



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV MIKROELEKTRONIKY

DEPARTMENT OF MICROELECTRONICS

PŘEVODNÍK SBĚRNICE SENT NA SPI

SENT TO SPI CONVERTER

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Petr Mikulášek

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Michal Pavlík, Ph.D.

BRNO 2016



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV MIKROELEKTRONIKY

DEPARTMENT OF MICROELECTRONICS

PŘEVODNÍK SBĚRNICE SENT NA SPI

SENT TO SPI CONVERTER

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Petr Mikulášek

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Michal Pavlík, Ph.D.

BRNO 2016

Abstrakt

Předkládaná práce se zabývá popisem sběrnice SENT dle standardu SAE-J2716 a jejího převodu na sběrnici SPI. V první části seznamuje čtenáře s fungováním obou sběrnic, poté se již věnuje řešení převodníku. Definuje komunikační protokol, použitý na sběrnici SPI, popisuje kompletní návrh hardware převodníku a v poslední části se zabývá návrhem firmware tohoto převodníku.

Abstract

This work deal with description of the SENT bus as defined in SAE-J2716 and its conversion to the SPI bus. In the first part there is brief introduction to both buses functionality and then its focused on the converter design. It defines the communication protocol used for the SPI communication, describes complete design of the converter hardware and it deal with the firmware in the last part of the document.

Klíčová slova

Převodník, MSP430, SAE-J2716, SENT, SPI

Keywords

Converter, MSP430, SAE-J2716, SENT, SPI

Prohlášení

Prohlašuji, že svou bakalářskou práci na téma „Převodník sběrnice SENT na SPI“ jsem vypracoval samostatně pod vedením vedoucího semestrálního projektu a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením tohoto projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení platných zákonů.

V Brně dne 30.05.2016

.....

Poděkování

Děkuji manželce Miroslavě a dceři Štěpánce za trpělivost a podporu při studiu

V Brně dne 30.05.2016

.....

podpis autora

BRNO 2016

Obsah

| | |
|--|----|
| Abstrakt..... | 0 |
| Seznam zkratk a pojmů..... | 0 |
| 1.Sběrnice SENT..... | 4 |
| 1.1.Fyzická vrstva..... | 4 |
| 1.2.Komunikační protokol..... | 5 |
| 1.3.Sériové zprávy..... | 5 |
| 1.3.1.Krátká sériová zpráva..... | 6 |
| 1.3.2.Rozšířená sériová zpráva..... | 6 |
| 2.Sběrnice SPI..... | 7 |
| 2.1.Fyzická vrstva sběrnice..... | 7 |
| 2.2.Komunikace po sběrnici..... | 8 |
| 3.Komunikační protokol sběrnice SPI..... | 9 |
| 3.1.Ovládání převodníku..... | 9 |
| 3.2.Čtení dat z převodníku..... | 10 |
| 3.2.1.Hlavička vysílaného paketu..... | 11 |
| 3.2.2.Formát bloku pro odeslání zprávy..... | 11 |
| 3.2.3.Formát pro odeslání zprávy ESM..... | 12 |
| 3.2.4.Formát pro odeslání zprávy SSM..... | 13 |
| 3.2.5.Formát zprávy pro konec dat..... | 14 |
| 3.3.Možné chyby při komunikaci..... | 14 |
| 4.Hardware převodníku..... | 15 |
| 4.1.Rozhraní SENT..... | 15 |
| 4.1.1.Obvod přijímače..... | 16 |
| 4.2.Rozhraní SPI..... | 18 |
| 4.2.1.Zapojení konektoru..... | 18 |
| 4.2.2.Zapojení rozhraní SPI..... | 19 |
| 4.3.Uživatelské rozhraní..... | 21 |
| 4.4.Zapojení bloku mikrokontroléru..... | 22 |
| 4.4.1.Časování mikrokontroléru..... | 23 |
| 4.4.2.Zapojení ladícího rozhraní..... | 23 |
| 4.4.3.RESET, detekce podpětí a kontrola běhu aplikace..... | 24 |
| 4.4.4.Schéma zapojení mikrokontroléru..... | 25 |
| 4.5.Napájecí zdroj..... | 25 |
| 4.5.1.Napájení sběrnice SENT..... | 25 |
| 4.5.1.Napájecí hladina mikrokontroléru..... | 26 |
| 4.5.2.Napájení sběrnice SPI..... | 27 |
| 4.5.3.Napájecí hladina +5 V..... | 29 |
| 4.5.4.Diagnostika napájení..... | 29 |
| 4.5.5.Zapojení napájecího zdroje..... | 29 |
| 4.6.Doplnění schématu..... | 30 |
| 4.7. Realizace prototypu..... | 31 |
| 4.7.1.Provedení desek s plošnými spoji..... | 32 |
| 4.7.2.Seznam součástek..... | 32 |
| 4.7.3.Gerber data..... | 33 |
| 5.Firmware..... | 34 |

| | |
|--|----|
| 5.1.Přijímač SENT..... | 34 |
| 5.1.1.Analýza časování..... | 34 |
| 5.1.2.Přerušení od časovače..... | 34 |
| 5.1.3.Vyrovňovací paměť..... | 35 |
| 5.1.4.Architektura přijímače..... | 35 |
| 5.1.5.Zpracování signálu..... | 37 |
| 5.1.6.Zpracování sériových zpráv..... | 39 |
| 5.1.7.Chyby přijímače a jejich ošetření..... | 40 |
| 5.2.Rozhraní SPI..... | 40 |
| 5.2.1.Příjem..... | 40 |
| 5.2.2.Vysílání..... | 41 |
| 5.3.Diagnostika a management napájení..... | 42 |
| 5.4.Výstup přerušení..... | 42 |
| 5.5.Práce s indikačními LED..... | 43 |
| 5.6.Časování jednotlivých úloh..... | 43 |
| 5.7.Podklady pro tvorbu firmware..... | 44 |
| 6.Závěr..... | 45 |
| 6.1.Realizace hardware..... | 45 |
| 6.2.Realizace firmware..... | 46 |
| 6.3.Celkové hodnocení projektu..... | 47 |

Seznam obrázků

| | |
|---|----|
| Obrázek 1.1: Schéma přenášeného paketu..... | 5 |
| Obrázek 2.1: Topologie sběrnice SPI..... | 7 |
| Obrázek 2.2: Režimy sběrnice SPI..... | 8 |
| Obrázek 3.1: Struktura vysílaných dat..... | 10 |
| Obrázek 4.1: Blokové schéma hardware..... | 15 |
| Obrázek 4.2: Schéma přizpůsobení rozhraní SENT..... | 18 |
| Obrázek 4.3: Schéma zapojení rozhraní SPI..... | 20 |
| Obrázek 4.4: Rozsahy napětí vstupů a výstupů pro základní technologie logických obvodů..... | 21 |
| Obrázek 4.5: Schéma zapojení mikrokontroléru, ladícího rozhraní a napájecího zdroje..... | 25 |
| Obrázek 4.6: Zapojení přepínače napájení s dvojicí tranzistorů PNP..... | 27 |
| Obrázek 4.7: Výsledky simulace přepínače napájení s dvojicí tranzistorů PNP..... | 28 |
| Obrázek 4.8: Schéma zapojení napájecí části převodníku..... | 29 |
| Obrázek 4.9: Zbývající části schématu zapojení převodníku..... | 30 |
| Obrázek 4.10: Mechanické provedení desky převodníku (není v měřítku)..... | 31 |
| Obrázek 4.11: Gerber data navržené desky..... | 33 |
| Obrázek 5.1: Minimální velikost FIFO pro přijímač SENT..... | 36 |
| Obrázek 5.2: Architektura přijímače SENT na nejvyšší úrovni s vyznačením datových toků..... | 37 |
| Obrázek 5.3: Stavový automat přijímače SENT..... | 37 |
| Obrázek 5.4: Princip hledání synchronizačního impulsu..... | 39 |
| Obrázek 5.5: Datové toky pro vysílač SPI..... | 41 |
| Obrázek 5.6: Stavový automat vysílače SPI..... | 42 |
| Obrázek 6.1: Modifikované zapojení fyzické vrstvy SENT..... | 46 |

Seznam tabulek

| | |
|--|----|
| Tabulka 1.1: Elektrické parametry sběrnice SENT na straně přijímače..... | 4 |
| Tabulka 1.2: Stavový nibble..... | 6 |
| Tabulka 1.3: Formát krátké sériové zprávy..... | 6 |
| Tabulka 1.4: Formát rozšířené sériové zprávy..... | 6 |
| Tabulka 2.1: Signály sběrnice SPI..... | 7 |
| Tabulka 3.1: Paket s povelom..... | 9 |
| Tabulka 3.2: Význam jednotlivých polí v tabulce 3.1..... | 9 |
| Tabulka 3.3: Rozlišení jednotlivých bloků dat při komunikaci..... | 10 |
| Tabulka 3.4: Hlavička dat, přenášených do SPI masteru..... | 11 |
| Tabulka 3.5: Význam jednotlivých polí v tabulce 3.4..... | 11 |
| Tabulka 3.6: Organizace dat v bloku zprávy..... | 12 |
| Tabulka 3.7: Význam jednotlivých částí zprávy..... | 12 |
| Tabulka 3.8: Organizace dat v bloku zprávy ESM..... | 12 |
| Tabulka 3.9: Význam jednotlivých částí zprávy ESM..... | 13 |
| Tabulka 3.10: Organizace dat v bloku zprávy SSM..... | 13 |
| Tabulka 3.11: Význam jednotlivých částí zprávy SSM..... | 13 |
| Tabulka 3.12: Indikátor konce dat..... | 14 |
| Tabulka 3.13: Přehled chyb komunikace na sběrnici SPI..... | 14 |
| Tabulka 4.1: Požadavky na vstup ze sběrnice SENT..... | 16 |
| Tabulka 4.2: Vhodné hodnoty rezistorů..... | 17 |
| Tabulka 4.3: Zapojení konektoru sběrnice SPI..... | 19 |
| Tabulka 4.4: Přehled možných diod LED..... | 22 |
| Tabulka 4.5: Funkce V/V bran mikrokontroléru..... | 22 |
| Tabulka 4.6: Zapojení konektoru CON3 ladícího rozhraní..... | 24 |
| Tabulka 4.7: Možné příčiny restartu mikrokontroléru..... | 24 |
| Tabulka 4.8: Rekapitulace napájecích napětí a proudů v převodníku..... | 25 |
| Tabulka 4.9: Zatížení napájení UDD..... | 26 |
| Tabulka 4.10: Seznam součástek převodníku SENT-SPI..... | 32 |
| Tabulka 5.1: Mezní hodnoty časování sběrnice SENT..... | 34 |
| Tabulka 5.2: Popis jednotlivých stavů přijímače..... | 37 |
| Tabulka 5.3: Přehled chybových stavů přijímače..... | 40 |
| Tabulka 5.4: Přehled přerušení na sběrnici SPI..... | 43 |
| Tabulka 5.5: Přehled úloh operačního systému..... | 43 |
| Tabulka 6.1: Hodnoty součástek změněné z důvodu výměny zdroje..... | 45 |
| Tabulka 6.2: Porovnání parametrů rozhraní SENT po výměně zdroje..... | 45 |
| Tabulka 6.3: Porovnání parametrů rozhraní SENT po výměně zdroje..... | 46 |
| Tabulka 6.4: Srovnání použitého mikrokontroléru s rodinou STM32..... | 48 |

Seznam příloh

| | |
|--|----|
| Příloha 1. Obsah přiloženého CD-ROM..... | I |
| Příloha 2. Použitý software..... | II |
| Příloha 3. Použitý hardware..... | IV |
| Příloha 4. Výpis klíčových částí programu..... | V |

Seznam zkratek a pojmů

| | |
|--------------------------------------|--|
| Brownout | Detekce podpětí, zabudovaná v mikrokontroléru. Při poklesu napájecího napětí restartuje mikrokontrolér. |
| CAN | <i>Controller Area Network</i> . Komunikační standard, vyvinutý firmou Bosch, často používaný v automobilech. |
| CMOS | <i>Complementary Metal Oxide Semiconductor</i> . Technologie výroby integrovaných obvodů. |
| CPU | <i>Central Processing Unit</i> . Centrální procesorová jednotka, zde jádro mikrokontroléru nebo SPI masteru. |
| CRC, CRC Polynom | <i>Cyclic Redundancy Code</i> . Polynom, vypočítaný z dat a přidaný obvykle na konec zprávy pro ověření, že zpráva není poškozena. |
| EMC | <i>Electro Magnetic Compatibility</i> . Schopnost nerušit ostatní zařízení a odolávat rušení. |
| ESM, Rozšířená sériová zpráva | <i>Enhanced Serial Message</i> . Jeden ze dvou druhů sériové zprávy, popsány v kapitole 1.3.2. |
| FIFO | <i>First In, First Out</i> . Druh paměti, ze které je přečten vždy nejstarší záznam. |
| FLASH | Technologie polovodičových pamětí, používaná u mikrokontrolérů k uložení programu. |
| Identifikátor, ID | <i>Identifier</i> . Unikátní číslo, kterým je možné rozlišit například obsah zprávy, příkaz a podobně. |
| LED | <i>Light Emitting Diode</i> . Dioda, emitující světlo. V tomto projektu jsou využity jen na indikaci stavu. |
| Mikrokontrolér, MCU | <i>Micro Controller Unit</i> . Počítač, který integruje jádro, paměti a periferie na jednom čipu. |
| Nibble | Čtveřice bitů. |
| Overflow | Přetečení. Stav, kdy je zaplněna vyrovnávací paměť a dojde k přepsání nebo ztrátě dat. |
| Paket | <i>Packet</i> . Blok dat, který je v celku přenesený komunikačním rozhraním. |
| Periferie | <i>Peripheral</i> . Vstupně-výstupní jednotka, připojená na sběrnici nebo integrovaná v mikrokontroléru. |

| | |
|-----------------------------------|--|
| Protokol | <i>Protocol</i> . Dohoda, popisující způsob přenosu informací po daném rozhraní. |
| Přerušení | <i>Interrupt</i> . Situace, kdy si periferie vyžádá obsluhu asynchronně k probíhající operaci. |
| pull up, pull down | Rezistor, který nastavuje logickou úroveň signálu v klidu. <i>Pull up</i> nastavuje jedničku, <i>pull down</i> nulu. |
| RAM | <i>Random Access Memory</i> . Paměť s náhodným přístupem |
| Reset | Stav, kdy je zařízení uvedeno do výchozího stavu po zapnutí napájení a podobně. |
| SENT, SAE-J2716 | <i>Single Edge Nibble Transmission</i> . Komunikační rozhraní, popsáné v kapitole 3. |
| Sériová zpráva | Zpráva, která je přenesena složením několika paketů na sběrnici SENT. Souhrnně označuje zprávy SSM a ESM |
| SPI | <i>Serial Peripheral Interface</i> . Standard pro komunikaci mezi obvody, popsáný v kapitole 2. |
| SPI Master | Zařízení, které řídí na sběrnici SPI komunikaci. |
| SPI Slave | Zařízení, které komunikuje na sběrnici SPI, ale nemůže iniciovat a řídit komunikaci. |
| SSM, Krátká sériová zpráva | <i>Short Serial Message</i> . Jeden ze dvou druhů sériové zprávy, popsáný v kapitole 1.3.1. |
| SWD | <i>Serial Wire Debug</i> . Ladící rozhraní mikrokontrolérů MSP430. |
| TU | <i>Time Unit</i> . Časová jednotka, použitá u sběrnice SENT. |
| TTL | <i>Transistor-Transistor Logic</i> . Technologie výroby číslicových integrovaných obvodů. |
| TVS | <i>Transient Voltage Suppressor</i> . Dioda, která má za úkol ochranu proti přepětí. Někdy se označuje obchodním názvem <i>Transil</i> . |
| Watchdog | Periferie, která v případě zacyklení programu restartuje CPU. Bývá založena na čítači, který je programově nulován a při přetečení dojde k restartu systému. |
| Zpráva | V kontextu sběrnice SENT se jedná o dva až šest nibblů s CRC polynomem. |

Typografická konvence

Pro snadnější orientaci v textu byla rozlišena slova v těchto významech:

- **Signál**, nebo **vývod integrovaného obvodu** je označen menším, tučným písmem.
- Negovaný signál je označený nadtržením.
- **Typ součástky** je označen tučným bezpatkovým písmem.
- Odkaz na součástku je označen tečkovaným rámečkem.
- **STAVY** ve stavovém automatu jsou zvýrazněny tučnými kapitálkami.
- *Soubory, softwarové balíky* a případně *příkazy* jsou označeny ...
- *Anglická slova* jsou označena kurzívou.

Úvod

Cílem této práce je návrh hardware a firmware převodníku z automobilové sběrnice SENT na sběrnici SPI, která je běžně používaná pro připojení pamětí a periférií v číslicové technice. Cílem byl jak návrh hardware, tak i návrh firmware tohoto převodníku a realizace funkčního prototypu. Navazuje na předcházející semestrální práci [1].

Protože v případě sběrnice SENT jde sice o standard, ale poměrně neznámý, je první kapitola věnována stručnému seznámení s tímto rozhraním a popisu jak požadavků na hardware, tak i komunikaci po tomto rozhraní.

Sběrnice SPI je sice poměrně běžný standard zvláště u mikrokontrolérů, ale základní princip fyzické vrstvy je zrekapitulován ve druhé kapitole, která čerpá z popisu v knize [2].

Protože pro realizaci převodníku je nutné mít oba převáděné protokoly definované na stejné úrovni, je komunikačnímu protokolu sběrnice SPI věnována třetí kapitola. a dále pokračuje definicí komunikačního protokolu na stejnou úroveň, jako je definice protokolu na sběrnici SENT. Bylo zde zohledněno jak formátování dat, tak i časování obou sběrnic a hardwarová omezení převodníku.

Čtvrtá kapitola je zaměřena na návrh hardware, a to v rozsahu od obecných principů jednotlivých bloků až po vlastní návrh plošného spoje a výrobu prototypu. Rozebírá jednotlivé části - komunikační rozhraní, napájecí zdroje nebo zapojení mikrokontroléru.

Byla požadována pokud možno univerzální konstrukce. Požadavky na rozhraní SENT byly jednoznačně dány příslušnou normou SAE-J2716 [3]. Naopak požadavky na rozhraní SPI jsou zadány velmi obecně, vychází z principu této sběrnice a z požadavku na použitelnost převodníku pro logiku 3,3 V i 5 V. Napájení převodníku bylo řešeno univerzálně, vše ostatní bylo nedefinováno.

Protože k provozu zařízení s mikrokontrolérem je vždy nutný firmware, je čtvrtá kapitola věnována právě jemu. Je v ní rozebrána architektura a zvoleny principy, které jsou omezeny protokoly nebo hardwarem. Jsou v ní rozebrána jednotlivá rozhodnutí, učiněná během návrhu firmware a diskutuje jejich dopad na funkci převodníku.

V poslední kapitole je zhodnocena realizace hardware i firmware a získané zkušenosti.

Mimo zadání zde byly stanoveny i mé vlastní cíle. Hlavním soukromým cílem bylo seznámení s rozhraními mikrokontrolérů **MSP430**, která jsem zatím nepoužil v praxi, a to jak ze strany hardware, tak i jejich obsluhy na úrovni firmware. Dalším cílem bylo praktické vyzkoušení řady mikrokontrolérů levné řady **MSP430G2xxx**.

1. Sběrnice SENT

SENT je sběrnice, která byla vyvinuta pro použití v automobilovém průmyslu. Její název je zkratkou ze „*Single Edge Nibble Transmission*“. Popisuje ji norma SAE-J2716 [3].

Sběrnice SENT byla navržena pro automobilový průmysl jako jednodušší, ale stejně spolehlivá náhrada sběrnice CAN. Byla standardizována v roce 2007 a poslední verze protokolu, která je implementována v tomto převodníku, pochází z roku 2010.

Sběrnice SENT pracuje také na úrovni 5 V jako sběrnice CAN, tím ale končí veškerá podobnost. SENT podporuje komunikaci jen dvou zařízení, omezenou jen na jeden směr. Oproti sběrnici CAN nepoužívá na fyzické vrstvě diferenciální pár, ale poskytuje napájení vysílače a vystačí si se třemi vodiči.

Norma pro sběrnici SENT byla navržena tak, aby nebylo nutné na straně přijímače používat speciální přijímače, ani speciální periferie. Na zpracování signálu stačí čítač/časovač v záchytném režimu a na převod a filtrování signálu postačuje šestice pasivních součástek.

1.1. Fyzická vrstva

Sběrnice propojuje vždy jeden vysílač a jeden přijímač pomocí trojice vodičů:

1. Napájení senzoru
2. Data
3. Společný vodič

Statické parametry rozhraní jsou uvedeny v tabulce 1.1. Dávají představu o základních elektrických parametrech.

