VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

USB MĚŘICÍ MODUL PRO VIRTUÁLNÍ INSTRUMENTACI

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE Bc. IVAN HANUŠ
AUTHOR

BRNO 2015

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

USB MĚŘICÍ MODUL PRO VIRTUÁLNÍ INSTRUMENTACI

USB MEASUREMENT MODULE FOR VIRTUAL INSTRUMENTATION

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE Bc. IVAN HANUŠ
AUTHOR

VEDOUCÍ PRÁCE Ing. ZDENĚK HAVRÁNEK, Ph.D.
SUPERVISOR

BRNO 2015

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav automatizace a měřicí techniky

Diplomová práce

magisterský navazující studijní obor
Kybernetika, automatizace a měření

Student:	Bc. Ivan Hanuš	***ID:***	136517
Ročník:	2	***Akademický rok:***	2014/2015

NÁZEV TÉMATU:

USB měřicí modul pro virtuální instrumentaci

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je navrhnout a realizovat USB měřicí modul se softwarovými ovladači pro vývojové prostředí LabVIEW.

1) Vypracujte literární průzkum na téma univerzální hardware pro generování a sběr analogových a digitálních signálů využívající virtuální instrumentaci a průzkum trhu tohoto druhu měřicího zařízení.
2) Definujte obvyklé vlastnosti a rozhraní komerčně dostupných univerzálních měřicích modulů – multifunkčních měřicích karet, zaměřte se na způsoby přístupu k modulům z prostředí LabVIEW.
3) Navrhněte a realizujte měřicí jednotku s mikrokontrolerem připojitelnou pomocí USB sběrnice k počítači a umožňující generování, sběr, zpracování a vizualizaci analogových a digitálních signálů s možností rozšíření o další obvody pomocí univerzální sériové sběrnice. Realizujte galvanické oddělení jednotlivých rozhraní a implementujte možnost volby napěťových úrovní na digitálních rozhraních.
4) Vytvořte softwarový ovladač pro realizovaný modul v jazyce C, který bude využitelný k ovládání měřicího modulu z prostředí LabVIEW, a prezentujte jeho použití pomocí jednoduché aplikace.
5) Určete metrologické parametry vstupních kanálů a ověřte funkčnost digitálních rozhraní s volitelnými napěťovými úrovněmi. Sestavte specifikační list vytvořeného USB modulu.

DOPORUČENÁ LITERATURA:

[1] Vlach, J. a kol. Začínáme s LabVIEW. 248 stran. 2008. ISBN 978-80-7300-2.
[2] MSP430. Datasheets and application notes. Texas Instruments.

Termín zadání:	9.2.2015	***Termín odevzdání:***	18.5.2015

Vedoucí práce: Ing. Zdeněk Havránek, Ph.D.
Konzultanti diplomové práce:

doc. Ing. Václav Jirsík, CSc.
Předseda oborové rady

ABSTRAKT

Cieľom tejto diplomovej práce je návrh univerzálneho USB meracieho modulu pre virtuálnu inštrumentáciu. Modul je pripojený k PC pomocou rozhrania USB. Modul obsahuje analógové vstupy a analógové výstupy. Vstupy majú niekoľko prepínateľných rozsahov, nastaviteľné hornopriepustné a doľnopriepustné filtre. Ďalej obsahuje digitálnu vstupno-výstupnú jednotku s 8 kanálmi s konfigurovateľným smerom a napäťovou úrovňou. Súčasťou tejto jednotky je fyzická vrstva pre digitálne sériové rozhranie SPI/UART a I2C. Modul je schopný komunikovať s prostredím LabVIEW.
Teoretická časť obsahuje prehľad komerčných USB modulov pre zber a generovanie dát od rôznych výrobcov s ich základnými parametrami. Vybrané moduly sú rozpísané podrobnejšie. Ďalej rozoberá možnosti prepojenia USB modulu s prostredím LabVIEW.
Praktická časť sa zaoberá samotným návrhom vstupno-výstupných obvodov a návrhom celého meracieho modulu. Ďalej je v nej popísaný komunikačný protokol v MCU, obslužná knižnica k modulu a demonštračný program. Na záver sú výsledky meraní overujúcich a zisťujúcich parametre modulu.
Výsledkom tejto diplomovej práce je zrealizovaný univerzálny USB merací modul s obslužnou knižnicou a programom v LabVIEW demonštrujúcim jej použitie.

KĽÚČOVÉ SLOVÁ

USB, DAQ, MSP430, Analógové vstupy, Analógové výstupy, LabVIEW

ABSTRACT

The objective of this thesis is design of the universal USB measuring module for virtual instrumentation. The module is connected to PC via USB interface. The module have analog inputs and analog outputs. The inputs have some switchable ranges, adjustable high-pass and low-pass fiters. Also the module have the digital input-output unit with 8 channels with configurable data direction and voltage level. Physical layer for digital digital serial interfaces SPI/UART and I2C is also part of this unit. The module is able to communicate with LabVIEW enviroment.
The teoretical part contains survey of the commercial USB modules for data acquisition and generation from various manufacturers with its basic parameters. Some selected modules are described in detail. Furthermore the thesis deals with possibilities of connecting the USB module with the LabVIEW enviroment.
The practical part deals with the design of input-output circuits and with design of the module itself. Furthemore, tho communication protocol, utility library and demonstration LabVIEW software are described in this part. At the end of this part, there are results of verification measurements.
The result of this thesis is completed universal USB measuring module with utility library and demonstration program.

KEYWORDS

USB, DAQ, MSP430, Analog inputs, Analog outputs, LabVIEW

HANUŠ, Ivan *USB měřicí modul pro virtuální instrumentaci*: diplomová práca. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2015. 104 s. Vedúci práce bol Ing. Zdeněk Havránek, Ph.D.

PREHLÁSENIE

Prehlasujem, že som svoju diplomovoú prácu na tému „USB měřicí modul pro virtuální instrumentaci“ vypracoval samostatne pod vedením vedúceho diplomovej práce, využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej diplomovej práce ďalej prehlasujem, že v súvislosti s vytvorením tejto diplomovej práce som neporušil autorské práva tretích osôb, najmä som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a/alebo majetkových a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., o autorskom práve , o právach súvisiacich s autorským právom a o zmene niektorých zákonov (autorský zákon), v znení neskorších predpisov, vrátane možných trestnoprávnych dôsledkov vyplývajúcich z ustanovenia druhej časti, hlavy VI. diel 4 Trestného zákoníka č. 40/2009 Sb.

Brno ....

(podpis autora)

POĎAKOVANIE

Ďakujem vedúcemu diplomovej práce Ing. Zdeňkovi Havránkovi, Ph.D. za účinnú metodickú, pedagogickú a odbornú pomoc a ďalšie cenné rady pri spracovaní mojej diplomovej práce.

Brno ....

(podpis autora)

OBSAH

Úvod **13**

1 Komerčné moduly **14**
1.1 National Instruments MyDAQ 14
1.2 National Instruments USB-6008/9 15
1.3 DAQ moduly od ďalších výrobcov 17

2 Požiadavky na modul **18**

3 Prepojenie modulu s prostredím LabVIEW **19**
3.1 Čo je to LabVIEW 19
3.2 Možnosti prepojenia USB modulu s prostredím LabVIEW 21
3.2.1 Virtual Instrument Software Architecture 21
3.2.2 Dynamická knižnica 21

4 Návrh obvodu **23**
4.1 Mikrokontrolér 24
4.2 Digitálne vstupy/výstupy 25
4.2.1 Oddelenie napájania a signálov 26
4.2.2 Nastavenie napäťovej úrovne 27
4.2.3 Poznámky k návrhu 28
4.3 USB 29
4.4 Analógové vstupy 29
4.4.1 Vstupný zosilňovač 29
4.4.2 Hornopriepustný filter 31
4.4.3 Dolnopriepustný filter 33
4.4.4 Napäťová referencia 34
4.4.5 Automatické kalibrovanie/kompenzovanie 35
4.5 Analógové výstupy 35
4.6 Napájanie 36
4.7 Možnosť pripojenia k nadradenému systému rozhraním iným než USB 37
4.8 Zmeny oproti pôvodnému návrhu 37

5 Činnosť modulu a komunikačný protokol **40**
5.1 Firmware modulu 40
5.1.1 Inicializácia 41
5.1.2 USB komunikácia 41
5.1.3 Štruktúra projektu 42

5.2 Nastavenia a módy činnosti modulu 43
5.2.1 Digitálne vstupy a výstupy 43
5.2.2 Analógové výstupy 43
5.2.3 Analógové vstupy - Analógová časť 44
5.2.4 Analógové vstupy - ADC 45
5.2.5 Predvolené nastavenia 45
5.3 Komunikačný protokol 46
5.3.1 Systémové správy 46
5.3.2 Digitálne vstupy a výstupy 47
5.3.3 Analógové vstupy 48
5.3.4 Analógové výstupy 52
5.4 Prvé pripojenie modulu k PC 53

6 Obslužná knižnica a demonštračný software 54
6.1 Trieda `module` 54
6.1.1 Verejné parametre 54
6.1.2 Verejné metódy 54
6.1.3 Odporúčania pri použití knižnice 61
6.2 Ukážkový program v LabVIEW 61
6.2.1 Hlavný program 61
6.2.2 Podprogramy 64

7 Overenie parametrov a vlastností zrealizovaného modulu 65
7.1 Digitálne I/O 65
7.2 Analógové výstupy 65
7.2.1 Statické charakteristika 66
7.2.2 Dynamické vlastnosti 66
7.3 Analógové vstupy 73
7.3.1 Statické charakteristiky 73
7.3.2 Dynamické charakteristiky 73
7.3.3 Dátová priepustnosť 75

8 Záver 79

Literatúra 81

Zoznam symbolov, veličín a skratiek 83

Zoznam príloh 84

A Špecifikácia modulu 85

B Obsah priloženého CD **88**

C Schémy zapojenia **89**

D Doska plošných spojov **95**

E Zoznam súčiastok **100**

ZOZNAM OBRÁZKOV

1.1 NI MyDAQ 14
1.2 NI USB-6009 16
3.1 Ukážka programu v LabVIEW 20
3.2 Funkcie na prácu s triedami z jazyka C# 22
4.1 Bloková schéma celého modulu 23
4.2 Bloková schéma bloku digitálnych vstupov a výstupov 25
4.3 Bloková schéma analógových vstupov 29
4.4 Teoretická schéma vstupného zosilňovača 30
4.5 Teoretická schéma hornopriepustného filtra 2. rádu 31
4.6 Teoretická frekvenčná charakteristika filtra horná prepusť s medznou frekvenciou 1Hz 32
4.7 Teoretická frekvenčná charakteristika filtra horná prepusť s medznou frekvenciou 100Hz 33
4.8 Bloková schéma napájacích obvodov 36
4.9 Deformovanie priebehu pri rozsahu ±5V 38
5.1 Zjednodušená hlavná slučka programu 40
6.1 Okno ukážkového programu 62
7.1 Závislosť napäťovej úrovne digitálnych I/O od kódového slova digitálneho potenciometra 65
7.2 Závislosť výstupného napätia analógových výstupov od kódového slova DAC 66
7.3 Analógový výstup: sínus, 1000 vzoriek, 1.818MS/s 67
7.4 Analógový výstup: sínus, 100 vzoriek, 1.818MS/s 68
7.5 Analógový výstup: sínus, 25 vzoriek, 1.818MS/s 68
7.6 Analógový výstup: trojuholník, 1000 vzoriek, 1.818MS/s 69
7.7 Analógový výstup: trojuholník, 100 vzoriek, 1.818MS/s 69
7.8 Analógový výstup: trojuholník, 25 vzoriek, 1.818MS/s 70
7.9 Analógový výstup: obdĺžnik, 1000 vzoriek, 198kS/s 70
7.10 Analógový výstup: obdĺžnik, 1000 vzoriek, 1.818MS/s 71
7.11 Analógový výstup: obdĺžnik, 100 vzoriek, 1.818MS/s 71
7.12 Analógový výstup: obdĺžnik, 25 vzoriek, 1.818MS/s 72
7.13 Prevodná charakteristika rozsahu ±1V 73
7.14 Relatívna chyba rozsahu ±1V 74
7.15 Prevodná charakteristika rozsahu ±0.2V 75
7.16 Frekvenčná charakteristika vstupného zosilňovača 76
7.17 Frekvenčné charakteristiky hornopriepustných filtrov 77

7.18 Frekvenčná charakteristika dolnopriepustného filtra v závislosti na veľkosti ladiaceho odporu 77
7.19 Frekvenčná charakteristika dolnopriepustného filtra v závislosti na zvolenom deliči frekvencie 78
7.20 Spektrum sínusového a obdĺžnikového signálu 78

ZOZNAM TABULIEK

1.1 Prehľad DAQ modulov od ostatných výrobcov 17
4.1 Nastavenie externého rezistora R_{EXT} a deliča *div* pre jednotlivé vzorkovacie frekvencie 34
5.1 Formát správy 46
5.2 Formát správy GetTime a SetTime 46
5.3 Formát správy GetSetControls a SetControls 47
5.4 Formát správy GetIOVoltage a SetIOVoltage 47
5.5 Formát správy GetIODirection a SetIODirection 47
5.6 Formát správy GetIO a SetIO 48
5.7 Formát správy GetAnInConfig a SetAnInConfig 48
5.8 Formát správy GetCalibration a SetCalibration 49
5.9 Formát správy GetCoupling a SetCoupling 49
5.10 Formát správy GetRange a SetRange 49
5.11 Formát správy GetHPF a SetHPF 50
5.12 Formát správy GetLPF a SetLPF 50
5.13 Formát správy GetADCConfig a SetADCConfig 51
5.14 Formát správy ADCSingleData 51
5.15 Formát správy ADCData 51
5.16 Formát správy DACWriteSingle a DACRead 52
5.17 Formát správy SetDACConfig a GetDACConfig 52
5.18 Formát správy DACWriteData 53

ÚVOD

V súčastnosti sa pri návrhu a hlavne pri oživovaní analógovo - číslicových obvodov stále častejšie stretávame s potrebou externého snímania výstupov a ovládania vstupov (či už analógových alebo digitálnych) oživovaného obvodu, čo môže byť, hlavne v neprofesionálnych podmienkach bez drahého profesionálneho vybavenia, značne problematické. Toto nás viedlo k rozhodnutiu navrhnúť si vlastný vstupno-výstupný modul, ktorý by bol schopný aspoň čiastočne nahradiť drahé profesionálne moduly.

Našim cieľom je navrhnúť modul obsahujúci viacero vstupných a výstupných analógových aj číslicových obvodov. Pripojiteľný by mal byť k PC pomocou rozhrania USB.

Mal by obsahovať aspoň 2 meracie kanály s voľbou aspoň 2 meracích rozsahov a so vzorkovaním 40kHz. Ďalej by boli celkom užitočné aj analógové výstupy, realizované napr. pomocou D/A prevodníka (ďalej len DAC[1]). Modul by mal ešte obsahovať aj digitálne vstupy a výstupy. Pri nich bude veľmi užitočná voľba napäťovej úrovne. Vstupy aj výstupy by mali od riadiaceho mikrokontroléra (ďalej len MCU[2]) oddelené, jednak z dôvodu ochrany MCU, ale aj kvôli možnosti pripojenia modulu k systému s iným vzťažným potenciálom (s inou „zemou“) oproti riadiacemu PC. A nemôžeme zabudnúť aj na sériové komunikačné rozhrania. Modul by mal obsahovať fyzickú vrstvu pre aspoň jedno sériové rozhranie I2C[3]/SPI[4], taktiež oddelené od MCU.

Ďalej by bolo dobré zvážiť možnosť ovládania modulu aj pomocou iného komunikačného rozhrania ako je USB[5], napr. Bluetooth alebo Ethernet.

Taktiež by bolo vhodné, aby samotné komunikačné rozhranie USB bolo tiež oddelené, opäť z dôvodu ochrany PC a možnosti práce na inom potenciáli, než je zem PC, ku ktorému bude modul pripojený.

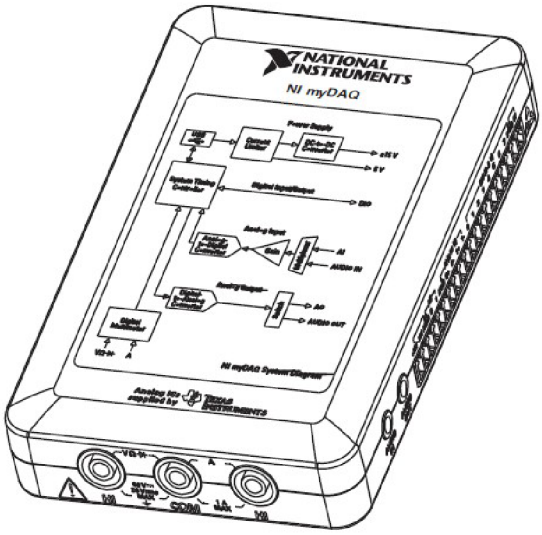
[1]Digital-to-Analog Converter – Číslicovo analógový prevodník
[2]Microcontroller unit – Mikrokontrolér
[3]Inter-Integrated Circuit – multi-masterová sériová zbernica
[4]Serial Peripheral Interface – sériové periférne rozhranie
[5]Universal Serial Bus – univerzálna sériová zbernica

1 KOMERČNÉ MODULY

V tejto kapitole je prehľad parametrov komerčných USB DAQ[1] modulov. Ako prvý som vybral do prehľadu modul MyDAQ od spoločnosti National Instruments. S týmto modulom som sa viackrát stretol vo výuke, bol vlastne prvotnou inšpiráciou pri výbere tejto témy a ktorý mi bol veľmi užitočným nástrojom pri oživovaní modulu navrhovaného v tejto práci. Ako druhý som vybral modul od National Instruments USB-6008/9, ktorý (narozdiel od prevažne výukového modulu MyDAQ) je už určený na praktické aplikácie a parametrami je podobný modulu navrhovanému v tejto práci. Ostatné moduly boli vybrané s ohľadom na parametre tohto modulu a zároveň som sa snažil pokryť čo najväčšie spektrum výrobcov.

1.1 National Instruments MyDAQ

Medzi najrozšírenejšie neprofesionálne USB DAQ moduly patrí výukový modul MyDAQ od spoločnosti National Instruments (ďalej len NI).



Obr. 1.1: NI MyDAQ

Jeho parametre sú [1]:

- Analógové vstupy:
  - 2 diferenciálne / 1 stereo audio
  - rozlíšenie 16bitov
  - maximálne vzorkovanie 200kS/s
  - rozsahy: ± 10V,± 2V, DC väzba (audio vstup ± 2V, AC väzba)

[1] Data Acquisition – získavanie dát

  - frekvenčný rozsah: DC až 400kHz (1,5Hz až 400kHz audio vstup)
- Analógové výstupy
  - 2 alebo 1 stereo audio
  - rozlíšenie 16bitov
  - maximálne vzorkovanie 200kS/s
  - rozsahy: ± 10V,± 2V, DC väzba (audio vstup ± 2V, AC väzba)
  - výstupný prúd: 2mA
- Digitálne I/O
  - 8 liniek s nezávisle konfigurovateľným smerom
  - výstupy 3,3V LVTTL
  - vstupy 5,5V LVTTL kompatibilné
- Digitálny multimeter
  - Meranie napätia
    * AC rozsahy: 200mV, 2V, 20V rms
    * DC rozsahy: 200mV, 2V, 20V, 60V
  - Meranie prúdu
    * AC rozsahy: 20mA, 200mA, 1A rms
    * DC rozsahy: 20mA, 200mA, 1A
  - Meranie odporu
    * Rozsahy: 200Ω, 2kΩ, 20kΩ, 200kΩ, 2MΩ, 20MΩ
  - Meranie diód

Medzi najväčšie výhody a asi hlavný dôvod veľkého rozšírenia v akademickej sfére je výborná prepojiteľnosť s programom LabVIEW (keďže sú od jedného výrobcu) a veľké množstvo rozširujúcich prípravkov na rôzne účely. Ďalšia jeho výhoda je možnosť použiť ho čisto ako laboratórny prístroj (generátor, osciloskop, ...) s pomocou sw NIElvis bez nutnosti tvoriť ovládací program v LV. Pre výukové a laborátorné účely je výborné, že obsahuje vstavaný základný multimeter s možnosťou merania napätia, prúdu, odporu a diód nezávisle od ostatných pripojených obvodov (podľa výrobcu by mal byť galvanicky oddelený od zvyšku modulu). Cena je približne 200€.

1.2 National Instruments USB-6008/9

Ďalší vybraný USB DAQ modul je USB-6008, resp. USB-6009 od spoločnosti National Instruments. Keďže sú od spoločnosti NI, predpokladám taktiež výbornú prepojiteľnosť s prostredím LabVIEW.

Ich parametre sú[2][2]:

- Analógové vstupy:

[2] V prípade rozdielnych parametrov, pre modul USB-6009 platia údaje v zátvorke

Obr. 1.2: NI USB-6009

  - 4 diferenciálne / 8 single-ended
  - diferenciálne 12bitov/ single-ended 11bitov (14bitov/13bitov)
  - maximálne vzorkovanie 10kS/s (48kS/s)
  - rozsahy:
    * Diferenciálne: ± 20V,± 10V, ± 5V, ± 4V, ± 2,5V, ± 2V, ± 1,25V, ± 1V
    * Single-ended: ± 10V
  - buffer: 512bytov
- Analógové výstupy
  - počet: 2
  - rozlíšenie 12bitov
  - maximálne vzorkovanie 150kS/s
  - rozsahy: 0 až +5V
  - výstupný prúd: 5mA
- Digitálne I/O
  - 12 liniek s nezávisle konfigurovateľným smerom
  - kompatibilita: TTL, LVTTL, CMOS
  - typ výstupu:
    * USB-6008: Otvorený kolektor
    * USB-6009: Každý kanál univerzálne programovateľný ako otvorený kolektor alebo aktívny výstup
- Cena približne 185€ (280€)

1.3 DAQ moduly od ďalších výrobcov

V tab. 1.1 je prehľad ďalších USB DAQ modulov, s podobnými parametrami ako vyššie spomínané, od ďalších výrobcov s ich základnými parametrami. Vybrané boli z pomerne širokého portfólia hlavne s ohľadom na parametre modulu navrhovaného v tejto práci (špecifikované ďalej). [3]

Výrobca	Data Translation	Measurement Computing	Keithley	Advantech
Typ	DT9813A	USB-1208FS	KUSB-3102	USB-4711A
AI	16SE	8SE/4D	16SE/8D	16SE/8D
Rozlíšenie [bit]	12	12	12	12
Sample rate [kS/s]	100	50	100	150
Merací rozsah [V]	±10; ±5; ±2,5; ±1,25	±1až±20(D), ±10(SE)	±10;5;2,5;1,25 0-10;5;2,5;1,25	±10; ±5; ±2,5; ±1,25; ±0,625
AO	2	2	2	2
Rozlíšenie [bit]	12	12	12	12
Sample rate [kS/s]	75	5	0.05	statická
Rozsah [V]	±10	0 až 4,096	±5, ±10, 0-5, 0-10	0-5, 0-10, ±5, ±10
Dig. I/O	4/4	16	8/8	8/8
Cena [$]	525	249		

Tab. 1.1: Prehľad DAQ modulov od ostatných výrobcov

[3] SE - signle-ended, D - diferenciálne

2 POŽIADAVKY NA MODUL

Keďže návrh nepočíta s použitím externého prevodníka a zároveň je požiadavka na pomerne vysokú vzorkovaciu frekvenciu, najlepší kompromis je použitie MCU so vstavaným 12-bitovým ADC. Modul bude mať po 2 vstupy a výstupy, aby bol pripadne použiteľný aj na určovanie prevodných a prenosových charakteristík. 8 digitálnych vstupov/výstupov (1 Byte) v kombinácii so štandardnými zbernicami (I2C/SPI/UART[1]) umožňujú pripojenie ďalších jednoduchých rozširujúcich modulov, napr. snímačov alebo iných vstupno-výstupných obvodov.

Ako hlavné komunikačné rozhranie s riadiacim systémom slúži zbernica USB, ktorá slúži zároveň aj na napájanie modulu. Modul by mal ďalej obsahovať fyzickú vrstvu pre možnosť pripojenia komunikačného modulu s rozhraním iným než USB, t.j. prevodníku rozhrania napr. WiFi/Bluetooth/Ethernet na niektorú štandardnú sériovú zbernicu (I2C/SPI/UART).

