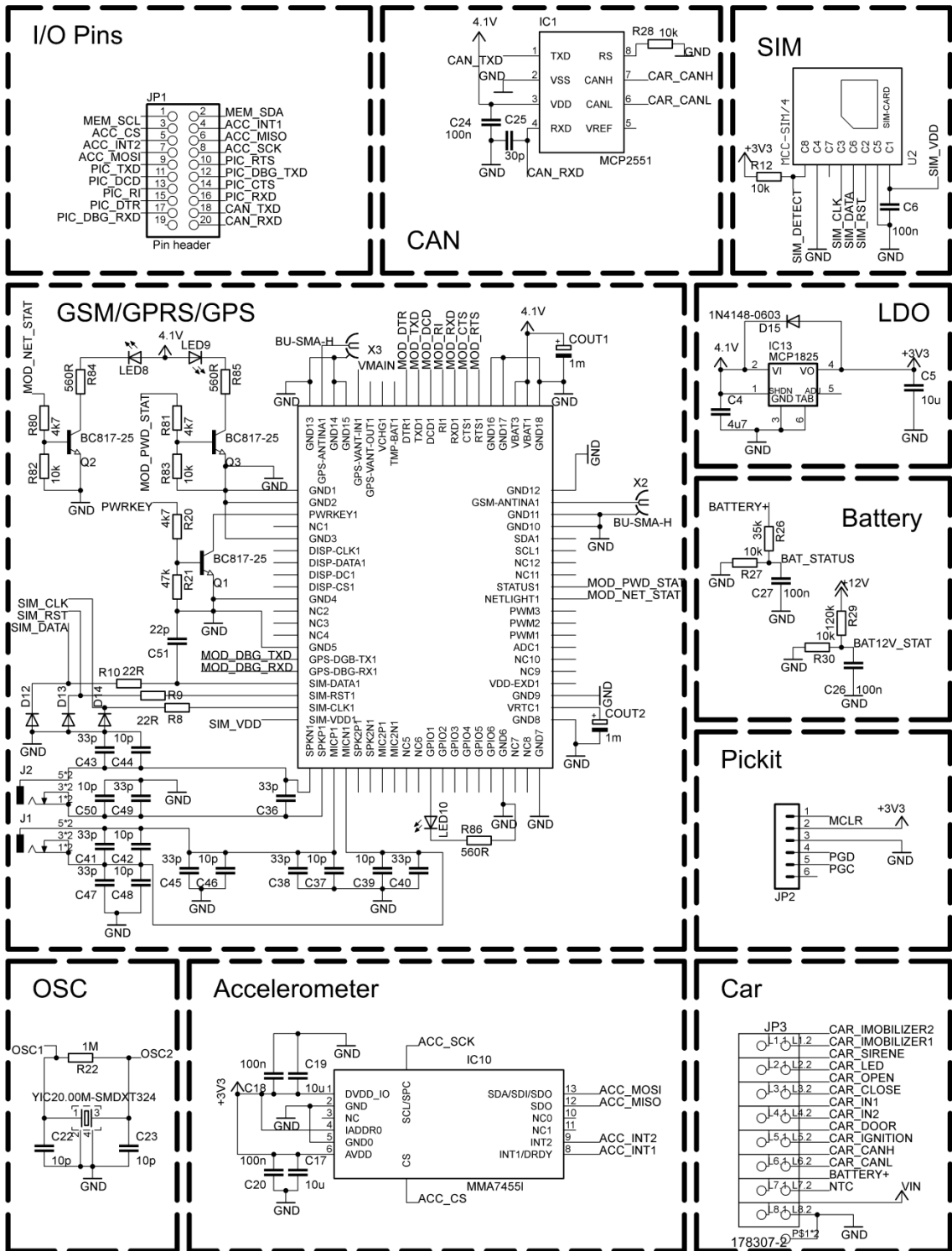
III. Přílohy

Seznam SMS příkazů pro autoalarm

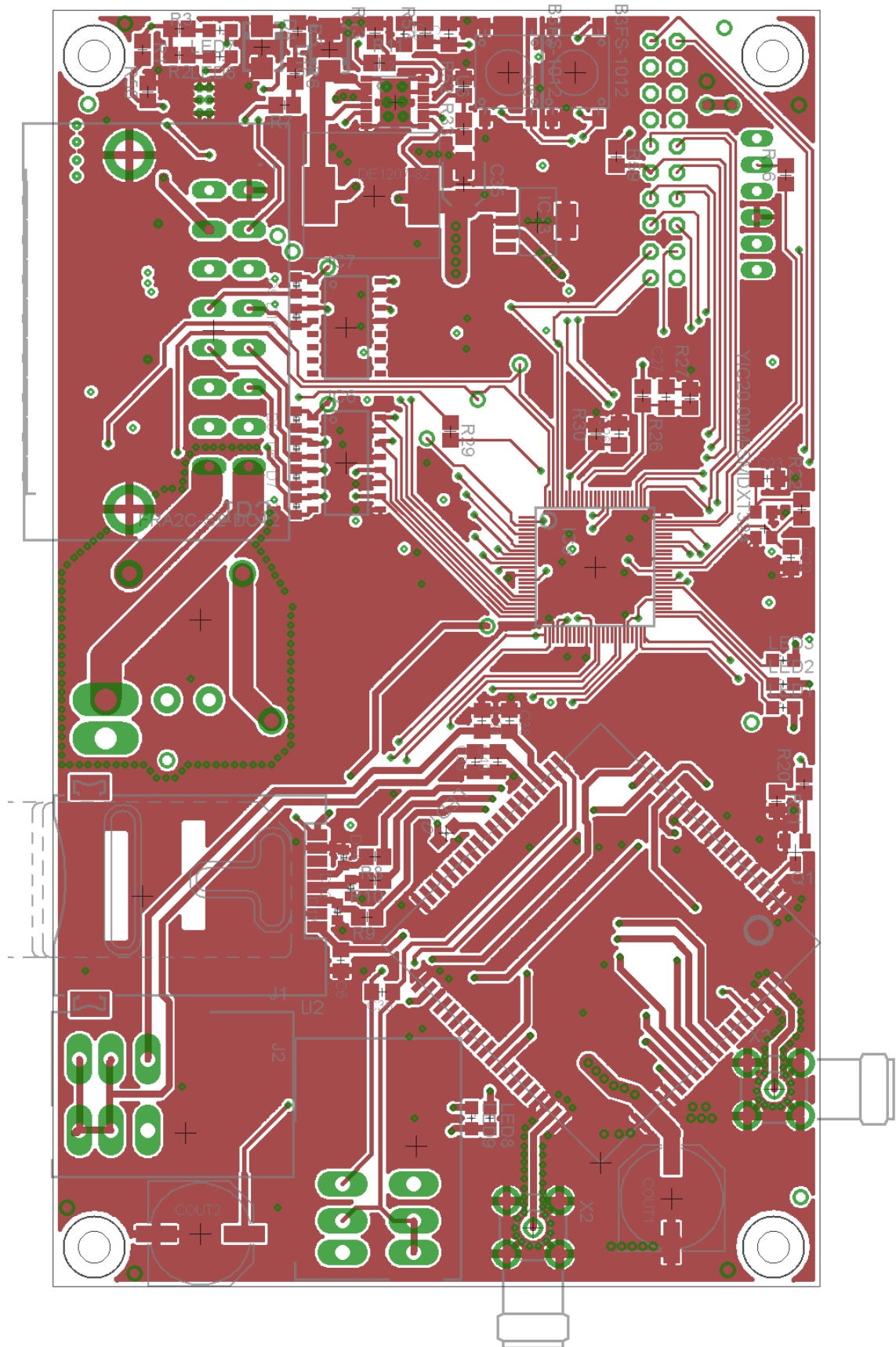
- **SIRENAON** – spustí sirénu
- **SIRENAOFF** – vypne sirénu
- **IMOBON** – sepne imobilizační okruh
- **IMOBOFF** – vypne imobilizační okruh
- **STATUS** – pošle SMS s informací o poplachových vstupech (sepnutý - rozepnutý)
- **GPSINFO** – pošle SMS s pozicí, rychlostí a směrem pohybu
- **SLEDUJ** – začne odesílat GPS data na FTP server, zobrazování pozice na webové adrese s Google Maps
- **NAHRAJ** – pošle informaci o aktuální poloze na FTP server
- **ZMENA1** – žádost o změnu prvního telefonního čísla, které ovládá autoalarm
- **ZMENA2** – žádost o změnu druhého telefonního čísla, které ovládá autoalarm
- **ZMENA3** – žádost o změnu třetího telefonního čísla, které ovládá autoalarm
- **HELLO** – pošle testovací SMS zprávu
- **RESET** – provede opětovný start autoalarmu
- **SLEEP** – vypne GPS a GPRS systémy
- **GPRSAPN** – změna jména přístupové brány GPRS
- **FTPSERVER** – adresa FTP serveru, kam chceme posílat GPS data
- **FTPUSER** – uživatelské jméno na FTP serveru, kam chceme posílat GPS data
- **FTPHEŠLO** – uživatelské heslo na FTP serveru, kam chceme posílat GPS data
- **HESLO** – změna systémového hesla pro změnu telefonního čísla ovládajícího autoalarm
- **ZAPNOUT** – uvede autoalarm do pohotovostního režimu
- **VYPNOUT** – autoalarm nebude hlásit poplach, ale ostatní funkce fungují nadále
- **INTERNET** – zahájí GPRS spojení