Tabulka 1.1: Elektrické parametry sběrnice SENT na straně přijímače

| Veličina | Hodnota | | | Jednotka | Poznámka |
|---------------------|---------|------|------|---------------|--|
| | Min | Typ | Max | | |
| U_{SUPPLY} | 4,85 | 5,00 | 5,15 | V | Napájecí napětí pro vysílač |
| I_{OUT} | 20 | | | mA | Napájecí proud pro vysílač |
| C_{LOAD} | | | 10 | μF | Kapacita kondenzátorů ve vysílači |
| U_{GO} | | | 20 | mV | Úbytek napětí na nulovém vodiči |
| R_{PU} | 10 | | 55 | k Ω | Velikost pull up rezistoru |
| C_{IN} | | | 100 | pF | Vstupní kapacita přijímače |
| U_{IL} | | | 1,39 | V | Maximální úroveň pro logickou nulu |
| U_{IH} | 3,8 | | | V | Minimální úroveň pro logickou jedničku |
| U_{HYST} | 0,3 | | | V | Minimální velikost hystereze |

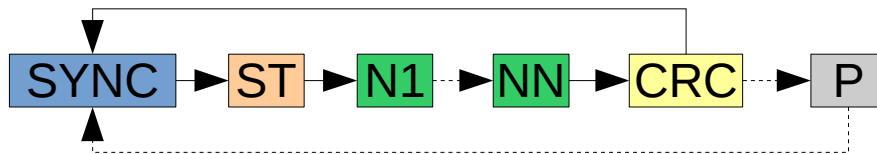
1.2. Komunikační protokol

Komunikační protokol obsahuje tři typy přenosů, jak je popsáno v následujících kapitolách. Vždy při přenosu paketu dochází k přenosu zprávy, během jejich přenosu může současně docházet i k přenosu některého standardního typu sériové zprávy.

Základní přenášenou jednotkou informace na sběrnici je nibble, neboli čtveřice bitů. Je přenášen jako impuls, jehož délka se měří na časové jednotky. Tato jednotka dle normy představuje čas 3 μ s až 90 μ s. Skutečná délka impulsu na sběrnici je vypočtena vysílačem dle vzorce

$$t_{NIBBLE} = TU \cdot (12 + N) [\mu s] \quad (1.1)$$

kde TU je délka časové jednotky a N je hodnota nibblu, tedy číslo v rozsahu nula až 15.



Obrázek 1.1: Schéma přenášeného paketu

Nibbly jsou, společně s povinným synchronizačním pulsem a nepovinnou pauzou po přenosu, sestaveny do paketů s pevně danou strukturou, která je vidět na obrázku 1.1. Paket vždy začíná synchronizačním pulsem délky 56 TU, označený SYNC. Následuje stavový nibble, označený ST a za ním už jsou data zprávy, vysílaná po nibblech. Může se jednat o dva až šest datových nibblů, označený N1 až N_N. Poté následuje CRC polynom, vypočtený z dat a použitý pro kontrolu.

Někdy je vhodné proložit pakety mezerou, nebo srovnat jejich délku, protože čas vysílání paketu se liší dle hodnoty. K tomuto účelu je v protokolu definována pauza, která může mít od 12 TU do 768 TU.

1.3. Sériové zprávy

Standard SENT podporuje dva standardy takzvaných sériových zpráv, v terminologii normy označených jako *serial message*. Sériových proto, že přichází v sérii, po jednom či dvou bitech v každém stavovém nibblu paketu, jak je naznačeno v tabulce 1.2. Smyslem těchto zpráv je například identifikace zařízení.

Byly definovány dva základní formáty těchto zpráv, jak je popsáno dále, a v jednom okamžiku může být vysílána jen jedna z nich, protože sdílí stejné bity stavového nibblu. Liší se od sebe jak počtem paketů, ze kterých jsou získány, tak i délkou dat a počtem hodnot, které je možné přenést.

Obsah těchto zpráv není nijak definován a je ponechána možnost definovat jejich obsah dle potřeby, proto jsou identifikátory a data těchto zpráv předávány tak, jak jsou zachyceny.

Stejně jako standardní zprávy, i sériové zprávy jsou zabezpečeny CRC polynomem.

Tabulka 1.2: Stavový nibble

| Bit | Funkce |
|---------|---------------------------------|
| 0 (LSB) | Rezervováno. Nastaveno na nulu. |
| 1 | |
| 2 | Sériová data |
| 3 (MSB) | Start zprávy / Sériová data |

1.3.1. Krátká sériová zpráva

Prvním způsobem doplňkové komunikace je krátká sériová zpráva (*Short serial message*, dále SSM). Je navržena k přenosu jednoho bajtu s čtyřbitovým identifikátorem a čtyřbitovým CRC polynomem. K přenosu této zprávy je nutných 16 paktů. Formát dat je v tabulce 1.3

Tabulka 1.3: Formát krátké sériové zprávy

| Status bit | Číslo zprávy | | | | | | | | | | | | | | | |
|----------------|--------------|---|---|---|---------------|---|---|---|---|----|----|----|-----|----|----|----|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 3 Start zprávy | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 Sériová data | ID zprávy | | | | Data (1 bajt) | | | | | | | | CRC | | | |

1.3.2. Rozšířená sériová zpráva

Tato zpráva (*Enhanced Serial Message*, ESM) je popsána v tabulce 1.4. Umožňuje přenos ve dvou formátech:

- Čtyřbitový identifikátor a 16 bitů dat (bit C = 0)
- Osmibitový identifikátor a 12 bitů dat (bit C = 1)

V obou případech trvá přenos 18 zpráv a je zabezpečeny šestibitovým CRC polynomem.

Tabulka 1.4: Formát rozšířené sériové zprávy

| Status bit | Číslo zprávy | | | | | | | | | | | | | | | | | | |
|-------------------------------|--------------|---|---|---|---|---|----------|---|----------|----|----|----|----|----------|----|----|-----------|----|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | |
| 3 Start zprávy / sériová data | 1 | 1 | 1 | 1 | 1 | 1 | 0 | C | ID[7..4] | | | | 0 | ID[3..0] | | | D[15..12] | | 0 |
| 2 Sériová data | CRC[5..0] | | | | | | D[11..0] | | | | | | | | | | | | |

2. Sběrnice SPI

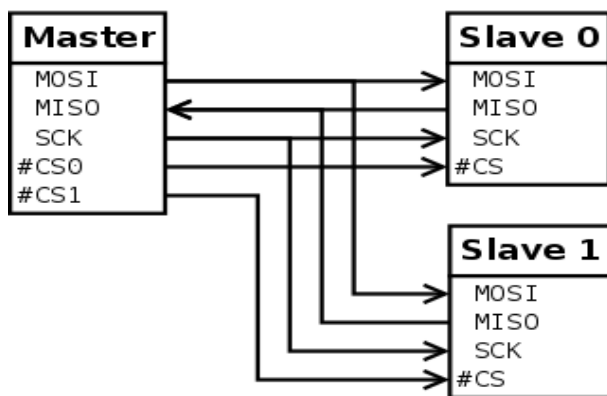
SPI znamená *Serial Peripheral Interface*. Jak prozrazuje její název, jde o sériovou sběrnici, určenou pro připojení periférií k mikrokontrolérům. V principu se jedná o páry posuvných registrů, které přenášejí data po jednom datovém vodiči pro každý směr.

2.1. Fyzická vrstva sběrnice

Typická topologie sběrnice se skládá z jednoho obvodu master a jednoho nebo několika obvodů slave (Obrázek 2.1). Master řídí komunikaci, vybírá slave obvod a generuje hodinový signál.

Vlastní signály jsou v logických úrovních procesoru a periférií, obvykle (LV)CMOS s napětím 5 V; 3,3 V; 2,5 V nebo 1,8 V. V případě potřeby je možné použít převodník úrovní nebo galvanické oddělení.

Značení signálů a jejich význam jsou v tabulce 2.1.



Obrázek 2.1: Topologie sběrnice SPI

Mimo těchto signálů, definovaných standardem SPI, mohou být užitečné další dva signály. Prvním z nich je **RESET**, který umožní uvedení periférie do definovaného stavu, například odpojení výstupů v situaci, kdy nejsou pod kontrolou mikrokontroléru. Na periférii jde většinou o vstup.

Druhým je $\overline{\text{INT}}$. Tento signál umožní vyvolat přerušení a tím eliminovat dotazy na stav periférie. Jeho přítomnost se projeví pozitivně jak zatížením CPU, tak i spotřebou a zlepšením parametrů EMC díky kratší době komunikace, nahrazené statickým signálem.

Tabulka 2.1: Signály sběrnice SPI

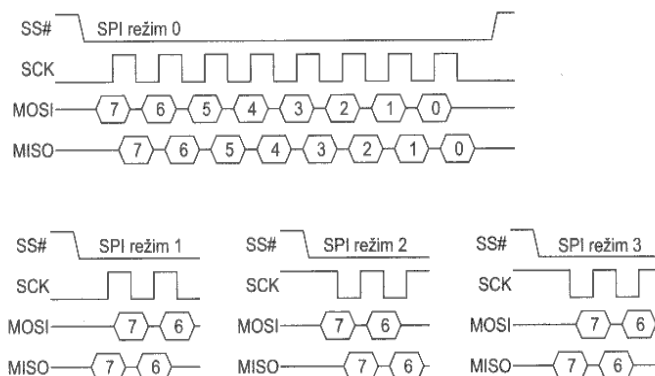
| Signál | Označení | Význam | Poznámka |
|------------------------|--------------------------------------|--|--|
| Povinné signály | | | |
| MOSI | <i>Master Out</i> <i>Slave In</i> | Přenáší data z obvodu Master na obvody Slave | Jeden z nich lze vynechat při jednosměrné komunikaci |
| MISO | <i>Master In</i> <i>Slave Out</i> | Přenáší data z vybraného obvodu Slave do obvodu Master | |

| Signál | Označení | Význam | Poznámka |
|--|--------------------|---|--------------------------------|
| SCK | <i>Slave Clock</i> | Hodiny pro obvody Slave | |
| Nepovinné signály | | | |
| $\overline{\text{CSn}}$ | <i>Chip select</i> | Výběr obvodu Slave, negovaný | n je číslo obvodu Slave |
| Často používané signály, které nejsou definovány standardem | | | |
| $\overline{\text{RESET}}$ | <i>Reset</i> | Uvede periférii do definovaného stavu při zapnutí zařízení nebo restartu | Obvykle negovaný |
| $\overline{\text{INT}}$ | <i>Interrupt</i> | Indikace události pro nadřazený obvod. Slouží k eliminování opakovaného dotazování na stav periférie. | Obvykle s otevřeným kolektorem |

2.2. Komunikace po sběrnici

Podle polariry a fáze hodinového signálu jsou definovány tři režimy provozu sběrnice. Jejich přehled je na obrázku 2.2. Důležitá vlastnost sběrnice je, že data jsou současně přenášena oběma směry. Protokol je tedy vhodné navrhnout tak, aby docházelo k zahzení minima přenášených bajtů a optimalizoval se tak přenos.

Komunikační protokol se liší dle využití sběrnice a fakticky neexistuje standardní formát přenosu dat po této sběrnici, vyjma NOR FLASH paměti a podobných, hromadně vyráběných obvodů. Návrh vlastního komunikačního protokolu je proto v následující kapitole.



Obrázek 2.2: Režimy sběrnice SPI

Rychlost komunikace je odvozena od frekvence signálu SCK. Tento signál obvykle generuje SPI Master.

3. Komunikační protokol sběrnice SPI

V této kapitole je popsán komunikační protokol, který rozšiřuje standard SPI na ekvivalentní úroveň rozhraní SENT. Bez jeho definice by nebylo možné realizovat firmware převodníku.

Protože rozhraní SPI používá dva nezávislé kanály, jeden pro příjem a druhý pro vysílání, je i navržený komunikační protokol rozdělený do dvou separátních protokolů. Jednoho řídicího, který ovládá funkci převodníku, a druhý se stará o vysílání stavu, chyb a dat do SPI masteru.

3.1. Ovládání převodníku

V tabulce 3.1 je celá struktura přijímané zprávy. Protože příjem i vysílání probíhají vždy současně a příjem řídicího slova není vždy žádoucí, je přenos započatý start bajtem s danou hodnotou. Pokud bude hodnota jiná než přesně daný klíč, je následující zápis ignorován. Jako klíč byla zvolena hodnota 0xA6.

Struktura řídicího slova (bajt 2) je v tabulce 3.2. V tomto slově jsou tři bity pro nastavení délky přijímané zprávy, jeden bit pro zapnutí nebo vypnutí převodníku a dva bity pro zapnutí jednotlivých kategorií přerušení.

Tabulka 3.1: Paket s povelem

| Bajt | Bit | | | | | | | | Poznámka |
|------|-----|---|---|---|----|---|---|---|--|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | Obsahuje klíč pro aktivaci zápisu. Při odlišné hodnotě jsou následující data ignorována. |
| 1 | EN | M | E | 0 | ML | | N | | Registr ovládání. Obsahuje bity se zapínáním a vypínáním jednotlivých funkcí převodníku. |

Tabulka 3.2: Význam jednotlivých polí v tabulce 3.1

| Položka | Význam |
|---------|--|
| EN | Povolení (<i>Enable</i>) EN = 0: Zakáže funkci převodníku EN = 1: Povolí funkci převodníku |
| M | Povolení přerušení při příchodu zprávy (<i>Message Receive Interrupt Enable</i>) M = 0: Přerušení zakázáno M = 1: Přerušení povoleno |
| E | Povolní přerušení při chybě (<i>Error Interrupt Enable</i>) MRIE = 0: Přerušení zakázáno MRIE = 1: Přerušení povoleno |

| Položka | Význam |
|---------|--|
| ML | Délka zprávy (<i>Message Length</i>) ML = 0: Rezervováno ML = 1..6: Nastavení délky ML = 7: Rezervováno |
| N | Indikace dalšího bajtu pro případ rozšíření funkcionality. NEXT=0: Konec zprávy. NEXT=1: Následuje další bajt. |

3.2. Čtení dat z převodníku

Při čtení dat je třeba zohlednit, že SPI master pravděpodobně nebude komunikovat po přijetí každé zprávy ze sběrnice SENT a vzhledem k chronologii přijatých zpráv je vhodné využít vyrovnávací paměť FIFO s přenosem několika zpráv v jednom paketu.

V praxi to znamená, že nejdříve musí být odeslána hlavička paketu, ve které je stav převodníku a počet čekajících datových bloků. Dále jsou přenášeny bloky dat z vyrovnávací paměti. Každý blok obsahuje jednu zprávu, sériovou zprávu nebo jinou informaci. První bajt v bloku rozlišuje jeho typ s pomocí dolní pěti bitů. Seznam těchto identifikátorů bloků je v tabulce 3.3.

Tabulka 3.3: Rozlišení jednotlivých bloků dat při komunikaci

| První bajt | Význam dat |
|------------|---|
| xxx1aaaa | SSM zpráva. „a“ je ID zprávy. |
| xxx01bbb | Přenesená data. „b“ je počet datových nibblů v rozsahu 2-6. |
| xxx001cc | Ladění. „c“ je typ ladící zprávy. |
| xxx0001d | ESM zpráva. „d“ určuje, zda se jedná o 12b nebo 16b data. |
| xxx00001 | Chybová zpráva. Není momentálně implementována |
| xxx00000 | Konec zpráv. |

Po vyčerpání dat, určených k přenosu v rámci aktuálního paketu, je třeba zamezit dalšímu čtení. Proto je definován prázdný blok s identifikátorem 0, který znamená, že v rámci paketu nejsou k dispozici další data.



Obrázek 3.1: Struktura vysílaných dat

V té době ale už může být v paměti připraven další blok dat pro vyčtení v dalším paketu. Pro je nutné v případě nuly ukončit komunikaci a je-li povolené a aktivní přerušování, zkusit opakované čtení dat.

Vlastní organizace vnitřku paketu je na obrázku 3.1. Komunikace začne aktivací signálu \overline{CS} , následuje stav a cyklicky všechny bloky dat, které jsou v daném okamžiku připraveny. Poté následují jednotlivé bloky zpráv a po jejich vyčerpání, neukončí-li master komunikaci, následuje vysílání prázdného bloku až do náběžné hrany signálu \overline{CS} .

3.2.1. Hlavička vysílaného paketu

Jednotlivé položky hlavičky jsou specifikovány v tabulce 3.4. Zašedlé nuly jsou rezerva a v příští verzi protokolu mohou být využity, přijímač by je tedy měl ignorovat.

Tabulka 3.4: Hlavička dat, přenášených do SPI masteru

| Bajt | Bit | | | | | | | | Poznámka |
|------|-----|-------|----------|----|----|---|---|-------------------------------------|---------------------------------------|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 0 | D | Verze | | | | | | | Bajt musí být konstantní – HW omezení |
| 1 | P5 | PS | PT | TO | TD | 0 | 0 | 0 | Stav systému převodníku |
| 2 | 0 | 0 | MESSAGES | | | | | Počet zpráv, připravených k přenosu | |

Tabulka 3.5: Význam jednotlivých polí v tabulce 3.4

| Položka | Význam |
|----------|---|
| D | Ladění (<i>Debug</i>). Standardně nastaveno na 0, je-li v 1, může se vyskytovat i jiný, nestandardní formát zprávy pro ladění. Tato volba v současné verzi firmware nebyla použita. |
| Verze | Konstanta, v současné době 1 |
| P5 | Jednička znamená přítomnost napájení +5V |
| PS | Jednička znamená přítomnost napájení sběrnice SPI |
| PT | Jednička znamená, že je zapnuto napájení vysílače SENT |
| TO | Jednička indikuje přetížení zdroje pro vysílač |
| TD | Jednička indikuje, že přichází data z vysílače SENT |
| MESSAGES | Počet zpráv, které čekají na vysílání |

3.2.2. Formát bloku pro odeslání zprávy

Nejčastější by za normálního provozu měla být standardní zpráva. Její formát je v tabulce 3.6 a popis jednotlivých polí v tabulce 3.7. Specifikem je zde variabilní délka, závislá na počtu nibblů.

Tabulka 3.6: Organizace dat v bloku zprávy

| Bajt | Bit | | | | | | | | Poznámka |
|------|-----|---|---|---|----|-----|---|---|---------------|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 0 | O | C | D | 0 | 1 | NIB | | | |
| 1 | N1 | | | | N2 | | | | |
| 2 | N3 | | | | N4 | | | | Pokud NIB > 2 |
| 3 | N5 | | | | N6 | | | | Pokud NIB > 4 |

Tabulka 3.7: Význam jednotlivých částí zprávy

| Pole | Význam |
|------|--|
| O | Přetečení dat (<i>Overflow</i>). Jednička značí, že přetekla vyrovnávací paměť a tato zpráva přepsala některou starší. |
| C | Příznak shody CRC. Pokud je jednička, CRC polynom souhlasí. |
| D | Příznak poškození (<i>Damage</i>). Je nastaven, pokud nastala nějaká chyba, například ztráta synchronizace. |
| NIB | Počet přenesených nibblů. Může být v rozsahu 2..6 |
| N1 | První přijatý nibble |
| N2 | Druhý přijatý nibble |
| N3 | Třetí přijatý nibble (bajt je přítomen, pokud NIB > 2) |
| N4 | Čtvrtý přijatý nibble, pokud NIB > 3. Jinak nuly. |
| N5 | Pátý přijatý nibble (bajt je přítomen, pokud NIB > 4) |
| N6 | Šestý přijatý nibble, pokud NIB > 5. Jinak nuly. |

3.2.3. Formát pro odeslání zprávy ESM

U zprávy ESM jsou možné dva formáty dat, popsané v kapitole 1.3.2. Jejich rozlišení je realizováno bitem 0 v identifikátoru bloku dat. V závislosti na typu zprávy je následně využita odpovídající část zprávy pro uložení identifikátoru a dat.

Blok dat pro zprávu ESM je popsán v tabulkách 3.8 a 3.9.

Tabulka 3.8: Organizace dat v bloku zprávy ESM

| Bajt | Bit | | | | | | | | Poznámka |
|------|-----|---|---|---|-----|---|---|---|----------|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 0 | O | C | D | 0 | 0 | 0 | 1 | F | |
| 1 | IDH | | | | IDL | | | | |
| 2 | DL | | | | | | | | |
| 3 | DH1 | | | | DH0 | | | | |

Tabulka 3.9: Význam jednotlivých částí zprávy ESM

| Pole | Význam |
|------|--|
| O | Přetečení dat (<i>Overflow</i>). Jednička značí, že přetekla vyrovnávací paměť a tato zpráva přepsala některou starší. |
| C | Příznak shody CRC. Pokud je jednička, CRC polynom souhlasí. |
| D | Příznak poškození (<i>Damage</i>). Je nastaven, pokud nastala nějaká chyba, například ztráta synchronizace. |
| F | Formát zprávy (<i>Format</i>).. 0 pro ESM16, 1 pro ESM12 |
| IDL | ID zprávy, dolní polovina. |
| IDH | ID zprávy, horní polovina. Pro ESM16 je vždy 0. |
| DL | Data zprávy, bity [0..7] |
| DH0 | Data zprávy, bity [8..11] |
| DH1 | Data zprávy, bity [12..15]. Pro ESM12 vždy 0. |

3.2.4. Formát pro odeslání zprávy SSM

Zpráva SSM je přenesena jako jeden bajt hlavičky, který mimo klasických bitů pro ošetření chyb obsahuje i identifikátor zprávy. Za hlavičkou už je jen jeden přijatý datový bajt. Kontrola CRC polynomu je provedena interně a předána příznakem v hlavičce.

Blok dat pro zprávu ESM je popsán v tabulkách 3.10 a 3.11.