Konkrétne požiadavky na modul:

- Analógové vstupy:
  - Počet: 2
  - Rozlíšenie: 12 bitov
  - Rozsahy: ±1V / ±200mV
  - Vzorkovanie: 40kHz, 20kHz, 10kHz, 5kHz, 2kHz, 1 kHz
  - Antialliasing filter
  - Hornopriepustný filter - DC, 1Hz, 100Hz
  - Možnosť kalibrácie
- Analógové výstupy:
  - Počet: 2
  - Rozlíšenie: 12 bitov
- Digitálne vstupy/výstupy:
  - Počet: 8
  - Voliteľná napäťová úroveň v rozsahu aspoň 1,8V až 5V
  - Oddelené od MCU
  - Fyzická vrstva pre zbernice SPI/I2C (taktiež oddelené od MCU)
- Zbernica USB:
  - Oddelenie modulu a PC

Rozsahy analógových vstupov boli zvolené tak, aby modul mohol slúžiť nie ako náhrada za modul MyDAQ, ale ako doplnok k nemu pre merania slabších signálov (keďže MyDAQ ma najmenší rozsah ±2V).

[1] UART – Universal Asynchronous Receiver/Transmitter – Univerzálny Asynchrónny Prijímač/Vysielač

3 PREPOJENIE MODULU S PROSTREDÍM LABVIEW

Táto kapitola obsahuje základný popis prostredia LabVIEW a sú v nej rozobraté možnosti pripojenia a ovládania USB modulu v tomto vývojovom prostredí.

3.1 Čo je to LabVIEW

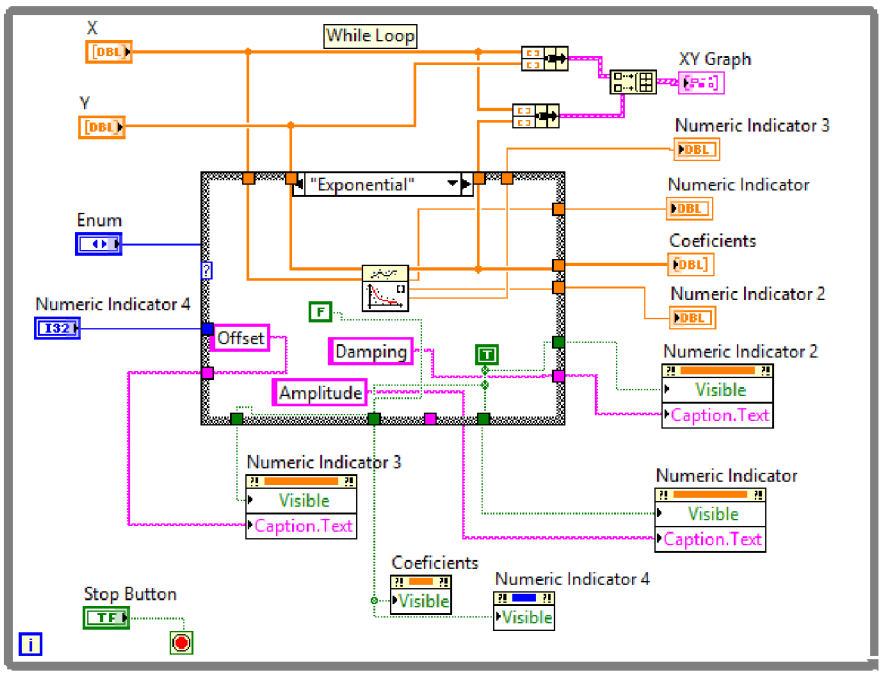
Názov prostredia LabVIEW je skratka zo slov Laboratory Virtual Instrumentation Engineering Workbench, voľne preložené ako Virtuálne laboratórne prístrojové inžinierske pracovisko. Je to vývojové prostredie vyvinuté spoločnosťou National Instruments založené na grafickom programovaní, t.j. algoritmy sú v ňom vytvárané prepájaním jednotlivých funkčných blokov. Vždy sa vykonáva blok, ktorý má práve platné vstupné dáta. Zmyslom tohto grafického programovania je rýchle, jednoduché a efektívne programovanie bez nutnosti znalosti niektorého klasického programovacieho jazyka. LabVIEW však obsahuje možnosť prepojenia na iné programovacie jazyky. [3]

Jeffrey Kodosky mal predstavu, že technik schopný zapísať svoju predstavu fungovania programu pomocou blokového diagramu, by mal mať možnosť túto predstavu intuitívne a jednoducho vložiť do programu. Jeffrey Kodosky teda začal vyvíjať grafické nástroje a implementoval ich do programu, ktorý nazval LabVIEW.Jednotlivé bloky sú reprezentované grafickými ikonkami, je možné ich spájať virtuálnymi vodičmi a výsledky je možné následne zapisovať napríklad do textových súborov, alebo vykresľovať do grafov. Hlavným cieľom virtuálnej interpretácie je nahradiť dočasne alebo aj trvale priestorové, finančné a často aj časovo náročné využitie technických prostriedkov virtuálnym – zdanlivým riešením za pomoci programových prostriedkov. Toto riešenie umožňuje rýchle navrhovanie nových aplikácií a vykonávanie zmien v konfigurácii, čo je pri realizácií skutočnými nástrojmi za pomoci reálnych súčiastok často veľmi nákladné alebo aj nemožné. [4]

Toto vývojové prostredie sa v súčasnosti najčastejšie používa na zber dát, kedy je možné prepojiť meracie zariadenie s počítačom prostredníctvom DAQ karty. Ďalšie široké využitie má v ovládaní meracích prístrojov alebo tiež v priemyselnej automatizácii na riadenie a vizualizáciu technologických procesov. Možno ním tiež programovať zložité systémy ako je robot, či dokonca sústava robotov. Programové vybavenie LabVIEW je veľkým hráčom v oblasti testovania a merania, priemyslovej automatizácie a analýzy údajov.

Program v LabVIEW (tzv. VI - Virtual Interface - Virtuálne rozhranie) pozostáva z dvoch častí - Front panel a Block diagram.

Front panel (Predný panel) slúži ako vstupno-výstupné rozhranie programu. Umiestňujú sa naň rôzne ovládacie prvky (Controls; tlačidlá, prepínače, rôzne textové a numerické vstupy, ...) a zobrazovacie prvky (Indicators; rôzne kontrolky, numerické a textové zobrazovače, rozličné 2D a 3D grafické zobrazovače, ...).



Obr. 3.1: Ukážka programu v LabVIEW

Block diagram (Bloková schéma) slúži na samotnú tvorbu algoritmu. Program sa tvorí vkladaním funkčných blokov a ich prepájaním virtuálnymi vodičmi, ktoré predstavujú premenné, a blok sa vykoná hneď ako má k dispozícií prípustné vstupné dáta. Pri viacvláknových / viacjadrových / viacprocesorových systémoch môžu výpočty blokov bežať paralelne, čo výrazne zrýchľuje výpočet. Bloky majú rozličné funkcie, od základných matematických a logických operácii, cez operácie s poliami, rôzne slučky (while, for), vetviace (case) a sekvenčné (vykonané postupne jedna po druhej) štruktúry, zložité matematické operácie, bloky pre signal processing a generovanie signálov až po špecializované vstupno-výstupné a ovládacie bloky pre rozličné zariadenia (meracie prístroje, DAQ moduly a iné). Na obr. 3.1 je ukážka programu v LabVIEW. Opakujúce sa časti blokovej schémy, prípadne rozsiahle blokové schémy sa dajú previesť do SubVI (podprogram, funkcia v klasickom programátorskom ponímaní) a vo vyššej úrovni zastúpené len ako malý blok so vstupmi a výstupmi.

3.2 Možnosti prepojenia USB modulu s prostredím LabVIEW

Možnosti prepojenia USB modulu s prostredím LabVIEW sú v podstate 2:

1. Virtual Instrument Software Architecture
2. Dynamická knižnica

3.2.1 Virtual Instrument Software Architecture

National Instruments Virtual Instrument Software Architecture (ďalej len VISA) je štandard pre konfiguráciu, programovanie a diagnostiku prístrojových systémov zahrnujúc GPIB, VXI, PXI, Sériovú linku, Ethernet a USB. VISA poskytuje programovacie rozhranie medzi hardware a vývojovým prostredím [5], ako napr. LabVIEW. NI VISA rozlišuje 2 triedy zariadení:

1. Trieda USB INSTR (USBTMC)
2. Trieda USB RAW

Trieda USB INSTR (USBTMC - USB Test and Measurement Class) využíva komunikáciu podobnú štandardu 488.2 [1] .[6] Toto by vyžadovalo v prvom rade jej dôkladné preskúmanie a overenie, či MCU podporuje tento typ USB komunikácie, a ak áno, implementáciu príkazov priamo v MCU. Na jednej strane by to bolo možno dobré (možnosť použitia štandardných komunikačných rozhraní a protokolov, ...), ale na druhej strane by to bola asi zbytočná komplikácia a vzhľadom na obmedzené možnosti MCU by to nemuselo MCU výpočetne zvládať.

Komunikácia s triedou USB RAW je trochu komplikovanejšia, pretože každé zariadenie môže mať svoj komunikačný protokol. Čiže je to v podstate komunikácia na úrovni nižších vrstiev, takže samotná komunikácia s MCU by nemala byť problém. Zložitejšie bude implementovať celý komunikačný protokol v LabVIEW. Ale stále je to asi schodnejšia cesta než trieda INSTR.

3.2.2 Dynamická knižnica

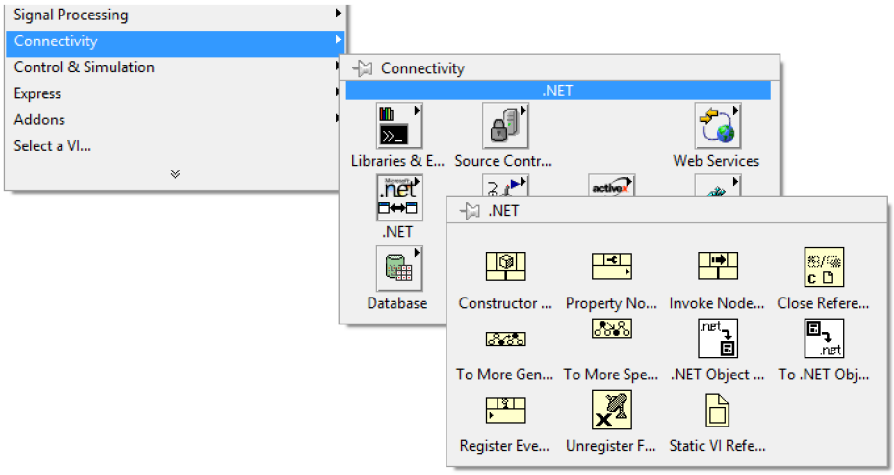
Ako najpraktickejšie sa javí použitie dynamickej knižnice napísanej v bežnom programovacom jazyku, napr. v C#, a jej následné použitie v LabVIEW.

Najväčšia výhoda spočíva asi v tom, že väčšina réžie spojenej s komunikáciou (tvorba a spracovanie balíkov dát, prevod medzi dátovými typmi a formátmi dát)

[1] IEEE-488 (HP-IB, GPIB) je štandardné rozhranie pre prepojenie meracích prístrojov s počítačom pre účely automatizácie merania. Navyše 488.2 zahrňuje minimálne požiadavky pre ovládače (kontroléry) a možnosti (funkcie) zariadení (rozprávač, poslucháč, kontrolér). Takisto je v novej špecifikácii (oproti IEEE 488.1) podrobnejšie definovaný formát a kódovanie údajov, štruktúra komunikačného protokolu a správ medzi kontrolérom a zariadením. [7]

sa dá v bežných programovacích jazykoch spraviť výrazne jednoduchšie ako v LV [2]. Ďalej sa tým zvýši použiteľnosť, keďže modul nebude viazaný na prostredie LV. Ale hlavne sa tým výrazne zjednoduší samotná tvorba komunikačného rozhrania a jeho odladenie. A tým sa zjednoduší aj tvorba a hlavne odlaďovanie firmware pre MCU. A ďalej to ešte zjednoduší prípadné modifikovanie a vylepšovanie rozhrania.

Konkrétne, pre použitie knižníc (resp. tried) vytvorených v jazyku C# (.NET) slúžia funkcie z karty Connectivity zo skupiny .NET (obr. 3.2). Funkcia `Constructor Node` slúži na vytvorenie inštancie danej triedy (vyberá sa v nej konštruktor danej triedy). Pomocou `Property Node` sa dá pristupovať k verejným premenným danej triedy. Funkcia `Invoke Node` sa používa na volanie verejných metód danej triedy.



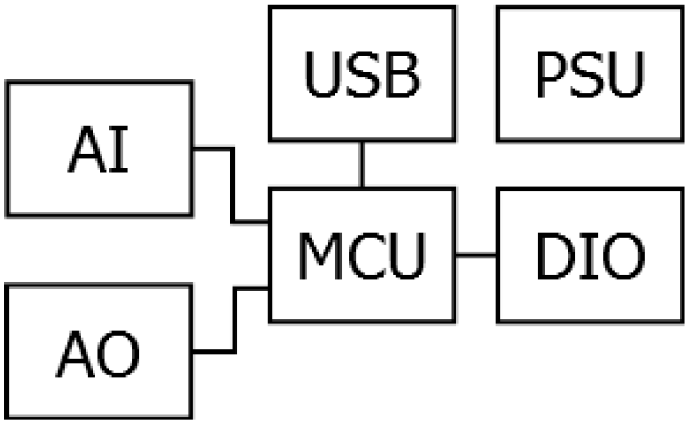
Obr. 3.2: Funkcie na prácu s triedami z jazyka C#

[2] LV – LabVIEW

4 NÁVRH OBVODU

Celý modul bude pozostávať z niekoľkých blokov, ktoré je možné navrhnúť viacmenej nezávisle od ostatných. Srdcom celého modulu je blok obsahujúci riadiaci MCU a obvody potrebné pre jeho chod. Ďalej obsahuje blok na oddelenie zbernice USB, blok obsahujúci obvody predspracovávajúce vstupný signál pre analógové vstupy, blok spracováva výstupný signál, a blok obsahujúci obvody digitálnych vstupov, výstupov a zberníc. A posledný, nemenej dôležitý, je blok starajúci sa o napájanie celého modulu.

Na obrázku 4.1 je bloková schéma celého modulu.



Obr. 4.1: Bloková schéma celého modulu

Popis jednotlivých blokov:

- MCU – blok s mikrokontrolérom a ďalšími odvodmi súvisiacimi s chodom samotného mikrokontroléra
- USB – blok obsahujúci obvody súvisiace so zbernicou USB a oddelením od PC
- DIO – blok obsahujúci digitálne vstupno-výstupné obvody, ich oddelenie od MCU, a ovládanie napäťovej úrovne výstupu.
- AI[1] – blok obsahujúci analógové vstupné obvody
- AO[2] – blok obsahujúci analógové výstupné obvody
- PSU[3] – oblok obsahujúci napájacie obvody

Nasleduje popis návrhu jednotlivých blokov.

[1] AI - Analog Inputs - Analógové vstupy
[2] AO - Analog Outputs - Analógové výstupy
[3] PSU - Power Supply Unit - Napájacia jednotka

4.1 Mikrokontrolér

Ako prvé bolo nutné vybrať MCU. Výber vhodného MCU môže ušetriť množstvo práce pri navrhovaní HW a následnú tvorbu FW. Bolo by dobré, aby MCU mal:

- Vstavaný ADC[4] s rozlíšením aspoň 12bitov a vzorkovaním aspoň 50kS/s
- Vstavaný DAC
- Vstavaný USB modul
- Čo najviac sériových komunikačných rozhraní
- Dostatok GPIO[5]
- Dostatok pamäte
- RTC [6]

Vzhľadom na požadované rozlíšenie 12bitov by bolo najvhodnejšie, aby bolo použité 16-bitové MCU. Pri 8bitovom MCU by šírka slova pôsobila veľké spomalenie práce s datami. Takže týmto odpadá možnosť použitia MCU rady 8051, 8bitových AVR a PIC a iných 8bitových MCU. Keďže samotný modul nemá vykonávať číslicové spracovanie signálu, je asi zbytočné použiť nejaké FPGA[7], DSP[8] alebo ARM. Použitie 32bitového MCU je tiež asi zbytočné, čím sa možnosť výberu MCU značne zúžila v rámci bežných MCU v podstate len na 16-bitové PIC/dsPIC od spoločnosti Microchip a radu MSP430 od spoločnosti Texas Instruments (ďalej len TI). Ďalšia selekcia bola podľa požiadavkov spomínaných vyššie. Najväčšiu váhu pri filtrovaní zohrala kombinácia ADC, DAC a USB. Z veľkého počtu dostupných MCU ostalo len zopár spĺňajúcich tieto požiadavky (zopár z rady PIC24FJ a MSP430F5). Rozhodol som sa pre MSP430F5 od TI, väčšina z nich má väčšiu pamäť, mali by byť rýchlejšie, majú viac potrebných periférii (komunikačných modulov), ďalej mám dobré skúsenosti s radou MSP430 a s výbornou technickou podporou od TI.

Ako najvhodnejšie mi prišli MCU MSP430F5658/5659. Líšia sa len veľkosťou pamätí, a keďže v tomto bode je obtiažne odhadnúť množstvo potrebnej pamäte, zvolil som radšej ten s väčšími pamäťami, t.j. MSP430F5659:

- 66kB SRAM, 512kB FLASH
- 74 GPIO
- 3xUSCI_A[9] a 3xUSCI_B[10]
- 12-kanálový 12bit ADC so vzorkovaním až 200kS/s

[4]Analog-to-Digital Converter – Číslicovo-analógový prevodník

[5]General-purpose input/output – Univerzálne vstupy/výstupy

[6]Real-Time Clock – Reálny čas

[7]Field programmable gate array – Preprogramovateľné hradlové pole

[8]Digital Signal Processor – digitálny signálový procesor

[9]Universal Serial Communication Interface – Univerzálne sériové komunikačné rozhranie, schopné pracovať v režime UART a SPI

[10]Schopné pracovať v režime SPI a I2C

- 2x 12bit DAC
- Full-Speed USB
- RTC so zálohovaním napájania

Zapojenie MCU je štandardné, podľa katalógu. Na zálohovanie RTC je použitý superkapacitor. Toto zálohovanie slúži zároveň aj ako zálohovanie malého bloku pamäte. Na programovanie bude použité dvojvodičové rozhranie Spy-By-Wire (SBW). Ako zdroj hodín pre RTC a pomocných hodín v MCU je slúži kryštál s frekvenciou 32,768kHZ, ako zdroj hodín pre USB je použitý kryštál s frekvenciou 12MHz.

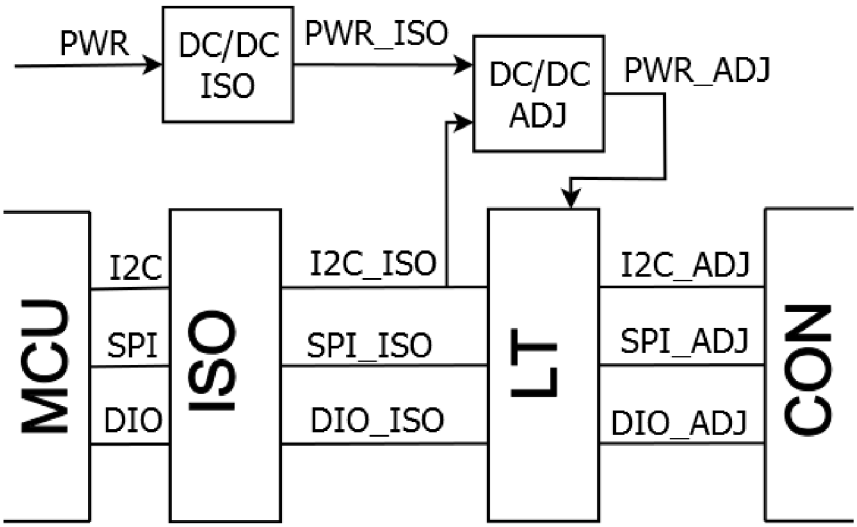
4.2 Digitálne vstupy/výstupy

Pri návrhu bloku obsahujúceho digitálne vstupy a výstupy bolo nutné vyriešiť hneď niekoľko problémov:

- Oddelenie napájania a signálov od MCU
- Nastavenie napäťovej úrovne

Na 4.2 je bloková schéma bloku digitálnych vstupov a výstupov. Popis jednotlivých blokov:

- MCU – Mikrokontrolér
- ISO – Izolátor – oddelenie signálov
- LT – Level Translator – Prevodník úrovní
- CON – Konektor
- DC/DC ISO – Oddeľovací DC/DC menič
- DC/DC ADJ – Nastaviteľný DC/DC menič



Obr. 4.2: Bloková schéma bloku digitálnych vstupov a výstupov

Signály s príznakom ISO za podtržítkom sú izolované od bloku s MCU, s príznakom ADJ sú nastaviteľnej úrovne, bez príznaku sú galvanicky spojené s blokom s MCU.

4.2.1 Oddelenie napájania a signálov

Na oddelenie zbernice I2C som v dobe návrhu našiel len jediný obvod na to určený, a to od spoločnosti Texas Instruments (ďalej len TI) ISO1540. Keďže obvod vie viezť oba signály oboma smermi, môže zbernica fungovať v multi-master režime. Adresa 0x2C je použitá pre digitálny potenciometer (vysvetlené ďalej).

Na oddelenie ostatných dátových signálov som pôvodne plánoval použiť obvod od spoločnosti Analog Devices (ďalej len AD) ADuM5400 v kombinácii s obvodmi ADuM140x. Obvod ADuM5400 obsahuje okrem digitálnych izolátorov aj vstavaný DC-DC menič s 0,5W výstupného výkonu. Obvody využívajú na oddelenie transformátorovú väzbu. Avšak majú jednu značnú nevýhodu, a to že na oboch stranách potrebujú napäťovú úroveň 5V. A keďže MCU používa len úroveň 3,3V, bolo by nutné použiť prevodník úrovní medzi MCU a izolátorom, takže som od variantu s obvodom ADuM5400 upustil. Obvody ADuM140x môžu pracovať s rôznymi napájacími úrovňami.

Obvody od TI ISO724x. Na oddelenie využívajú kapacitnú izoláciu s SiO_2 izolačnou bariérou. Keďže majú väčšinu parametrov podobnú, z ekonomických dôvodov som zvolil obvody ISO724x.

Na oddelenie ôsmych diskrétnych I/O sú použité 4 obvody ISO7240 (4bity rovnakým smerom), 2 a 2 anti-paralelne (jeden vedie signály smerom od MCU, druhý smerom k MCU. Aktívny smer je zvolený pomocou Enable vstupu obvodu. Táto konfigurácia umožňuje nezávislé nastavenie smeru dát pre 4 a 4 bity a zároveň je to asi konfigurácia vyžadujúca minimum obvodov a režijných signálov z MCU. Ako riadiace signály sú na izolovanej strane využité 2 digitálne výstupy digitálneho potenciometra (spomenutý bude ďalej).

Na oddelenie zbernice SPI/UART je použitý obvod ISO7241 (3bity smerom od MCU, 1bit smerom k MCU). Takéto zapojenie, žiaľ, limituje použitie zbernice na režim MCU–Master, pripojené zariadenie–Slave. Iné kombinácie by vyžadovali použitie ďalšieho izolátora pre opačný smer a ďalšieho pre signály ovládajúce smer prenosu (keďže tie by museli byť prenášané nezávislé od nastaveného smeru prenosu). Tieto vývody je možné použiť aj čisto ako digitálne vstupy/výstupy bez možnosti zmeny smeru prenosu dát.

Na oddelenie napájania som použil obvod DCR010505 od spoločnosti TI. Je to izolovaný DC-DC menič, so vstupným napájaním 5V, regulovaným výstupným napätím 5V a výstupným výkonom 1W. Na funkčnosť mú stačia 4 externé súčiastky:

vstupný blokovací kondenzátor, vyhladzovací kondenzátor medzi usmerňovacím členom a výstupným regulátorom, filtrovací kondenzátor na výstupe a zdvíhací rezistor na Enable vstupe.