Schéma a DPS autoalarmu

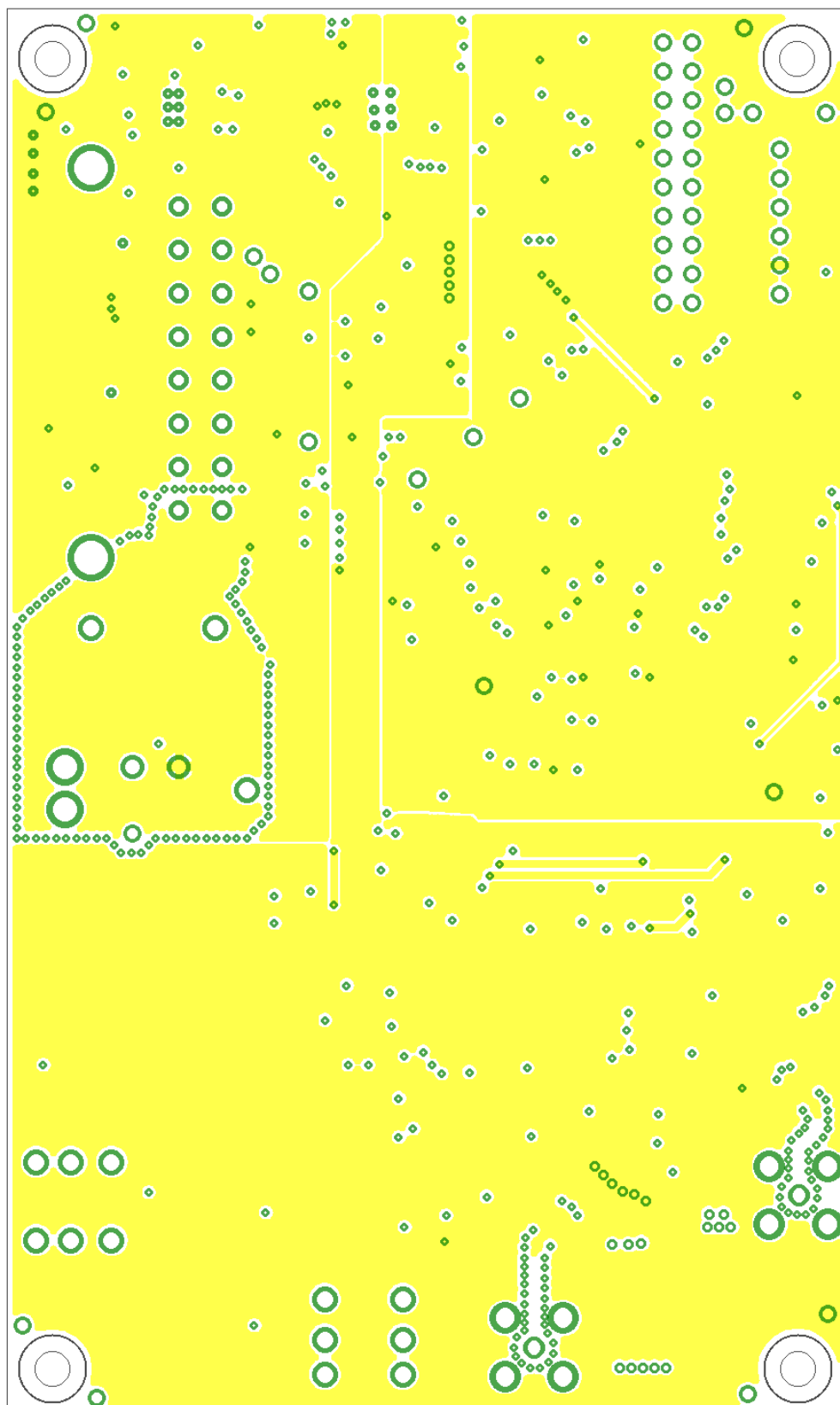




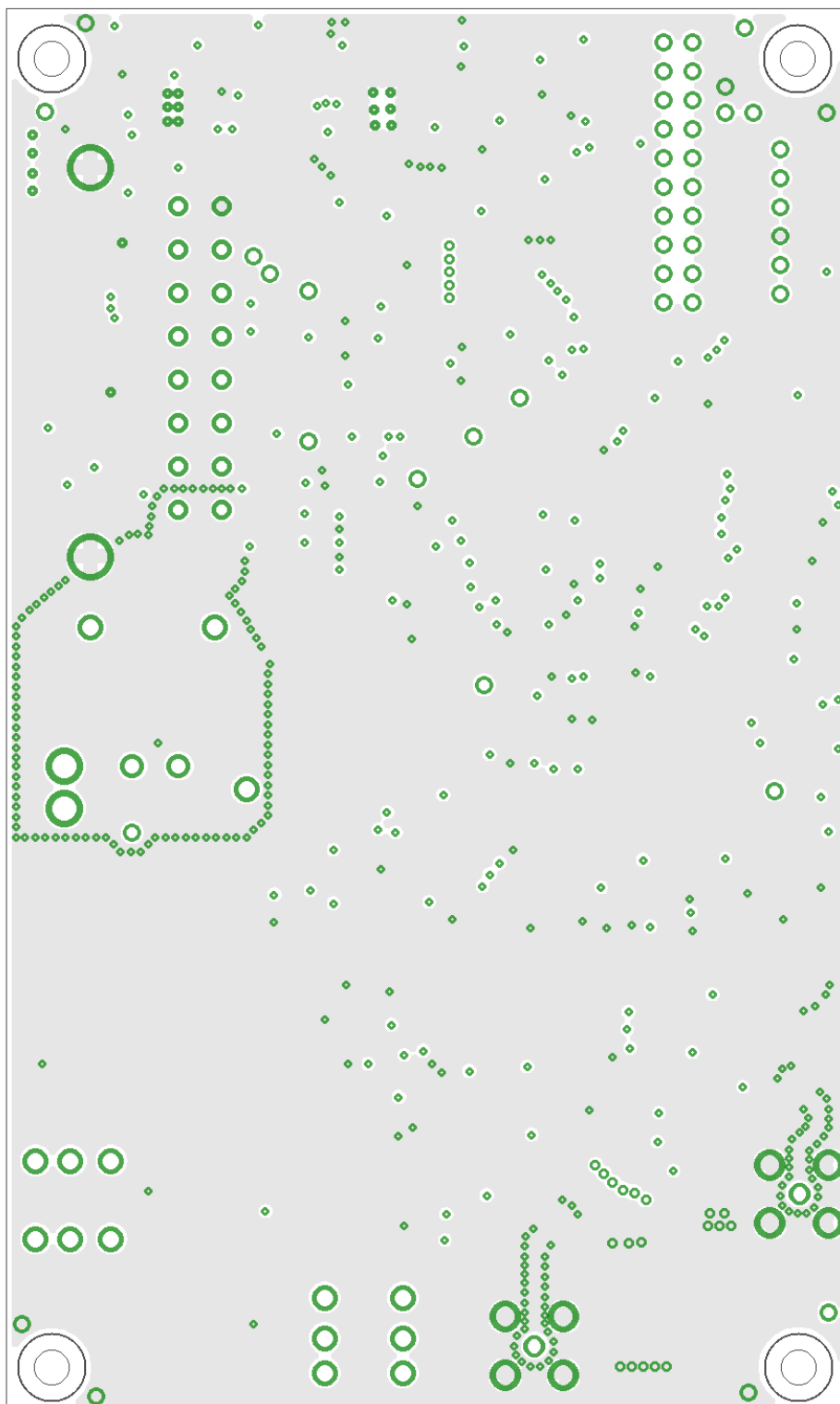
Vrchní vrstva DPS



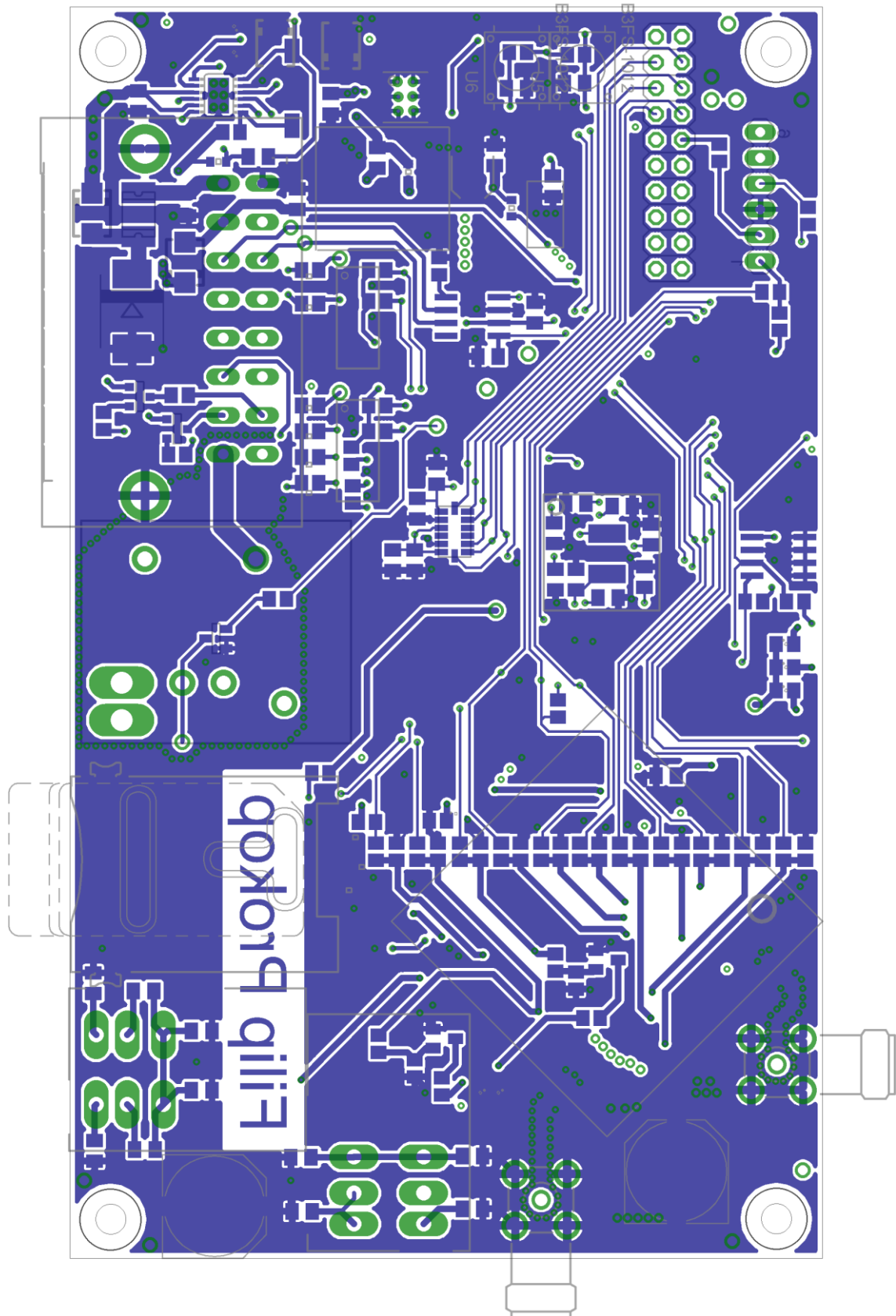