Tabulka 3.10: Organizace dat v bloku zprávy SSM

| Bajt | Bit | | | | | | | | Poznámka |
|------|------|---|---|---|----|---|---|---|----------|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 0 | O | C | D | 1 | ID | | | | |
| 1 | DATA | | | | | | | | |

Tabulka 3.11: Význam jednotlivých částí zprávy SSM

| Pole | Význam |
|------|--|
| O | Přetečení dat (<i>Overflow</i>). Jednička značí, že přetekla vyrovnávací paměť a tato zpráva přepsala některou starší. |
| C | Příznak shody CRC. Pokud je jednička, CRC polynom souhlasí. |
| D | Příznak poškození (<i>Damage</i>). Je nastaven, pokud nastala nějaká chyba, například ztráta synchronizace. |
| ID | Identifikátor zprávy |
| DATA | Přenesená data zprávy |

3.2.5. Formát zprávy pro konec dat

Jedním ze základních požadavků na implementaci jakéhokoliv protokolu je zabránění ve čtení dat mimo povolený rozsah. Mohou tak být přenesena data z paměti, ke kterým nemá mít protistrana přístup, nebo v našem případě můžou být čtena znovu již odvyšlána data, ale jinak posunutá a špatně interpretovaná, neboť se jedná o různé délky dat.

Jako prevence tohoto stavu je na konci vysílán bajt, který má nulovou identifikaci zprávy. Horní tři bity jsou také nulové, ale v budoucnu mohou být využity pro kontrolu stavu a je doporučeno je ve SPI masteru maskovat.

Tabulka 3.12: Indikátor konce dat

| Bajt | Bit | | | | | | | | Poznámka |
|------|-----|---|---|---|---|---|---|---|----------|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

3.3. Možné chyby při komunikaci

Při komunikaci se SPI masterem může nastat k některým chybám, které je nutné ošetřit na úrovni komunikace. Možné chyby a jejich ošetření jsou v tabulce 3.13

Tabulka 3.13: Přehled chyb komunikace na sběrnici SPI

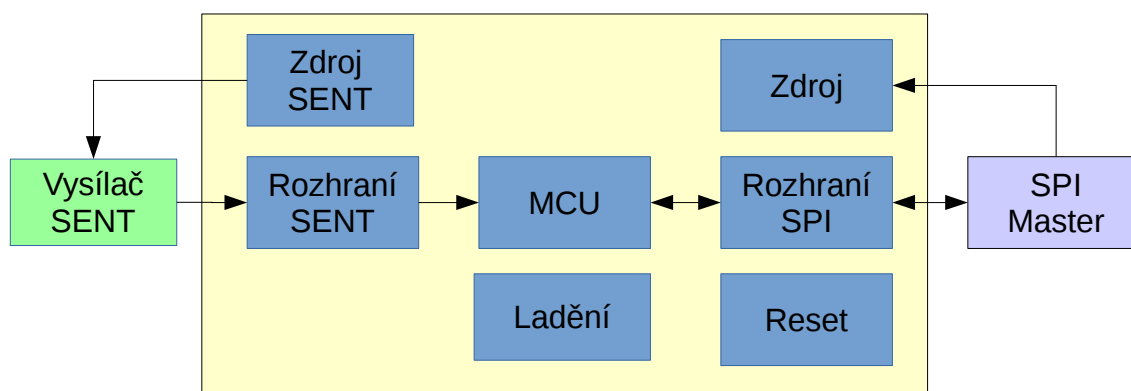
| Číslo | Chyba | Řešení |
|-------|--|---|
| 1 | Výpadek napájení | Reportován jako příznak ve stavovém slově. Přerušuje se komunikace, pokud není možné pokračovat. Přichází-li data, udržují se nejnovější. |
| 2 | Přerušování komunikace uprostřed bajtu | Restart rozhraní. Je-li to uprostřed zprávy, je tato zahozena. |
| 3 | Přečtení nekompletních dat | Bloky, které nebyly přečteny, budou předány při další komunikaci. |
| 4 | Přečtení nekompletního bloku | Blok je zahozen. Nepřečtené následující bloky budou přečteny v dalším paketu. |
| 5 | Přečtení více dat, než bylo ve frontě | Po posledním bloku jsou čteny nuly. |
| 6 | Přetečení vyrovnávací paměti | Každý blok, který přepíše jiný nevyčtený, je označen příznakem „Overflow“. |

4. Hardware převodníku

Základem hardware je mikrokontrolér, vykonávající všechny klíčové funkce převodníku. Ten je doplněn několika dalšími obvody pro zajištění jeho správné funkce. Byl zvolen typ **MSP430G2553IPW20** [4] s pouzdrém TSSOP20. Použitý mikrokontrolér má von Neumannovu architekturu s šířkou datové sběrnice 16 bitů a taktem maximálně 16 MHz. Velikost paměti programu FLASH je 16 kiB a je vybaven celkem 512 B RAM.

Základní bloky jsou na obrázku 4.1. K provozu mikrokontroléru (MCU) a jeho ladění jsou nezbytné obvody pro reset, časování a ladění.

Protože oscilátory jsou u zvoleného mikrokontroléru integrovány přímo na čipu, ale jsou omezeny jeho možností hlídání napájení vzhledem ke zvolené frekvenci, je zde nutné externí hlídání napájení a generování signálu $\overline{\text{RESET}}$ při podpětí. Obvody pro ladění obsahují rozhraní pro programování a ladění mikrokontroléru, společně s několika indikačními LED.



Obrázek 4.1: Blokové schéma hardware

Z pohledu mechaniky nebyla dána žádná omezení, zařízení je tedy koncipováno jako jednodeskový modul s možností integrace do desky, nebo samostatné montáže a připojení pomocí vodičů.

4.1. Rozhraní SENT

Zapojení tohoto rozhraní vychází z doporučení normy a parametrů použitého mikrokontroléru. Toto rozhraní je realizováno pomocí pouhých osmi součástek a konektoru.

Sběrnice SENT byla vyvedena na konektoru se třemi kontakty. Protože zde není normou dané provedení a zapojení konektoru, byl zvolen konektor se třemi vývody a roztečí 2,54 mm. Kontakty jsou zapojeny takto:

1. Napájení +5 V (signál **USENT**)
2. Zem
3. Data

4.1.1. Obvod přijímače

Zde je popsán obvod k přizpůsobení signálu sběrnice SENT na úroveň, které je možné přímo zpracovat mikrokontrolérem. Norma mimo přizpůsobení úrovní požaduje i ochranu proti přepětí a proti zákmitům. Podmínky jsou shrnuty v tabulce 4.1. Výsledné schéma je na obrázku 4.2. Všechny vzorce se odkazují na součástky na tomto schématu.

Tabulka 4.1: Požadavky na vstup ze sběrnice SENT

| Číslo | Popis | Matematický popis |
|-------|--|---|
| 1 | Při zkratu vstupu na napájení nebude překročeno napětí +3,0V na vstupu MCU | $U_{SENT} \cdot \frac{R_V}{R_V + R_F} \leq 3V$ |
| 2 | Nepřipojený vstup zajistí vstup MCU na úrovni log. 1 | $2,25V \leq U_{SENT} \left(\frac{R_V}{R_{PU} + R_F + R_V} \right) \leq 3V$ |
| 3 | Vstup MCU se překlopí do log. 1 při $U_I < 3,8V$ | $2,25V \leq 3,8V \left(\frac{R_V}{R_F + R_V} \right) \leq 3V ; R_{PU} \gg R_Z$ |
| 4 | Vstup MCU se překlopí do log. 0 při $U_I > 0,3V$ | $0V \leq 0,5V \left(\frac{R_V}{R_F + R_V} \right) \leq 0,75V ; R_{PU} \gg R_Z$ |
| 5 | Velikost RPU v rozsahu 10k až 55k | $10k \Omega \leq R_{PU} \leq 55k \Omega$ |
| 6 | Velikost hystereze je min. 0,3V | $U_{IH} - U_{IL} > 0,3V$ |
| 7 | Na vstupu je zařazená kaskáda dvou RC filtrů | |
| 8 | Časová konstanta filtru 1 je 0,74-1,73us | $0,74 \mu s \leq \tau_1 \leq 1,73 \mu s$ |
| 9 | Velikost R21 je 448-672 Ohmů | $448 \Omega \leq R_{F1} \leq 672 \Omega$ |
| 10 | Velikost C11 je 1,54-2,86nF | $1,54 nF \leq C_{F1} \leq 2,86 nF$ |
| 11 | Časová konstanta filtru 2 je 0,6-1,4us | $0,6 \mu s \leq \tau_2 \leq 1,4 \mu s$ |

První z podmínek, kterou je nutné splnit, je maximální napětí +3,0 V na signálu **SENTRX**, a to jak v případě odpojení vysílače, tak i v případě jeho připojení a výstupního napětí 5 V. Pro nepřipojený přijímač platí vztah

$$U_O = U_I \cdot \frac{R_{23}}{R_{21} + R_{22} + R_{23}} \Rightarrow \frac{U_O}{U_I} = \frac{R_{23}}{R_{21} + R_{22} + R_{23}} \quad (4.1)$$

a pro připojený vysílač platí

$$U_{OF} = U_N \cdot \frac{R_{23}}{R_{20} + R_{21} + R_{32} + R_{23}} \Rightarrow \frac{U_{OF}}{U_N} = \frac{R_3}{R_{20} + R_{21} + R_{22} + R_{23}} \quad (4.2)$$

Protože jde o lineární obvod, může být zvolena hodnota jednoho z rezistorů zvolena libovolně a po dopočítání ostatních hodnot můžou být vypočteny ostatní rezistory z vhodné řady. Byla tedy zvolena hodnota $R_{23} = 60 \Omega$ a k němu byla dopočítána hodnota součtu R_{21} a R_{22} na základě podmínky 1 z tabulky 4.1.

$$\frac{U_O}{U_I} = \frac{R_{23}}{R_{21} + R_{22} + R_{23}} \Rightarrow R_{21} + R_{22} = R_{23} \left(\frac{U_I}{U_O} - 1 \right) = 60 \left(\frac{5}{3} - 1 \right) = 40 \Omega \quad (4.3)$$

S těmito rezistory a s podmínkou 2 z tabulky 4.1 byla vypočtena hodnota rezistoru R_{20} .

$$\frac{U_O}{U_N} = \frac{R_{23}}{R_{20} + R_{21} + R_{22} + R_{23}} \quad (4.4)$$

$$R_{20} = \frac{R_{23} U_N}{U_O} - (R_{21} + R_{22} + R_{23}) = \frac{60 \cdot 5}{2,25} - (60 + 40) = 33,33 \Omega$$

Na základě podmínky 5 z tabulky 4.1 byly určeny možné kombinace těchto rezistorů v řadě E24. Tyto možnosti jsou shrnuty v tabulce 4.2, kde první řádek jsou vypočtené hodnoty a ve sloupci „k“ je hodnota koeficientu, kterým jsou vypočtené hodnoty vynásobeny pro dosažení daných hodnot rezistorů. Protože jedna z hodnot označuje kombinaci dvou rezistorů, je zvolena nejvyšší použitelná kombinace, aby se zlepšila tolerance obvodu.

Tabulka 4.2: Vhodné hodnoty rezistorů

| k | R_{20} [Ω] | $R_{21} + R_{22}$ [Ω] | R_{23} [Ω] |
|-----|-----------------------|--------------------------------|-----------------------|
| 1 | 33,33 | 40 | 60 |
| 300 | 10 000 | 12 000 | 18 000 |
| 450 | 15 000 | 18 000 | 27 000 |
| 600 | 20 000 | 24 000 | 36 000 |

Návrh prvního stupně filtru vychází z podmínek 8, 9 a 10 v tabulce 4.1. Zvolena byla hodnota uprostřed doporučeného rozsahu, tedy $R_{21} = 560 \Omega$ a $C_{11} = 2,2 \text{ nF}$.

Z požadované hodnoty $R_{21} + R_{22}$ je to cca 2,3% ve prospěch snížení výstupního napětí. Hodnotu R_{22} je tedy možné ponechat 24 k Ω . Pro druhý filtr tak platí vztah

$$\tau_2 = \frac{R_{22} R_{23}}{R_{22} + R_{23}} C_{12} \Rightarrow C_{12} = \tau_2 \frac{R_{22} + R_{23}}{R_{22} R_{23}} \quad (4.5)$$

Minimální hodnota C_{12} je potom daná vztahem

$$C_{12min} = \tau_{2min} \frac{R_{22} + R_{23}}{R_{22} R_{23}} = 0,6 \cdot 10^{-6} \frac{24000 + 36000}{24000 \cdot 36000} = 41,67 \text{ pF} \quad (4.6)$$

a jeho maximální hodnota

$$C_{12max} = \tau_{2max} \frac{R_{22} + R_{23}}{R_{22} R_{23}} = 1,4 \cdot 10^{-6} \frac{24000 + 36000}{24000 \cdot 36000} = 97,22 \text{ pF} \quad (4.7)$$

Z tohoto rozsahu byla zvolena hodnota $C_{12} = 68 \text{ pF}$, která je zhruba uprostřed.

Zbývá ověření podmínek 3, 4 a 5 z tabulky 4.1. Pro napěťový zisk filtru v ustáleném stavu platí vztah

$$k_U = \frac{R_{23}}{R_{21} + R_{22} + R_{23}} = \frac{36000}{36000 + 24000 + 560} = 0,595 \quad (4.8)$$

Pro korektní vstupní napětí v logické jedničce pak musí platit nerovnost

$$U_{SENTH} = \frac{U_{IH}}{k_U} = \frac{2,25}{0,595} = 3,785 V < 3,8 V \quad (4.9)$$

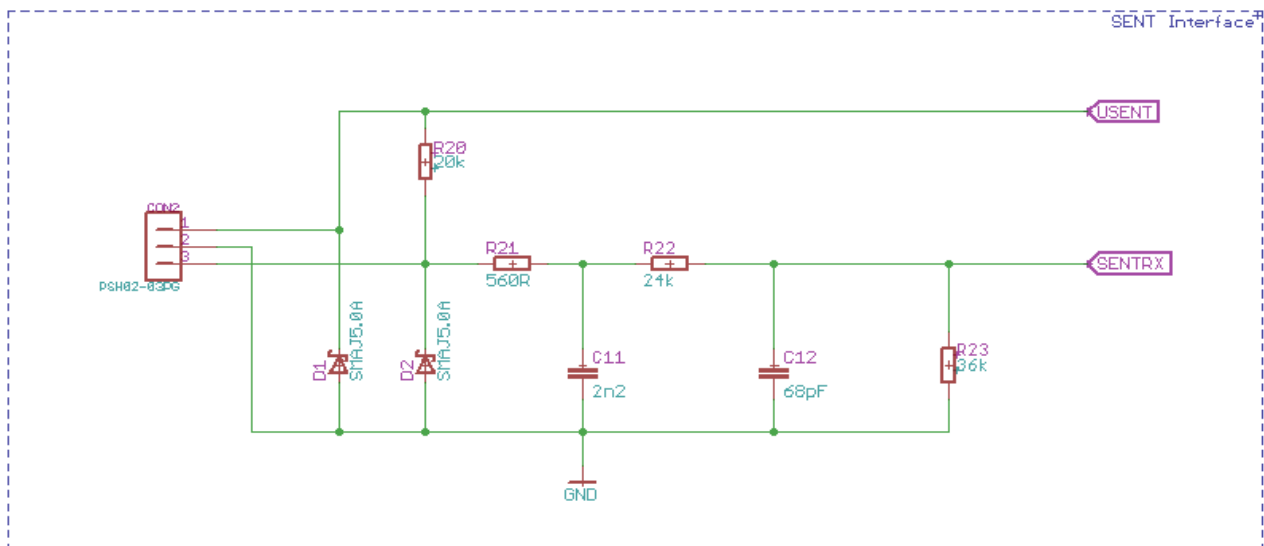
a pro logickou nulu

$$U_{SENTL} = \frac{U_{IL}}{k_U} = \frac{0,75}{0,595} = 0,4458 V < 0,5 V \quad (4.10)$$

Dále je definována nerovnost pro hysterezi vstupu. Zde platí nerovnost

$$U_{HYST} = \frac{U_{IHYST}}{k_U} = \frac{0,3}{0,595} = 0,505 V > 0,3 V \quad (4.11)$$

Všechny tři nerovnosti jsou tedy splněny. Posledním krokem je doplnění přepětových ochran v podobě TVS na signálový i napájecí signál, a to co nejdříve ke konektoru.



Obrázek 4.2: Schéma přizpůsobení rozhraní SENT

Pro příjem dat z rozhraní SENT je nutné měřit délku impulsů. To je jednoduše realizovatelné s pomocí čítače/časovače v záchytném (*Capture*) režimu. Pro danou aplikaci je požadováno minimálně 16 bitů, možnost děličky referenčního signálu, rozlišení hran a možnost vyvolat přerušení. Proto je signál z tohoto obvodu přiveden do čítače/časovače TA mikrokontroléru.

4.2. Rozhraní SPI

Rozhraní SPI slouží ke komunikaci s nadřazeným systémem. Oproti standardním signálům je toto rozhraní rozšířeno o signály pro přerušení a pro restart převodníku. Rozhraní sdílí jeden konektor společně s napájecími signály.

4.2.1. Zapojení konektoru

Toto rozhraní potřebuje k činnosti celkem devět signálů. Protože je běžně k dispozici konektor s deseti kontakty pro plochý kabel, byl zvolený tento typ konektoru.

Při návrhu rozložení signálů bylo zohledněno minimální rušení, tj. střídání rychlých a „pomalých“ signálů, jako je napájení a zem. Za tímto účelem byly signály standardně rozděleny do čtyř skupin:

1. Rychlé signály, se základním kmitočtem až v oblasti MHz. Sem patří **SCK**, **MISO**, **MOSI** a $\overline{\text{CS}}$.
2. Pomalé signály, řádově od kHz níže. Ty nejsou takovým zdrojem rušení a mohou vést samostatně. Zde se jedná o signály $\overline{\text{INT}}$ a $\overline{\text{RESET}}$.
3. Diferenciální signály. Tyto signály musí být vedeny v párech vedle sebe. V řešeném rozhraní se takové signály nevyskytují.
4. Napájecí signály. Tyto signály jsou stabilní a mohou sloužit jako stínění. I v případě země, i v případě napájení, na kterém bývají blokovací kondenzátory proti zemi, které pro složky rušení představují zkrat.

Protože jeden vývod vychází volný, je zdvojnásobena zem. Mezi zemními vodiči je natažen signál s nejvyšší úrovní rušení, **SCK** se stabilní frekvencí.

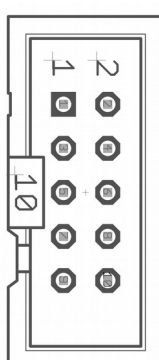
Datové signály **MISO** a **MOSI** mají při stejné úrovni signálů poloviční maximální frekvenci a spektrum je rozloženo více do šířky, proto vyhoví jejich pozice mezi zemním a napájecím signálem.

Pro signál **SCK** platí, že pokud by na něm teoreticky byla střída 1:1, je frekvence $f_{\text{SCK}}/80$. V reálu se časování mění a opět se jedná o rozprostřené spektrum. Protože ale pro tento signál nezbyvá druhý vodič ze skupiny napájení, je vedle něj vedený $\overline{\text{RESET}}$, který má stabilní úroveň a obvykle i keramický kondenzátor proti zemi.

Posledním zbývajícím signálem je $\overline{\text{INT}}$. Zde nezbyvá, než jej dát na vodič vedle napájení.

Rozložení signálů je zrekapitulováno v tabulce 4.3. Logické signály jsou vztaženy k U_{SPI} .