4.2.2 Nastavenie napäťovej úrovne

Pri riešení nastavenia napäťovej úrovne bolo nutné vyriešiť 2 problémy:

- Nastavenie veľkosti napätia
- Prenos tohto napätia na logické vstupy/výstupy

Jedno z riešení nastavenia veľkosti napätia by bolo prepínanie stabilizátorov, resp výstupov stabilizátorov. To by ale vyžadovalo použitie veľa súčiastok (stabilizátorov, prepínačov, pasívnych súčiastok...) a tým pádom by rapídne stúpla aj spotreba. Celkovo použitie klasických stabilizátorov by znamenalo zbytočné výkonové straty, hlavne pri nižších napätiach. Ako vhodnejšie sa javí použitie nastaviteľného DC/DC meniča.

Spoločnosť TI vyrába obvod TPS62200. Je to Step-down menič so vstupným napätím od 2,5 do 6V a s výstupným napätím nastaviteľným v rozsahu od 0,7V až po veľkosť vstupného napätia s prúdom až 300mA s účinnosťou až 95%. Výstupné napätie sa nastavuje deličom v spätnej väzbe:

$$V_{\mathrm{OUT}} = 0,5V * \frac{R_1 + R_2}{R_2} \tag{4.1}$$

kde R_1 a R_2 tvoria delič v spätnej väzbe.

V našom prípade bude delič tvorený sériovou kombináciou 150kΩ pevného rezistora a digitálneho potenciometra od spoločnosti Analog Devices (ďalej len AD) AD5241 s celkovým odporom 100kΩ. Touto kombináciou vieme dosiahnuť na výstupe napätie v rozsahu 1,25V až po takmer vstupné napätie. Znížením hodnoty pevného rezistora sa dá spodná hranica ešte znížiť, teoreticky až po hodnotu 0,7V udávanú výrobcom (pri použití 40kΩ rezistora); avšak minimálna použiteľná hodnota na výstupe je 1,2V (140kΩ), čo je minimálne pracovné napätie použitého prevodníka úrovní (riešené ďalej). Horná hranica je daná úbytkom na výkonovom spínacom tranzistore v obvode TPS62200.

Obvod AD5241 má rozlíšenie 8bitov (256 krokov), je ovládaný cez zbernicu I2C. Navyše ešte má 2 digitálne výstupy, ktoré využívam na ovládanie smeru prenosu digitálnych dát na izolovanej strane. Jeho adresa je 0x2C, a preto na zbernici nesmie byť externe pripojené nič s touto adresou.

Ďalším krokom bolo vybrať vhodný prevodník úrovní. Najťažšie bolo nájsť taký, ktorý nevyžaduje pre chod väčší rozdiel napätia medzi primárnou a sekundárnou stranou (keďže väčšina obvodov vyžaduje rozdiel aspoň 0,3V) a nevyžaduje externé riadenie smeru prenosu dát.

Ako prvý som našiel obvod ADG3300 od AD, ktorý spĺňa všetko vyššie spomínané. Avšak po dôkladnom preštudovaní dokumentácie som zistil, že vyžaduje na vybudenie jednotlivých kanálov prúd až 36mA, čo je pre našu aplikáciu nepoužiteľné. Potom som objavil obvod ADG3308, ktorý už túto nevhodnú vlastnosť nemá a mal by zvládať prenosovú rýchlosť až 50Mb/s. Potom som ešte objavil obvod od spoločnosti Maxim Integrated MAX3002E, ktorý taktiež spĺňa požiadavky uvedené vyššie. Prenosovú rýchlosť má síce len 20Mb/s (čo ale vzhľadom na rýchlosť MCU akurát postačuje), ale oproti AD3308 navyše poskytuje na všetkých I/O pinoch ESD[11] ochranu až na 15kV. Takže som sa rozhodol pre tento obvod.

V súčasnom usporiadaní majú všetky vstupy/výstupy (diskrétne vstupy, zbernice SPI a I2C) rovnakú napäťovú úroveň.

4.2.3 Poznámky k návrhu

Rôzne napäťové úrovne na rôznych pinoch

Použitie rôznych napäťových úrovní na rôznych výstupoch by bola značná komplikácia. Vyžadovalo by to použitie toľkých meničov spomínaných vyššie, koľko by bolo treba I/O s nezávislou napäťovou úrovňou. A ďalej by bolo treba odpovedajúce množstvo prevodníkov úrovní. S počtom meničov by taktiež súvisel aj počet digitálnych potenciometrov, ktorých počet je obmedzený hlavne množstvom možných adries na zbernici I2C (keďže počet nastaviteľných bitov adresy na strane slave zariadenia nebýva vyšší než 2). Ale teoreticky, v prijateľnej miere, to možné je.

Viac pinov s nezávisle nastaviteľným smerom

Keďže sa zatiaľ asi nevyrábajú obojsmerné izolátory (aspoň som taký, teda až na zopár určených na oddelenie zbernice I2C, nenašiel), je daný počet pinov s nezávislým smerom závislý od počtu nezávislých izolátorov. Avšak vyšší počet nezávislých izolátorov vyžaduje vyššie množstvo ovládacích signálov z MCU, a tým pádom aj ďalšie izolátory.

Výstupy typu otvorený kolektor

V súčasnej koncepcii má obvod len aktívne výstupy. V niektorých aplikáciach by ale boli žiadúce výstupy typu otvorený kolektor. Rozšírenie o výstupy typu otvorený kolektor by znamenalo použitie množstva elektronických prepínačov (ich počet a množstvo bitov by záviselo od počtu nezávislých možností nastavenia). A na ich ovládanie by bolo treba ďalšie ovládacie signály z MCU, a tým pádom aj ďalšie izolátory. V prípade potreby nie je problém tento typ výstupov implementovať externe.

[11] Electrostatic discharge - výboj elektrostatického náboja

4.3 USB

Ďalší krok bolo vyriešiť oddelenie USB. Keďže ide o komplikovanejšiu zbernicu než napr. SPI, na jej oddelenia nestačia jednoduché izolátory ako v prípade SPI,

Zistil som, že spoločnosť AD vyrába obvod ADUM4160, USB izolátor. Podporuje full-speed USB 2.0, s rýchlosťou až 12Mb/s. Umožňuje vybrať rýchlosť: low-speed (1,5Mb/s) alebo full-speed (12Mb/s). Ďalej obsahuje možnosť ovládania prenosu Pull-up na vodiči D+ (pre potrebu ručného ovládania nadviazania spojenia - enumerácie).

Zvolil som režim full-spped. Pin ovládajúci zdvíhací rezistor linky D+ (PIN) je pripojený na odpovedajúci pin na MCU (PUR) a cez pull-up rezistor privedený na linku D+. Zvyšok je zapojený podľa katalógu.

4.4 Analógové vstupy

Bloková schéma analógových vstupov je na obr.4.3. Popis jednotlivých blokov a

Obr. 4.3: Bloková schéma analógových vstupov

signálov:

- AIx - Analog input x – Analógový vstup x
- IA - Input amplifier – Vstupný zosilňovač
- HPF - High-pass filter – Hornopriepustný filter
- LPF - Low-pass filter – Dolnopriepustný (Antialliasingový) filter
- ADCx - Signál do ADC

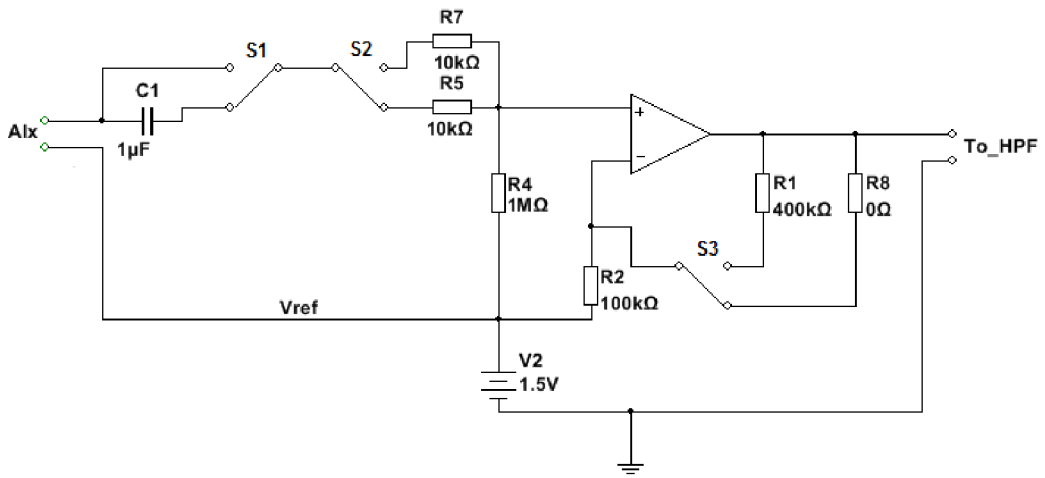
4.4.1 Vstupný zosilňovač

Úlohou vstupného zosilňovača je upraviť veľkosť vstupného signálu tak, aby bol v rozsahu ADC a posunúť bipolárny signál tak, aby bol zmerateľný pomocou single-ended ADC. Ďalej sa vstupný zosilňovač stará o zmenu rozsahu a o prípadné odstránenie jednosmernej zložky.

Keďže sa v celom reťazci môže meniť, vzhľadom na aktuálne zvolenú konfiguráciu, množstvo aktívnych operačných zosilňovačov (ďalej len OZ), použil som len neinvertujúce zapojenia OZ. Ďalej, z dôvodu obmedzení rozkmitu napätia pre obvod slúžiaci ako dolnopriepustný filter (popísané a vysvetlené ďalej), je nutné signál

upraviť na rozsah 0,5 až 2,5V, inak povedané, nulovému napätiu na vstupe bude odpovedať 1,5V na výstupe, maximálnemu kladnému napätiu bude odpovedať 2,5V, maximálnemu zápornému napätiu bude odpovedať 0,5V na výstupe. Takže je nutné celé zapojenie riešiť oproti virtuálnej zemi odpovedajúcej 1,5V proti systémovej zemi. Taktiež aj vstupné napätie musí byť privedené proti virtuálnej zemi, a nie systémovej. Našťastie, použitý ADC umožňuje nastaviť aj hornú aj spodnú referenciu, takže to umožňuje využiť celý rozsah ADC bez nutnosti ďalšieho upravovania signálu po dolnopriepustnom filtri (popísanom ďalej).

Teoretická schéma vstupného zosilňovača je na obr.4.4. O prepínanie rozsahov



Obr. 4.4: Teoretická schéma vstupného zosilňovača

sa starajú prepínače S2 a S3 (prepínané naraz).

Spínač S2 v kombinácii s rezistormi R5, resp. R7, je v podstate zbytočný, je to pozostatok pôvodného návrhu (spomínané ďalej). Rezistor R5, resp. R7, slúžia ako ochrana vstupu OZ. Rezistor R4 slúži ako vstupná impedancia analógového kanálu. Keď zanedbáme deliaci pomer deliča vstupe, môžeme veľkosť vstupné signálu privedeného na OZ považovať za nezmenenú.

Pre rozsah ±1V je signál operačným zosilňovačom posunutý o 1,5V, t.j. na rozsah 0,5 až 2,5V. A keďže to je požadovaná úroveň signálu, operačný zosilňovač má zosilnenie nastavené spätnou väzbou z rezistorov R2 a R8 na 1 ($k = 1+\frac{R8}{R2} = 1+\frac{0}{100k}$=1).

Pre rozsah ±0,2V je signál operačným zosilňovačom s rezistormi R1 a R2 v spätnej väzbe zosilnený 5krát ($k = 1 + \frac{R1}{R2} = 1 + \frac{400k}{100k}$=5) a posunutý o 1,5V na požadovanú úroveň.

Vstupný odpor na oboch rozsahoch je približne 1MΩ.

Prepínačom S1 sa volí jednosmerná alebo striedavá väzba zo vstupu. Na oddelenie jednosmernej zložky slúži kondenzátor C1 s kapacitou 1μF. Kapacita musí

byť pomerne vysoká vzhľadom na pomerne vysoký odpor vstupného deliča, keďže kombinácia CR tvorí hornopriepustný filter s medznou frekvenciou [10]:

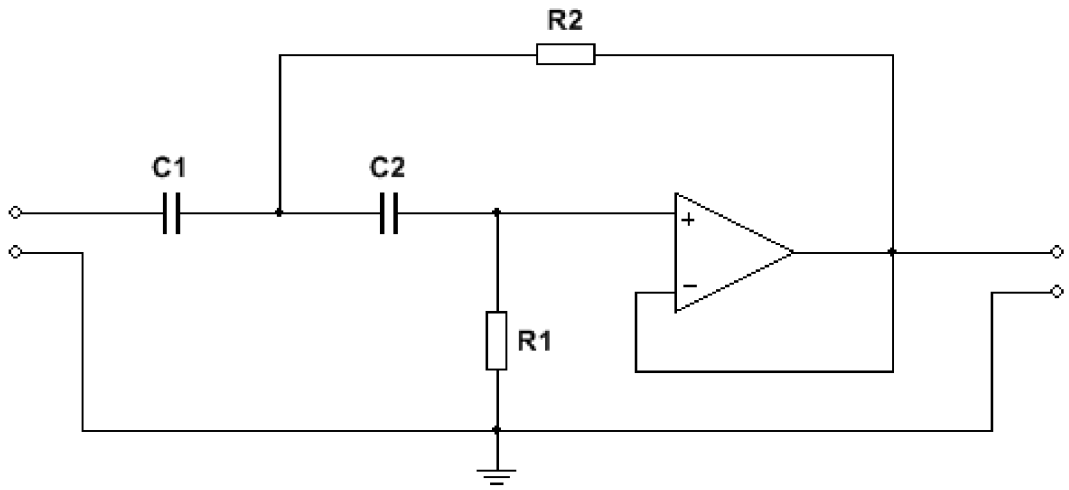
$$f_{medz} = \frac{1}{2\pi RC} \tag{4.2}$$

Kapacita musí byť veľká preto, aby bola medzná frekvencia čo najnižšia a signál na nízkych frekvenciách čo najmenej zmenený. Pre oba rozsahy je medzná frekvencia približne 0,16Hz.

Ako ochrana celého meracieho reťazca je na vstupe použitá obojsmerná TVS[12] dióda s napätím 5V.

4.4.2 Hornopriepustný filter

Teoretická schéma hornopriepustného filtra druhého rádu [8] je na obr.4.5. Použil



Obr. 4.5: Teoretická schéma hornopriepustného filtra 2. rádu

som topológiu Sallen-Key. Medzná frekvencia takéhoto filtra je

$$f_{medz} = \frac{1}{2\pi RC\sqrt{mn}} \tag{4.3}$$

kde:

$$R_1 = R$$

$$C_2 = C$$

$$R_2 = mR$$

$$C_1 = nC$$

[12]Transient-voltage-suppression

Faktor kvality takéhoto filtra je:

$$Q = \frac{\sqrt{mn}}{m+1} \tag{4.4}$$

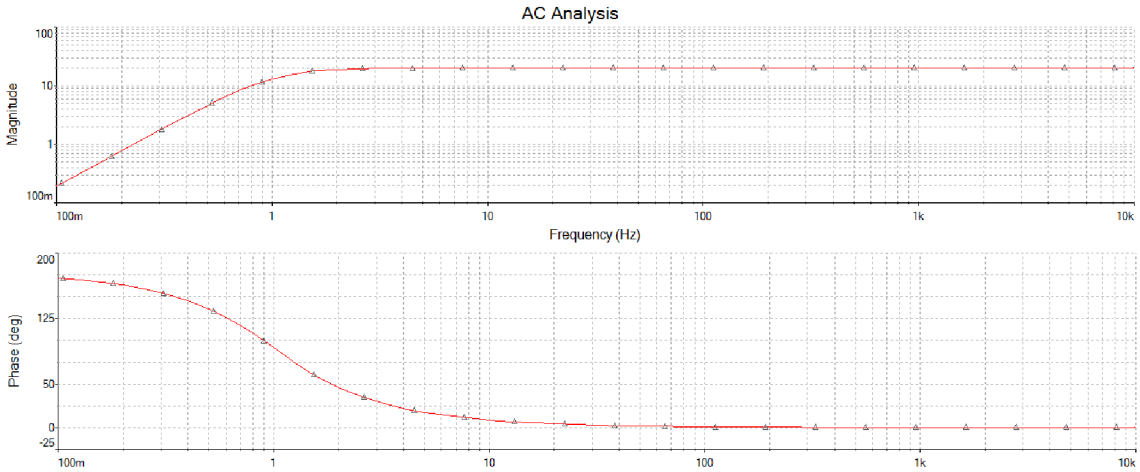
Alternatívna možnosť výpočtu medznej frekvencie takéhoto filtra je:

$$f_{medz} = \frac{1}{2\pi\sqrt{R_1 R_2 C_1 C_2}} \tag{4.5}$$

V našom zapojení sú požadované 2 filtre, a to s medznou frekvenciou 1Hz a 100Hz. Prepínanie medznej frekvencie pomocou elektronických frekvencie pomocou prepínačov, ako je tomu pri vstupnom zosilňovači, by bolo problematické, lebo by vyžadovalo použitie veľkých odporov a veľkých kapacít. Preto som zvolil radšej prepínanie samotných filtrov. Prepínanie je riešené pomocou dvojitého štvorpolohového elektronického prepínača od TI TS3A5017. Jeden kanál je bez filtra (premostenie), na ďalších dvoch sú filtre s medznou frekvenciou 1Hz a 100Hz, posledný kanál je prichystaný pre ľubovoľný filter topológie Sallen-Key, zatiaľ bez špecifikácie.

Výpočet filtra

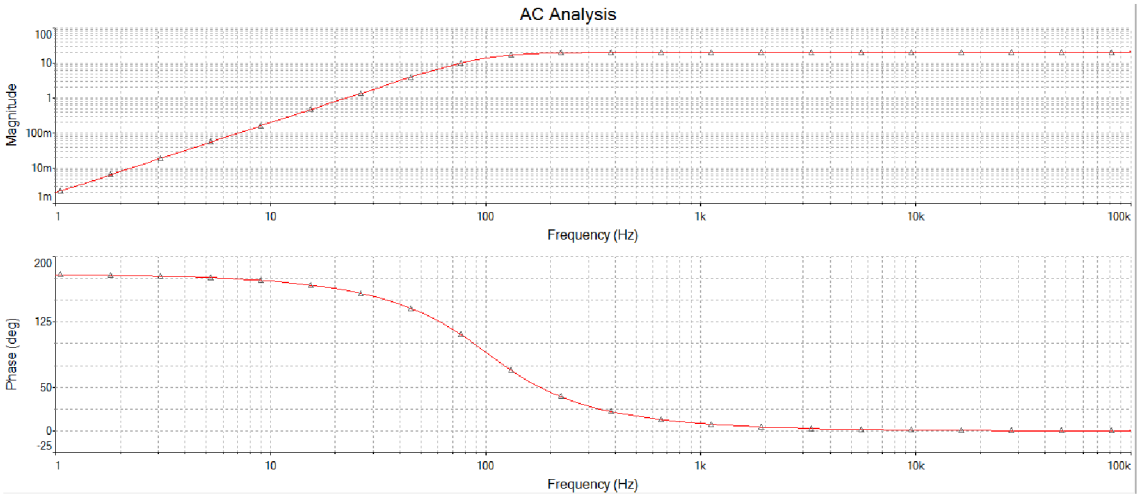
Pre frekvenciu 1Hz zvolím $mn = 1$ a $C = C_2 = 270$nF. Podľa vzťahu 4.3 dopočítam $R = R_1 = 589k\Omega$, zvolil som hodnotu $576k\Omega$. C_1 som zvolil približne dvojnásobné, t.j. 560nF, n vyjde tým pádom približne $n = 2,07$, z toho určím $m = \frac{1}{n} = 0,47$, z toho som určil $R_2 = 284k\Omega$. Z okolitých hodnôt som zvolil $R_2 = 280k\Omega$. Podľa vzťahu 4.5 vyjde medzná frekvencia tohto filtra $f_{medz} = 1,02$Hz a podľa vzťahu 4.4 vyjde $Q = 0,69$. Frekvenčná charakteristika tohto filtra je na obr. 4.6.



Obr. 4.6: Teoretická frekvenčná charakteristika filtra horná prepusť s medznou frekvenciou 1Hz

Pre frekvenciu 100Hz bol postup výpočtu rovnaký. Výsledné hodnoty: $C_1 = 220$nF, $C_2 = 100$nF, $R_1 = 16k\Omega$, $R_2 = 7,15k\Omega$. Podľa vzťahu 4.5 vyjde medzná

frekvencia tohto filtra $f_{medz} = 100,3$Hz a podľa vzťahu 4.4 vyjde $Q = 0,69$. Frekvenčná charakteristika tohto filtra je na obr. 4.7.



Obr. 4.7: Teoretická frekvenčná charakteristika filtra horná prepusť s medznou frekvenciou 100Hz

4.4.3 Dolnopriepustný filter

Dolnopriepustný filter pred ADC plní prioritne úlohu antialliasingového filtra. Preto je užitočné, aby tento filter mal čo najväčšiu strmosť. A vzhľadom na množstvo požadovaných vzorkovacích frekvencii riešenie použité pri hornopriepustných filtroch (prepínanie filtrov elektronickým prepínačom) by vyžadovalo veľké množstvo súčiastok. Preto bolo nutné nájsť iné riešenie, a to integrovaný, čo najjednoduchšie laditeľný filter s potrebným frekvenčným rozsahom a s možnosťou ladenia v čo najväčšom rozsahu.

Spoločnosť Linear Technology vyrába obvod LTC1569-6. Je to dolnopriepustný filter 10.rádu s lineárnou fázou. Nevyžaduje externé hodiny a medzný kmitočet sa nastavuje pomocou jedného externého rezistora a s nastavení deličky frekvencie pomocou jedného externého pinu (1/1, 1/4, 1/16). Nastavenie frekvencie je možné až v rozsahu 6 oktáv. Obvod je určený na prácu do medznej frekvencie až 64kHz pri napájaní 3V. Existuje aj variant LTC1569-7 s medznou frekvenciou 128kHz (256kHz pri napájaní 5V). Medzná frekvencia sa určí podľa vzťahu:

$$f_{medz} = \frac{64\text{kHz}(\frac{10k}{R_{EXT}})}{div} \tag{4.6}$$

kde:

- R_{EXT} – externý rezistor
- div – delič: 1, 4 alebo 16

Pre delič 1 musí byť pin DIV uzemnený, pre delič 4 musí byť plávajúci (odporúčajú naň pripojiť 100pF kondenzátor voči zemi), pre delič 16 musí byť privedený na napájacie napätie. Obvod má však jedno veľmi nepríjemné obmedzenie, a to rozkmit vstupného napätia. Obvod nevie pracovať so signálom v rozsahu napájacieho napätia, ale len v rozsahu od 0,3V po (napájacie napätie-0,8V) (v prípade unipolárneho napájania).

Jemná zmena medznej frekvencie (externý rezistor) je realizovaná pomocou digitálneho potenciometra AD5241 zapojeného ako premenlivý odpor. Hrubá zmena (prepínanie pinu DIV) je realizovaná pomocou jednej skupiny kontaktov elektronického prepínača TS3A5017. Druhá skupina kontaktov realizuje prípadný premostenie tohto obvodu. Ovládaný je pomocou digitálnych výstupov digitálneho potenciometra AD5241. Zvyšok zapojenia je podľa katalógu. Optimálna medzná frekvencia vzhľadom na vzorkovaciu frekvenciu je:

$$f_{medz} \doteq 0,7 * \frac{f_{vz}}{2} \tag{4.7}$$

f_{vz}	f_{medz}	R_{EXT} [Ω]		
[kHz]	[kHz]	div = 1	div = 4	div = 16
40	16	40k	10k	2,5k
20	8	80k	20k	5k
10	4	160k	40k	10k
5	2	320k	80k	20k
2	0,8	800k	200k	50k
1	0,4	16M	400k	100k

Tab. 4.1: Nastavenie externého rezistora R_{EXT} a deliča *div* pre jednotlivé vzorkovacie frekvencie

Požadované hodnoty externého odporu a nastavenia deličky pre zadané vzorkovacie frekvencie sú v tab. 4.1. Medzná frekvencia je určená podľa vzťahu 4.7. Mohlo by sa zdať, že by prepínanie deličky nebolo ani nutné, ale výrobca udáva, že externý rezistor nesmie byť menší než 3,8kΩ. Ďalej to umožňuje dosiahnuť aj vyšších vzorkovacích frekvencii.