Napájecí vrstva DPS



GND vrstva DPS



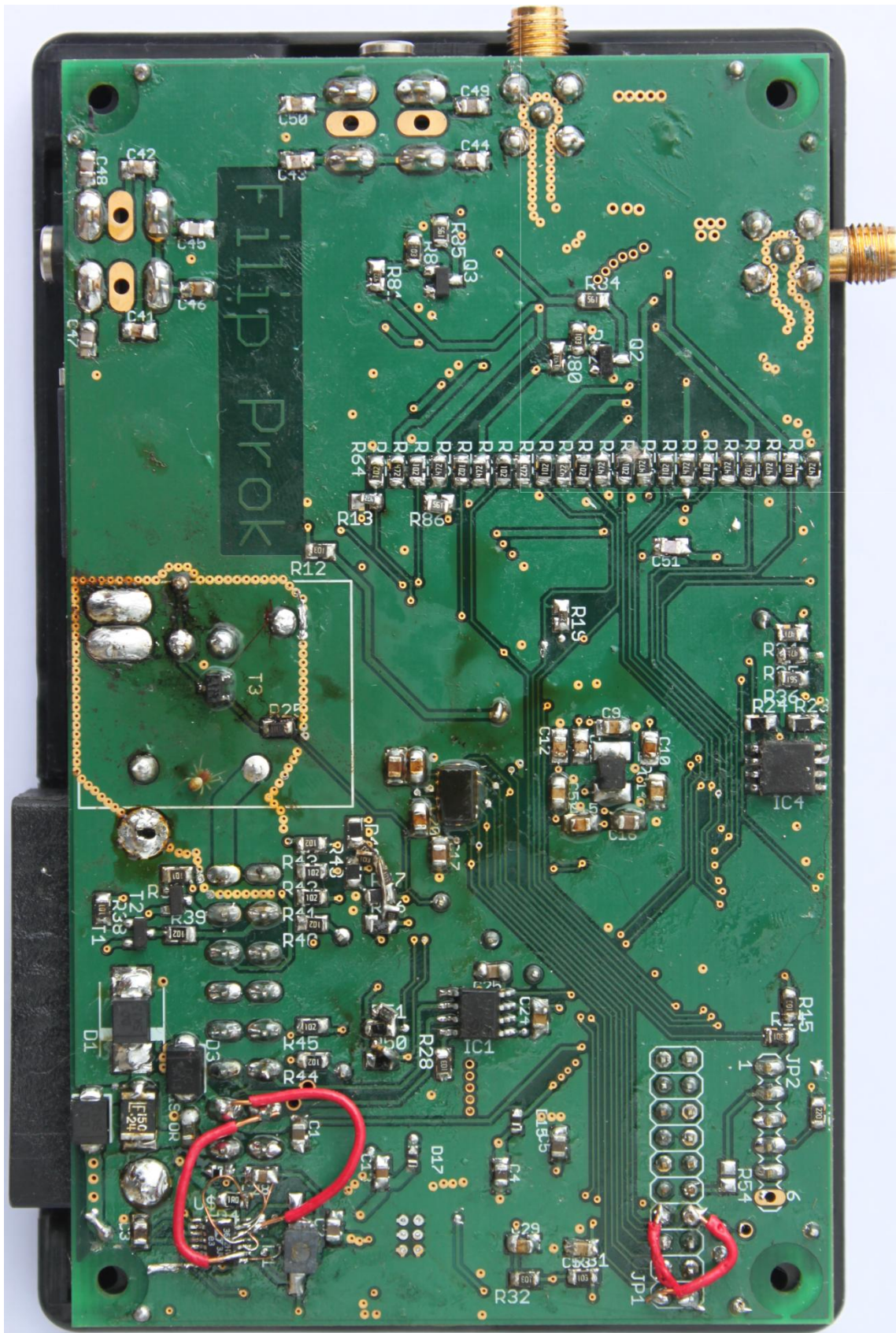
Spodní vrstva DPS



Fotografie autoalarmu







Obsah příloženého CD

- CAN – Zdroje ke sběrnici CAN
- Datasheety součástek – Dokumentace použitých součástek v autoalarmu
- Firmware – Obsahuje materiály potřebné k vývoji programu
- Jablotron manuály – Dokumentace k produktům firmy Jablotron
- Schema – Schéma a DPS autoalarmu
- SIM908 – Dokumentace k modulu SIM908
- Zdroje – Obsahuje většinu použitých zdrojů