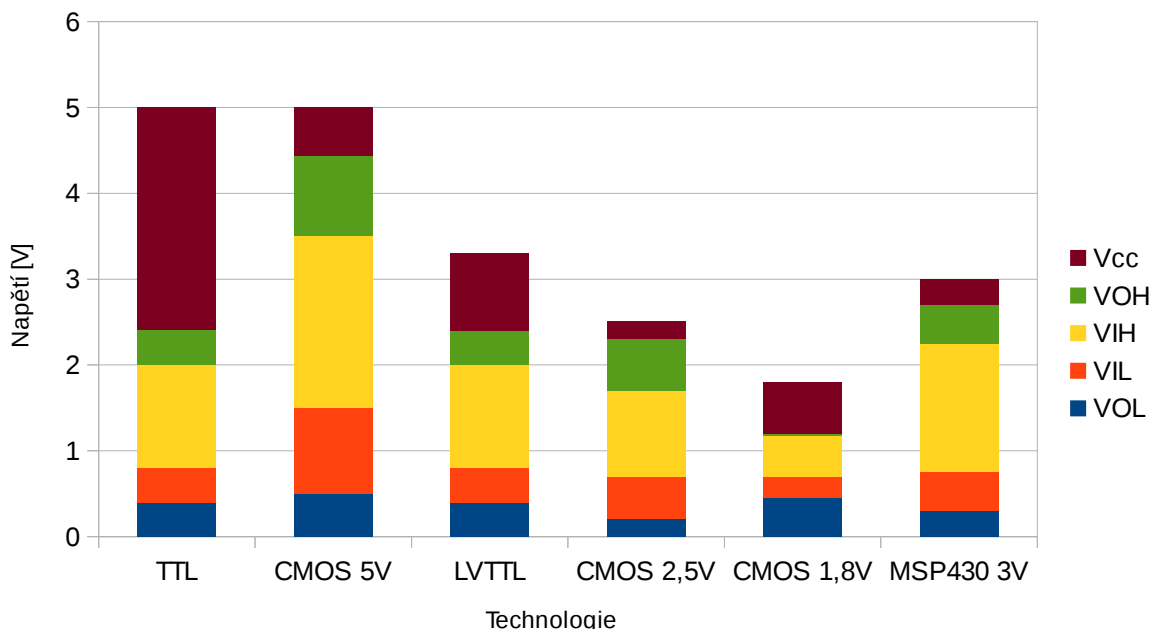
Tabulka 4.3: Zapojení konektoru sběrnice SPI

| Konektor | SPI + Napájení | Reference | CON1 | Typ | ML10 |
|--|----------------|---|------|-----------|---|
| $\overline{\text{INT}}$ Výstup, otevřený kolektor | 1 |  | | 2 | +5 V Napájení vysílače |
| MISO Třístavový výstup | 3 | | | 4 | Gnd Společná zem |
| SCK Hodinový vstup | 5 | | | 6 | Gnd Společná zem |
| MOSI Vstup | 7 | | | 8 | USPI Napájení – úroveň sběrnice |
| $\overline{\text{CS}}$ Vstup | 9 | | | 10 | $\overline{\text{RESET}}$ Vstup |

4.2.2. Zapojení rozhraní SPI

Schéma zapojení bloku rozhraní SPI je na obrázku 4.3.

V tomto zapojení je důležitá i kompatibilita vstupních a výstupních napětí u IC2. Jak již bylo řečeno výše, je třeba mít pro výstup přerušení napájení budiče minimálně na úrovni USPI. Ze stejného důvodu je třeba mít napájení 5 V i pro IC2D, kde je také zapojena závěrně polarizovaná dioda na napájení.



Obrázek 4.4: Rozsahy napětí vstupů a výstupů pro základní technologie logických obvodů

Nutnou podmínkou je i kompatibilita vstupů s úrovní +3 V, kde klasická 5 V CMOS logika není kompatibilní. Jak je vidět na obrázku 4.6, nejlépe v této situaci vyhovují klasické úrovně TTL. Tedy v technologii CMOS s nízkým příkonem je nejlepší volba obvod **74HCT125** [6], **74AHCT125** nebo **74ACT125**. Výstup pro blokování IC1 je pak ošetřen odporovým děličem z R3 a R4. Protože poměr napětí na jeho rezistorech je stejný jakou u R22 a R23, byly v rámci redukce počtu položek na seznamu součástek použity stejné hodnoty.

Parametry jednotlivých řad logiky jsou převzaty ze strany 4 příručky [7].

4.3. Uživatelské rozhraní

Pro indikaci stavu a pro ladění firmware je vhodné mít k dispozici několik LED. Tyto mohou, ale nemusí být osazeny a příprava desky pro ně není velkou komplikací. Převodník byl navržen s možností použít indikaci dle tabulky 4.4.

LED2 je sdílena s indikací přetížení zdroje. Toto je použito pro úsporu vývodů. Pokud dojde k přetížení zdroje, je signál v log. 0 a LED svítí. Pokud ne, jde jen o spoj mezi mikrokontrolérem a LED. Krátké zhasnutí LED, kdy mikrokontrolér přepne tento vývod na vstup a zkontroluje jeho stav, oko nepostřehne a pokud se potkají dva výstupy v log. 0, jde jen o dva paralelně zapojené spínače.

Maximální povolený výstupní proud na vývodu mikrokontroléru je 6 mA. Tato hodnota je dostatečná pro připojení indikačních LED s omezovacím rezistorem.

Tabulka 4.4: Přehled možných diod LED

| LED | Význam | Indikovaný stav | Zapojení |
|------|---------|------------------------------------|--|
| LED1 | SENT | Příjem dat ze SENT | U_{DD} proti výstupu MCU |
| LED2 | SPI | Komunikace po SPI | U_{DD} proti výstupu MCU |
| LED3 | Provoz | Procesor naběhl a vykonává program | U_{DD} proti signálu \overline{ACT} |
| LED4 | Porucha | Svítil, pokud došlo k poruše | U_{DD} proti indikaci přetížení zdroje |

4.4. Zapojení bloku mikrokontroléru

Jedním z problémů při výběru mikrokontroléru je, že vývody mohou být sdíleny mezi několika periférii a potřebnou periférii nelze použít. Dobrou metodou, jak zajistit, že k této kolizi nedojde, je použití tabulky vývodů, ve které jsou ještě před finálním výběrem mikrokontroléru vyznačeny jednotlivé vývody a jejich zapojení do obvodu.

Zde hrozilo riziko konfliktu mezi rozhraním SPI a komparátorem pro kontrolu přítomnosti 5 V. Proto je i zde provedena kontrola s pomocí tabulky 4.5.

Tabulka 4.5: Funkce V/V bran mikrokontroléru

| Vývod | Funkce procesoru | Periferie | Signál | Popis |
|-------|--|-----------|------------------------|--|
| 1 | DVCC | | | Napájení procesoru +3 V |
| 2 | P1.0, TA0CLK, ACLK, A0, CA0 | AC | DETSV | Detekce přítomnosti +5 V s pomocí komparátoru |
| 3 | P1.1, TA0.0, UCA0RXD, UCA0SOMI, A1, CA1 | AC | DETSPI | Detekce přítomnosti napájecího napětí sběrnice SPI |
| 4 | P1.2, TA0.1, UCA0TXD, UCA0SIMO, A2, CA2 | GPIO | SENTON | Log. 1 zapne napájení pro vysílač SENT. |
| 5 | P1.3, ADC10CLK, CAOUT, VREF-, A3, CA3 | GPIO | SENTOK | Log. 0 indikuje nadproud v zařízení napájeném ze sběrnice. V normálním stavu log. 1. |
| 6 | P1.4, SMCLK, UCB0STE, UCA0CLK, VREF+, A4, CA4, TCK | USCI | $\overline{USCI_STE}$ | Signál CS sběrnice SPI |
| 7 | P1.5, TA0.0, UCB0CLK, UCA0STE, A5, CA5, TMS | USCI | UCSI_CLK | Signál SCK sběrnice SPI |
| 8 | P2.0, TA1.0 | GPIO | \overline{SENT} | Indikační LED pro sběrnici SENT. Aktivní v nule. |
| 9 | P2.1, TA1.1 | GPIO | \overline{SPI} | Indikační LED pro sběrnici SPI. Aktivní v nule |

| Vývod | Funkce procesoru | Periferie | Signál | Popis |
|-------|---|-----------|---------------------------|--|
| 10 | P2.2, TA1.1 | GPIO | $\overline{\text{ACT}}$ | Výstup, který je v nule, pokud je procesor aktivní. Napájí indikační LED a povoluje komunikaci po SPI. |
| 11 | P2.3, TA1.0 | TA1 | DATA | Vstupy ze sběrnice SENT |
| 12 | P2.4, TA1.2 | TA1 | | |
| 13 | P2.5, TA1.2 | GPIO | $\overline{\text{INTO}}$ | Výstup přerušení |
| 14 | P1.6, TA0.1, CA6, TDI, TCLK, UCB0SOMI | USCI | USCI_SOMI | SPI, signál MISO |
| 15 | P1.7, CAOUT, UCB1SIMO, UCB0SDA, A7, CA7, TDO, TDI | USCI | USCI_SIMO | SPI, signál MOSI |
| 16 | $\overline{\text{RST}}$, NMI, SBWTDIO | SBW | $\overline{\text{RESET}}$ | Ladící a programovací rozhraní |
| 17 | TEST, SBWTCK | SBW | SBWTCK | |
| 18 | XOUT, P2.7 | OSC | | Krystalový oscilátor (rezerva, nezapojeno) |
| 19 | XIN, P2.6 / TA0.1 | OSC | | |
| 20 | DVSS | | | Napájení procesoru – 0 V |

4.4.1. Časování mikrokontroléru

Zapojení obvodů pro časování převodníku je v podstatě interní věcí mikrokontroléru. Co se časování vně pouzdra týká, je jen možnost zapojení externího krystalového oscilátoru. U daného typu to ovšem postrádá smysl, protože u zvoleného typu mikrokontroléru lze připojit externě pouze hodinový krystal. Vývody, sloužící pro krystal, byly v zapojení ponechány volné a nastavené jako vstupy, jejichž hodnota je ignorována.

Mikrokontrolér je tedy limitovaný použitím interního oscilátoru a maximální frekvencí 14 MHz při napájecím napětí +3,0 V. Byla zvolena hodinová frekvence 12 MHz, protože v paměti FLASH je již z výroby uložena kalibrační hodnota RC oscilátoru pro tuto frekvenci. Zásadní je zde časování periferie TA touto frekvencí. Od něj se pak odvíjí návrh přijímače SENT.

4.4.2. Zapojení ladícího rozhraní

Ladící rozhraní těchto mikrokontrolérů je typu SWD. Vyžaduje pouze dva vývody mikrokontroléru, jeden z nich je přitom sdílený s se signálem $\overline{\text{RESET}}$ a druhý se signálem TEST, který by byl jinak nepoužitelný. Mimo to si vystačí pouze s napájením a zemí, tj. dostačují čtyři kontakty. Tuto roli zastane lánací lišta s roztečí 2,54mm, její zapojení je v tabulce 4.6. V zapojení je výhodou, pokud se podaří signál $\overline{\text{RESET}}$ zapojit vedle zemního signálu, aby bylo možné mikrokontrolér resetovat zkratovací propojkou.

Tabulka 4.6: Zapojení konektoru CON3 ladícího rozhraní

| Vývod | Signál | Význam |
|-------|--------------|---|
| 1 | Ucc | Napájecí napětí mikrokontroléru +3,0 V |
| 2 | TEST/SBWTCK | Hodinový signál pro ladící rozhraní mikrokontroléru |
| 3 | RESET/SBWDIO | Kombinace signálu RESET a dat debuggeru rozhraní SPI. |
| 4 | Gnd | Společná zem |

4.4.3. RESET, detekce podpětí a kontrola běhu aplikace

Mikrokontrolér je hardwarově resetován v několika situacích, jak je uvedeno v tabulce 4.7.

Interní zapojení hlídače podpětí a watchdogu je definováno vnitřním zapojením mikrokontroléru. Podstatným omezením zde je, že $\overline{\text{RESET}}$ je pouze vstupní signál, proto interní reset není nijak indikován navenek a také to, že je sdílený s rozhraním SWD.

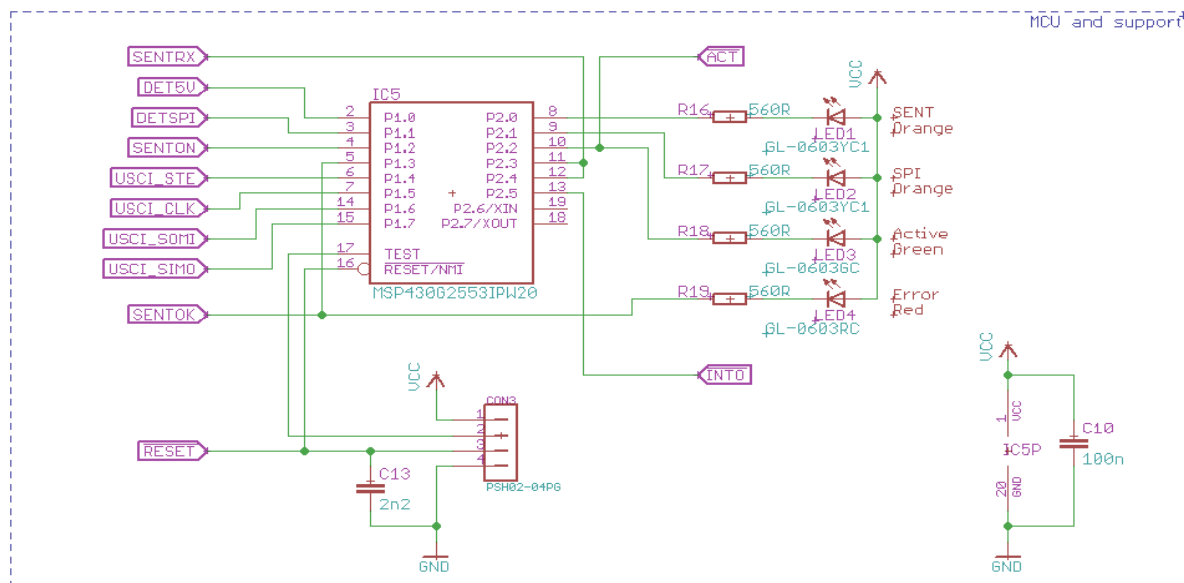
První z těchto problémů jde v případě potřeby obejít výstupem, který je softwarově nastavený do log. 0 při startu aplikace. K jeho inicializaci na vstup dojde právě během restartu mikrokontroléru a v jedničce jej drží R18 a LED3. Tento signál je označený jako $\overline{\text{ACT}}$.

Druhý problém souvisí spíše s omezeními ohledně zapojení – omezená zatěžovací kapacita, předepsaná hodnota pull up rezistoru a podobně. Samozřejmě také není možné takový signál vyvést přímo na konektor rozhraní SPI, protože tam nejde zaručit ani tyto parametry, ani vyloučit narušení ladícího rozhraní při sdílení signálu $\overline{\text{RESET}}$ například se SPI masterem během ladění. Vnější vstup byl popsán v kapitole týkající se zapojení rozhraní SPI a z pohledu interního signálu se jedná jen o další otevřený kolektor, který se většinu času neuplatní.

Tabulka 4.7: Možné příčiny restartu mikrokontroléru

| Důvod restartu | Podmínky | Zdroj resetu |
|---|--|----------------|
| Náběh napájecího napětí. Je nutné správně inicializovat MCU. | Udd < 1,8V | Brownout |
| Podpětí. Při nízkém napájecím napětí se snižuje maximální pracovní frekvence mikrokontroléru a hrozí chyby a porušení dat v paměti. Riziko narušení komunikace po SPI | Udd < 2,8V | Napájecí zdroj |
| Zacyklení programu. V případě chyby se musí převodník zotavit bez přerušení napájecího napětí a bez fyzické manipulace s ním. | Uváznutí programu ve smyčce | Watchdog |
| Reset na žádost nadřazeného systému. V případě, že nadřazený systém detekuje chybu, má možnost restartovat mikrokontrolér pomocí jednoho ze signálů, přidaných ke sběrnici SPI. | Aktivace signálu $\overline{\text{RESET}}$ na konektoru. | SPI Master |
| Nástroje během ladění aplikace. | Akce vývojáře | SWD |

4.4.4. Schéma zapojení mikrokontroléru



Obrázek 4.5: Schéma zapojení mikrokontroléru, ladícího rozhraní a napájecího zdroje

Kompletní zapojení bloku mikrokontroléru včetně uživatelského rozhraní je na obrázku 4.5.. Konektor **CON3** souží k ladění a programování mikrokontroléru.

Signál **RESET** je blokován proti zážitům kondenzátorem 2,2 nF. Napájení mikrokontroléru, který jako každý číslicový obvod CMOS zatěžuje zdroj proudovými impulsy, má napájení blokováno kondenzátorem 100 nF pro potlačení těchto špiček.

4.5. Napájecí zdroj

V převodníku jsou použita celkem čtyři napájecí napětí, jak je uvedeno v tabulce 4.8. Každá z úrovní je rozebrána samostatně a v uvedené tabulce je odkaz na příslušnou kapitolu, věnovanou této napájecí úrovni.

Tabulka 4.8: Rekapitulace napájecích napětí a proudů v převodníku

| Označení | Napětí [V] | | | Odběr [mA] | | | Popis a poznámka |
|-------------------|------------|------|------|------------|------|-----|--|
| | Min | Nom | Max | Min | Nom | Max | |
| +5 V | | 5,00 | | | 25,5 | 140 | 4.5.3 Napájecí hladina +5 V |
| U _{SPI} | 3,0 | | 5,5 | | | 40 | 4.5.2 Napájení sběrnice SPI |
| U _{SENT} | 4,85 | 5,00 | 5,15 | 0 | 20 | 80 | 4.5.1 Napájení sběrnice SENT |
| U _{DD} | 2,9 | 3,0 | 3,1 | | 5,5 | 36 | 4.5.1 Napájecí hladina mikrokontroléru |

4.5.1. Napájení sběrnice SENT

Velikost napájecího napětí je odvozena z požadavků [3], rekapitulovaných v tabulce 4.3. Převodník musí poskytnout výstupní proud minimálně 20 mA.

Protože převodník může pracovat i se SPI sběrnici, která nedosahuje úrovně napájení potřebné pro napájení sběrnice SENT, byl v rozhraní převodníku definován napájecí signál +5 V. Technicky je to jednodušší řešení, než nábojová pumpa nebo zvyšující měnič.

Zůstává ale problém s tím, že případný zkrat nebo přetížení převodníku může znamenat problém pro celé zařízení. Pro spínání napájení byl zvolen obvod **TPS22493**. Jedná se o spínač napájení, určený pro spínání napětí od tří do pěti voltů, s velmi malým vnitřním odporem, automatickým vypnutím při přetížení a výstupem indikujícím přetížení, to vše v pouzdře SC-70. Parametry jsou uvedeny v [8].

4.5.1. Napájecí hladina mikrokontroléru

Toto napájecí napětí je využíváno interně pro mikrokontrolér a obvody převodníku. Bylo zvoleno poněkud netradiční napětí +3 V, a to z těchto důvodů:

1. Převodník **TXS0104E** vyžaduje na straně mikrokontroléru maximálně takové napětí, jaké je na straně SPI Masteru. Použitím nižší hladiny napájení je tak dosaženo větší tolerance U_{SPI} .
2. Do převodníku jsou přivedena dvě napájecí napětí, +5 V a U_{SPI} . Napětí +3 V může být získáno z obou z nich, přepnutí vstupů bude provedeno automatickým přepínačem.

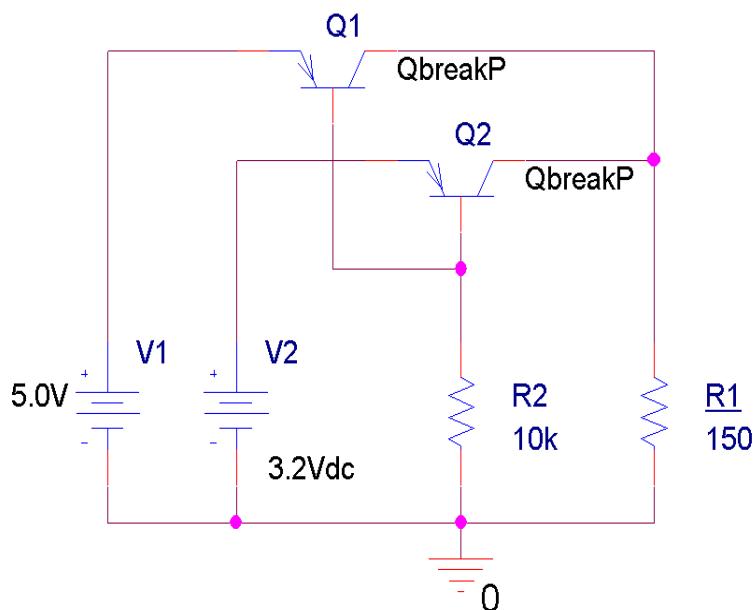
Z této hladiny jsou napájeny obvody v tabulce 4.9. Celkový odběr z této větve by neměl překročit hodnotu 45 mA, se kterou je počítáno při konstrukci zdroje. Pro získání tohoto napájecího napětí je vhodné použít stabilizátor napětí s malým úbytkem (LDO). Byl zvolen stabilizátor **TPS79730DCKR** [9], který má i výstup indikující podpětí. Tím odpadá nutnost použití dalšího obvodu pro hlídání velikosti napájecího napětí.

Tabulka 4.9: Zatížení napájení U_{DD}

| Napájený obvod | Odebíraný proud | | Poznámka |
|--------------------|-----------------|-------------|---|
| | Typ. | Max. | |
| MSP430G2553 | 3 mA | 5 mA | Odpovídá 3 V, 12 MHz, trvale aktivní stav |
| TXS0104E | | 2,5 μ A | Pro $U_{SPI} > 3,2$ V |
| Indikace | 5,5 mA | 15 mA | Svítlí maximálně tři LED po 5 mA |
| Programátor | 0 | 25 mA | Odhad |
| Celkem | 8,5 mA | 45 mA | |

Jak již bylo zmíněno, napájecí napětí mikrokontroléru napětí může být získáno z obou vstupů a je tedy nutné jejich automatické přepnutí. Původně mělo být přepínání zdrojů realizováno dvojicí Schottkyho diod. Při napájení z U_{SPI} na dolní hranici 3,2 V a při absenci +5 V ale není možné dosáhnout dostatečného napětí pro řízení stabilizátoru. Tedy nezbylo nic jiného, než použít tranzistorový přepínač dle obrázku 4.6. Toto zapojení využívá dvojici tranzistorů PNP v netradičním zapojení se společnou bází ve spínacím režimu.