4.4.4 Napäťová referencia

Ako napäťovú referenciu pre ADC a pre virtuálnu zem som použil obvod od TI REF2030. Veľká výhoda tohto obvodu je to, že má 2 výstupy (1,5 a 3V), vyžaduje veľmi malý úbytok napätia a výstupný prúd má až ±20mA.

Výstup 1,5V som použil ako virtuálnu zem. Výstup 3V je cez odporový delič rozdelený približne na napätia 0,5V a 2,5V a tie sú privedené ako spodná a vrchná referencia pre ADC.

4.4.5 Automatické kalibrovanie/kompenzovanie

Vzhľadom na to, že sa v obvode nikde nevyskytuje presné napätie odpovedajúce maximálnej hodnote niektorého rozsahu, možnosť automatického kalibrovania nie je možná. Takže kalibrovanie bude možné len pomocou presného externého zdroja za pomoci software.

Toto ale neplatí pre možnosť kompenzovania nuly, keďže tá je daná napäťovou referenciou. Na vstupe celého obvodu je elektronický prepínač, ktorý môže túto referenciu priviezť na vstup obvodu a program bude schopný si sám vykompenzovať prípadný posun nuly.

4.5 Analógové výstupy

Analógové výstupy pozostávajú z jednoduchého pasívneho RC rekonštrukčného filtra a výkonového posilnenia v podobe neinvertujúceho sledovača z operačného zosilňovača [9]. Medzná frekvencia použitého filtra sa určí podľa vzťahu:

$$f_{medz} = \frac{1}{2\pi RC} \tag{4.8}$$

Pre vzorkovaciu frekvenciou 200kHz vychádza podľa vzťahu 4.7 medzná frekvencia filtra $f_{medz} \doteq 70\text{kHz}$. Pre kapacitu C = 50pF vychádza podľa vzťahu 4.8 hodnota odporu $R \doteq 45k\Omega$.

Od možnosti prepínania rozsahov som upustil. Na 5V rozsah by bolo nutné do analógovej oblasti dosky doviesť napätie 5V, ktoré sa v nej nikde nevyužíva. Ďalej by bolo nutné riešiť prepínanie rozsahu prepínačmi, podobne ako je tomu pri analógových vstupoch, na čo už ale doske nie je miesto. Ďalej by bolo nutné tam doviesť ovládacie signály, čo by tiež znamenalo značné skomplikovanie návrhu plošného spoja, už len kvôli tomu, že na strane MCU priľahlej k analógovej časti nie sú voľné žiadne, rozumne dostupné, digitálne výstupy. Z rovnakého dôvodu som upustil aj od možnosti loopbacku[13].

V súčasnom návrhu sú možné 2 zapojenia výstupu:

1. *Zapojenie výstupu proti systémovej zemi* - rozsah výstupného napätia je od 0V až po napájacie napätie (prípadne inú referenciu zvoliteľnú v MCU)

[13]Privedenie signálu z výstupu na vstup priamo na doske, ovládateľné softwarovo

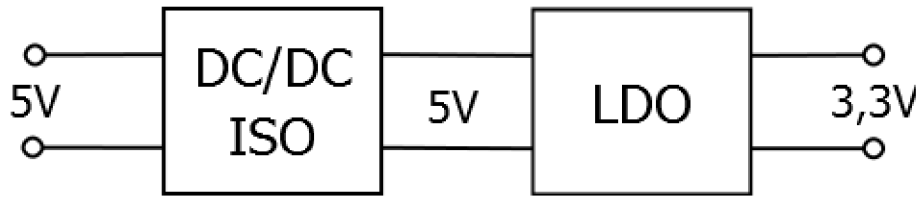
2. *Zapojenie výstupu proti analógovej zemi* - rozsah výstupného napätia je od -1,5V až po (napájacie napätie - 1,5V) (prípadne inú referenciu zvoliteľnú v MCU).

Variant 2 aspoň čiastočne kompenzuje obmedzenia spôsobené jedným napájacím napätím. Pozor však treba dať na to, že **analógová a systémová zem nesmú byť za žiadnych okolností prepojené!** Mohlo by to spôsobiť poškodenie obvodov napäťovej referencie!

4.6 Napájanie

Bloková schéma napájacích obvodov je na obr. 4.8. Význam jednotlivých blokov:

- *DC/DC ISO* – izolujúci DC/DC menič
- *LDO* – Low drop-out – stabilizátor s nízkym úbytkom napätia



Obr. 4.8: Bloková schéma napájacích obvodov

Ako hlavný zdroj napájania bude slúžiť USB port. Kvôli ochrane PC musí byť napájanie, podobne ako dátová zbernica, oddelené. Ďalší dôvod oddelenia napájania je oddelenie od globálnej zeme, čo umožní meranie na rôznych potenciálov, takže nie sme viazaný na meranie voči zemnému potenciálu.

Izolovaných 5V slúži na napájanie sekundárnej strany USB izolátora, izolujúceho DC/DC meniča pre modul digitálnych vstupov a výstupov a pre stabilizátor napájací zvyšok obvodov.

Na oddelenie som použil izolujúci menič IL0505S s výkonom 2W. Doska je navrhnutá na 2 takéto meniče paralelne. Na doske je vyvedený aj konektor, pre prípadný stand-alone mód (mód bez pripojenia k PC rozhraním USB). O napätie 3,3V sa stará stabilizátor od TI TPS76733. Napájanie je na každom stupni chránené Zenerovou diódou.

4.7 Možnosť pripojenia k nadradenému systému rozhraním iným než USB

Než sa pokúšať hardwarovo implementovať niektoré iné komunikačné rozhranie než USB také, čo nemá použitý MCU vstavané, javí sa ako najschodnejšia cesta použitie hotového modulu (či už na bluetooth, ethernet alebo iné) s niektorým štandardným sériovým rozhraním SPI/I2C/UART), a pre to následne implementovať len komunikačný protokol.

Na tento účel som na konektor vyviedol port P9 MCU, na ktorý sú pripojené USCI_A a USCI_B moduly[14].

4.8 Zmeny oproti pôvodnému návrhu

V priebehu oživovania modulu som narazil na niekoľko chýb prvotného návrhu spôsobených najčastejšie zlým pochopením katalógového listu a nepozornosťou.

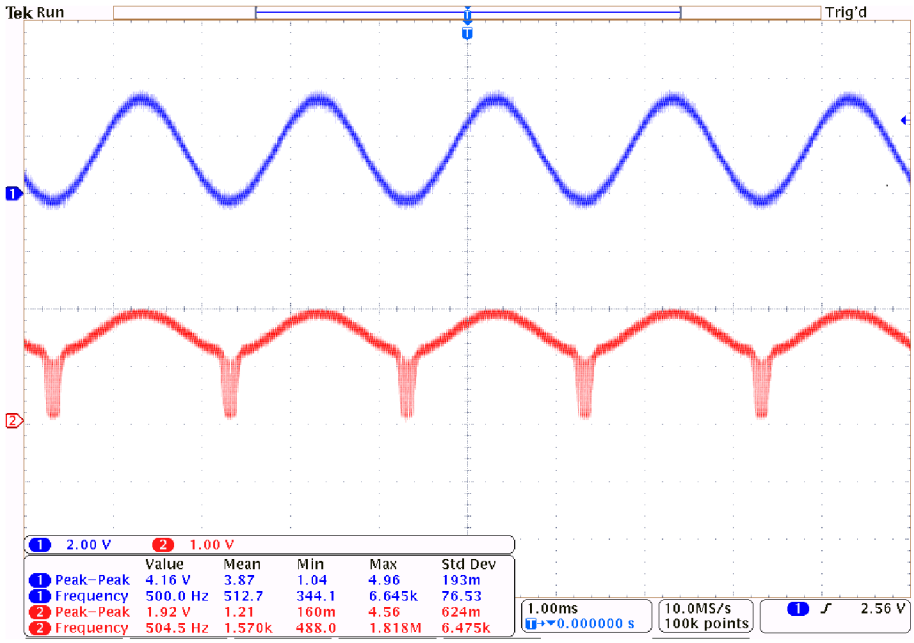
ADuM4160 a pull-up na linke D+

Vstup PIN obvodu ADuM4160 je datasheete popísaný ako Upstream Pull-up Enable (povolenie pull-up-u na upstream[15] linke) a má ovládať výkonové pripojenie Pull-up rezistora na upstream linke a má byť pripojený buď na napájacie napätie pre okamžitú činnosť po privedení napájania alebo na externý ovládací signál pre aplikácie vyžadujúce oneskorenú enumeráciu[16]. Tak som pripojil na výstup PUR MCU, ktorý toto obstaráva. Po nejakom čase strávenom pátraním po príčine som sa na nejakom diskusnom fóre dočítal, že pin akurát ovláda prenos pull-upu, avšak privedenie PIN signálu na log. 1 neaktivuje pull-up na výstupe. Pull-up treba fyzicky realizovať na downstreame[17], čo si vyžiadalo pridanie rezistoru R54 medzi piny PUR a D+.

Napäťové rozsahy analógových vstupov

Pôvodný návrh počítal s rozsahmi ±5V a ±0.5V. Menší rozsah fungoval bez problémov, avšak keď som na väčší rozsah priviedol napätie s amplitúdou väčšou než 1.5V, priebeh na výstupe analógového reťazca (resp. už na výstupe OZ vstupného zosilňovača) sa začínal postupne deformovať (pre signál ±2V ukážka na obr. 4.9, modrá - vstupný signál, červená - výstupný signál). Po vylúčení rôznych príčin (napr. odplávanie virtuálnej zeme, nestabilita OZ, ...) sa ukázalo, že prepínače na vstupe reťazca nezvládajú prenos signálu mimo rozsah ich napájacieho napätia. A keďže

[14] Sériové komunikačné rozhrania schopné pracovať v režimoch: A) SPI a UART, B) SPI a I2C
[15] Časť linky medzi izolátorom a USB portom PC
[16] Nadviazanie USB spojenia
[17] Spojenie medzi MCU a izolátorom



Obr. 4.9: Deformovanie priebehu pri rozsahu ±5V

nebolo možné im priviesť napájacie napätie napätie aspoň na úrovni vstupného signálu, museli sme od týchto rozsahov upustiť a zvoliť súčasné rozsahy.

Buffer na napäťovej referencii

V pôvodnom návrhu bol 1,5V výstup napäťovej referencie posilnený OZ zapojeným ako napäťový sledovač. Pri meraní (spomínanom vyššie) reálneho stavu tejto referencie som zistil mierne oscilácie. Po dôkladnom preštudovaní datasheetu k OZ a zistení, čo všetko je pripojené na tento potenciál som zistil, že použitý OZ nezvláda kapacitnú záťaž v podobe blokovacích kondenzátorov pri obvode LT1569. Našťastie, samotná referencia túto záťaž zvláda, tak bol OZ odstránený a virtuálna zem pripojená priamo na výstup referencie.

Zbernica I2C na analógovej strane

Pri oživovaní dolnopriepustného filtra som narazil na problém nadviazania spojenia s digitálnymi potenciometrami určujúcimi nastavenie filtra. Zistil som, že na zvolenú adresu ani jeden obvod nezareagoval. Po vylúčení chybných adries a chyby vo obslužnom programe (keďže rovnaké funkcie som použil na rovnaký digitálny potenciometer na digitálnej strane kde bez problémov fungovali), som sa zameral na hardware. Zistil som, že som pri návrhu zabudol na pull-upy na zbernici I2C. Zapnúť pull-upy v MCU nepomohlo. Našťastie sa mi na DPS[18] podarilo nájsť miesto, kde sa dali pripojiť bez nutnosti narušenia DPS. Problém to však nevyriešilo. Po uistení sa, že problém nie je v hardware som dôkladne zanalyzoval dianie v prog-

[18]Doska plošných spojov

rame a na doske. Všetko nasvedčovalo tomu, že sú medzi sebou prehodené signály SDA a SCL. Na porte 2 MCU sú všetky vývody mapovateľné, čo som využil pri návrh a v obslužnom programe som bral do úvahy. V programe vyzeralo byť všetko v poriadku, no po nazretí do reálnych nastavení mapovacích registrov sa ukázalo, že sú nastavené naopak, než ich nastavujem v programe. V datasheete som zistil, že ostali v predvolenom stave. Ukázalo sa, že problém je v odomknutí týchto registrov na zápis, čo je značne mätúco popísané v používateľskej príručke k MCU.

Spy-by-wire

Rozhranie Spy-by-wire je zapojené podľa datasheetu, resp podľa príručky k programovaniu MCU MSP430. Avšak pri prvom pokuse o naprogramovanie MCU ho programátor nevedel rozpoznať. S týmto problémom som sa však už stretol v minulosti pri inom projekte, stačilo odstrániť kondenzátor C65 (reset a zároveň súčasť dátovej linky pri SBW[19]). Len neviem, či je toto nejaký globálny problém MSP430 alebo je problém v mojom (neoriginálnom) programátore.

Našťastie, všetky problémy sa zaobišli bez nutnosti narušenia DPS (porušenia izolácie, prerušovania ciest, . . .). Aj keď vyriešenie väčšiny z nich zabralo dosť času. . .

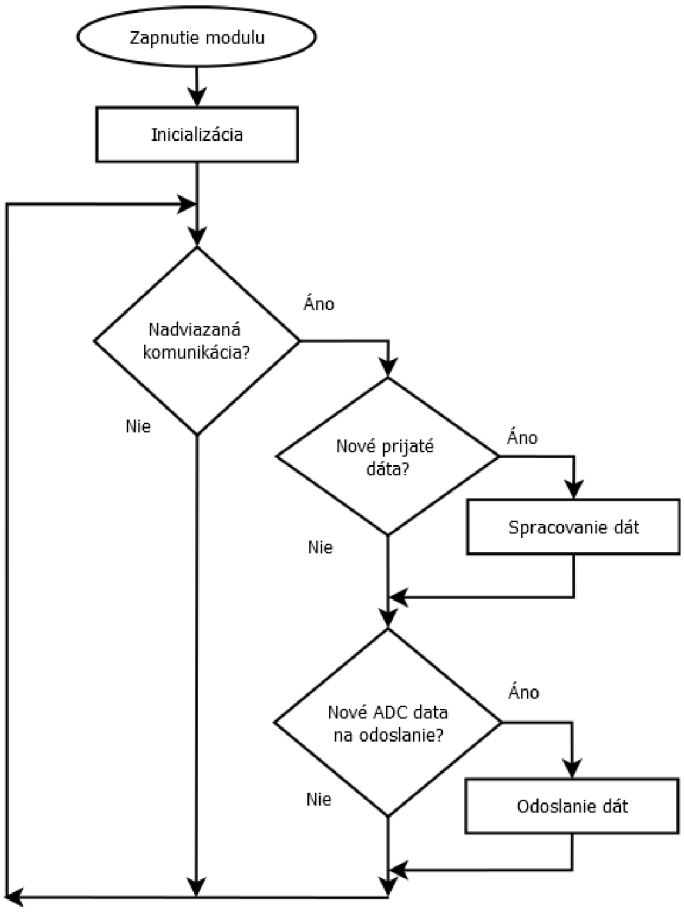
[19] Spy-by-wire

5 ČINNOSŤ MODULU A KOMUNIKAČNÝ PROTOKOL

V tejto kapitole je rozobratá základná štruktúra firmware modulu, možnosti nastavenia modulu a režimy činnosti. Na záver je popísaný komunikačný protokol.

5.1 Firmware modulu

Základná činnosť firmware by sa dala zjednodušene popísať diagramom na obr.5.1. Po spustení modulu prebehne najprv inicializácia samotného MCU a jeho periférii



Obr. 5.1: Zjednodušená hlavná slučka programu

a následne prebehne inicializácia (privedenie do definovaného stavu) celého modulu. Nasledujúca časť prebieha v nekonečnej slučke. Program skontroluje, či je úspešne nadviazaná komunikácia. Ak nie, program skočí na koniec slučky. Ak je nadviazaná komunikácia, program skontroluje, či modul prijal nejaké nové dáta. Ak áno, dáta sa spracujú a prípadne sa odošle odpoveď. Ďalej program skontroluje, či nie sú k

dispozícii nové dáta z analógových vstupov pripravené na odoslanie. Ak sú, tak ich program odošle. Na konci slučky sa CPU uspí, čím sa pozastaví vykonávanie hlavnej programovej slučky. Pokračovanie vykonávania hlavnej slučky môže zahájiť buď zmena na USB zbernici, prijatie nových dát, alebo keď sú k dispozícii nové dáta z analógových vstupov.

5.1.1 Inicializácia

Inicializácia modulu prebieha v niekoľkých po sebe idúcich krokoch:

1. Nastavenie napätia jadra – od veľkosti napätia jadra závisí maximálna taktovacia frekvencia MCU a tým aj výkon niektorých periférii
2. Inicializácia vstupov a výstupov MCU
3. Inicializácia hodín – Spustenie kryštálových oscilátorov, nastavenie MCLK[1] a SMCLK[2] na 20MHz a ACLK[3] na 32768Hz.
4. Inicializácia RTC a backup napájania
5. Nastavenie USB rozhrania
6. Inicializácia digitálnej časti modulu
7. Inicializácia analógovej časti modulu

Počas inicializácie prebieha prvý pokus o nadviazanie USB spojenia (enumeráciu).

5.1.2 USB komunikácia

Modul pracuje ako trieda Communication Device Class (ďalej len CDC). MSP430 používajúce CDC API po pripojení do hostiteľského systému vytvorí na ňom virtuálny COM port. COM porty sú populárny jednoduchý softwarový mechanizmus prostredníctvom ktorého môže hostiteľský systém komunikovať s periférjami. Aj keď bol tento mechanizmus pôvodne navrhnutý pre sériové RS232 porty, je aj dnes často využívaný ďalšími protokolmi, napr. USB alebo Bluetooth. Keďže sa porty RS232 už prakticky nepoužívajú, sú tieto COM porty nazývané Virtuálne COM porty.

Trieda CDC je jedna zo štandardných USB tried protokolov a je natívne podporovaná väčšinou operačných systémov.

Toto API podporuje len časť CDC špecifikácie, pretože rozsah CDC špecifikácie siaha výrazne ďalej než Virtuálne COM porty, zahrňujúc širokú škálu telekomunikačného vybavenia[11].

Identifikácia zariadenia

- Vendor ID (VID): 0x2047
- Product ID (PID): 0x3ee

[1] Master Clock – hodiny pre CPU a niektoré periférie
[2] Subsystem Master Clock – hodiny pre niektoré periférne moduly
[3] Auxiliary Clock – pomocné hodiny, využívané niektorými periférjami

- Názov zariadenia (Product string): USB UniModule
- Výrobca (Vendor String): Ivan Hanus
- Trieda (Class): Ports
- Max. spotreba z host systému: 500mA

VID a PID nemám registrované.

Komunikačný stack

Komunikačný stack a niektoré nastavenia MCU (špecifiká potrebné pre správnu činnosť USB) sú prevzaté z USB Developer Package[11], ktorú poskytuje Texas Instruments ku všetkým MCU vybaveným USB modulom. Balík okrem zdrojových kódov poskytuje aj nástroj Descriptor Tools, ktorý jednak umožňuje jednoduché nastavenie niektorých špecifických nastavení pre USB modul (trieda zariadenia, identifikácia, spotreba, ...), vygeneruje konfiguračné súbory pre firmware (descriptory, eventy pre príslušnú triedu zariadenia, ...) a identifikačné súbory pre PC (aby bol operačný systém PC schopný prideliť k modulu správny ovládač).

Všetky aspekty USB komunikácie obstarávajú funkcie z Developer Package, doplnené a upravené na potreby tejto aplikácie.

5.1.3 Štruktúra projektu

Projekt bol vytvorený vo vývojovom prostredí Code Composer Studio 5.5.0 od spoločnosti Texas Instruments.

Podpriečinok `drivelib` obsahuje základné knižnice pre periférie MCU. Pre svoju činnosť ich vyžaduje USB API, ja z nich využívam len funkciu na nastavenie clock systému. Podpriečinok `USB_API`obsahuje samotnú USB API, nezasahoval som doň. Podpriečinok `USB_app` obsahuje ďalšie súbory vyžadované API. Nezasahoval som do nich, okrem súboru `usbEventHandling.c`, kde som doplnil, čo sa má prí niektorých udalostiach zbernice stať. Podpriečinok `USB_config` obsahuje súbory konfigurujúce identifikáciu zariadenia (descriptory) vygenerované nástrojom Descriptor Tools[11] z USB Developer Package.

Súbor `main.c` obsahuje hlavnú programovú slučku vykonávajúcu hrubé spracovanie prijatých dát a odosielanie navzorkovaných dát z analógových vstupov. Súbory `hal.c/.h` obsahujú funkciu nastavujúcu hodiny MCU. Súbory `ports.c/.h` obsahujú nastavenie GPIO[4] MCU, funkcie súvisiace s RTC a s ovládacími tlačidlami a indikátormi. Súbory `DIO.c/.h` obsahujú funkcie ovládajúce digitálne I/O modulu. Súbory `analog.c/.h` obsahujú funkcie ovládajúce analógové vstupy a výstupy.

[4]zjednodušene, vstupno-výstupných pinov MCU

5.2 Nastavenia a módy činnosti modulu

Modul umožňuje viaceré nastavenia a módy činnosti pre jednotlivé bloky. Konkrétne postupy nastavenia sú rozobraté ďalej v práci.

5.2.1 Digitálne vstupy a výstupy

Blok digitálnych vstupov a výstupov umožňuje nastaviť napäťovú úroveň v rozsahu od 1.25V po 5V. Z katalógového listu obvodu AD5241 vieme, že pre $100k\Omega$ variant obvodu určíme odpor medzi bežcom a terminálom A podľa vzťahu 5.1 a medzi bežcom a terminálom B podľa vzťahu 5.2.

$$R_{WA} = \frac{256 - n}{256} * 100[k\Omega] \tag{5.1}$$

$$R_{WB} = \frac{n}{256} * 100[k\Omega] \tag{5.2}$$

Dosadením vzťahov 5.1 a 5.2 do vzťahu 4.1 aplikovanom na tento modul, výstupné napätie sa nastaví približne podľa vzťahu 5.3, kde n je celé číslo v rozsahu 0 až 192 (pre väčšie n je už výstupné napätie na úrovni vstupného napätia meniča a výstup sa viac nemení).

$$U_{DIO} = 0.5 * \frac{R_{WA} + R_{WB} + 150}{R_{WA}} = \frac{320}{256 - n}[V] \tag{5.3}$$

Ďalej je možné pre 4 a 4 bity nezávisle nastaviť smer prenosu dát.

Sériová linka z modulu USCI_A je nakonfigurovaná ako UART s rýchlosťou 115,2kb/s, bez parity s jedným stop bitom. V súčasnej verzii funguje len ako echo[5].

5.2.2 Analógové výstupy

Veľkosť výstupného napätia proti zemi je daná vzťahom 5.4, kde 3.3 je dané zvolenou vrchnou referenciou DAC (v súčasnej konfigurácii napájacím napätím), 4096 je dané zvoleným rozlíšením DAC (v súčasnej konfigurácii 12bitov, DAC umožňuje aj prácu v 8bitovom režime, kde je namiesto 4096 hodnota 256) a x je hodnota zapísaná do registra DAC.

$$U_{DAC} = 3.3 * \frac{x}{4096}[V] \tag{5.4}$$

Analógové výstupy môžu pracovať v dvoch režimoch:

1. Ručné nastavenie vzorku po vzorke
2. Kontinuálny mód

[5] Posiela naspäť to, čo prijal

Ručné nastavenie vzorku po vzorke

V tomto režime užívateľ posiela jednotlivé vzorky a modul ich okamžite nastaví na výstup. Tento režim je vhodný statické nastavenia alebo pre pomalé priebehy.