Chování tohoto zapojení je vidět na obrázku 4.7, který podrobuje zmíněné schéma přepínače *DC Sweep* analýze v programu *Spice*, kde zdroj V2 měl pevné napájecí napětí +3,2 V (minimální napájecí napětí rozhraní SPI) a napětí zdroje V1 bylo rozmitáno v rozsahu 0 V až 5,2 V s krokem 10mV.



Obrázek 4.6: Zapojení přepínače napájení s dvojicí tranzistorů PNP

Graf má tři části, kde v první je napětí zdroje $V1$ menší, než napětí zdroje $V2$. V tomto okamžiku teče proud přechodem BE $Q1$ rezistorem $R2$. Tím je otevřen tranzistor $Q1$ a na výstupu je napětí $V2$, ponížené o U_{CE} $Q1$. U tranzistoru $Q2$ je v tomto okamžiku přechod BE závěrně polarizovaný, tj. $Q2$ je zavřený a zdroj $V1$ je odpojený od výstupu.

Ve druhé části se napětí obou zdrojů přiblíží a rezistorem $R2$ teče součet bázových proudů obou tranzistorů. Jeden tranzistor se přivírá a druhý otevírá, na výstupu je pak proud z obou zdrojů a napětí, odpovídající vyššímu z nich.

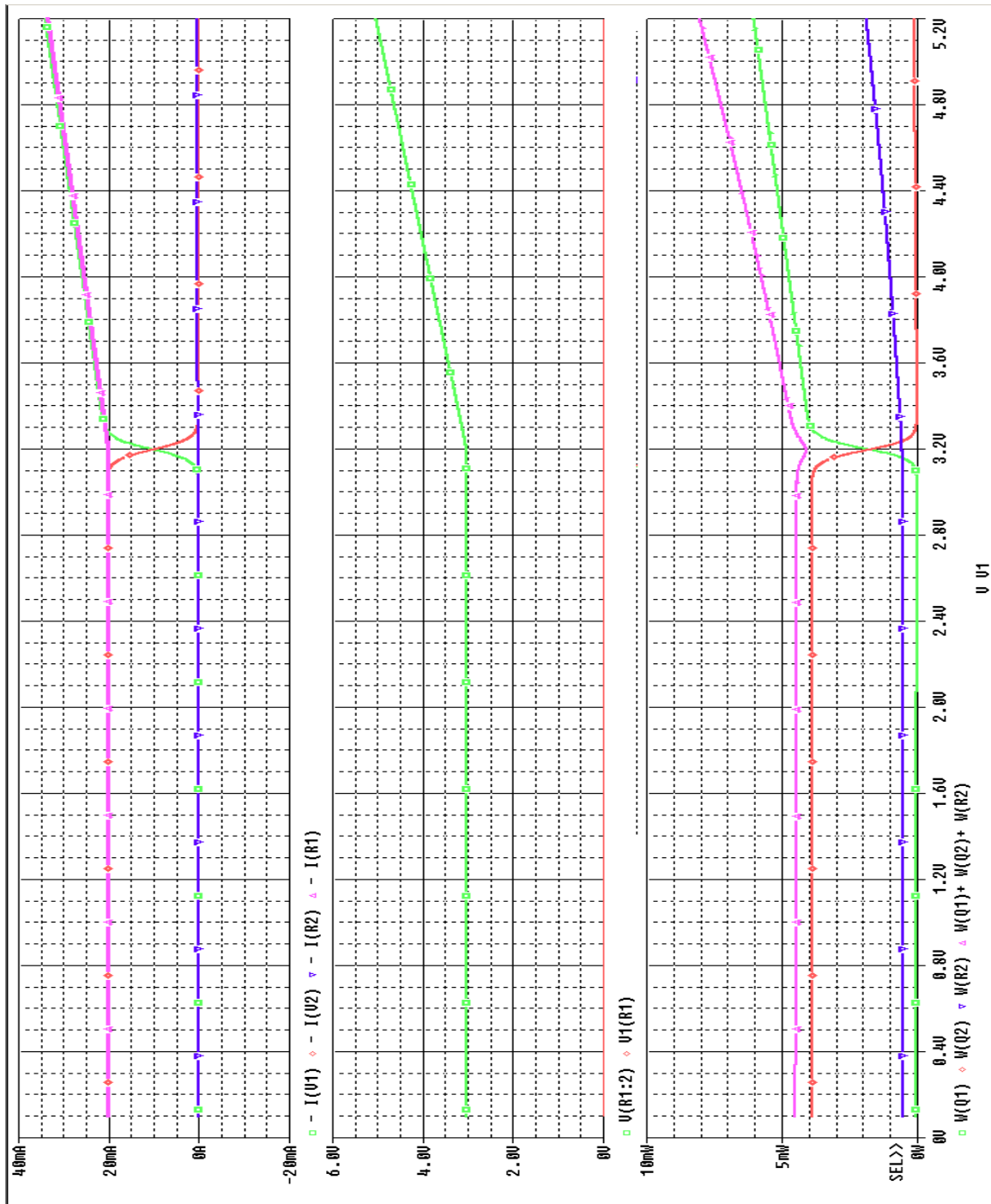
Konečně ve třetí části se situace obrátí a je zablokovaný $Q1$, zatímco všechen výstupní proud protéká tranzistorem $Q2$.

Výstupní napětí, jak je vidět ve střední části obrázku 4.7, se drží těsně nad úrovni 3 V. Při řešení s diodami Schottky je třeba počítat s poklesem až na 2,8 V. Výstupní napětí je možné ovlivnit zesílením tranzistorů a velikostí jejich U_{CESAT} . Použitý tranzistor je z důvodu dostupnosti **BC807-40** (dle [10] má parametry $h_{21} = 250 - 600$, U_{CESAT} cca 200 mV při proudu 20 mA a teplotě 25°C).

Zapojení se chová zajímavě i z pohledu výkonové ztráty, jak ilustruje třetí část obrázku 4.7. Výkonová ztráta tranzistoru závisí na kvadrátu protékajícího proudu, a tak pokud je napětí obou zdrojů shodné, teče každým tranzistorem poloviční proud a tranzistory tak mají čtvrtinovou výkonovou ztrátu proti situaci, kdy je jeden z nich zavřený. Je zde vidět i celková výkonová ztráta, která ukazuje, že by na $R2$ neměl být překročený výkon 2mW a tranzistory se pravděpodobně vejdou do 10mW.

4.5.2. Napájení sběrnice SPI

Toto napájecí napětí musí být přivedeno zvenčí. Definuje napěťové úrovně rozhraní SPI. Rozsah napájení je omezený



Obrázek 4.7: Výsledky simulace přepínače napájení s dvojicí tranzistorů PNP

1. Provozním napětím obvodu **TXS0104E**, které pro správnou funkci musí být minimálně na úrovni napětí na úrovni +3 V.
2. Funkcí napájení z U_{SP1} , kde je navíc požadavek na 85 mV rezervy pro stabilizátor, 100 mV pro přepínač napájení a 15 mV tolerance.
3. Shora je omezeno maximálním napětím připojených obvodů. Zde je omezením **TXS0104E**, limit je tedy 5,5 V

Výsledkem je rozsah U_{SPI} od 3,2 V do 5,5 V. Proudový odběr je předpokládán maximálně 50 mA, jedná se součet proudů pro napájení převodníku, vstupní proud stabilizátoru a I_B přepínače.

4.5.3. Napájecí hladina +5 V

Dle tabulky 4.3 je požadované napětí a proud na výstupu 4,85 V až 5,15 V., min. 20 mA. Tento rozsah je plně dostačující i pro zdroj U_{DD} .

Spínač TPS22495 má dle dokumentace odpor v sepnutém stavu při 5 V kolem 40 mΩ. Při maximálním proudu před vypnutím 80 mA to představuje úbytek napětí

$$U_S = I_{SENTLIM} \cdot R_{DS(ON)} = 0,08 \cdot 0,04 = 3,2 \text{ mV} \quad (4.12)$$

Vliv spínače je na úrovni přívodních vodičů a je možné jej zanedbat. Rozsah napájení tedy zůstává v mezích normy SENT s tím, že odebíraný proud může být bez programátoru a bez vypnutí při nadproudu nejvýše 100 mA, s programátorem 140 mA.

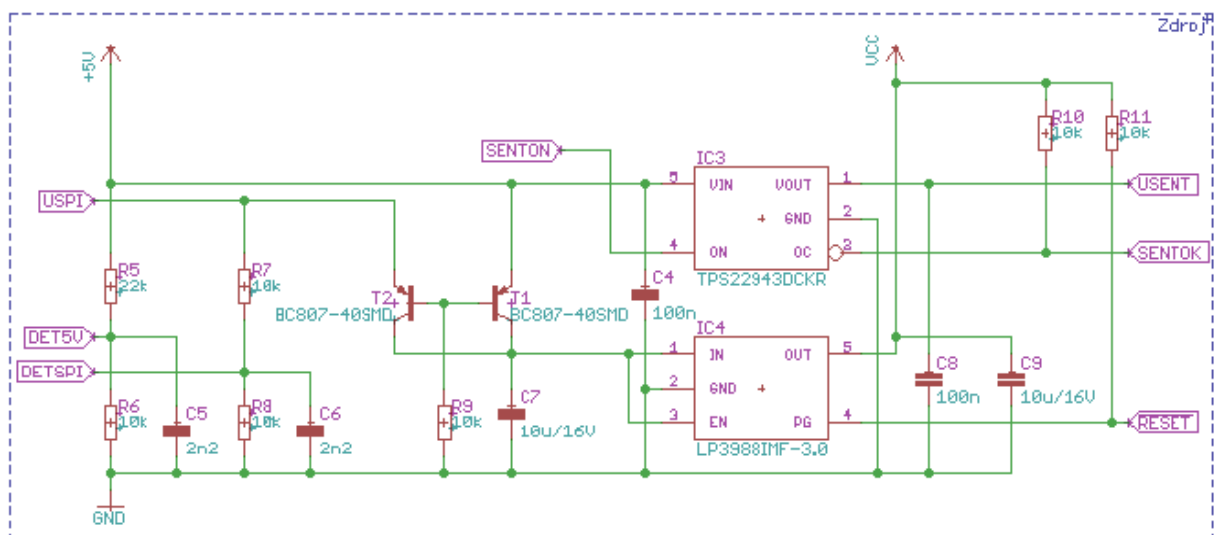
4.5.4. Diagnostika napájení

Pro diagnostiku napájení dostačuje zjištění, že jsou přítomna napětí U_{SPI} a +5 V. V obou případech je nejjednodušší možná metoda použití odporového děliče a analogového komparátoru. Proti použití vstupu s hysterezí vychází obvodově stejně, ale je zde rozdíl v přesnosti a v tom, že jsou definována prahová napětí pro detekci. Jako referenci je možné použít $\frac{1}{2}$ nebo $\frac{1}{4}$ z napájecího napětí MCU s možností změny reference a zdroje za běhu.

U zvoleného mikrokontroléru je na stejných vývodech i A/D převodník, takže z pohledu hardware zadarmo je možné i změřit velikost napájecích napětí s přesností na 10 bitů.

4.5.5. Zapojení napájecího zdroje

Napájecí zdroj obsahuje obvody pro detekci napětí na obou napájecích úrovních, spínač napětí pro napájení vysílače SENT a přepínač zdroje napájení společně s se stabilizátorem. Jeho kompletní zapojení je na obrázku 4.8.



Obrázek 4.8: Schéma zapojení napájecí části převodníku

Oba dva vstupní obvody jsou zapojeny shodně. Na vstupu jsou dva odporové děliče, jeden pro detekci +5 V (R5, R6, C5), druhý pro detekci přítomnosti U_{SPI} (R7, R8, C6). Výstupní napětí těchto děličů musí být větší než 1,5 V při minimální úrovni napájení a nesmí překročit 3 V při maximu. Výstupy jsou zavedeny do analogového komparátoru v mikrokontroléru. C5 a C6 jsou zde jako ochrana proti případným zákmitům napájecí linky, které by mohly způsobovat detekci falešných výpadků. Společně s děličem tvoří RC článek.

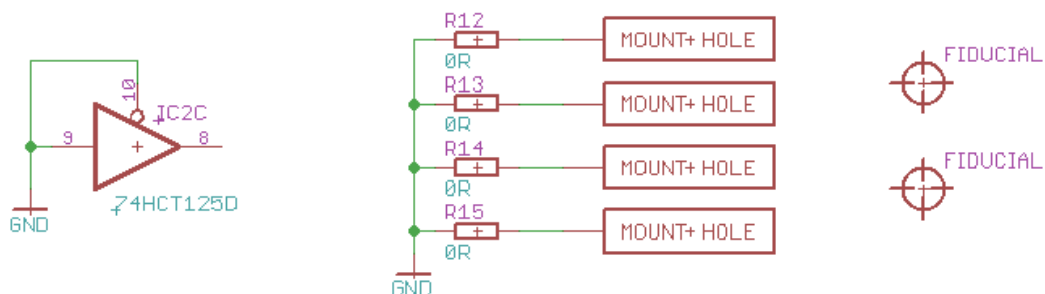
Další částí obvodu je přepínač zdrojů, tvořený T1, T2 a R9, funkce tohoto spínače byla vysvětlena v kapitole Napájecí hladina mikrokontroléru. Z něj jde napájecí proud do stabilizátoru IC4. Ten je blokován proti zákmitům kondenzátory C7 a C9. Výstup PG, indikující správnou hladinu napájení, je použitý pro signál \overline{RESET} . Není-li \overline{RESET} aktivní, je udržován v log. 1 rezistorem R11.

Pro vytvoření U_{SENT} je větev +5 V odbočena do spínače IC3, který současně plní i funkci nadproudové ochrany a indikátoru přetížení. Signálem SENTON v log. 1 je zapnuto napájení vysílače. Je-li signál SENTOK (výstup s otevřeným kolektorem, udržovaný v klidové úrovni log. 1 rezistorem R10) v log. 0, znamená to, že vysílač je vypnutý z důvodu přetížení.

4.6. Doplnění schématu

Mimo součástek, které jsou nutné k zajištění funkce převodníku, je nutné i zapojen součástí či jejich částí, které nezaručují samotnou funkci převodníku, ale jsou důležité z pohledu EMC, technologie a podobně. Jejich zapojení je na obrázku 4.9. Jedná se o tyto tři části:

1. Ošetření nezapojených vstupů. V tomto případě se jedná o IC2C, který by se při nezapojených vstupech mohl náhodně rozkmitat a způsobit tak rušení, nebo zůstat v zakázané oblasti s vysokým odběrem. Jako prevence jsou oba nevyužité vstupy připojeny na potenciál 0 V (v tomto případě není úroveň rozhodující, byla zvolena 0, protože je dostupná bez prokoveného otvoru).
2. Montážní otvory je možné nechat plovoucí, připojit přímo na zem nebo uzemnit přes definovanou impedanci. Toto mají na starost rezistory R12 až R15, které jsou standardně neosazeny.
3. Fiduciární značky, které slouží k navádění osazovacího automatu. Pro desku v této přesnosti a rozměru jsou dostačující dvě.



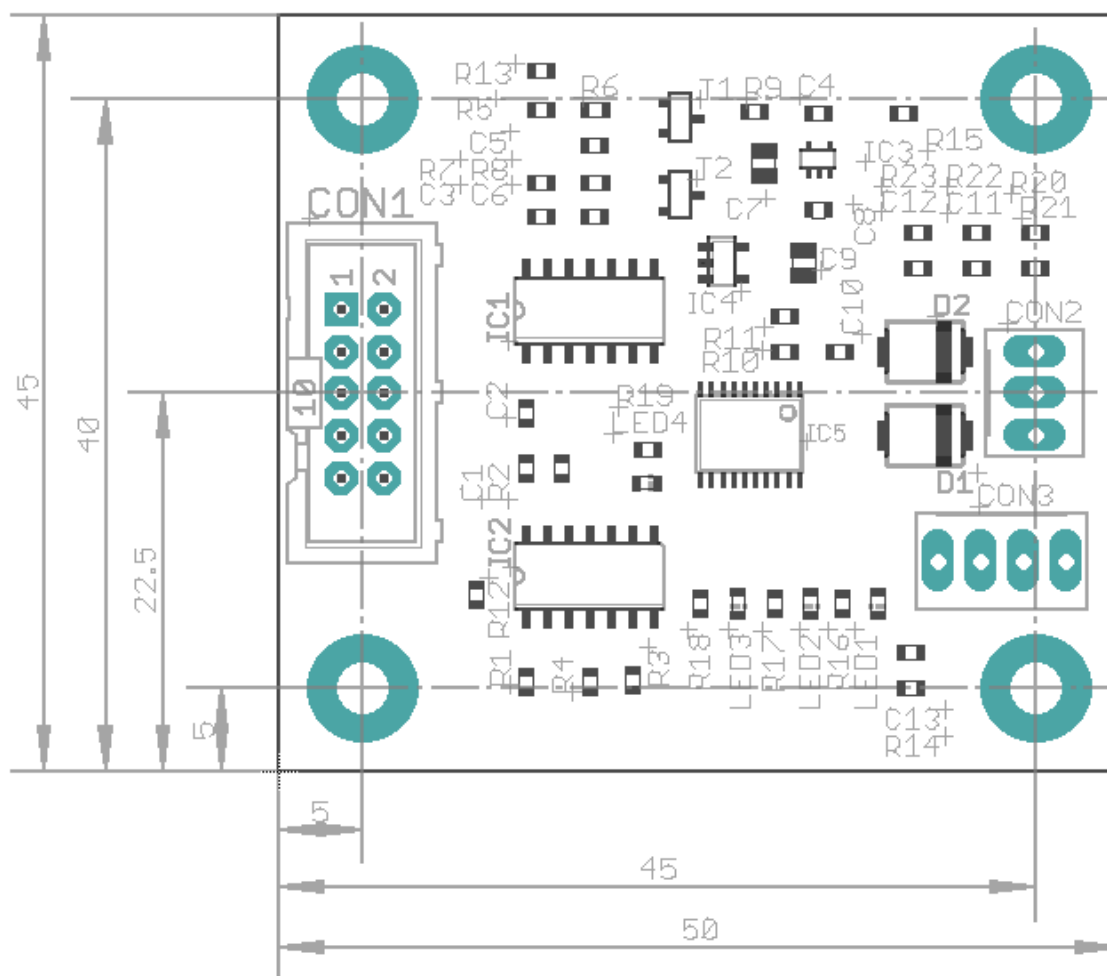
Obrázek 4.9: Zbývající části schématu zapojení převodníku

4.7. Realizace prototypu

Jak již bylo zmíněno na začátku popisu hardware, mechanicky se jedná o jednodeskový modul, připojený pomocí konektorů. Při jeho konstrukci byla vzata do úvahy jak univerzálnost montáže, tak i technologie výroby a cena. Celý modul byl navržen tak, aby:

- Byl obdélníkového tvaru, což umožní levnější dělení přířezu drážkováním. Rozměr je volen tak, aby v přířezu bylo maximum kusů s co nejmenším odpadem.
- Měl v rozích otvory pro upevnění. Tyto otvory jsou prokovené (ušetří se operace vrtání) s možností připojení na **Gnd** (v případě problémů s EMC). Předpokládané použití je se šroubky M3.
- Mimo konektorů byly všechny součástky SMT, osazené z jedné strany. Je tak umožněno osazení při jednom průchodu SMT + reflow s tím, že pro konektory je možné použít technologii PIP, ruční pájení nebo pájecí vlnu.

Výsledné rozměry jsou na obrázku 4.10. Tyto rozměry vychází z technologie firmy Gatema Boskovice. Standardní rozměr přířezu je 255 x 317 mm. Zde je možné rozmístit tyto desky v matici 5 x 4 ks s okrajem 2 mm na delší straně a 12,5 mm na kratších stranách pro manipulaci.



Obrázek 4.10: Mechanické provedení desky převodníku (není v měřítku)

4.7.1. Provedení desek s plošnými spoji

Desky na prototypy byly vyrobeny na standardní materiál FR4 s plátováním 18 μm metodou *pattern plating*. Jde o desku v konstrukční třídě IV (cesta 0,2 mm, mezera 0,2 mm, prokovy 0,5 mm). Tato konstrukční třída je zvolena tak, aby bylo možné vést cestu pod pasivní součástkou 0603 a s ohledem na rozteč vývodů mikrokontroléru 0,65 mm.

4.7.2. Seznam součástek

Seznam součástek převodníku je v tabulce 4.10. Neobsahuje rezistory pro připojení montážních otvorů na zem.

Tabulka 4.10: Seznam součástek převodníku SENT-SPI

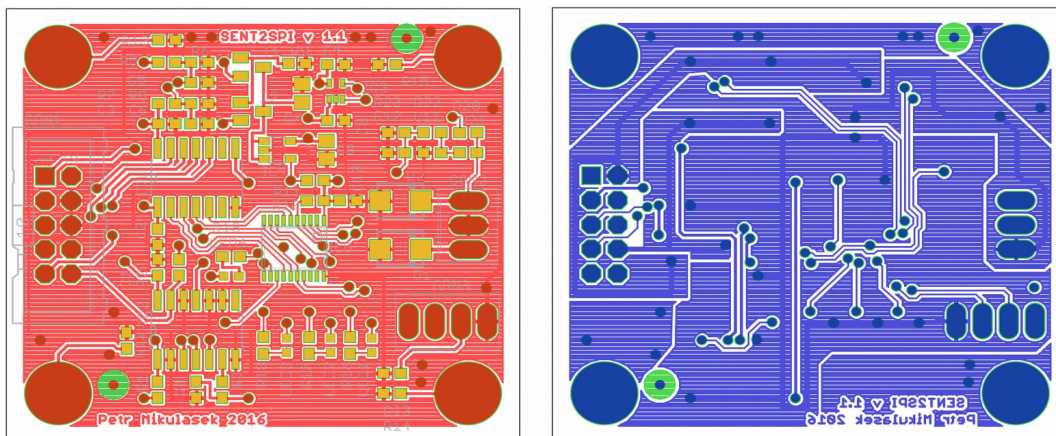
| Počet | Hodnota | Pouzdro | Popis | Pozice |
|-------|------------------|---------|---|----------------------------------|
| 1 | TXS0104ED | SO14 | 4-Bit Bidirectional Voltage-Level Translator For Open-Drain and Push-Pull | IC1 |
| 1 | 74HCT125D | SO14 | Quad bus BUFFER, 3-state | IC2 |
| 1 | TPS22943DCKR | SC70-5 | Low-input-voltage Current-limited Load Switch | IC3 |
| 1 | LP3988IMF-3.0 | SOT23-5 | Micropower, 150-mA Ultra-Low-Dropout CMOS Voltage Regulator | IC4 |
| 1 | MSP430G2553IPW20 | TSSOP20 | Microcontroller | IC5 |
| 2 | BC807-40 | SOT23 | PNP Transistor, universal | T1, T2 |
| 2 | SMAJ5.0A | SMA | TVS Diode, SMA | D1, D2 |
| 2 | GL-0603YC1 | 0603 | LED, orange | LED1, LED2 |
| 1 | GL-0603GC | 0603 | LED, red | LED3 |
| 1 | GL-0603RC | 0603 | LED, green | LED4 |
| 8 | 10k | 0603 | Resistor, 10k | R1, R2, R6, R7, R8, R9, R10, R11 |
| 2 | 24k | 0603 | Resistor, 24k | R3, R22 |
| 2 | 36k | 0603 | Resistor, 36k | R4, R23 |
| 1 | 22k | 0603 | Resistor, 22k | R5 |
| 5 | 560R | 0603 | Resistor, 560R | R16, R17, R18, R19, R21 |
| 1 | 20k | 0603 | Resistor, 20k | R20 |
| 6 | 100n | 0603 | Ceramic capacitor, 100nF, X5R | C1, C2, C3, C4, C8, C10 |

| Počet | Hodnota | Pouzdro | Popis | Pozice |
|-------|------------|----------|-------------------------------|------------------|
| 4 | 2n2 | 0603 | Ceramic capacitor, 2,2nF, NP0 | C5, C6, C11, C13 |
| 2 | 10u/16V | 0805 | Ceramic capacitor, 100nF, X7R | C7, C9 |
| 1 | 68pF | 0603 | Ceramic capacitor, 68pF, NP0 | C12 |
| 1 | ML10W10G | ML10 | 2-rows header, 10pin | CON1 |
| 1 | PSH02-03PG | PSH02-03 | Header with lock, 3 pin | CON2 |
| 1 | PSH02-04PG | PSH02-03 | Header with lock, 4 pin | CON3 |

4.7.3. Gerber data

Gerber data desky ve formátu pro výrobu ve firmě Gatema jsou k dispozici na přiloženém disku CD-ROM (podrobnosti o jejich umístění jsou v příloze 1).

Na obrázku 4.11 jsou zobrazena gerber data pro spodní stranu desky (modrá je měď, zelená je nepájivá maska) a horní stranu desky (červená je měď, modrozelená je nepájivá maska, šedá servisní potisk). K vygenerování obrázků byl použit program Gerbv.



Obrázek 4.11: Gerber data navržené desky

5. Firmware

Nedílnou součástí převodníku je firmware, který řídí činnost mikrokontroléru. Vzhledem k ceně, dostupnosti a použité platformě byl zvolen jazyk C [11] a kompilátor *MSP-GCC*.

5.1. Přijímač SENT

Stěžejní částí celé aplikace, od které se vše odvíjí, jsou komunikační rozhraní. A z obou rozhraní je nejkritičtější návrh rozhraní SENT, protože při chybě dojde ke ztrátám dat a tím i k výpadkům funkce celého převodníku. Je proto třeba pečlivě dimenzovat přijímač tak, aby spolehlivě pracoval i při nejhorsích podmínkách, které norma [3] dovoluje.

Podrobnosti o fungování tohoto rozhraní lze najít v kapitole Sběrnice SENT.

5.1.1. Analýza časování

Návrh časování sběrnice vychází z předpokladů v tabulce 5.1. Aby nedocházelo ke ztrátě dat, musí být zpracování zprávy rychlejší, než její příjem, se zohledněním sdílení času jednoho jádra. Tento fakt se podepisuje v architektuře celého přijímače.

Tabulka 5.1: Mezní hodnoty časování sběrnice SENT

| Parametr | Význam | Min | Max | Jednotka |
|-------------------|-----------------------------|-----|-----|----------|
| TU | Základní jednotka | 3 | 90 | μs |
| T _{SYNC} | Doba synchronizačního pulsu | 56 | 56 | TU |
| T _{NIB} | Doba přenosu jednoho nibble | 12 | 27 | TU |
| N _{NIB} | Počet nibblů | 4 | 8 | |
| T _{DLY} | Zpoždění po přenosu paketu | 0 | 768 | TU |

Z tabulky 5.2 vyplývá, že přerušení mohou přicházet v nejhorsím případě s periodou

$$T_{INTMIN} = TU_{MIN} \cdot T_{NIBMIN} = 36 \mu s \quad (5.1)$$

a minimální doba příjmu paketu je

$$T_{PKMIN} = TU_{MIN} \cdot (t_{SYNCMIN} + t_{NIBMIN} \cdot N_{NIBMIN} + t_{DLYMIN}) = 3 \cdot (56 + 4 \cdot 12 + 0) = 312 \mu s \quad (5.2)$$

Dále je důležité vědět, že za dobu zpracování paketu přijde minimálně pět a maximálně 10 přerušení, ale hodnota nad pět přerušení již znamená delší pakety. Tyto parametry se promítají do návrhu přijímače, kde je nutné zvolit architekturu v závislosti na těchto časech a parametrech obvodu.

5.1.2. Přerušení od časovače

Hrany jsou detekovány pomocí přerušení. Aby mikrokontrolér stíhal jejich zpracování, bylo rozhodnuto nesledovat náběžné hrany, čímž se počet přerušení sníží na polovinu. Protože čas, nutný pro obsluhu přerušení, je součinem počtu přerušení a doby jeho zpracování ani četnost přerušení nelze měnit, je nutné minimalizovat čas přerušení a tím i počet operací. Přerušení od časovače tedy bylo omezeno na pouhý zápis vzorku do vyrovnávací paměti.

Klíčové pro časovač je zamezit ztrátě dat a dosáhnout rychlé reakce. Pro časovač jsou k dispozici hodiny s frekvencí 12 MHz a jeho délka je 16 b. Pro rozhodnutí, zda je nutné přepínat rozsahy měření, je klíčová maximální měřená délka impulsu, tedy synchronizačního délky 56TU. Při časové jednotce 90 μ s tato hodnota

$$N_{MAX} = 56 \cdot TU \cdot f_s = 56 \cdot 3 \cdot 10^{-6} \cdot 12 \cdot 10^6 = 60480 \quad (5.3)$$

Je tedy zřejmé, že není nutné přepínat rozsahy, ale během příjmu zprávy dojde s velkou pravděpodobností k přetečení. Při použití neznaménkových (*unsigned*) čísel je ovšem jejich rozdíl shodný bez ohledu na přetečení, takže tato skutečnost nepředstavuje problém. Nepovinné pauzy jsou zahazovány a zde případná odchylka od skutečné hodnoty nevádí.

5.1.3. Vyrovnávací paměť

Jako vyrovnávací paměť byl zvolen kruhový buffer. Důvodem pro tuto volbu je možnost jeho opakovaného použití bez přesunů dat nebo dynamické alokace paměti a jednoduchost obsluhy. Také nabízí velmi efektivní využití přidělené RAM.

Pro velikost této vyrovnávací paměti je důležité, aby mezi zpracováním dat nedošlo k jeho přetečení. Závisí tedy jak na datovém toku, tak i na periodě zpracování dat, kterou jedinou lze ovlivnit. Ostatní parametry jsou dány vysílačem SENT.

Vyšší frekvence spouštění vede ke zmenšení vyrovnávací paměti, ale také k vyšší režii při přepínání úloh a omezení délky jejich běhu.

Minimální velikost potřebné vyrovnávací paměti je možné určit dle vzorce

$$M_{MIN} = B \cdot \left(\frac{T_{TASK}}{TU_{MIN} \cdot (56 + 12 \cdot (2 + N))} \right) \quad (5.4)$$

kde B je počet bajtů na jeden impuls (2 B), T_{TASK} je perioda zpracování a N je počet přijatých nibblů v rozsahu 2 až 6. Hodnotu je nutné zaokrouhlit nahoru na celá čísla.

Pro rozsah 1 až 10 ms jsou paměťové nároky na obrázku 5.1 Jak je vidět, minimální velikost paměti pro zpracování po 1 ms je 54 B a při 2 ms jsou již nároky minimálně 90 B. Jako nejlepší možnost bylo zvoleno zpracování s periodou 2 ms a vyrovnávací paměť 128B (64 hran neboli $\frac{1}{4}$ RAM).

5.1.4. Architektura přijímače

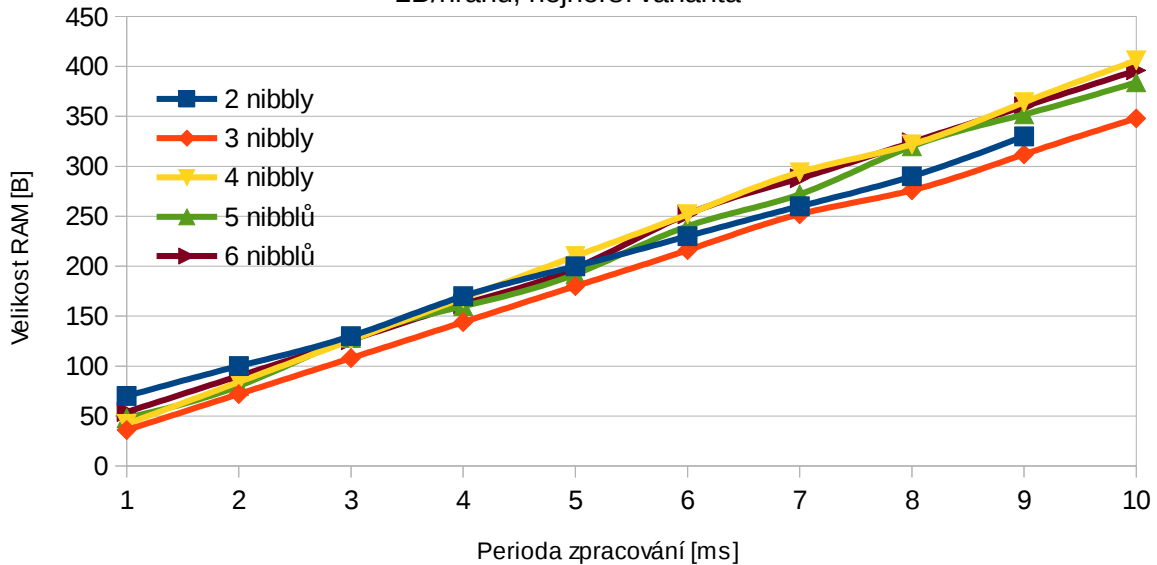
Architektura přijímače SENT na nejvyšší úrovni je na blokovém schématu na obrázku 5.2. Z výše popsaných důvodů optimalizace přerušení a vyrovnávací paměti je zvoleno ukládání změřených hodnot do paměti FIFO a jejich zpracování v periodicky spouštěné úloze každé 2ms.

Z pohledu architektury je nutné ještě znát alespoň přibližně datový tok na výstupu. Ten se skládá ze zpráv, které mohou teoreticky přijít až čtyři každou milisekundu. Datový tok je nejvyšší v okamžiku, kdy se s minimální časovou jednotkou přijímají zprávy, složené z pěti nibblů. Při předpokladu, že zpráva ve FIFO obsahuje 1 B hlavičku a 3 B dat a její příjem trvá 420 μ s, je možné datový tok určit jako

$$BR_M = N_{BYTES} \cdot \frac{1}{t_{MSG}} = 4 \cdot \frac{1}{4,2 \cdot 10^{-4}} = 9524 \text{ B/s} \quad (5.5)$$

Minimální velikost vyrovnávací paměti

2B/hranu, nejhorší varianta



Obrázek 5.1: Minimální velikost FIFO pro přijímač SENT

Dále zde mohou být i zprávy SSM a ESM. Jejich datový tok je

$$BR_{SSM} = N_{BSSM} \cdot \frac{1}{t_{SSM}} = 3 \cdot \frac{1}{4 \cdot 10^{-3}} = 750 \text{ B/s} \quad (5.6)$$

pro SSM a

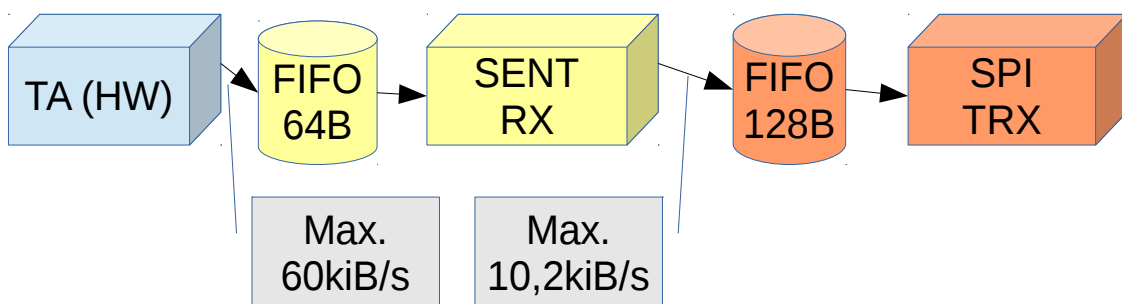
$$BR_{ESM} = N_{BESM} \cdot \frac{1}{t_{ESM}} = 4 \cdot \frac{1}{4,5 \cdot 10^{-3}} \approx 890 \text{ B/s} \quad (5.7)$$

pro ESM. Celkově je tedy datový tok dekodovaných zpráv, tvořený součtem toku dat a toku ESM dat přibližně 10,2 kiB/s.

Protože jsou zprávy zpracovávány po dávkách a nelze zajistit, že SPI master vyčte data ihned po každé dávce, je nutné za dekodér vložit ještě druhou vyrovnávací paměť na dekodované zprávy. Opět je výhodné použití FIFO, minimální kapacita je závislá na četnosti komunikace se SPI masterem. Přidělíme-li na tuto paměť 128B RAM, je možné do ní ukládat data po dobu

$$t_{BUFF} = \frac{N_{BYTES}}{BR_M + BR_{ESM}} = \frac{128}{9524 + 890} = 12,3 \text{ ms} \quad (5.8)$$

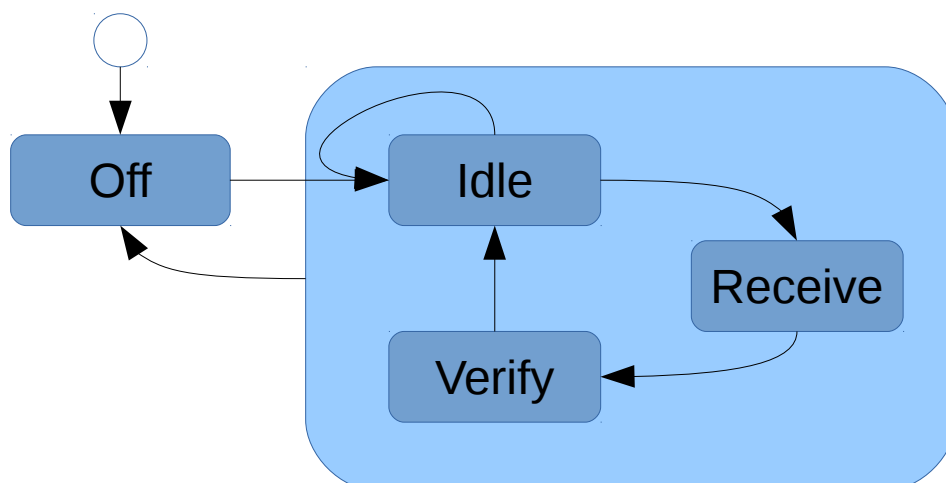
Je tedy možné použít tuto velikost paměti za předpokladu, že bude specifikováno vyčítání dat s periodou 10 ms s tím, že 2,3 ms jsou rezerva pro případ zpoždění na straně masteru.



Obrázek 5.2: Architektura přijímače SENT na nejvyšší úrovni s vyznačením datových toků

5.1.5. Zpracování signálu

Zpracování signálu probíhá ve stavovém automatu, kde má každá vstupní hodnota svůj specifický význam a podle něj je zpracována. Stavový diagram je na obrázku 5.3, popis jednotlivých stavů v tabulce 5.2.



Obrázek 5.3: Stavový automat přijímače SENT

Tabulka 5.2: Popis jednotlivých stavů přijímače

| Stav | Význam | Zpracování dat |
|------------|---|--|
| OFF | Neprobíhá komunikace. Napájení vysílače může být vypnuto nebo přerušeno, reset přijímače nebo vysílače. | Čeká na první hranu, kterou vezme jako základ pro výpočet délky impulsu. Přepne se do stavu NOTSYNC. |

| Stav | Význam | Zpracování dat |
|----------------|---|--|
| IDLE | Přišla minimálně jedna hrana, ale není známé, která hrana ve zprávě to je a není zatím navázána komunikace. | Vezme další hranu a vypočítá na základě předchozí délky impulsu. Výsledek zkusí dekodovat na základě časové jednotky, vypočtené z předchozího impulsu. Pokud vyhovuje daným mezím, je aktuální impuls stavovým nibblem. Předá jej ke zpracování a přejde do stavu RECEIVE |
| RECEIVE | Přijímač je synchronizovaný a přijímá data zprávy | Přijatý nibble je dekodovaný na základě známé časové jednotky. Uloží se a přidá jej k CRC. Je-li dosaženo požadovaného počtu nibblů, přejde do stavu VERIFY |
| VERIFY | Zkontrolování a odeslání zprávy | Přijme CRC, zkontroluje zprávu a odešle k dalšímu zpracování. Poté se vrátí do stavu IDLE . |

Dekódování nibblu spočívá, jednoduše řešeno, ve třech krocích:

1. Výpočet délky impulsu ze dvou hran
2. Vydělení délky impulsu časovou jednotkou
3. Odečtení 12.

Poněkud problematické je získání délky časové jednotky, která musí být detekována ze synchronizačního impulsu. Ten má délku 56 TU, tedy se zde nabízí vydělení délky impulsu touto hodnotou. Zbývá jen určit, o jaký druh impulsu jde. Naštěstí na to existuje jednoduchý algoritmus, zachycený na obrázku 5.4

Tento algoritmus se používá i pro eliminaci volitelné pauzy na konci zprávy a pro synchronizaci na novou zprávu. Zde není, z důvodu návaznosti, nulování první hrany a je vhodné, aby předchozí časová jednotka alespoň řádově odpovídala předchozí časové jednotce pro vyloučení kolize s poměrem pauza:synchronizace 14:3.

Poslední trik, který se dekodování nibblů týká, je zavedení hystereze. Takto realizovaný přijímač by, pokud by napočítal i jen o jeden tik časovače méně, dekodoval nižší hodnotu nibblu, nebo chybu. Proti tomuto byla použita jednoduchá obrana, která ve finále stojí jen dvě instrukce. Spočívá v tom, že přijímač používá pro dekodování polovinu časové jednotky. Data pro dekodování tak mají dvojnásobnou hodnotu a v nejnižším bitu je pak 0, pokud čítač dodal správnou nebo vyšší hodnotu, nebo 1, pokud byla hodnota nižší. K tomuto výsledku stačí přičíst jedničku (inkrement), u kratšího času se pouze neguje tento bit, který je při dělení dvojkou (rotace vpravo) zahozen a u kratšího času o onu chybějící jedničku posune výsledek výpočtu nahoru.

Zpracované nibbly se následně počítají a po dosažení požadovaného počtu jsou ukládány do výstupní vyrovnávací paměti. Průběžně je počítána i hodnota CRC polynomu. Po dosažení zvoleného počtu nibblů se přijme poslední, kontrolní nibble a výsledek je uložen do vyrovnávací paměti.