Kontinuálny mód

V tomto režime užívateľ nahrá do modulu navzorkovaný priebeh (s rozlíšením 12bitov). Súčasnej konfigurácii má modul pamäť na 1024 vzoriek pre každý kanál. Vzorkovanie je riadené časovačom. V tomto režime sa využívajú dva DMA[6] kanály (každý výstup má svoj DMA kanál). DAC v MSP430 má jadro s dvojitým bufferovaním [7], pričom pri ručnom režime je hodnota vstupu zapísaná priamo do pracovného registra jadra. V kontinuálnom móde je zápis zo záchytného registra do jadra riadený časovačom. Keď modul dostane signál z časovača, hodnota je zo záchytného registra zapísaná do pracovného registra a nastavený výstup. Po skončení tohto zápisu sa spustí DMA prenos, ktorý zapíše novú hodnotu z pamäte do vstupného záchytného registra. Pri ďalšom signály z časovača sa tento proces opakuje. Vzorkovacia frekvencia je daná vzťahom 5.5, kde $20 * 10^6$ je daná vstupnou frekvenciou časovača a n je 16bitové neznamienkové číslo (rozsah <0:65535>), čo umožňuje nastaviť nastaviť vzorkovaciu frekvenciu v rozsahu približne 305Hz až po teoreticky 20MHz, avšak výrobca udáva pre malé zmeny maximálnu frekvenciu približne 1MHz.

$$f_{vz} = \frac{20 * 10^6}{n + 1}[Hz] \tag{5.5}$$

5.2.3 Analógové vstupy - Analógová časť

Implementované sú všetky možnosti, ktoré umožňuje hardware, tj. prepínanie napäťových rozsahov, prepínanie jednosmernej a striedavej väzby, pripnutie napäťovej referencie na vstup reťazca, prepínanie hornopriepustných filtrov a ladenie dolnopriepustného filtra.

Ladenie dolnopriepustného filtra

Vzhľadom na obvodové zapojenie, skombinovaním vzťahom 5.1 a 4.6 získame vzťah 5.6 na výpočet medznej frekvencie filtra, kde n je celé číslo v rozsahu približne od 154 do 246 (v tomto rozsahu sa pohybujú hodnoty odporu digitálneho potenciometra v odporúčanom rozsahu pre obvod LTC1569), čo umožňuje nastaviť medzný kmitočet od približne 1kHz do približne 64kHz. Mimo týchto hodnôt výrobca negarantuje vlastnosti obvodu.

$$f_{medz} = \frac{1638400}{(256 - n) * div}[Hz] \tag{5.6}$$

[6]Direct Memory Access – Priamy prístup do pamäte – možnosť pristupovať perifériám do pamäte nezávisle od CPU

[7]Medzi pracovným registrom jadra a vstupom je ešte jeden záchytný register

5.2.4 Analógové vstupy - ADC

Keďže MCU má len jedno ADC jadro s prepínaním vstupov, prebieha vzorkovanie a prevod postupne, najprv kanál 0, potom kanál 1. Čas vzorkovania je 6.4μs, konverzia trvá 2.6μs (oba časy platia pre každý kanál). ADC môže pracovať v dvoch režimoch:

1. Jednotlivé vzorkovanie
2. Kontinuálne vzorkovanie

Režim sa dá nastaviť nezávisle pre oba kanály. Ale v prípade, že je jeden kanál v kontinuálnom režime, interne idú oba kanály v kontinuálnom režime. Líšia sa len v tom, kedy a koľko dát je odoslaných nadradenému systému.

Jednotlivé vzorkovanie

Pri jednotlivom vzorkovaní začne prebiehať najprv proces vzorkovania a prevodu na číslo pre kanál 0 a následne pre kanál 1. Po skončení konverzie druhého kanálu sú dáta odoslané do nadradeného systému. V prípade, že je jeden z kanálov v kontinuálnom móde, po externom signály na vzorkovanie sú odoslané posledné získané vzorky z oboch kanálov.

Kontinuálne vzorkovanie

Časovanie vzorkovania je riadené časovačom. Vzorkovacia frekvencia sa určí podľa vzťahu 5.7, kde n je celé číslo v rozsahu od 8 do 65535, čo umožňuje nastavenie vzorkovacej frekvencie približne od 5Hz do približne 34kHz (vyššie frekvencie nie sú možné vzhľadom na čas vzorkovania a konverzie - 18μs a ďalšiu réžiu súvisiacu s ADC, vyžadovali by zníženie času vzorkovania, čo by, vzhľadom na kapacity v obvode, nebolo moc dobré). Vzorkovacia frekvencia je spoločná pre oba kanály.

$$f_{vz} = \frac{312500}{n+1}[Hz] \tag{5.7}$$

Pre každý kanál sú vyhradené dva buffere, každý o veľkosti 1022 vzoriek. Dva sú z dôvodu zachovania kontinuity vzorkovania v čase odosielania. Po získaní novej vzorky je táto vzorka uložená do aktívneho buffera. Keď je v buffery zvolené množstvo vzoriek, za aktívny sa prepne druhý buffer (tj. nové vzorky sú ukladané doňho) a dáta z naplneného buffera sú medzičasom odosielané nadradenému systému.

5.2.5 Predvolené nastavenia

Po spustení má modul nasledujúce nastavenia:

- Všetky indikačné led zhasnuté
- Digitálne vstupy a výstupy
  - všetky nastavené ako výstup v log. 0
  - napäťová úroveň približne 3.3V
- Analógové výstupy

  - Ručný mód
  - Oba výstupy 0V
- Analógové vstupy
  - DC väzba, rozsah ±1V, vypnuté filtre
  - Jednotlivé vzorkovanie
  - Buffer 160 vzoriek

5.3 Komunikačný protokol

Mojim cieľom bolo spraviť komunikačný protokol, ktorý súčasne bude čo najmenej vyťažovať linku, bude implementačne čo najjednoduchší a bude prehľadný.

Ako najjednoduchší spôsob komunikácie sa mi javí použitie správ, v ktorých prvý byte obsahuje operačný kód a zvyšok sú dáta, v ktorých sú zakódované údaje (tab. 5.1). Podľa operačného kódu systém vie, čo a v akom formáte správa obsahuje. Pričom, všetky správy nemusia obsahovať dáta a konkrétne zloženie správy sa môže líšiť od smeru prenosu správy (iný obsah pri smere do a z MCU).

1. Byte	Ostatné byty
Operačný kód	Dáta

Tab. 5.1: Formát správy

Nasleduje zoznam a presný formát dát pre jednotlivé operačné kódy.

5.3.1 Systémové správy

Operačné kódy od 0 do 3 využívajú správy, ktoré nesúvisia s I/O funkcionalitou modulu (nastavenie a zistenie času v module, zistenie alebo nastavenie stavu indikačných a ovládacích prvkov modulu).

Operačný kód 0 - GetTime – slúži na zistenie času modulu. Formát správy je v tab. 5.2.Údaje sú v binárnom formáte. Smerom do MCU stačí odoslať operačný kód, MCU vráti správu v tomto formáte.

1. Byte	2. Byte	3. Byte	4. Byte
Hodiny	Minúty	Sekundy	Deň v týždni
5. Byte	6. Byte	7. Byte	8. Byte
Deň	Mesiac	Rok - MSB	Rok - LSB

Tab. 5.2: Formát správy GetTime a SetTime

Operačný kód 1 - SetTime – slúži na nastavenie času modulu. Formát správy je v tab. 5.2.Údaje sú v binárnom formáte. Modul vráti údaje, ktoré prijal.

Operačný kód 2 - GetControls – slúži na zistenie stavu indikátorov a ovládacích modulu. Formát správy je v tab. 5.3. Smerom do MCU stačí odoslať operačný kód, MCU vráti správu v tomto formáte. Značenie jednotlivých bitov sedí s označením na DPS.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	S1	X	L1	L2	L3	L4	X	S2

Tab. 5.3: Formát správy GetSetControls a SetControls

Operačný kód 3 - SetControls – slúži na nastavenie indikačných LED modulu. Formát správy je v tab. 5.2. Modul vráti údaje, ktoré prijal.

5.3.2 Digitálne vstupy a výstupy

Operačné kódy 4 až 9 slúžia na ovládanie digitálnych vstupov a výstupov.

Operačný kód 4 - GetIOVoltage – slúži na zistenie napäťovej úrovne digitálnych I/O. Formát správy je v tab. 5.4. Smerom do MCU stačí odoslať operačný kód, MCU vráti správu v tomto formáte. Napätia sa dá prepočítať podľa vzťahu 5.3.

1. Byte
Napätie

Tab. 5.4: Formát správy GetIOVoltage a SetIOVoltage

Operačný kód 5 - SetIOVoltage – slúži na nastavenie napäťovej úrovne digitálnych I/O modulu. Formát správy je v tab. 5.4. Modul vráti údaje, ktoré prijal.

Operačný kód 6 - GetIODirection – slúži na zistenie smeru digitálnych I/O. Formát správy je v tab. 5.5. Smerom do MCU stačí odoslať operačný kód, MCU vráti správu v tomto formáte. Ak sú všetky bity v LSB/MSB Smer v log. 0, daná časť je nastavená ako vstup, inak výstup.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	LSB Smer				MSB Smer			

Tab. 5.5: Formát správy GetIODirection a SetIODirection

Operačný kód 7 - SetIODirection – slúži na nastavenie smeru digitálnych I/O modulu. Formát správy je v tab. 5.5. Modul vráti údaje, ktoré prijal. Ak sú všetky bity v LSB/MSB Smer v log. 0, daná časť je nastavená ako vstup, inak výstup.

Operačný kód 8 - GetIO – slúži na zistenie stavu digitálnych I/O. Formát správy je v tab. 5.6. Smerom do MCU stačí odoslať operačný kód, MCU vráti správu v tomto formáte.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	IO1	IO2	IO3	IO4	IO5	IO6	IO7	IO8

Tab. 5.6: Formát správy GetIO a SetIO

Operačný kód 9 - SetIO – slúži na nastavenie stavu digitálnych I/O modulu. Formát správy je v tab. 5.6. Modul vráti údaje, ktoré prijal.

5.3.3 Analógové vstupy

Operačné kódy 20 až 39 slúžia na prácu s analógovými vstupmi.

Operačný kód 20 - GetAnInConfig – slúži na zistenie nastavení analógovej časti analógových vstupov. Formát správy je v tab. 5.7. Smerom do MCU je nutné odoslať aspoň prvý byte správy, aby MCU vedelo odoslať nastavenia správneho kanálu. MCU vráti správu v tomto formáte.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Kanál	Kalib.	Väzba	Rozsah	HPF		DPF - Rozsah	
2	Medzná frekvencia							

Tab. 5.7: Formát správy GetAnInConfig a SetAnInConfig

Význam jednotlivých bitov:

- Kanál – voľba kanálu
- Kalib – 1 - Pripnutie nulovej úrovne na vstup kanálu
- Väzba – 1 - Zapnutá DC väzba, 0 - AC väzba
- Rozsah – 0 - rozsah ±0.2V, 1 - rozsah ±1V
- HPF – Hornopriepustný filter:
  - 0 – Vypnutý
  - 1 – 1Hz
  - 2 – 100Hz
  - 3 – Nevyužitý filter
- DPF Rozsah – Rozsah dolnopriepustného filtra:
  - 0 – delič 1
  - 1 – delič 4
  - 2 – delič 16
  - 3 – Vypnutý

- Medzná frekvencia – V kombinácii s predchádzajúcim bodom sa určí medzná frekvencia podľa vzťahu 5.6.

Operačný kód 21 - SetAnInConfig – slúži na nastavenie analógovej časti analógových vstupov. Formát správy je v tab. 5.7. Modul vráti údaje, ktoré prijal.

Operačný kód 22 - GetCalibration – slúži na zistenie stavu zapnutia kalibrácie (pripnutie nulovej úrovne na vstup). Formát správy je v tab. 5.8. Smerom do MCU je nutné odoslať celú správu, aby MCU vedelo odoslať nastavenia správneho kanálu. MCU vráti správu v tomto formáte. Ak sú všetky bity časti Kalibrácia v log. 0, je režim kalibrácie vypnutý.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Kanál	X			Kalibrácia			

Tab. 5.8: Formát správy GetCalibration a SetCalibration

Operačný kód 23 - SetCalibration – slúži na nastavenie stavu zapnutia kalibrácie. Formát správy je v tab. 5.8. Modul vráti údaje, ktoré prijal.

Operačný kód 24 - GetCoupling – slúži na zistenie zvolenej väzby vstupu. Formát správy je v tab. 5.9. Smerom do MCU je nutné odoslať celú správu, aby MCU vedelo odoslať nastavenia správneho kanálu. MCU vráti správu v tomto formáte. Ak sú všetky bity časti Väzba v log. 0, je zvolená AC väzba.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Kanál	X			Väzba			

Tab. 5.9: Formát správy GetCoupling a SetCoupling

Operačný kód 25 - SetCoupling – slúži na nastavenie väzby vstupu. Formát správy je v tab. 5.9. Modul vráti údaje, ktoré prijal.

Operačný kód 26 - GetRange – slúži na zistenie zvoleného nap. rozsahu vstupu. Formát správy je v tab. 5.10. Smerom do MCU je nutné odoslať celú správu, aby MCU vedelo odoslať nastavenia správneho kanálu. MCU vráti správu v tomto formáte. Ak sú všetky bity časti Rozsah v log. 0, je zvolený rozsah ±0.2V, inak ±1V.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Kanál	X			Rozsah			

Tab. 5.10: Formát správy GetRange a SetRange

Operačný kód 27 - SetRange – slúži na nastavenie rozsahu vstupu. Formát správy je v tab. 5.9. Modul vráti údaje, ktoré prijal.

Operačný kód 28 - GetHPF – slúži na zistenie zvoleného hornopriepustného filtra vstupu. Formát správy je v tab. 5.11. Smerom do MCU je nutné odoslať celú správu, aby MCU vedelo odoslať nastavenia správneho kanálu. MCU vráti správu v tomto formáte.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Kanál	X					HPF	

Tab. 5.11: Formát správy GetHPF a SetHPF

HPF – Hornopriepustný filter:

- 0 – Vypnutý
- 1 – 1Hz
- 2 – 100Hz
- 3 – Nevyužitý filter

Operačný kód 29 - SetHPF – slúži na výber hornopriepustného filtra vstupu. Formát správy je v tab. 5.11. Modul vráti údaje, ktoré prijal.

Operačný kód 30 - GetLPF – slúži na zistenie nastavenia dolnopriepustného filtra vstupu. Formát správy je v tab. 5.12. Smerom do MCU je nutné poslať aspoň prvý byte správy, aby MCU vedelo odoslať nastavenia správneho kanálu. MCU vráti správu v tomto formáte.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Kanál	X					LPF Rozsah	
2	Medzná frekvencia							

Tab. 5.12: Formát správy GetLPF a SetLPF

Význam jednotlivých bitov:

- DPF Rozsah – Rozsah dolnopriepustného filtra:
  - 0 – delič 1
  - 1 – delič 4
  - 2 – delič 16
  - 3 – Vypnutý
- Medzná frekvencia – V kombinácii s predchádzajúcim bodom sa určí medzná frekvencia podľa vzťahu 5.6.

Operačný kód 31 - SetLPF – slúži na nastavenie dolnopriepustného filtra vstupu. Formát správy je v tab. 5.12. Modul vráti údaje, ktoré prijal.

Operačný kód 35 - SetADCConfig – slúži na nastavenia ADC modulu. Formát správy je v tab. 5.13. Modul vráti údaje, ktoré prijal.

Význam jednotlivých bitov/bytov:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	ADC0 mód				ADC1 mód			
2	Vzorkovacia frekvencia MSB							
3	Vzorkovacia frekvencia LSB							
4	Veľkosť bufferu MSB							
5	Veľkosť bufferu LSB							

Tab. 5.13: Formát správy GetADCConfig a SetADCConfig

- ADCx mód – Mód ADC - ak sú všetky bity 0 - mód jednotlivého vzorkovania, inak kontinuálne vzorkovanie
- Vzorkovacia frekvencia MSB/LSB– byty tvoriace vzorkovaciu frekvenciu určenú podľa vzťahu 5.7.
- Veľkosť bufferu MSB/LSB – byty tvoriace veľkosť bufferov ADC

Operačný kód 36 - GetADCConf – slúži na zistenie nastavenie ADC modulu. Formát správy je v tab. 5.13.Smerom do MCU stačí odoslať operačný kód. MCU vráti správu v tomto formáte.

Operačný kód 37 - Trigger – slúži ako spúšť pre vzorkovanie ADC (popísané vyššie). Smerom do MCU stačí odoslať operačný kód, MCU vráti aktuálne vzorky signálu (operačný kód 38). Tento operačný kód sa nepoužíva v smere z modulu.

Operačný kód 38 - ADCSingleData – slúži na posielanie jednotlivých vzoriek z MCU, tj. tento operačný kód sa nepoužíva v smere do MCU. Formát správy je v tab. 5.14. Prvý byte je bez významu, slúži len ako výplň na zarovnanie dát v pamäti

Byte	1	2	3	4	5
Obsah	výplň	ADC0 LSB	ADC0 MSB	ADC1 LSB	AD1 MSB

Tab. 5.14: Formát správy ADCSingleData

MCU. Za ním nasledujú 2 a 2 byty, v ktorých sú uložené hodnoty jednotlivých vstupov. Keďže MSP430 je typu little endian, menej významný byte ide prvý.

Operačný kód 39 - ADCData – slúži na posielanie balíku vzoriek z jedného kanálu z MCU, tj. tento operačný kód sa nepoužíva v smere do MCU. Formát správy je v tab. 5.15. Prvý byte obsahuje kanál, z ktorého sú dáta. Byty 2 a 3 obsahujú dĺžku

Byte	1	2	3	4	5	6	...
Obsah	Kanál	dĺžka LSB	dĺžka MSB	D0 LSB	D0 MSB	D1 LSB	...

Tab. 5.15: Formát správy ADCData

balíka dát. V ďalších bytoch sú už samotné vzorky. Všetky viacbytové dáta sú opäť typu little endian.

5.3.4 Analógové výstupy

Operačné kódy od čísla 50 slúžia na obsluhu analógových výstupov.

Operačný kód 50 - DACWriteSingle – slúži na zápis jednej hodnoty na výstup DAC. Formát správy je v tab, 5.16. Modul vráti údaje, ktoré prijal. Napätie na výstupe sa dá vypočítať podľa vzťahu 5.4.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Kanál	X			Dáta MSB			
2	Dáta LSB							

Tab. 5.16: Formát správy DACWriteSingle a DACRead

Operačný kód 51 - DACRead – slúži na prečítanie aktuálnej hodnoty na výstupe daného kanálu. Formát správy je v tab.5.16. Do modulu stačí poslať prvý byte, v ktorom je číslo kanálu, ktorého výstup nás zaujíma. Modul vráti kompletnú správu.

Operačný kód 52 - SetDACConfig – slúži na nastavenie módov DAC modulu. Formát správy je v tab.5.17. Modul vráti prijaté údaje. Ak je bit DAC mód v log.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Kanál	X						DAC mód
2	Vzorkovacia frekvencia MSB							
3	Vzorkovacia frekvencia LSB							

Tab. 5.17: Formát správy SetDACConfig a GetDACConfig

0, modul je nastavený v móde ručného nastavenia. Inak je v kontinuálnom móde. Vzorkovacia frekvencia sa určí podľa vzťahu 5.5.

Operačný kód 53 - GetDACConfig – slúži na zistenie nastavení módu DAC modulu. Formát správy je v tab.5.17. Do modulu stačí poslať prvý byte s nastaveným požadovaným kanálom. Modul vráti kompletnú správu.

Operačný kód 54 - DACWriteData – slúži na nahratie priebehu pre daný výstupný kanál do modulu. Formát správy je v tab. 5.18.

Najvyšší bit prvého bytu obsahuje voľbu kanálu. Druhý a tretí byte obsahuje počet vzoriek. Ďalšie byty obsahujú samotné vzorky. Vo všetkých viacbytových dátach ide prvé LSB. Modul vracia len prvé tri byty.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	Kanál	X						
2	Dĺžka LSB							
3	Dĺžka MSB							
4	Vzorka0 LSB							
5	Vzorka0 MSB							
6	Vzorka1 LSB							
...	...							

Tab. 5.18: Formát správy DACWriteData

5.4 Prvé pripojenie modulu k PC

Pri prvom pripojení modulu k PC je nutné operačnému systému poskytnúť súbor identifikujúci modul (`MSP430_CDC.inf`), pomocou ktorého vie modulu priradiť správny ovládač.

Pre OS Windows 7 a staršie, po výzve systému na inštalovanie ovládačov, treba zvoliť ručnú inštaláciu, počas ktorej treba vybrať `.inf` súbor. Zvyšok OS vykoná sám. Následne by sa modul mal objaviť v Správcovi zariadení v záložke Porty.

Pri Windows 8 je situácia trošku zložitejšia, keďže systém defaultne neumožňuje nainštalovanie digitálne nepodpísaných ovládačov a okrem administrátorských práv môže vyžadovať aj bezpečnostný kód systému. Operačný systém treba reštartovať do režimu Advanced startup, v ňom zvoliť Troubleshoot -> Advanced Options -> Startup Settings a tam zvoliť možnosť Disable Driver Signature Enforcement. Ďalej je postup rovnaký ako pri Win7.

Na iných operačných systémoch som modul neskúšal.

6 OBSLUŽNÁ KNIŽNICA A DEMONŠTRAČNÝ SOFTWARE

Ako som už spomínal skôr v práci, namiesto tvorby ovládača k modulu komplet v LabVIEW som sa rozhodol pre vytvorenie knižnice (resp. triedy zapuzdrenej v knižnici) v jazyku C#. Má to viacero výhod, jednou z hlavných je to, že modul nebude viazaný na použitie s prostredím LabVIEW (príp. jeho Runtime Engine), pre chod mu bude stačiť počítač s nainštalovaným .NET Framework 4.5 (v systémoch Windows 8 a novších natívne, minimálne Windows Vista SP2), prípadne by nemal byť problém knižnicu prerobiť na niektorú trochu staršiu verziu .NET Frameworku. Knižnica nie je použiteľná na operačných systémoch iných výrobcov.

Knižnica bola vytvorená vo vývojovom prostredí Microsoft Visual Studio 2013.

6.1 Trieda `module`

Názov triedy je `module` a uložená je v knižnici `UniModule.dll`. Jej jadro tvorí trieda `SerialPort`, ktorá je využitá na komunikáciu s virtuálnym COM portom vytvoreným pri pripojení modulu k host systému.

6.1.1 Verejné parametre

Pri použití triedy `module` má užívateľ prístup k niekoľkým verejným parametrom.

`bool DAC0Loaded` a `bool DAC1Loaded`

Tieto parametre sa nastavia po úspešnom nahraní balíka dát pre daný DAC.

`bool ADC0_dr` a `bool ADC1_dr`

Tieto parametre indikujú prijatie nového balíka dát z daného ADC (v kontinuálnom móde). Užívateľ ich môže zmazať, aby vedel, že už daný prijatý balík dát spracoval. Príznaky sa nastavia po každom prijatí balíka dát.

6.1.2 Verejné metódy

Keďže LabVIEW by asi malo problém so spracovaním návratovej hodnoty v podobe štruktúry alebo triedy, sú v prípade viacerých metód zložité návratové hodnoty zakódované v základných dátových typoch, prípadne poliach týchto typov. Metódy typu Get odošlú požiadavku modulu, ak nedostanú v požadovanom čase odpoveď, vrátia poslednú známu hodnotu. Užívateľ môže využívať nasledujúce verejné metódy:

`module()`

Implicitný konštruktor triedy. Vytvára inštanciu triedy `module` a je nutné ho zavolať pred prvým použitím hociktorej metódy alebo prístupom k hociktorému parametru (pred zavolaním konštruktora fakticky neexistujú).

`~module()`

Deštruktor triedy. Je automaticky zavolaný po ukončení používania triedy. Ak je aktívne spojenie s virtuálnym COM portom, ukončí ho a uvoľní port pre použitie.

`bool Connect(string name)`

Táto metóda slúži na nadviazanie spojenia s modulom. Parameter `name` musí obsahovať platný názov virtuálneho COM portu (zistiteľný napr. v Správcovi zariadení). Metóda nadväzuje spojenie s virtuálnym COM portom daného názvu. Ak bolo spojenie úspešne nadviazané alebo spojenie s portom daného názvu už existuje, funkcia vráti `true`. V prípade, že trieda má nadviazané spojenie s portom odlišného názvu, toto spojenie sa ukončí a metóda sa pokúsi pripojiť na zadaný virtuálny COM port. Pri neúspechu pri nadviazaní spojenia, funkcia vráti `false`.