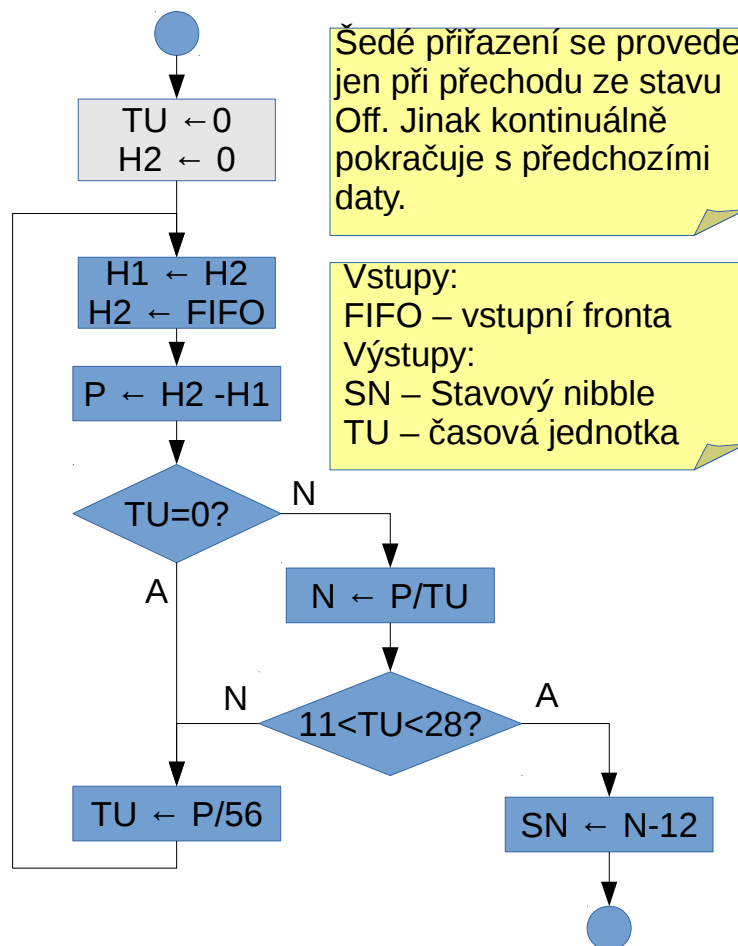
5.1.6. Zpracování sériových zpráv

Tento systém pracuje jako paralelní větev ke zpracování standardních zpráv. Jako zdroj dat využívá stavové nibbly po sobě jdoucích paketů. Dva nejvyšší bity jsou ukládány do dvou posuvných registrů. Datový tok při uvažované periodě 2 μs je pro každý z těchto registrů

$$DR = \frac{\text{bits} \cdot 1 \text{ ms}}{T_{PKMIN}} = \frac{2 \cdot 1 \cdot 10^{-3}}{312 \cdot 10^{-6}} = 6,41 \text{ b/ms} \quad (5.9)$$

Zaplnění celého 16 b posuvného registru trvá tedy necelých 5 ms. Je tedy nutné zpracovat tyto data s menší periodou, a to maximálně 4 ms.

Dekodér po spuštění bere jednu dvojici bitů za druhou a přesouvá je do hlavních posuvných registrů délky 32 b. Tam jsou jen zamaskovány variabilní bity (ID, data, CRC) a porovnány s předlohou pro SSM a stejná operace je opakována i pro ESM. Při nalezení shody jsou data předána ke kontrole a hlavní registr vynulován. Během tohoto zpracování je z časových důvodů možné zpracovat pouze nejvýše 16 párů bitů. Po nalezení sériové zprávy je ověřeno její CRC a následně je uložena do vyrovnávací paměti pro vysílač SPI.



Obrázek 5.4: Princip hledání synchronizačního impulsu

5.1.7. Chyby přijímače a jejich ošetření

Během zpracování přijímaných dat může dojít k několika možným chybám. Jejich přehled je v tabulce 5.3. Jak z této tabulky vyplývá, ve většině případů je generována chyba.

V případě, že se SPI master rozhodne chybu napravit prostřednictvím restartu vysílače SENT, může tak učinit pomocí povelů a sběrnici SPI a sám si určit dobu, po kterou vypne napájení..

Tabulka 5.3: Přehled chybových stavů přijímače

| Číslo | Popis chyby, příčiny | Reakce |
|-------|--|---|
| 1 | Nepřichází vstupní signál. Po dobu několika průchodů je prázdná FIFO | Zhasnutí LED SENT Opakovaná inicializace časovače Informování SPI masteru o výpadku |
| 2 | Příliš rychlá komunikace na sběrnici. Dekódováno záporné číslo, nebo je časová jednotka mimo rozsah. | Rozsvícení chybové LED Informování SPI masteru |
| 3 | Příliš pomalá komunikace. Dekódováno číslo větší než 15, nebo je časová jednotka mimo rozsah. | Rozsvícení chybové LED Informování SPI masteru |
| 4 | Ztráta synchronizace. Datový nibble je mimo povolený interval. | Návrat do stavu NOTSYNC a předání chybové zprávy s informací, co se stalo. |
| 5 | Přetečení FIFO časovače. | Vede k chybě komunikace, není ošetřeno separátně. |
| 6 | Chyba CRC zprávy | Informování SPI masteru v bloku zprávy |
| 7 | Chyba CRC sériové zprávy | Informování SPI masteru v bloku zprávy |
| 8 | Přetečení vyrovnávací paměti. | Blok, který přepsal data, má nastaven příznak přetečení. |

5.2. Rozhraní SPI

Pro rozhraní SPI je použito odpovídající rozhraní mikrokontroléru, tedy **USCI**. Obsahuje samotný vysílací a přijímací registr, taktované signálem **SCK**, Pro signál \overline{CS} je výhodné mít detekci jeho stavu, ale tato funkce není periferií podporována a ani příslušný řadič portu neposkytuje při změně tohoto signálu přerušeni, pokud je přiřazen k periferii **USCI**. Z tohoto důvodu musel být tento signál odpojen od **USCI** a jeho funkce je simulována firmwarem za pomoci přerušeni.

5.2.1. Příjem

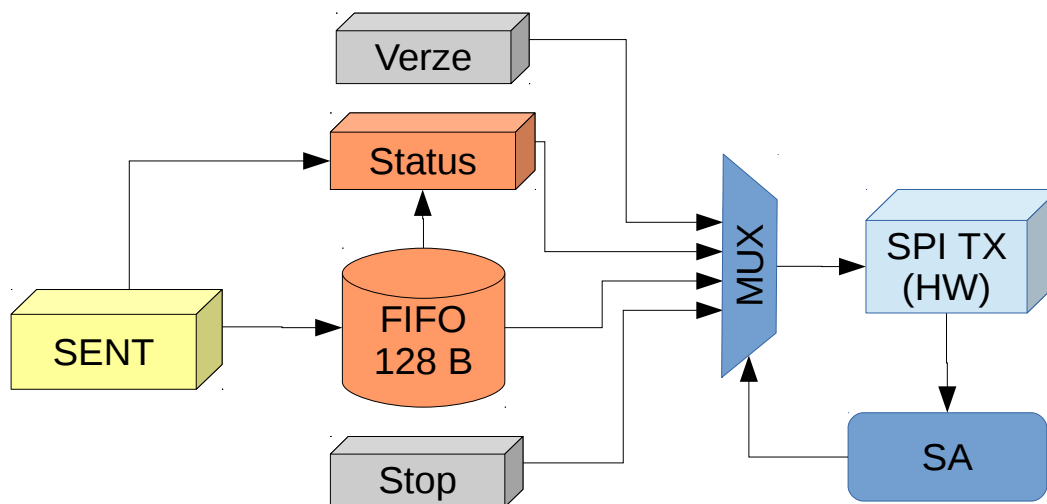
Příjem je postavený na jednoduchém stavovém automatu. Tento je po inicializaci ve stavu **READY**. V tomto stavu je po přerušeni přijímače SPI vyhodnocen první bajt a jedná-li se o klíč, je přijímač poslán do stavu **RECEIVE**. Další přijatý bajt je vyhodnocen a použit jako nastavení.

Protože nastavení smí být pouze na začátku dat, je přijetím příkazu nebo neplatného klíče přijímač přepnut do stavu **DONE**, ve kterém zahazuje přijatá data do sestupné hrany signálu \overline{CS} . Tou je přijímač restartován zpět do stavu **READY**.

Zapsané změny jsou aplikovány od okamžiku zpracování zpráv (pro délku zprávy), s výjimkou **ENABLE**, který je aplikován při další kontrole napájení.

5.2.2. Vysílání

Vysílací protokol byl specifikovaný v kapitole 3.2 a struktura paketu je na obrázku 3.1. Pro odvysílání tohoto paketu jsou třeba celkem čtyři zdroje dat, jak ukazuje obrázek 5.5. Data z rozhraní SENT jdou do vyrovnávací paměti (viz kapitola 5.1.4). Současně je přijímačem SENT reportován stav zprávy, například chyba CRC. FIFO také reportuje do stavového registru, zde je uložen počet nevyzvednutých zpráv. Dále jsou pro sestavení paketu nutné dvě konstanty, úvodní verze protokolu a na konci označení konce paketu. Z těchto čtyř zdrojů jsou přepínána data během vysílání v závislosti na vysílacím stavovém automatu.

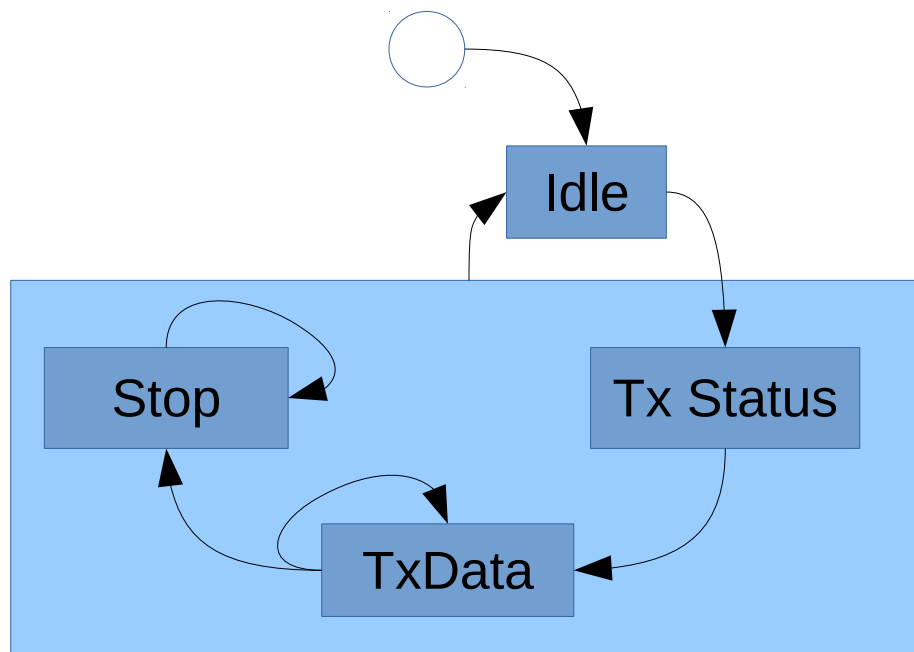


Obrázek 5.5: Datové toky pro vysílač SPI

Stavový automat je řízený přerušeními z hardwarového modulu **USCI**. Tento modul poskytuje informace o náběžné a sestupné hraně signálu \overline{CS} a přerušení pro prázdný vysílací buffer. Stavový diagram vysílače je na obrázku 5.6.

Po inicializaci a během nečinnosti se stavový automat nachází ve stavu **IDLE**. Při vstupu do tohoto stavu se restartuje či inicializuje periferie **USCI** a do vysílače je nastavena konstanta s verzí protokolu. V tom stavu čeká na sestupnou hranu signálu \overline{CS} . Ta jej přepne do stavu **TxSTATUS**, kde je vysílaný stavový bajt a počet bloků dat.

Ihned po odvysílání hlavičky následuje vysílání bloků dat ve stavu **TxDATA**. Vždy je odvysílán celý blok dat a poté se opakuje vysílání pro další blok, dokud není odeslán počet bajtů, který byl ve vyrovnávací paměti na začátku komunikace. Po odvysílání celého bloku je pak vysílač plněn konstantou, indikující konec.



Obrázek 5.6: Stavový automat vysílače SPI

Je-li přenos deaktivován náběžnou hranou signálu \overline{CS} , ať se jedná o kterýkoliv stav, je stavový automat vrácen do stavu **IDLE**, restartována periférie a připravena konstanta s verzí do vysílače.

Praktická realizace se snaží co nejvíce zrychlit přerušení od prázdného vysílače, proto je použita univerzální přerušovací rutina, která vždy udělá tři věci v uvedeném pořadí:

1. Vezme znak z definované adresy a uloží do vysílače.
2. Dekrementuje čítač zbývajících bajtů.
3. Zavolá vyhodnocovací funkci pro zjištění, zda má přejít do jiného stavu.

Všechny parametry pro tuto funkci - adresa zdroje dat, čítač a ukazatel na funkci – jsou ve formě proměnných, které se mění v závislosti na stavu vysílače a tvoří tak pomyslný multiplexer a řízení stavového automatu.

Přerušení od signálu \overline{CS} vždy neguje hranu, kterou má příště zachytit, a v závislosti na zachycené hraně povolí nebo zakáže přerušení od modulu **USCI**. Současně mění požadovaný stav stavového automatu a u náběžné hrany restartuje příslušný hardware.

5.3. Diagnostika a management napájení

Napájení je sledováno s periodou 4 ms v k tomu určené úloze. Ta zkontroluje oba komparátory a vstup indikující přetížení napájení. Na jejich základě, pokud došlo k chybě, je ovládána indikační LED pro chybu a zapsán stav napájení do stavového slova.

Tato úloha je současně využívána k vypínání a zapínání převodníku pomocí povelu ze sběrnice SPI, kdy tato úloha zkontroluje příslušný bit přijatého povelu a podle něj zapne nebo vypne napájení pro vysílač SENT.

5.4. Výstup přerušeni

Výstup přerušeni pro SPI master má dva významy a je tedy ovládán ze dvou míst programu, jak naznačuje tabulka 5.4 Každá z těchto funkcí může být zamaskována odpovídajícím bitem v řídicím povelu převodníku.

Přerušeni je vždy nulováno odvysíláním stavového slova na sběrnici SPI.

Tabulka 5.4: Přehled přerušeni na sběrnici SPI

| Příčina | Zdroje | Maskování |
|------------|--|------------------------|
| Příjem dat | - Přijetí zprávy - Přijetí sériové zprávy | Bitem 4 řídicího slova |
| Chyba | - Výpadek napětí +5 V - Výpadek napájení pro SPI - Přetížení zdroje pro SENT vysílač - Výpadkem dat na sběrnici SENT po dobu minimálně 2 ms | Bitem 5 řídicího slova |

5.5. Práce s indikačními LED

V zapojení převodníku je čtveřice LED, které jsou uvedeny v tabulce 4.4, zapojených dle obrázku 4.5. Jak zapojení, tak i ovládání se liší u každé z nich.

LED1 slouží k indikaci aktivity na sběrnici SENT.. K jejímu rozsvícení dojde nastavením příslušného výstupu do nuly vždy, když modul přijímače SENT najde nějaká data z přijatých hran ve vyrovnávací paměti. Naopak, nepřijme-li po dobu 2ms žádná data neboje-li vypnuté napájení vysílače, je výstup nastaven do jedničky a LED1 nesvítí.

LED2 plní funkci indikátoru rozhraní SPI a je rozsvícena vždy, když je aktivováno rozhraní SPI. K její deaktivaci dojde z úlohy pro kontrolu napájení v případě, že nedošlo od minulého spuštění ke komunikaci se SPI masterem.

Dále je zde LED3 pro indikaci aktivity programu. Ta je aktivována v okamžiku startu programu a je aktivována pouze restartem firmware. To, že nesvítí, indikuje problém s mikrokontrolérem.

Poslední v pořadí je LED4. Tato indikuje chyby v běhu převodníku, které jsou indikovány i přerušeni, doplněné informací o přetečení vyrovnávacích pamětí. Je řízena sloučením všech zdrojů chyb do jednoho. Jako všechny LED, i tato je spínána logickou nulou na výstupu.

5.6. Časování jednotlivých úloh

V tabulce 5.5 jsou zrekapitulovány jednotlivé úlohy, které se periodicky opakují, jak byly navrženy v předchozích kapitolách. Jejich časování je odvozeno z druhého, nepoužitého časovače TA.

Tabulka 5.5: Přehled úloh operačního systému

| Úloha | Význam | Perioda |
|-------------------|--|---------|
| Příjem SENT | Přijetí a dekodování zpráv ze sběrnice SENT včetně synchronizace a kontroly dat. | 2 ms |
| Příjem SSM, ESM | Zpracování stavového nibble, sestavení bitového proudu, rozeznání a zpracování zprávy. | 4 ms |
| Kontrola napájení | Kontroluje stav napájecího napětí a aplikuje nastavení napájení. | 4 ms |

Jak vyplývá z tabulky 5.5, úlohy se střídají v pořadí Příjem SENT → Zpracování ESM → Příjem SENT → Kontrola napájení s krokem 2 ms. Běh těchto úloh je přerušován periferiemi, tj. úlohy mají po započítání režie a přerušení cca 1ms na zpracování dat.

Jak již bylo uvedeno, všechna přerušení jsou minimalizována na nezbytně krátkou dobu. Úlohy jsou proto spouštěny jen z hlavní smyčky. Z přerušení je sem přenesena jen informace o tom, že došlo k přerušení od systémového časovače. Byla zvolena prostá metoda sdílení volatilní proměnné mezi přerušením a hlavní smyčkou. Přerušení pouze inkrementuje tuto proměnnou a o nic víc se nestará.

Je-li zaručeno, že je na vstupu sudá hodnota, je po přerušení na výstupu lichá hodnota. Tedy pro předání do hlavní smyčky stačí testovat nejnižší bit a je-li jedničkový, inkrementovat jej. Ve vyšších bitech je pak uložený počet přerušení od časovače a dle něj je možné realizovat spouštění jednotlivých úloh. V tomto případě se o vše postará příkaz *switch*.

Z hlavní smyčky je také nulován watchdog. Perioda jeho nulování je 2 ms.

5.7. Podklady pro tvorbu firmware

Firmware byl realizovaný na základě další literatury, která nebyla zmíněna ohledně architektury a funkce firmware. Nejdůležitějším dodatečným podkladem je v tomto ohledu manuál k mikrokontroléru „MSP430x2xx Family User's Guide“ [12], kde jsou popsány periferie i jádro mikrokontroléru včetně jejich ovládání. Kompilátor byl nakonfigurován dle manuálu k MSP GCC od Steve Underwooda [13].

Jako datová struktura všech vyrovnávacích pamětí je použita FIFO. Základní implementaci je možné najít například v [14] na stranách 59-61. Zbytek kódu byl vytvořen přímo na míru této aplikaci.

6. Závěr

Při realizaci bakalářské práce byl brán ohled na moje předchozí zkušenosti. Proto byla práce orientována na součástkovou základnu *Texas Instruments*. Tento výrobce se v minulosti osvědčil jak kvalitou, tak i cenou a dostupností jejich produktů a v neposlední řadě, technická podpora byla schopná jakýkoliv problém vyřešit do tří pracovních dnů. Tento projekt byl proto vhodná příležitost, jak se seznámit s levnou, nízkopříkonovou řadou mikrokontrolérů **MSP430G2xxx**. Po předchozí zkušenosti s DSP **TMS320LF2407** a s řadou **MSP430F41x**, kde jsem dělal jen vyšší vrstvy firmware, to byla příležitost osahat si tyto mikrokontroléry i na úrovni hardware.

6.1. Realizace hardware

Největším problémem během realizace byla nemožnost sehnat obvod **LP3988IMF** v provedení na 3,0 V, a to ani jako vzorek od výrobce. I přes to, že je v dokumentaci toto provedení uváděno, a to včetně objednáčích čísla. Nakonec byl tedy použit obvod ze stejné řady, jen s o stupeň nižším napětím. Vyžádalo si to změny několika součástek dle tabulky 6.1.

Tabulka 6.1: Hodnoty součástek změněné z důvodu výměny zdroje

| Počet | Hodnota | Pouzdro | Popis | Pozice |
|-------|----------------|---------|---|--------|
| 1 | LP3988IMF-2,85 | SOT23-5 | Micropower, 150-mA Ultra-Low-Dropout CMOS Voltage Regulator | IC4 |
| 2 | 36k | 0603 | Resistor, 36k | R22 |

Změna napájecího napětí a R22 byla ověřena výpočtem a výsledné hodnoty jsou uvedeny v tabulce 6.2. Napětí byla uvedena pro vstup mikrokontroléru. Jak je vidět, po změně je překročeno napětí na vstupu o 60mV. Dojde k tomu na horní hranici napájecího napětí sběrnice SENT a rezerva vstupu je zde 300mV.