`void Disconnect()`

Metóda ukončí aktívne spojenie triedy s virtuálnym COM portom.

`bool IsConnected()`

Metóda zisťuje, či má trieda aktívne spojenie s niektorým virtuálnym COM portom. Ak áno, vráti `true`, inak `false`.

`string GetTime()`

Metóda vráti čas a dátum v module v reťazci vo formáte „hodiny:minúty:sekundy deň_v_týždni deň.mesiac.rok“.

`void SetLocalTime()`

Metóda nastaví modulu aktuálny lokálny čas hostujúceho systému.

`byte GetControls()`

Metóda vráti stav ovládacích prvkov (tlačidiel) a indikátorov modulu. Formát návratovej hodnoty je v tab.5.3.

`void SetControls(byte data)`

Metóda nastaví indikátory (LED) modulu. Formát vstupných dát je v tab.5.3.

`void SetControls(bool LED1, bool LED2, bool LED3, bool LED4)`

Oproti predchádzajúcemu variantu, táto metóda umožňuje nastaviť indikátory priamo, bez nutnosti formátovania do špeciálneho tvaru.

`byte GetIOVoltageRaw()`

Metóda vráti napäťovú úroveň digitálnych I/O v neupravenom tvare, na napätie prepočítateľná podľa vzťahu 5.3.

`double GetIOVoltage()`

Metóda vráti napäťovú úroveň digitálnych I/O.

`void SetIOVoltage(byte data)`

Metóda nastaví napäťovú úroveň digitálnych I/O, vstupné dáta sú v neupravenom tvare, tj. na napätie prepočítateľné podľa vzťahu 5.3.

`void SetIOVoltage(double voltage)`

Metóda nastaví napäťovú úroveň digitálnych I/O.

`byte GetIODirection()`

Metóda vráti nastavenie smeru digitálnych I/O. Dáta sú zakódované podľa tab.5.5.

`void SetIODirection(byte data)`

Metóda nastaví smer digitálnych I/O. Dáta sú zakódované podľa tab.5.5.

`void SetIODirection(bool msb_dir, bool lsb_dir)`

Metóda nastaví smer digitálnych I/O. Tento variant umožňuje nastaviť smer priamo, bez nutnosti kódovania. Ak je odpovedajúci parameter `true`, daný kanál je nastavený ako výstup. Inak vstup.

`byte GetIO()`

Metóda vráti stav I/O. Dáta sú zakódované podľa tab.5.6.

`void SetIO(byte data)`

Metóda nastaví I/O. Dáta sú zakódované podľa tab.5.6.

`void SetIO(bool d8, bool d7, bool d6, bool d5, bool d4, bool d3, bool d2, bool d1)`

Metóda nastaví I/O priamo, bez nutnosti kódovania.

`byte[] GetAnInConfig(ushort input)`

Metóda vráti nastavenia analógovej časti analógových vstupov. Dáta sú zakódované podľa tab. 5.7.

`void SetAnInConfig(byte input, byte data, byte freq)`

Metóda nastaví analógovú časť analógových vstupov. Parameter `input` udáva, o ktorý kanál sa jedná. Parametre `data` a `freq` sú zakódované podľa prvého (resp. druhého) bytu z tab.5.7.

`void SetAnInConfig(byte input, bool cal, bool coupling, bool range, byte HP, byte LP, byte LP_Freq)`

Tento variant umožňuje nastavenie parametrov analógovej časti analógových vstupov bez nutnosti kódovania.

`bool GetAnInCal(byte input)`

Metóda vráti stav nastavenia kalibračného módu daného vstupu (`true` - zapnuté, `false` - vypnuté).

`void SetAnInCal(byte input, bool calibration)`

Metóda nastavení kalibrační mód daného vstupu (`true` - zapnuté, `false` - vypnuté).

`bool GetAnInCoupling(byte input)`

Metóda zistí väzbu vstupu daného kanálu (`true` - jednosmerná väzba, `false` - striedavá väzba).

`void SetAnInCoupling(byte input, bool coupling)`

Metóda nastaví väzbu daného kanálu (`true` - jednosmerná väzba, `false` - striedavá väzba).

`bool GetAnInRange(byte input)`

Metóda zistí, aký má daný kanál zvolený napäťový rozsah (`true` - $\pm$1V, `false` - $\pm$0.2V).

`void SetAnInRange(byte input, bool range)`

Metóda nastaví danému kanálu požadovaný napäťový rozsah (`true` - $\pm$1V, `false` - $\pm$0.2V).

`byte GetAnInHP(byte input)`

Metóda zistí aktuálne zvolený hornopriepustný filter pre daný kanál (0 - vypnuté, 1 - 1Hz, 2 - 100Hz, 3 - nepoužité)

`void SetAnInHP(byte input, byte HP)`

Metóda nastaví zvolený hornopriepustný filter pre daný kanál (0 - vypnuté, 1 - 1Hz, 2 - 100Hz, 3 - nepoužité)

`byte[] GetAnInLP(byte input)`

Metóda zistí aktuálne nastavenie dolnopriepustného filtra pre zvolený kanál. Formát návratovej hodnoty je podľa tab.5.12.

`double GetLPFFreq(byte input)`

Metóda vráti medzný kmitočet dolnopriepustného filtra pre zvolený kanál. Ak je návratová hodnota 0, filter je vypnutý.

`void SetAnInLP(byte input, byte LP_range, byte LP_freq)`

Metóda nastaví dolnopriepustný filter zvoleného kanála. Formát dát je v tab.5.12, kde prvý byte odpovedá parametru `LP_range` a druhý byte parametru `LP_freq`.

`void SetLPFFreq(byte input, double freq)`

Metóda nastaví dolnopriepustný filter daného kanálu na približne[1] zvolenú frekvenciu. Ak užívateľ zvolí frekvenciu menšiu než 1kHz, program zvolí frekvenciu približne 1kHz, ak zvolí frekvenciu vyššiu než 64kHz, program nastaví približne 64kHz [2]. Voľba frekvencie 0 znamená vypnutie filtra.

`byte[] GetADCConfig()`

Metóda vráti nastavenia ADC modulu. Formát vrátených dát je v tab.5.13.

`void SetADCConfig(bool adc0_mode, bool adc1_mode, ushort SR, ushort buff_size)`

Metóda nastaví ADC modulu. Parameter `adc0_mode` (resp. `adc1_mode`) v `true` prepne odpovedajúci kanál na kontinuálny mód, vo `false` na mód jednotlivého vzorkovania. Parameter `SR` udáva vzorkovaciu frekvenciu (prepočítateľnú podľa vzťahu 5.7. Parameter `buff_size` udáva veľkosť bufferu na vzorkovanie (max 1000).

[1] Približne, lebo modul nie je schopný prelaďovať filter úplne plynule - dané krokom digitálneho potenciometra

[2] Dané limitmi obvodu LTC1569-6

`public void SetADCConfig(bool adc0_mode, bool adc1_mode, double SR, ushort buff_size)`

Pre tento variant je vzorkovacia frekvencia zadaná priamo. Musí byť v rozsahu[3] <4.8:44642>. Frekvencia je nastavená len približne[4].

`void ADCTrigger()`

Metóda zaháji získavanie jednej vzorky. Po zavolaní tejto vzorky modul vráti po vzorke z každého kanálu. Je vhodné, aby po zavolaní tejto metódy bola zavolaná nasledujúca metóda.

`ushort[] ADCGetSingleData()`

Metóda vráti posledné dve prijaté vzorky z oboch kanálov. Je nutné, aby bola pred ňou zavolaná metóda `ADCTrigger`. Dáta obsahujú priamy výstup z ADC. Napätie sa prepočíta podľa vzťahu 6.1 (kde n je výstupná hodnota ADC) pre rozsah ±1V. Pre rozsah ±0.2V treba túto hodnotu vydeliť 5.

$$V_{ADC} = 4.889 * 10^{-4} * n - 1.0002[V] \tag{6.1}$$

`ushort ADCGetSingleData(byte channel)`

Metóda vykonáva to isté ako predchádzajúca s tým rozdielom, že užívateľ vyberá kanál, ktorého hodnotu chce vedieť. Inak pre ňu platí všetko, čo pre predchádzajúcu.

`double[] ADCGetSingleVoltage()`

Metóda vráti posledné vzorky z oboch kanálov prepočítané na napätie. Je nutné, aby bola pred ňou zavolaná metóda `ADCTrigger`. Pre korektný výpočet je nutné, aby bola pred jej prvým použitím zavolaná niektorá z metód zisťujúca alebo nastavujúca napäťový rozsah.

`double ADCGetSingleVoltage(byte channel)`

Metóda vykonáva to isté ako predchádzajúca s tým rozdielom, že užívateľ vyberá kanál, ktorého hodnotu chce vedieť. Inak pre ňu platí všetko, čo pre predchádzajúcu.

`ushort[] ADCGetData(byte channel)`

Metóda vráti posledný prijatý balík neupravených dát zo zvoleného kanálu pri kontinuálnom vzorkovaní. V prípade, že ešte nebol prijatý žiadny balík z modulu,

[3] Limity dané parametrami modulu

[4] Vzhľadom na krok časovača

metóda vráti `null`. Na napätie je nutné ich prepočítať podľa vzťahu 6.1 (podobne ako pri metóde `ADCGetSingleData`).

`double[] ADCGetVoltage(byte channel)`

Metóda vráti posledný prijatý balík dát zo zvoleného kanálu pri kontinuálnom vzorkovaní prepočítaný na napätia. Pre korektný výpočet je nutné, aby bola pred jej prvým použitím zavolaná niektorá z metód zisťujúca alebo nastavujúca napäťový rozsah. V prípade, že ešte nebol prijatý žiadny balík z modulu, metóda vráti `null`.

`void DACWriteSingle(byte channel, ushort data)`

Metóda nastaví na zvolený analógový výstup hodnotu v parametre `data`. Na napätie sa dá prepočítať podľa vzťahu 5.4.

`void DACWriteSingleVoltage(byte channel, double voltage)`

Metóda nastaví na zvolený analógový výstup požadované napätie (voči systémovej zemi; oproti analógovej zemi k nemu treba pripočítať 1.5V. Metóda kontroluje, či je napätie v rozsahu DAC (0-3.3V). V prípade, že by bolo mimo rozsah, nastaví maximum (resp. minimum, podľa toho, čo je bližšie).

`ushort DACGetData(byte channel)`

Metóda zistí aktuálnu hodnotu na zvolenom analógovom výstupe. Vrátená hodnota sa na napätie sa dá prepočítať podľa vzťahu 5.4.

`double DACGetVoltage(byte channel)`

Metóda vráti aktuálnu hodnotu napätia na zvolenom analógovom výstupe.

`byte[] GetDACConfig(byte channel)`

Metóda vráti nastavenia zvoleného analógového výstupu. Výstupné dáta sú vo formáte podľa tab.5.17. Vzorkovacia frekvencia sa dá určiť podľa vzťahu 5.5.

`void SetDACConfig(byte channel, bool contin_mode, ushort sr)`

Metóda nastaví zvolený kanál do požadovaného módu. Keď je parameter `contin_mode` `true`, daný kanál sa nastaví do kontinuálneho módu, inak ručný mód. Parameter `sr` udáva vzorkovaciu frekvencie (prepočítateľnú podľa vzťahu 5.5).

`void SetDACConfig(byte channel, bool contin_mode, double sr)`

V tomto variante sa zadáva vzorkovacia frekvencia priamo v [Hz]. Metóda kontroluje, či je frekvencia v rozsahu $<305:10^7>$ Hz.

`void DACLoad(byte channel,ushort length, byte[] data)`

Táto metóda slúži na nahratie navzorkovaného priebehu do modulu pre daný kanál. Parameter `length` určuje počet vzoriek. Tento variant vyžaduje pripravené vstupné dáta dáta rozdelené na byty systémom Little endian.

`void DACLoad(byte channel, ushort length, ushort[] data)`

Pre tento variant sú vstupné dáta nerozdelené na byty. Na napätie sa dajú prepočítať podľa vzťahu 5.4.

`void DACLoad(byte channel, ushort length, double[] voltage)`

Tento variant má ako vstupné dáta napätie. Metóda kontroluje, či sú dáta v rozsahu DAC.

6.1.3 Odporúčania pri použití knižnice

Odporúčané poradie krokov pri použití knižnice je nasledujúce:

1. Zavolanie konštruktora `module()`
2. Zavolanie metódy `Connect()`, parameter funkcie je názov virtuálneho COM portu. Ak je návratová hodnota `true` pokračovať ďalej, inak skúsiť tento bod znova.
3. Vlastný užívateľský program. V prípade, že by došlo k odpojeniu modulu, je nutné sa vrátiť na bod 2.
4. Pred ukončením programu zavolať metódu `Disconnect()`, ktorá uzavrie virtuálny COM port a uvoľní ho pre ostatné programy.

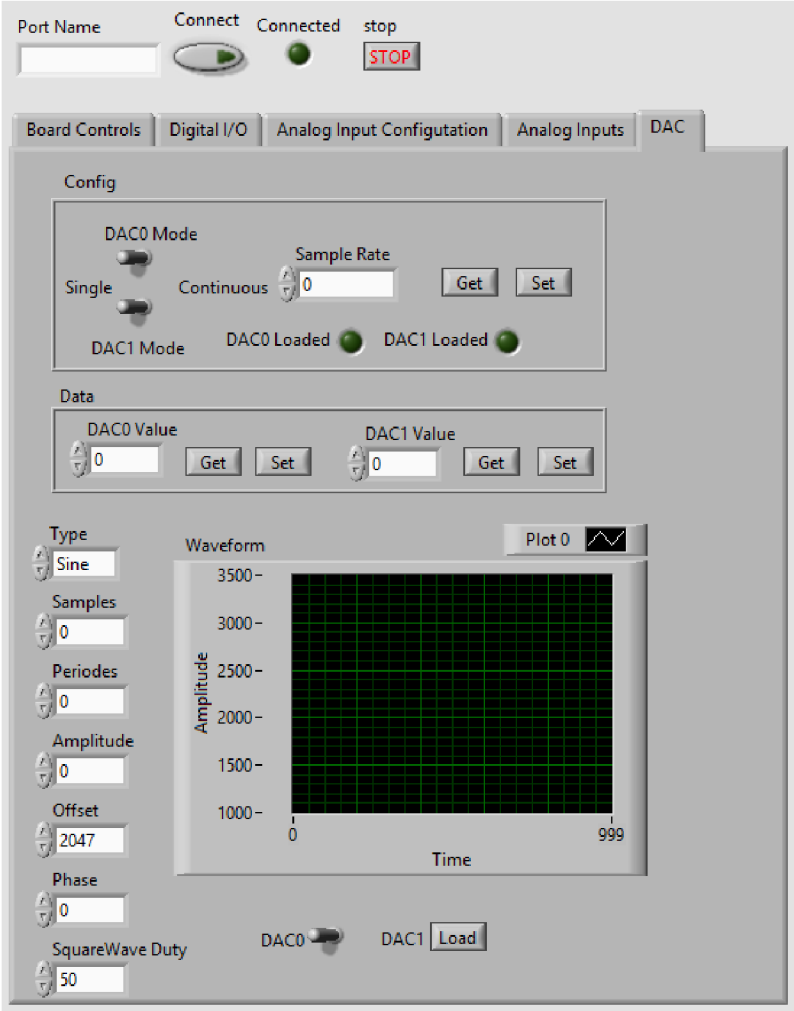
6.2 Ukážkový program v LabVIEW

Schopnosť knižnice pracovať s LabVIEW demonštruje ukážkový program. Program umožňuje využívať všetku funkcionalitu modulu. Kvôli názornosti princípov činnosti, program pracuje som základným formátovaním, t.j. formátovanie odpovedajúce priamo konfigurácii modulu, bez metód s prepočtami do základného formátovania, okrem časti zobrazujúcej veľkosť napätí pri analógových vstupoch.

6.2.1 Hlavný program

Na obr.6.1 je okno ukážkového programu. Vo vrchnej časti programu je textové pole `Port Name`, ktoré súži na zadanie názvu virtuálneho COM portu. Tlačítko `Connect` slúži na nadviazanie spojenia s modulom. Úspešné nadviazanie spojenia signalizuje

kontrolka `Connected`. Tlačítko `Stop` slúži na ukončenie spojenia a ukončenie programu. Funkčné začne byť až po úspešnom nadviazaní spojenia.



Obr. 6.1: Okno ukážkového programu

Hlavná časť okna programu je rozdelená na 5 záložiek:

- Board Control
- Digital I/O
- Analog Input Configuration
- Analog Inputs
- DAC

Záložka Board Controls

Záložka Board Controls obsahuje indikátory stavu LED a tlačidiel na doske modulu. Prepínače slúžia na nastavenie stavu jednotlivých LED. Tlačidlom `Get` sa načíta stav, tlačidlom `Set` sa prenesie stav prepínačov na LED.

Políčko `Board Time` slúži na zobrazenie času modulu. Tlačidlom `Get` sa načíta čas z modulu, tlačidlom `Set` sa nastaví modulu aktuálny čas hostovacieho systému.

Záložka Digital I/O

Táto záložka slúži na ovládanie digitálnych I/O. V spodnej časti okna je políčko `Voltage`, ktoré slúži na nastavenie napäťovej úrovne.

Vrchná časť záložky je rozdelená na dve časti – `MSB` a `LSB`, tie obsahujú indikátory a prepínače ovládajúce jednotlivé digitálne vstupy a výstupy. Ďalej obsahujú ovládacie políčka na výber smeru toku dát. Smer sa zisťuje a nastavuje tlačidlami `Get Direction` a `Set Direction`.Stav jednotlivých I/O liniek sa zisťuje a nastavuje tlačidlami `Get I/O`, resp. `Set I/O`.

Záložka Analog Input Configuration

Táto záložka slúži na ovládanie analógovej časti analógových vstupných kanálov. Záložka je rozdelená na dve časti, podľa kanálov. Tlačidlom `Calibration` sa vyberá kalibračný mód. Tlačidlom `DC Coupling` sa vyberá väzba kanálu. Kontrolky indikujú zapnutý stav. Prepínačom `Range` sa vyberá merací rozsah. Prvkom `High Pass` sa vyberá hornopriepustný filter. Prvkami `Low Pass Filter Range` a `Low Pass Filter Constant` sa nastavuje dolnopriepustný filter (podľa vzťahu 5.6, kde n odpovedá konštante filtra, rozsah 3 odpovedá deliču 1, rozsah 2 deliču 4 a rozsah 1 deliču 16). Tlačidlami `Get` sa zistí nastavenie kanálu, tlačidlami `Set` sa nastaví kanál.

Záložka Analog Inputs

Záložka Analog Inputs slúži na nastavenie ADC a na zobrazenie získaných dát. Políčkom `ADC Sample Rate` sa nastavuje vzorkovacia frekvencia (podľa vzťahu 5.7). Políčkom `Samples` sa nastavuje dĺžka vzorkovacieho bufferu. Prepínačmi sa volí režim vzorkovania. Tlačidlom `Trigger` sa získa jedna vzorka z oboch kanálov a výsledky sa zobrazia v políčkach `Channel 0` a `Channel 1`. Pod týmito políčkami sú políčka `Mean`, v ktorom je pri kontinuálnom vzorkovaní spočítaná cyklická stredná hodnota; `RMS` s cycklickou RMS hodnotou; a `Vpp`, kde je spočítaná hodnota špička-špička. Všetky napätia sú vo voltoch. V pravej časti programu sú pri kontinuálnom vzorkovaní zobrazené priebehy napätí na vstupoch.

Záložka DAC

Záložka DAC slúži na obsluhu analógových výstupov. V časti `Config` sa nastavuje mód analógových výstupov a vzorkovacia frekvencia (podľa vzťahu 5.5). Kontrolky `DAC0 Loaded` (resp. `DAC1 Loaded`) indikujú, že bol do modulu nahratý priebeh pre daný kanál. V časti `Data` sa dajú zistiť a nastaviť jednotlivé výstupy. Výstupné napätie sa spočíta podľa vzťahu 5.4.

V spodnej časti je generátor priebehov. Umožňuje zvoliť typ priebehu (sínus, kosínus, trojuholník, obdĺžnik, píla, rastúca a klesajúca rampa), počet vzoriek, počet periód signálu, amplitúdu, offset, fázu a pri obdĺžnikovom priebehu striedu. Pre-

pínačom sa volí, do ktorého kanálu sa priebeh nahrá. Tlačidlom `Load` sa zaháji nahrávanie. Po úspešnom nahratí priebehu sa rozsvieti odpovedajúca kontrolka `DAC Loaded`.

6.2.2 Podprogramy

Keďže viaceré metódy vyžadujú zložitejšie formátovanie vstupných dát alebo dekódovanie návratových hodnôt, vytvoril som niekoľko podprogramov (SubVI), ktoré pripravujú vstupné dáta alebo dekódujú návratové hodnoty. Jednak to sprehľadní program, ale hlavne zo zjednoduší opakované použitie a umožní jednoduchšie budúce použitie modulu v LabVIEW.

GetControls.vi

Toto SubVI obstaráva načítanie a dekódovanie stavu ovládacích a indikačných prvkov modulu.

SetControls.vi

Toto SubVI obstaráva kódovanie dát a samotné nastavenie indikátorov modulu.

GetIODir.vi

Načítanie a dekódovanie konfigurácie smeru toku dát digitálnych I/O obstaráva toto SubVI.

SetIODir.vi

Nastavenie smeru toku dát digitálnych I/O rieši toto SubVI.

GetIO.vi

Na zistenie stavu digitálnych I/O súži toto SubVI.

GetAnConf.vi

Toto SubVI obstaráva zistenie a dekódovanie konfigurácie analógových kanálov.

GetAdcConf.vi

Toto SubVI zistí a dekóduje konfiguráciu ADC.

ADCTriggerAndRead.vi

Toto SubVI zaháji získavanie vzorky dát z oboch kanálov a vráti ich hodnotu vo voltoch.

ADCGetSingleData.vi

Toto SubVI vráti poslednú prijatú vzorku pre oba kanály v raw podobe. Na napätie sa dá prepočítať podľa vzťahu 6.1.

ADCGetSingleVoltage.vi

Toto SubVI vráti poslednú prijatú vzorku pre oba kanály vo voltoch.

GetDACConf.vi

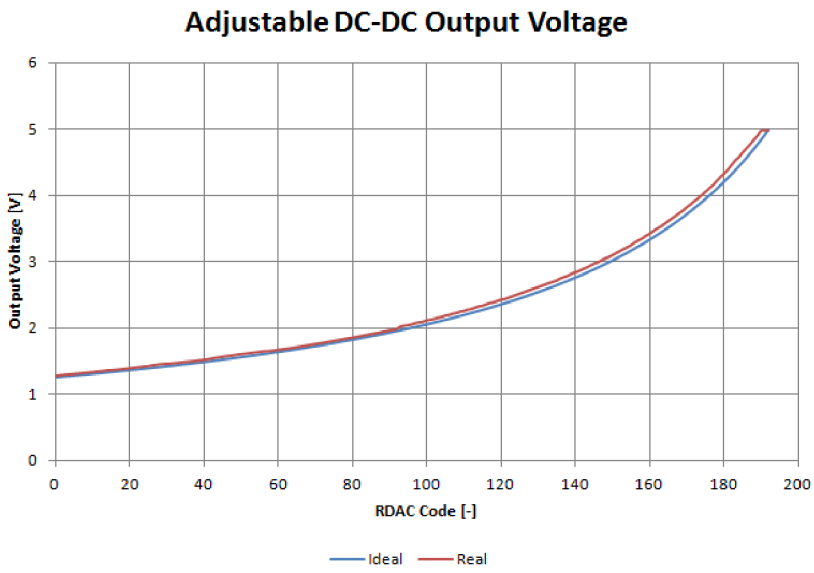
Na zistenie a dekódovanie nastavení DAC slúži toto SubVI.

7 OVERENIE PARAMETROV A VLASTNOSTÍ ZREALIZOVANÉHO MODULU

Táto kapitola obsahuje výsledky meraní zisťujúcich a overujúcich parametre a vlastnosti modulu.

7.1 Digitálne I/O

Na obr. 7.1 je závislosť napäťovej úrovne digitálnych I/O od kódu odoslaného do digitálneho potenciometra. Krivka *Ideal* (modrá) je vypočítaná podľa vzťahu 5.3, krivka



Obr. 7.1: Závislosť napäťovej úrovne digitálnych I/O od kódového slova digitálneho potenciometra

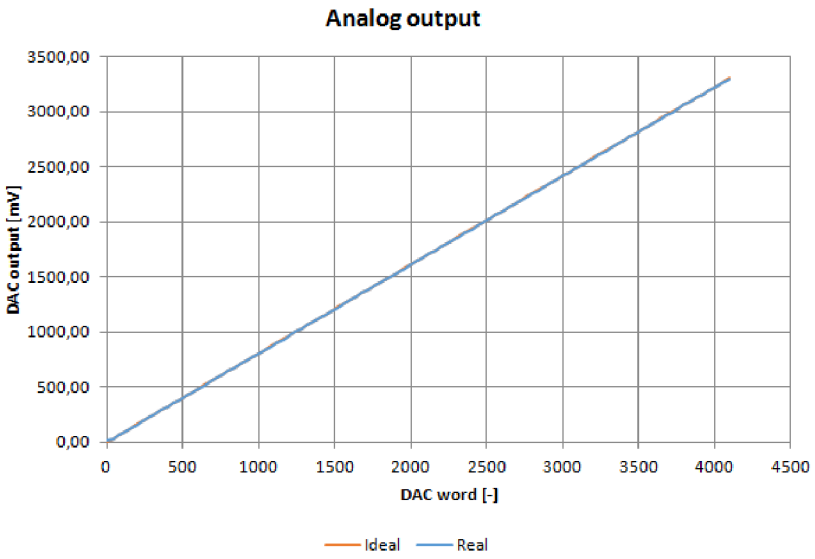
Real (červená) je reálne zmeraná. Drobná odchýlka medzi krivkami je najpravdepodobnejšie spôsobená nepresnosťou spätnoväzobného deliča nastaviteľného DC-DC meniča určujúceho napäťovú úroveň. Meranie potvrdilo, že približne od hodnoty 190 je už výstupné napätie na úrovni vstupného napätia a tým pádom už nemôže viac stúpať.

7.2 Analógové výstupy

Pri analógových výstupoch je vhodné určiť statickú prevodnú charakteristiku a dynamiku výstupu.

7.2.1 Statické charakteristika

Na obr. 7.2 je závislosť výstupného napätia analógových výstupov od kódového slova DAC (statická prevodná charakteristika). Z obrázku vidno, že závislosť je lineárna.



Obr. 7.2: Závislosť výstupného napätia analógových výstupov od kódového slova DAC

Krivka *Ideal* (červená) je vypočítaná podľa vzťahu 5.4, krivka *Real* (modrá) je reálne zmeraná. Krivky sa takmer dokonalé zhodujú.

7.2.2 Dynamické vlastnosti

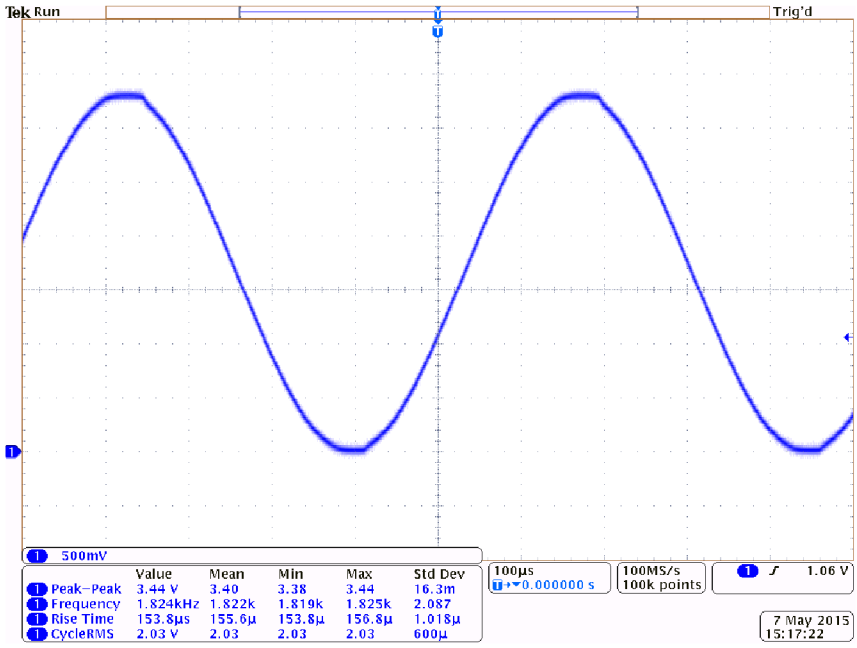
Všetky nasledujúce priebehy boli volené so špičkovým napätím odpovedajúcou plnému rozsahu výstupu DAC. Merania slúžili na overenie dynamických vlastností DAC a celého výstupného kanálu. Údajom *počet vzoriek* je myslený počet vzoriek na periódu signálu.

Na obr. 7.3 je sínusový priebeh o 1000 vzorkách a vzorkovaní 1.818MS/s. Toto nastavenie analógové výstupy zvládajú.

Na obr. 7.4 je sínusový priebeh o 100 vzorkách a vzorkovaní 1.818MS/s. Pri tomto nastavení už analógové výstupy pomaly prestávajú stíhať.

Na obr. 7.5 je sínusový priebeh o 25 vzorkách a vzorkovaní 1.818MS/s. Z grafu vidno, že pri tejto konfigurácii už výstup nestíha plný rozsah výstupu. Vzhľadom na frekvenciu výstupného signálu, na obmedzení výstupu sa začína prejavovať rekonštrukčný filter.

Na obr. 7.6 je trojuholníkový priebeh o 1000 vzorkách a vzorkovaní 1.818MS/s. Toto nastavenie analógové výstupy zvládajú.



Obr. 7.3: Analógový výstup: sínus, 1000 vzoriek, 1.818MS/s

Na obr. 7.7 je trojuholníkový priebeh o 100 vzorkách a vzorkovaní 1.818MS/s. Pri tomto nastavení už analógové výstupy pomaly prestávajú stíhať.

Na obr. 7.8 je trojuholníkový priebeh o 25 vzorkách a vzorkovaní 1.818MS/s. Priebeh je značne deformovaný, keďže základná frekvencia sa blíži medznému kmitočtu rekonštrukčného filtra, tým pádom sú vyššie harmonické signálu výraznejšie potlačené.

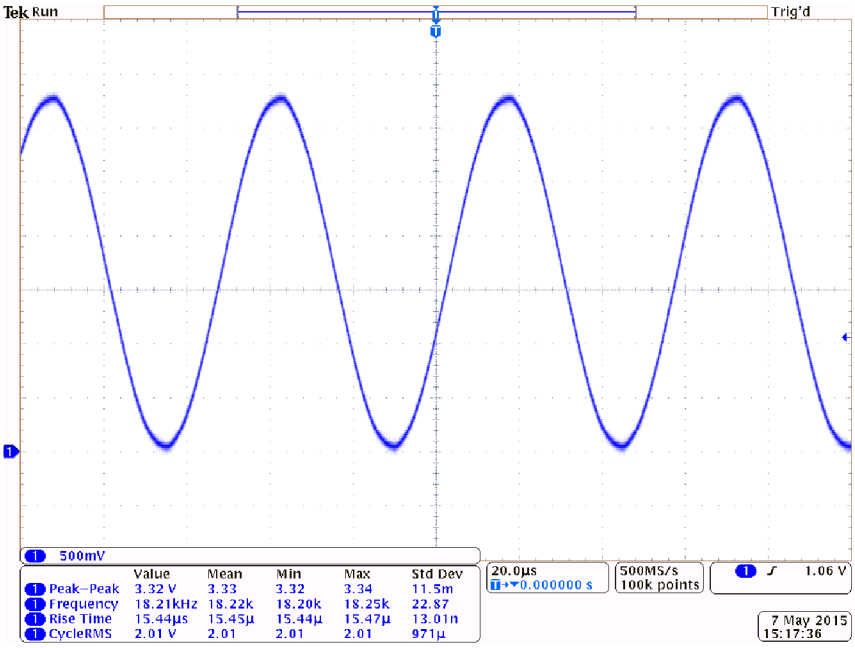
Na obr. 7.9 je obdĺžnikový priebeh o 1000 vzorkách a vzorkovaní 198kS/s. Toto nastavenie analógové výstupy bez problémov zvládajú.

Na obr. 7.10 je obdĺžnikový priebeh o 1000 vzorkách a vzorkovaní 1.818MS/s. Pri tomto nastavení sa už začínajú mierne prejavovať dynamické obmedzenia DAC.

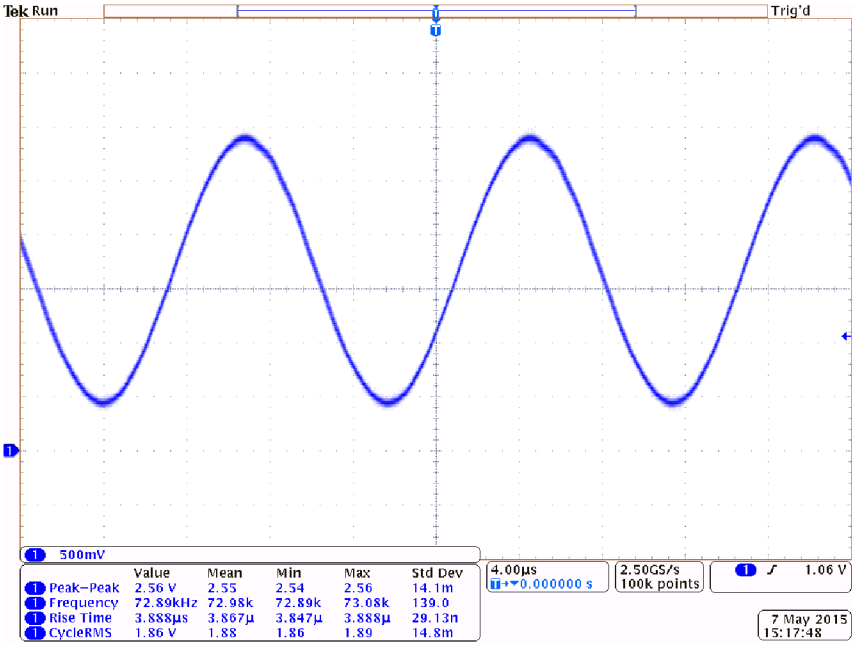
Na obr. 7.11 je obdĺžnikový priebeh o 100 vzorkách a vzorkovaní 1.818MS/s. Pri tomto nastavení sa jednak výraznejšie prejavujú obmedzenia dynamiky DAC, ale aj orezanie vyšších harmonických rekonštrukčným filtrom.

Na obr. 7.12 je obdĺžnikový priebeh o 25 vzorkách a vzorkovaní 1.818MS/s. Priebeh je výrazne deformovaný, keďže základná frekvencia sa blíži medznému kmitočtu rekonštrukčného filtra, tým pádom sú vyššie harmonické signálu výraznejšie potlačené; a taktiež sa výrazne prejavuje obmedzená dynamika DAC.

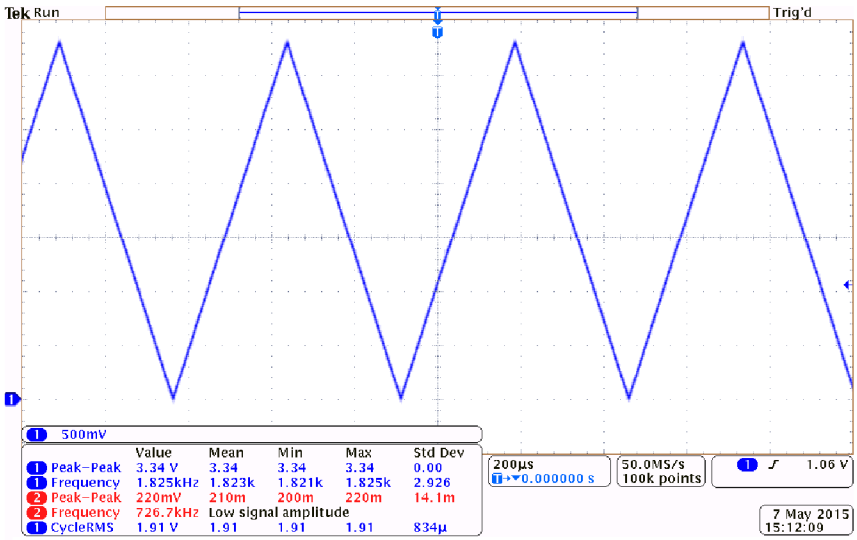
Merania s obdĺžnikovým priebehom ukázali, že pri zmene výstupu v celkom rozsahu, je čas nábehu približne 4.3μs. Z toho vyplýva, že strmosť nábehu je približne 0.613V/μs pri zmene v plnom rozsahu výstupu. Prípadnou zmenou rekonštrukčného filtra by sa mala dať táto hodnota ešte zvýšiť.



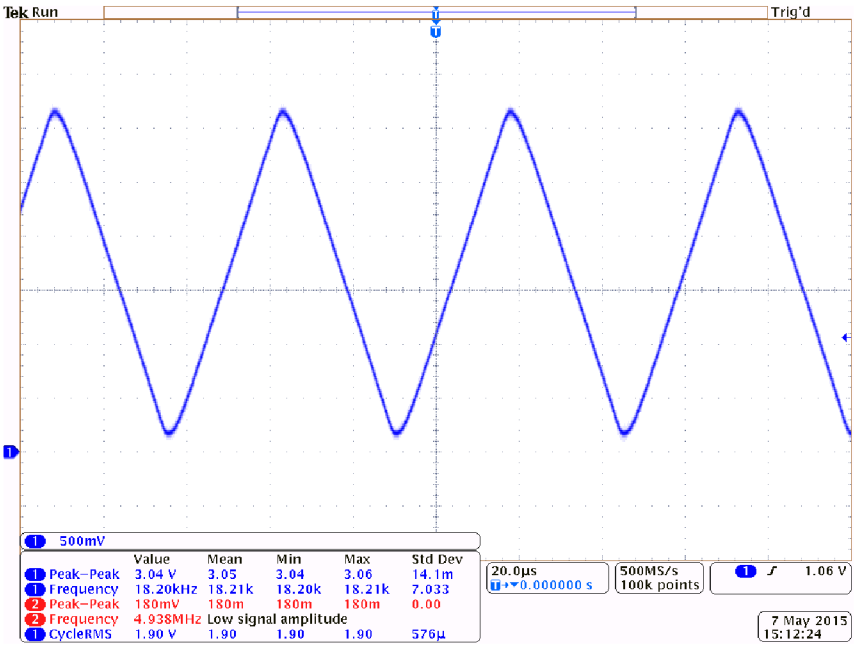
Obr. 7.4: Analógový výstup: sínus, 100 vzoriek, 1.818MS/s



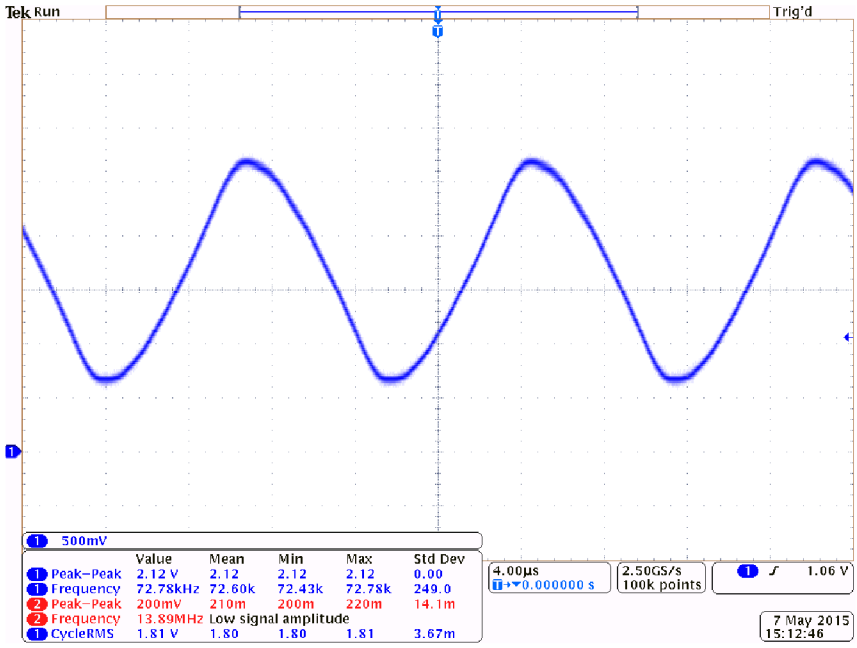
Obr. 7.5: Analógový výstup: sínus, 25 vzoriek, 1.818MS/s



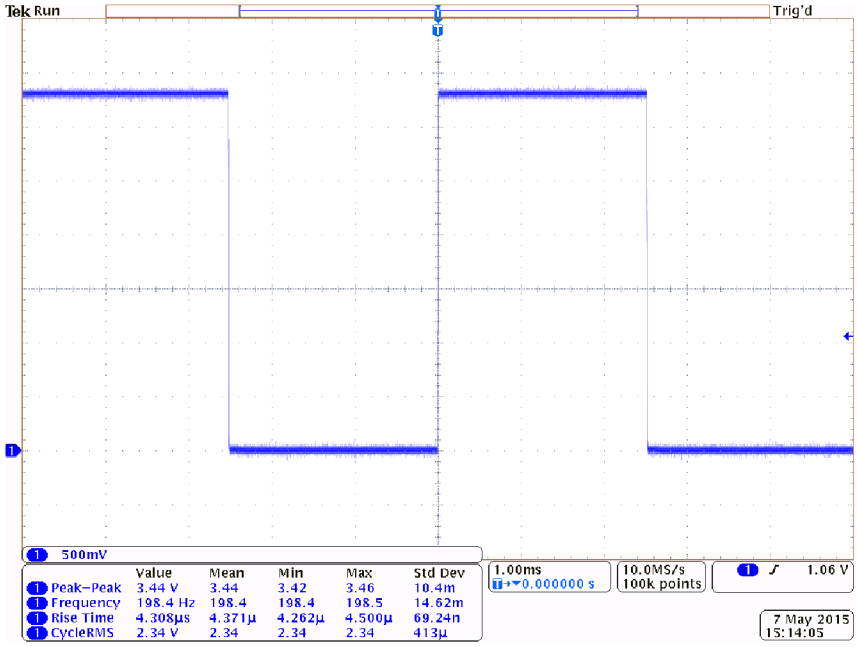
Obr. 7.6: Analógový výstup: trojuholník, 1000 vzoriek, 1.818MS/s



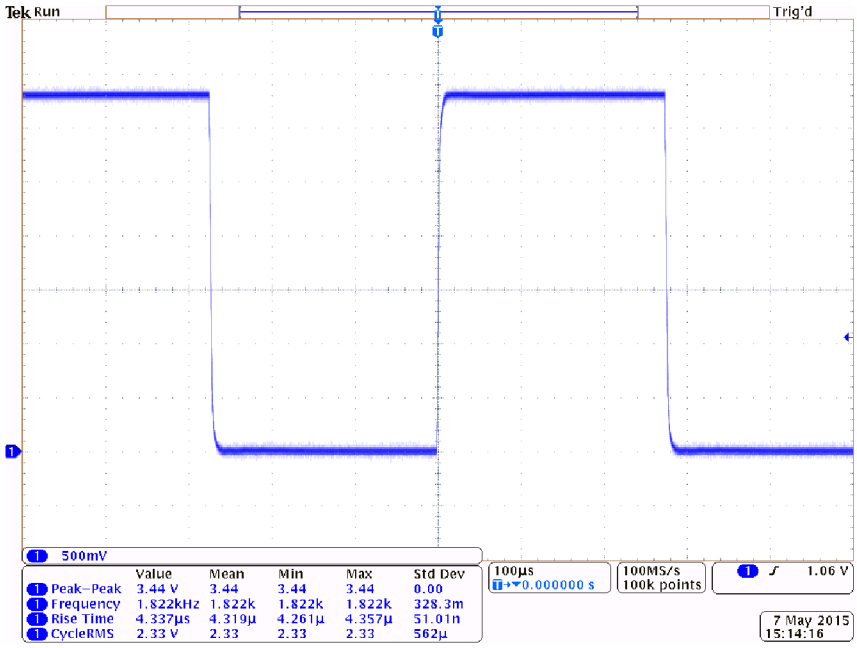
Obr. 7.7: Analógový výstup: trojuholník, 100 vzoriek, 1.818MS/s



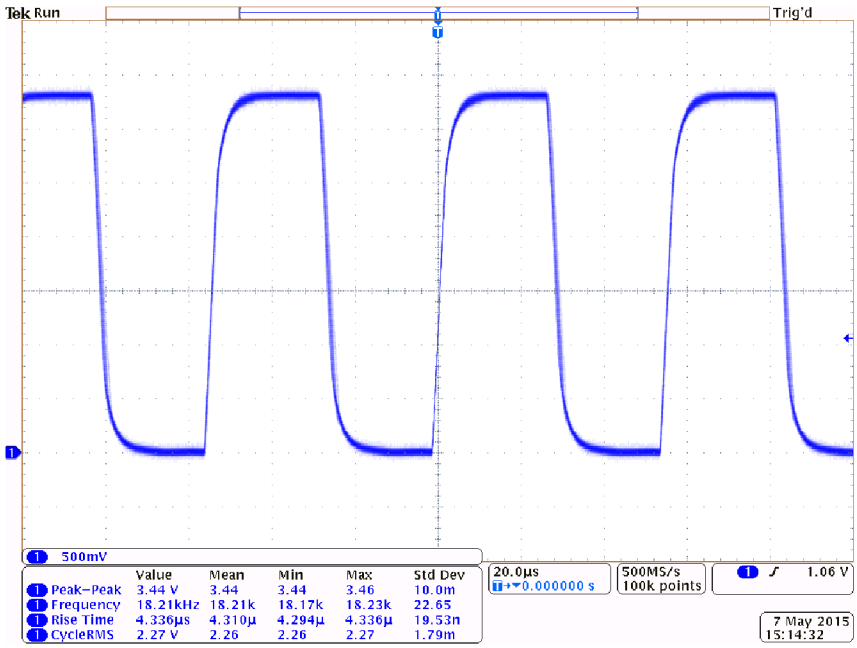
Obr. 7.8: Analógový výstup: trojuholník, 25 vzoriek, 1.818MS/s



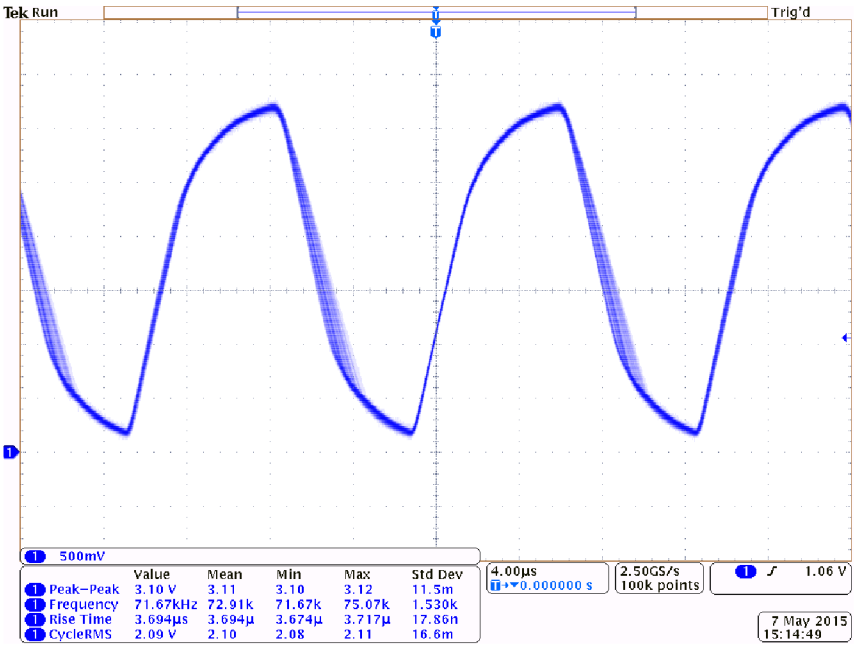
Obr. 7.9: Analógový výstup: obdĺžnik, 1000 vzoriek, 198kS/s



Obr. 7.10: Analógový výstup: obdĺžnik, 1000 vzoriek, 1.818MS/s



Obr. 7.11: Analógový výstup: obdĺžnik, 100 vzoriek, 1.818MS/s



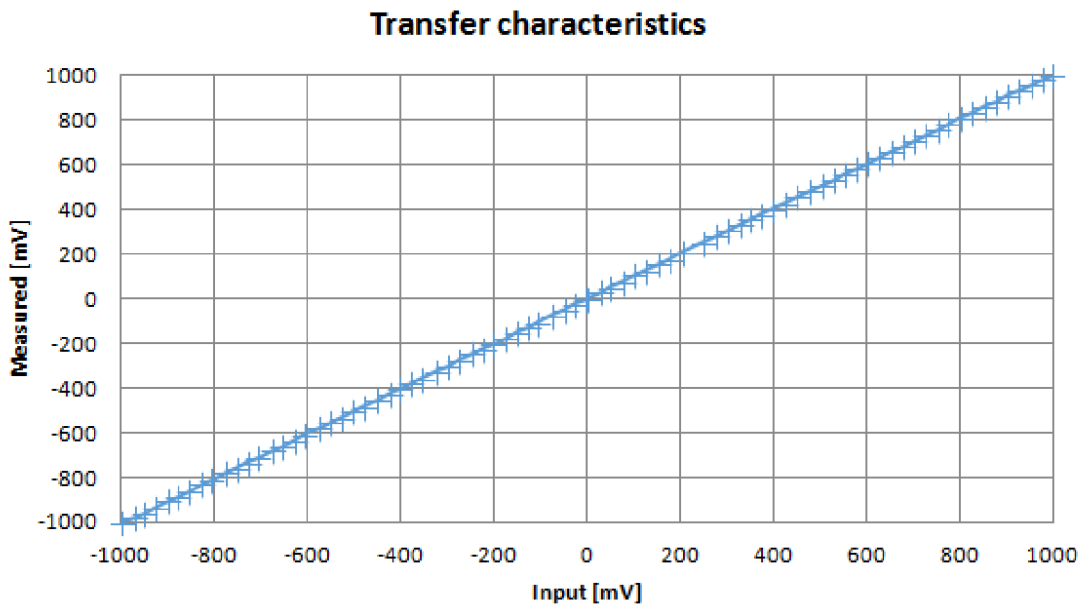
Obr. 7.12: Analógový výstup: obdĺžnik, 25 vzoriek, 1.818MS/s

7.3 Analógové vstupy

Pri analógových vstupov je dobré určiť statickú prevodnú charakteristiku a chybu meracieho kanálu a ďalej určiť frekvenčné vlastnosti celého reťazca. Pri statických meraniach bolo napätie získané programom v LV, pri dynamických meraniach bolo napätie merané multimetrom na vstupe ADC.