Tabulka 6.2: Porovnání parametrů rozhraní SENT po výměně zdroje

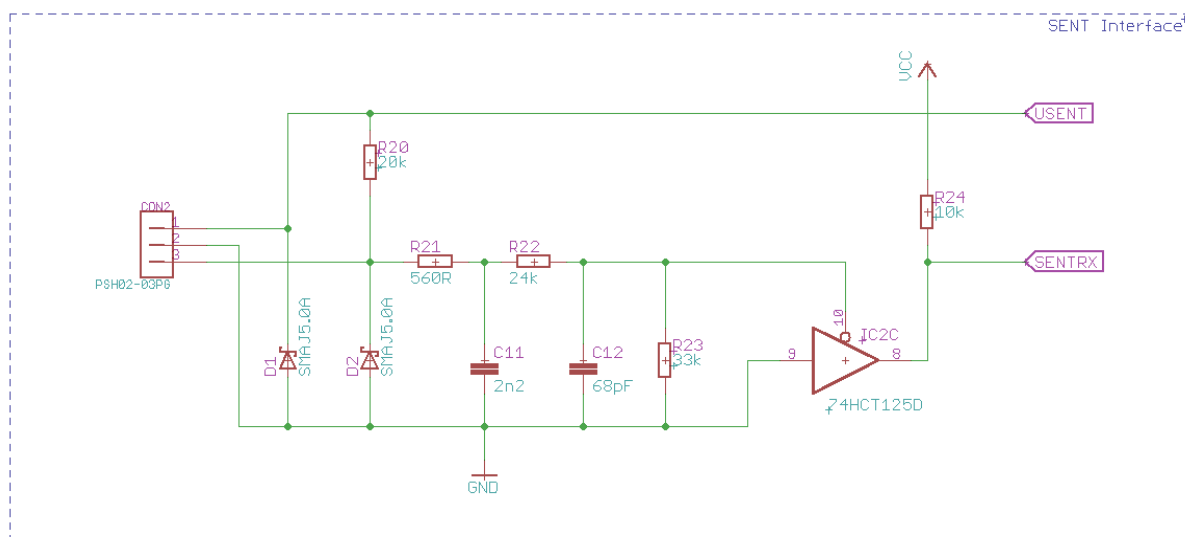
| Veličina | Limit | | 2,85V | | | Poznámka |
|------------|-------|------|-------|------|------|---------------|
| | Min | Max | Min | Typ | Max | |
| U_{MCU0} | 1,24 | 3,0 | 1,89 | | 2,3 | V |
| U_{MCUL} | 0 | 0,75 | 0 | | 0,83 | V |
| U_{MCUH} | 2,25 | 3,0 | 1,89 | | 3,06 | V |
| τ_1 | 0,74 | 1,73 | | 1,23 | | μs |
| τ_2 | 0,6 | 1,4 | | 1,22 | | μs |

Ostatní parametry, kterých se změna dotkne, jsou maximální frekvence mikrokontroléru, kompatibilita úrovně s TTL a napájení indikačních LED. Žádná z těchto věcí není nijak negativně ovlivněna.

V další verzi hardware je vhodné odstranit závislost vstupního obvodu sběrnice SENT použitím volného třístavového budiče IC2C v zapojení simulujícím otevřený kolektor. Byla

by tak odstraněna závislost přizpůsobovacích obvodů sběrnice SENT na napájecím napětí mikrokontroléru.

Snížené napětí se ukázalo jako výhoda při přepínání napájecích zdrojů, kde 3,0 V je akceptovatelné maximum. Při napětí pod 2,8 V už není zaručen běh mikrokontroléru na požadované frekvenci 12 MHz. Bylo by proto výhodné použít regulovatelný stabilizátor a jeho výstup nastavit kolem 2,9 V.



Obrázek 6.1: Modifikované zapojení fyzické vrstvy SENT

Tabulka 6.3: Porovnání parametrů rozhraní SENT po výměně zdroje

| Veličina | Limit | | IC2-10 | | | | Poznámka |
|------------|-------|------|--------|------|------|---------------|----------|
| | Min | Max | Min | Typ | Max | | |
| U_{MCU0} | 2,00 | 5,15 | 2,06 | | 2,19 | V | |
| U_{MCUL} | 0,00 | 0,80 | 0,00 | | 0,80 | V | |
| U_{MCUH} | 2,00 | 5,15 | 2,18 | | 2,95 | V | |
| τ_1 | 0,74 | 1,73 | | 1,23 | | μs | |
| τ_2 | 0,60 | 1,4 | | 0,94 | | μs | |

6.2. Realizace firmware

Nejhorší, s čím jsem se během práce na projektu setkal, je zoufalá podpora pro tvorbu firmware tohoto mikrokontroléru ze strany jeho výrobce. Vše se omezilo na poskytnutí kompilátoru a hlavičkového souboru s opisem hardware. A i u tohoto hlavičkového souboru je evidentní, že jeho autor má hodně zkušeností s návrhem digitálních obvodů, ale nulovou praxi v psaní software.

Zatím co u ostatních platform je standardem hierarchický popis, kde je periferie na úrovni struktury, popisující vnitřní registry, a jednotlivé instance stejné periferie jsou odkazovány pomocí ukazatele, zde je pro každý registr každé periferie definováno jedno makro preprocesoru bez ohledu na to, že patří logicky ke stejné periférii. Zatím co na jiné platformě

by stačilo napsat „TA0->“ a editor dá automaticky k výběru řídicí registry periferie, zde to znamenalo překliknutí do zdrojového souboru s makry a hledání. V tomto firma *Texas Instruments* kulhá za konkurenčními firmami jako *ST Microelectronics*, *Renesas*, *Atmel* a dalšími.

Ani popis bitů k registrům není moc intuitivní. Pro každý bit je definováno zvlášť makro, naštěstí již sdílené napříč periferiemi. Pozoruhodný je i způsob, jak jsou řešeny vícebitové hodnoty. V tom případě jsou hodnoty ne jako výčet (*enum*), který by se objevil v našeptávání, ale sada maker, od bitů v registru odlišený jen podtržítkem. Samozřejmostí je i to, že název bitu je tří- až pětípísmenný kód z čísel a velkých písmen, jen velmi málo říkající o tom, jaký má význam. Kdo by ostatně potřeboval při čtení programu vědět bez manuálu, co přesně znamená řádek jako „`CACTL1 = CAEX | CAREF_2 | CAON;`“ ?

Velmi zajímavý je i linker skript, který popisuje organizaci paměti a jek má být výsledný firmware slinkován. Vždy se jednalo o prostý soubor se jménem odpovídajícím typu mikrokontroléru a příponou *.ld*. Zde ne, zde je soubor *msp430.x*, který do sebe zahrne některý z adresářů se jménem, odpovídajícím typu mikrokontroléru. V tomto adresáři jsou dva soubory – *memory.x* a *peripheral.x*. Jsou voleny z příkazového řádku parametrem `-mmcuc=msp430g2553` a použity vždy, když je v projektu tento typ mikrokontroléru. Původní představa vlastní sekce 256 B pro vyrovnávací paměti, aby posledních sedm bitů adresy na konci bylo sedm jedniček, tak raději nebyla realizována.

O vyšších vrstvách, jako je CMSIS u procesorů s jádrem ARM, si zde může člověk nechat jen zdát a dávat necelých 500 USD za jejich IDE s nadějí ve získání kusu použitého kódu, je s takovým základem spíše projevem zoufalství.

Co se vlastní aplikace týká, zde bylo před štábní kulturou, abstrakcí a přehledností dostala přednost rychlost. Vše bylo podřízeno tomu, aby bylo co nejrychleji zpracováno přerušeni od časovače i od rozhraní SPI. U dekódování rozhraní SENT je nutné co nejrychlejší zpracování, aby nedošlo ke ztrátě dat. Rychlosti je podřízeno i zpracování dat ze sériových zpráv, opět aby nepřetekl vyrovnávací registr. Jediné, co nebylo časově kritické, byla poslední ze tří úloh, aplikace změn a kontrola napájení v power manageru. Na tuto úlohu ale zbyly i ostatní činnosti, proto zde opět nebylo moudré příliš plýtvat výkonem.

6.3. Celkové hodnocení projektu

Projekt byl bezproblémový po dobu příprav, návrhu schématu a desky a bez potíží byla i její výroba. První potíže nastaly až se sháněním součástek. Ukázalo se, že zvolený stabilizátor není možné sehnat s požadovaným výstupním napětím, a to ani u výrobce, ač se dle dokumentace nabízí. Poměrně dlouho, cca dva týdny, trval nákup ostatních součástek, paradoxně největším problematické je shánění rezistorů a kondenzátorů velikosti 0603.

U architektury a tvorby firmware bylo nutné použít jiný postup, než na který jsem normálně zvyklý. Operační systém je zredukovaný na jeden příkaz *switch*, defenzivní postupy a kontroly běhu byly vypuštěny z důvodu rychlosti. Utrpěla i přehlednost a abstrakce, ale nakonec se podařilo najít způsob, jak vytvořit použitelný firmware.

Co se nepodařilo, bylo testování. Vzhledem k tomu, že jde o převodník dvou rozhraní a nebylo k dispozici nic co by generovalo signál sběrnice SENT, nepodařilo se firmware plně odladit ani otestovat, nicméně to ani nebylo v zadání a v praxi musí vždy testovat někdo jiný, než autor řešení. Firmware, který je součástí práce, je tak nutné dodatečně odladit a otestovat.

Zvolený mikrokontrolér **MSP430G2553IPWR20** je použitelný, i když z pohledu hardware je těsně na hraně a zůstává zde riziko, že s nejrychlejším časováním nastanou problémy (nebylo ověřeno). Jako vhodnější mikrokontrolér se jeví například **STM32F030F4P6** [15]. Srovnání těchto dvou mikrokontrolérů je v tabulce 6.4, ceny pochází z e-shopu Farnell..

Tabulka 6.4: Srovnání použitého mikrokontroléru s rodinou STM32

| Parametr | MSP430G2553 | STM32F030F4P6 |
|--------------------|--|--|
| Jádro | MSP430 | Cortex-M0 |
| Frekvence | 16 MHz | 48MHz |
| Architektura | Von Neumann, RISC | Von Neumann, RISC |
| Šířka sběrnice | 16b | 32b |
| Velikost FLASH | 16k B | 16kB |
| Velikost RAM | 512 B | 4 kB |
| Čítače (32b) | 0 | 0 |
| Čítače (16b) | 2x, 3 moduly | 5x, 11 kanálů |
| A/D převodník | 8 kanálů, 10 bitů | 12b, 11 vstupů |
| Analog. komparátor | 8 kanálů | Ne |
| Rozhraní SPI | 2x | 1x |
| DMA | Ne | 5 kanálů |
| Vstupu/výstupy | 16 | 15 |
| Watchdog | 1x | 2x |
| Napájení | 1,8 až 3,6V, max. frekvence roste od 6MHz do 16MHz dle napětí. | 2,4 až 3,6V pro celý rozsah frekvencí bez omezen |
| Pouzdro | TSSOP 20 | TSSOP 20 |
| SW podpora | GCC + Minimum | GCC + STM32 Cube |
| Cena 1ks / 100 ks | 59,08 / 39,20 Kč | 38,65 / 22,36 Kč |

Získané zkušenosti s řadou **MSP430G2xxx** jsou jednoznačné a nejen z pohledu ceny, ale i výbavy, velikosti paměti, vybavení a tvorby firmware si v této chvíli neumím představit aplikaci, kde by tento typ mikrokontroléru představoval významnou výhodu proti jiným řešením.

Změna mikrokontroléru v tomto převodníku by pravděpodobně vedla ke zlepšení jeho parametrů, ale za cenu změny zapojení desky a rozsáhlejších změn firmware. Vyhoví-li proto při testech, jedinou požadovanou úpravou bude regulovatelný stabilizátor s výstupem 2,9 V a vstup připojený s pomocí IC2C.

Seznam použité literatury

- [1] Mikulášek, P., Převodník sběrnice SENT na SPI., Semestrální práce, FEKT VUT Brno 2016, 42s.
- [2] Gook, M.: Hardwarová rozhraní. Průvodce programátora, Computer Press 2006, 463 s., ISBN 80-251-1019-2
- [3] Single Edge Nibble Transmission for Automotive Applications, Technická norma, SAE-J2716:2010, SAE International 2010
- [4] Datasheet MSP430G2553, Texas Instruments, <http://www.ti.com/lit/ds/symlink/msp430g2553.pdf>
- [5] Datasheet TXS0104E, Texas Instruments, <http://www.ti.com/lit/ds/symlink/txs0104e.pdf>
- [6] Datasheet 74HC125, 74HCT125, NXP Semiconductors, http://cache.nxp.com/documents/data_sheet/74HC_HCT125.pdf?pspll=1
- [7] Logic Selection Guide, Online příručka, Texas Instruments 2014, <http://www.ti.com/lit/sg/sdyu001aa/sdyu001aa.pdf>
- [8] Datasheet TPS2294x, Texas Instruments, <http://www.ti.com/lit/ds/symlink/tps22943.pdf>
- [9] Datasheet TPS79730, Texas Instruments, <http://www.ti.com/lit/ds/symlink/tps797.pdf>
- [10] Datasheet BC807, NXP Semiconductors, http://www.nxp.com/documents/data_sheet/BC807_BC807W_BC327.pdf
- [11] Kernigham, B.W.; Ritchie, D.M.: Programovací jazyk C - ANSI C99, Computer Press 2008, 286 s., ISBN 80-251-0897-X
- [12] MSP430x2xx Family User's Guide, Rev. J, Texas Instruments, 643s., <http://www.ti.com/lit/ug/slau144j/slau144j.pdf>
- [13] Underwood, S., mspgcc A port of the GNU tools to the Texas Instruments MSP430 microcontrollers, Online kniha, 2003, <http://www.eecs.harvard.edu/~konrad/projects/motetrack/mspgcc-manual-20031127.pdf>
- [14] Prokop, J.: Algoritmy v jazyku C a C++, 3. vydání, Grada Publishing, a.s. 2015, 200 s., ISBN 978-80-247-5467-3
- [15] Datasheet STM32F030F4P6, ST Microelectronics, <http://www.st.com/content/ccc/resource/technical/document/datasheet/a4/5d/0b/0e/87/c4/4d/71/DM00088500.pdf/files/DM00088500.pdf/jcr:content/translations/en.DM00088500.pdf>

Příloha 1. Obsah přiloženého CD-ROM

V této příloze jsou uvedeny jednotlivé adresáře relativně k adresáři, do kterého je CD-ROM přimontován (nebo ve Windows k jednotce).

| | |
|--------------------|---|
| <i>./bp</i> | Bakalářská práce ve formátu .odt a .pdf |
| <i>./bp/obr</i> | Obrázky, použité v bakalářské práci |
| <i>./bp/obj</i> | Objekty, použité v bakalářské práci (kresby apod.) |
| <i>./calc</i> | Výpočty v programu LibreOffice Calc (formát .ods) |
| <i>./eagle-lib</i> | Knihovna součástek pro program Eagle |
| <i>./foto</i> | Fotografie prototypu a desek plošných spojů |
| <i>./fw</i> | Zdrojové soubory firmware |
| <i>./fw/bin</i> | Binární soubory s firmware, objektové soubory atd. |
| <i>./hw</i> | Schéma zapojení a deska v programu Eagle, Seznam součástek ve formátu .csv |
| <i>./hw/gerber</i> | Gerber data pro výrobu desek. Formát, používaný firmou Gatema Boskovice Náhled v programu Gerbv + exportované obrázky vrstev |
| <i>./scan</i> | Naskenované podepsané zadání bakalářské práce |
| <i>./sp</i> | Semestrální práce ve formátu .odt a .pdf |
| <i>./sp/obr</i> | Obrázky, použité v semestrální práci |
| <i>./sp/obj</i> | Objekty, použité v semestrální práci |

Příloha 2. Použitý software

| | |
|-------------------------|---|
| Software | Textový editor, tabulkový procesor, kreslení grafů a blokových schémat |
| Označení a verze | Libre Office, verze 5.0.6.2 |
| Zdroj software | https://cs.libreoffice.org/download/ |
| Instalace | Fedora 23: Předinstalováno, nebo příkaz <code>sudo dnf install libreoffice</code> MS Windows: Standardní instalátor |
| Použité doplňky | Český slovník pro kontrolu pravopisu MacroFormatterADP pro automatické zvýraznění syntaxe |
| Nastavení | Výchozí. Doplněn jen automatický text pro vložení číslovaného vzorce. |
| Poznámky | Zdarma, formát souborů dle ISO/IEC 26300:2006, nadstandardní práce se vzorečky, propracovaný export do formátu PDF, funguje i bez přístupu k Internetu. |

| | |
|-------------------------|---|
| Software | Editor schémat a desek plošných spojů |
| Označení a verze | Eagle Lite, verze 7.2. pro MS Windows |
| Zdroj software | https://www.cadsoft.de/download-eagle/ |
| Instalace | Fedora 23: <code>sudo dnf install wine</code> , poté spuštění binárního instalátoru pro MS Windows. |
| Použité doplňky | CAM procesor z http://pcb.gatema.cz/files/ckeditor/ke_stazeni/ostatn%C3%AD/cam.rar |
| Nastavení | Zkopírování adresáře Eagle z příloženého CD-ROM a nastavení cest k projektu a knihovně. Rozbalení staženého CAM procesoru do adresáře a nastavení cesty k tomuto adresáři. |
| Poznámky | Linuxová verze má závislosti na nekompatibilní, zastaralé knihovny, proto je lepší použít verzi pro MS Windows společně s Wine. |

| | |
|-------------------------|---|
| Software | Prohlížeč Gerber dat |
| Označení a verze | Gerbv 2.6.0 |
| Zdroj software | Repozitář Fedora |
| Instalace | Fedora23: <code>sudo dnf install gerbv</code> |
| Použité doplňky | |
| Nastavení | Výchozí |
| Poznámky | Verze pro MS Windows neexistuje, použijte jiný prohlížeč. |

Software Simulace zdroje

Označení a verze OrCad Capture CIS Lite 16.6

Zdroj software

Instalace

Použité doplňky

Nastavení

Poznámky Ve škole, nemám Windows

Software Vývojové prostředí pro mikrokontrolér

Označení a verze Code::Blocks 13.12 rev. 9501

Zdroj software Repozitář Fedora23

Instalace Fedora23: *sudo dnf install codeblocks*

Použité doplňky

Nastavení

Poznámky

Software Kompilátor pro MSP430

Označení a verze MSP430-GCC 4.0.1.0

Zdroj software http://software-dl.ti.com/msp430/msp430_public_sw/mcu/msp430/MSPGCC/latest/index_FDS.html

Instalace Fedora 23: spuštění staženého souboru

Použité doplňky

Nastavení

Poznámky

Příloha 3. Použitý hardware

Přístroj Laboratorní zdroj 0-30V, 0-1A

Typ BS 554

Výrobce Tesla

Sériové číslo 013394

Příslušenství

Kalibrace do (orientační měřidlo)

Poznámky

Přístroj Multimetr

Typ VA 16

Výrobce V&A Instruments

Sériové číslo VA140501402

Příslušenství

Kalibrace do (orientační měřidlo)

Poznámky

Přístroj Programátor / emulátor

Typ MSP430 LaunchPad

Výrobce Texas Instruments

Sériové číslo

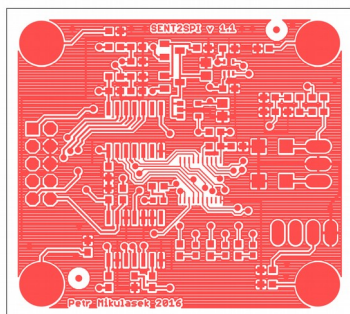
Příslušenství

Kalibrace do

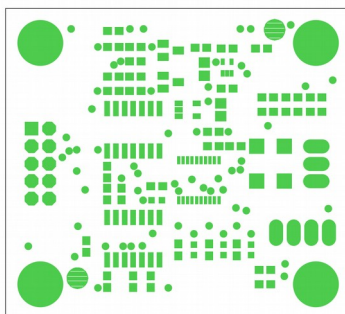
Poznámky Vývojový kit s vyvedenou sběrnici SWD pro ladění.

Příloha 4. Gerber data

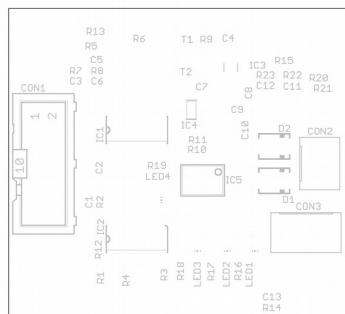
Je zobrazen pohled na gerber data z programu Gerbv. Samotná gerber data jsou na příloženém CD. Pojmenování souborů a jejich obsah odpovídá standardu firmy Gatema Boskovice, popsáném na webu <http://pcb.gatema.cz/vseobecne-pozadavky-na-data/>.



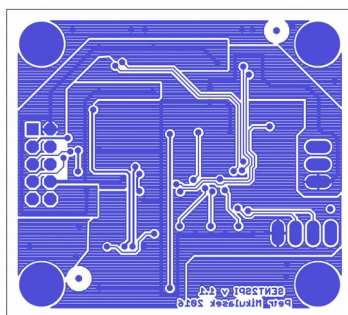
Měď str. součástek



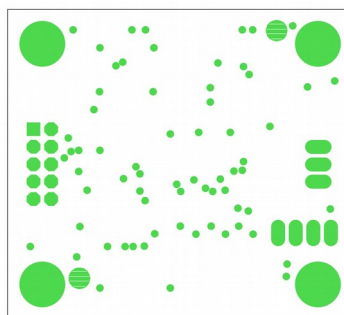
Maska str. součástek



Servisní potisk



Měď strana spojů



Maska strana spojů