7.3.1 Statické charakteristiky

Na obr. 7.13 je prevodná charakteristika rozsahu ±1V. Meranie ukázalo, že je celý rozsah lineárny.



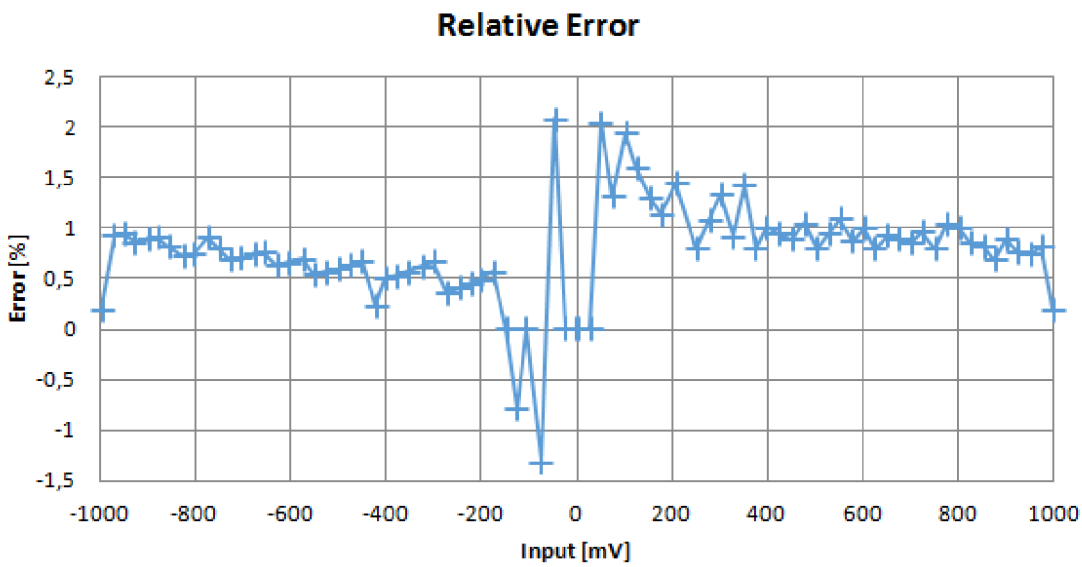
Obr. 7.13: Prevodná charakteristika rozsahu ±1V

Na obr. 7.14 je relatívna chyba merania. Meranie ukázalo, že po väčšine rozsahu (okrem niektorých malých hodnôt) sa chyba pohybuje do 2%.

Na obr. 7.15 je prevodná charakteristika rozsahu ±0.2V. Meranie ukázalo, že na krajoch rozsahu saturuje ADC a chyba narastá úmerne hodnote, to znamená nepresne nastavené zosilnenie, takže určovať chybu na tomto rozsahu nemá význam.

7.3.2 Dynamické charakteristiky

Nasledujúce charakteristiky boli zmerané na rozsahu ±1V s napätím 500mV$_{RMS}$ (približne -3dBFS).



Obr. 7.14: Relatívna chyba rozsahu ±1V

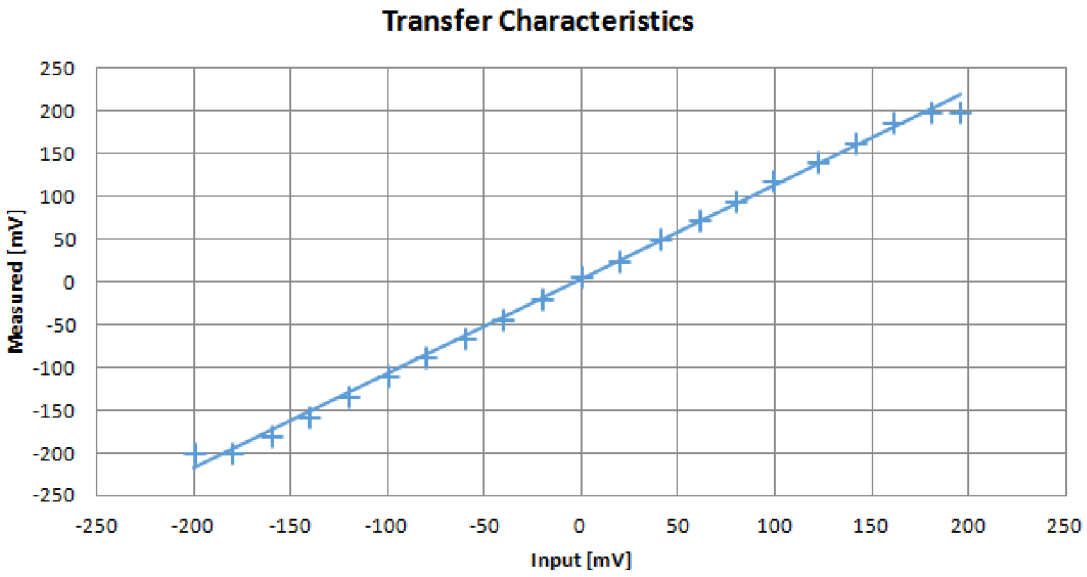
Na obr.7.16 je frekvenčná charakteristika vstupného zosilňovača pri zapnutej jednosmernej väzbe. Pri najvyšších frekvenciách mohlo byť meranie skreslené obmedzeným frekvenčným rozsahom multimetra (výrobca garantuje presnosť merania do 300kHz).

Na obr.7.17 sú frekvenčné charakteristiky hornopriepustných filtrov. Modrá krivka *AC Coupling* je krivka vstupného zosilňovača so zapnutou striedavou väzbou. Meranie potvrdzuje, že medzný kmitočet je mierne pod 0.2Hz so sklonom 20dB/dek. Zelená krivka *1Hz HPF* je krivka hornopriepustného filtra s medzným kmitočtom približne 1Hz so sklonom 40dB/dek. Červená krivka *100Hz HPF* je krivka hornopriepustného filtra s medzným kmitočtom približne 100Hz so sklonom 40dB/dek.

Merať nasledujúce charakteristiky do vyšších frekvencií, než boli zmerané nemalo význam, lebo už boli na úrovni vlastného šumu modulu.

Na obr.7.18 sú frekvenčné charakteristiky dolnopriepustného filtra v závislosti na veľkosti ladiaceho odporu. Všetky krivky boli merané pri deliči /16. Modrá krivka *Rext=39k* bola meraná pri kóde 156, čo odpovedá odporu približne 39kΩ, čo by malo odpovedať medznej frekvencii približne 1kHz, čo približne odpovedá zistenému. Oranžová krivka *Rext=19,5k* bola meraná pri kóde 206, čo odpovedá odporu približne 19,5kΩ, čo by malo odpovedať medznej frekvencii približne 2kHz, čo meranie potvrdilo.

Na obr.7.19 sú frekvenčné charakteristiky dolnopriepustného filtra v závislosti na zvolenom deliči. Merané boli pri kóde 156, ktorý odpovedá odporu približne 39kΩ. Modrá krivka */16* bola meraná pri deliči /16, čo by malo odpovedať medznej



Obr. 7.15: Prevodná charakteristika rozsahu ±0.2V

frekvencii približne 1kHz, čo meranie potvrdilo. Oranžová krivka */4* bola meraná pri deliči /4, čo by malo odpovedať medznej frekvencii približne 4kHz, čo meranie potvrdilo. Zelená krivka */1* bola meraná pri deliči /1, čo by malo odpovedať medznej frekvencii približne 16kHz, čo meranie taktiež potvrdilo.

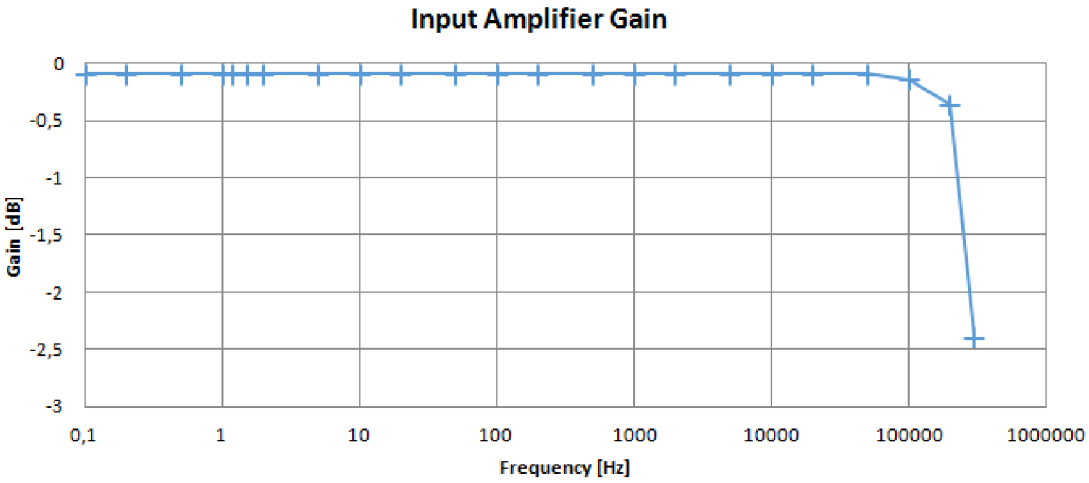
Na obr.7.20 je spektrum sínusového a obdĺžnikového signálu (oba s frekvenciou 1kHz) zmeraného modulom a vypočítané pomocou LabVIEW (blokom FFT Spectrum (Mag-Phase)). Z grafu vidno, že sínusový signál (modrá krivka) neobsahuje nežiadúce frekvencie, čo potvrdilo aj zistené harmonické skreslenie 0.04% (spočítané blokom Harmonic Distortion Analyzer v LV). Spektrum obdĺžnikového signálu (oranžový priebeh) obsahuje výraznú tretiu harmonickú (podľa predpokladu). Harmonické skreslenie tohto signálu je približne 42%, čo taktiež približne odpovedá teoretickému predpokladu.

7.3.3 Dátová priepustnosť

Je vhodné overiť, aké množstvo dát je schopný modul preniesť do PC. Meranie spočívalo v tom, že 10sek. boli zaznamenávané dáta. Bolo zrealizovaných 10 nezávislých meraní, s ktorých bol následne spočítaný priemer a smerodajná odchýlka. Tie ešte boli vydelené desiatimi, aby som dostal údaj za sekundu. Vzorkovacia frekvencia približne 34kHz a buffer dĺžky 1000 vzoriek.

Jednokanálové meranie

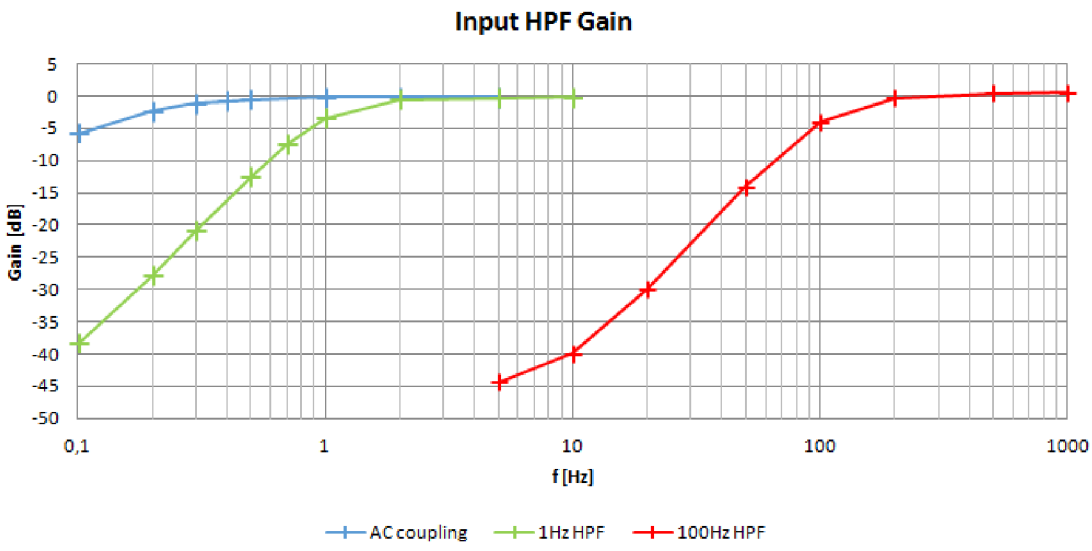
Pri jednokanálovom meraní som dostal priemer 33500 vzoriek/s so smerodajnou odchýlkou približne 90.



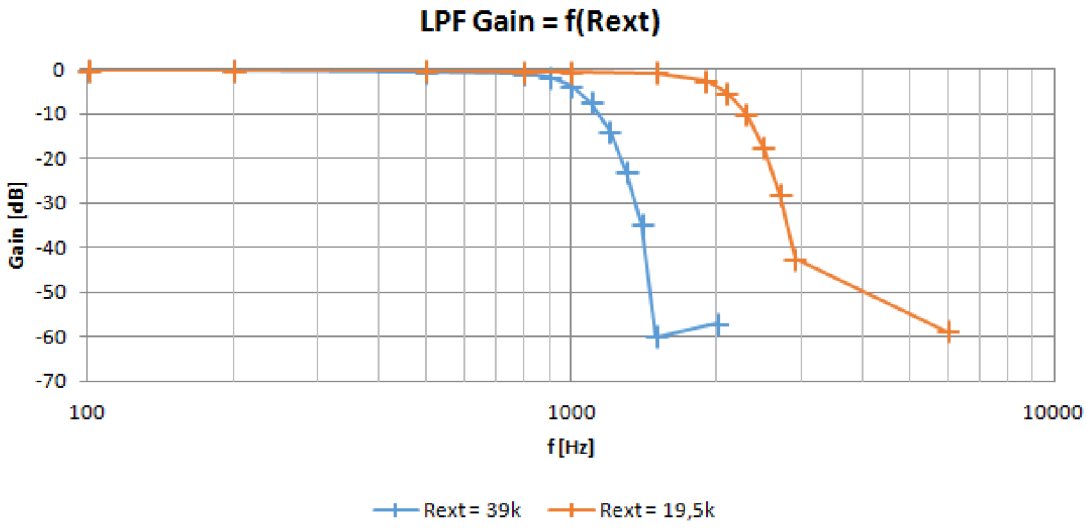
Obr. 7.16: Frekvenčná charakteristika vstupného zosilňovača

Dvojkanálové meranie

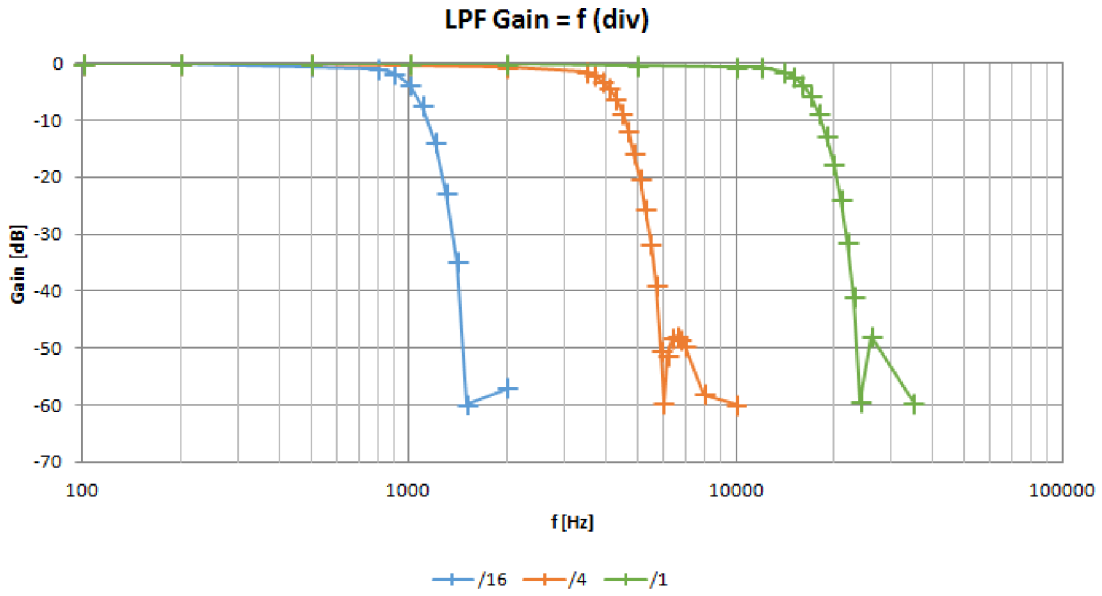
Pri dvojkanálovom meraní som dostal pre prvý kanál priemer 30670 vzoriek/s so smerodajnou odchýlkou približne 809, pre druhý kanál 32170 vzoriek/s so smerodajnou odchýlkou 200.



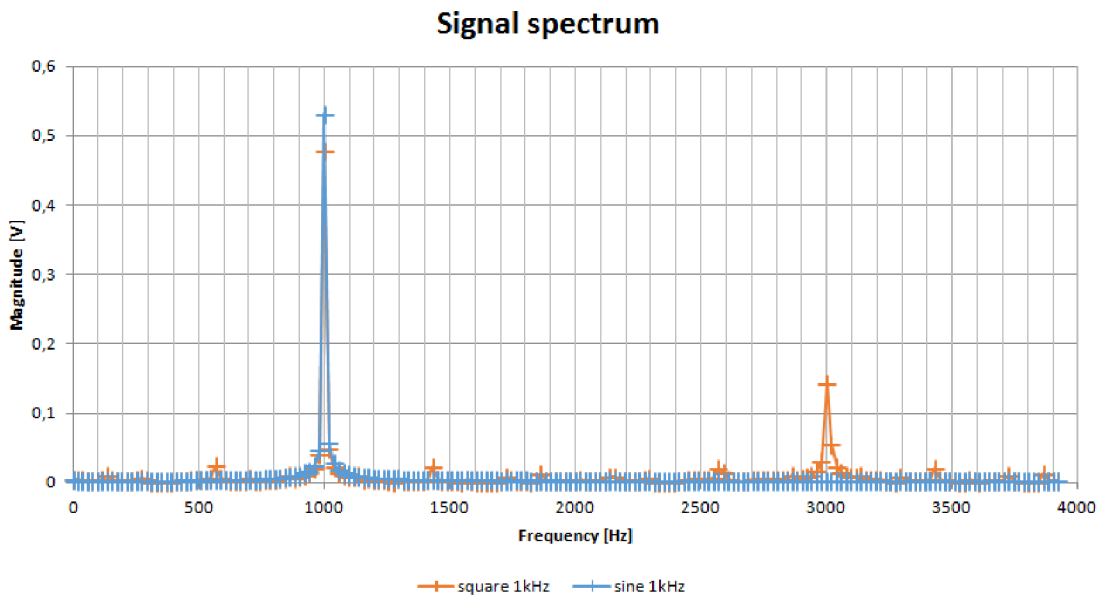
Obr. 7.17: Frekvenčné charakteristiky hornopriepustných filtrov



Obr. 7.18: Frekvenčná charakteristika dolnopriepustného filtra v závislosti na veľkosti ladiaceho odporu



Obr. 7.19: Frekvenčná charakteristika dolnopriepustného filtra v závislosti na zvolenom deliči frekvencie



Obr. 7.20: Spektrum sínusového a obdĺžnikového signálu

8 ZÁVER

Cieľom práce bolo navrhnúť univerzálny USB merací modul pre virtuálnu inštrumentáciu, ktorý by mal byť ovládateľný z prostredia LV.

Srdce modulu tvorí MCU MSP430F5659 od spoločnosti Texas Instruments. Modul je cez oddeľovače pripojený k PC zbernicou USB, ktorá slúži aj na napájanie modulu. Modul je oddelený, aby ho bolo možné použiť aj na inom potenciály ako je zemný potenciál hosťovacieho systému. Na moduly je pripravená fyzická vrstva pre ovládanie aj iným rozhraním, ktoré by sa dalo pripojiť k MCU zbernicou UART/SPI/I2C (napr. pomocou prevodníka na bluetooth, ethernet a iné). Modul ďalej obsahuje 4 indikačné LED a 2 ovládacie tlačidlá, ktoré zatiaľ nemajú špecifikovanú funkciu a sú dostupné pre užívateľa. Modul obsahuje aj hodiny reálneho času vstavané v MCU so zálohovaním času superkapacitorom.

Modul ďalej obsahuje subsystém digitálnych I/O. Tento subsystém je galvanicky oddelený od MCU, aby mohli byť I/O použiteľné na inom potenciály ako je zem modulu. Tvorí ho 8 digitálnych liniek s možnosťou nastaviť smer prenosu dát po štvoriciach. Ďalšou súčasťou subsystému sériová linka UART/SPI (s možnosťou použiť ako digitálne I/O s pevne nastaveným smerom) a zbernica I2C. Subsystém má plynule nastaviteľnú napäťovú úroveň v rozsahu 1.2V až 5V. Toto nastavenie je spoločné pre celý subsystém. Všetky časti tohto subsystému sú na spoločnom potenciáli.

Ďalší subsystém sú 2 analógové výstupy s rozlíšením 12bitov využívajúci vstavaný DAC MCU a vzorkovaním až 1.8MS/s a strmosťou pri zmene v plnom rozsahu až 0.613V/μs. Tento parameter by sa teoreticky mal dať zvýšiť zmenou rekonštrukčného filtra.

Posledný subsystém obsahuje 2 analógové vstupy využívajúci vstavaný ADC MCU. Vstupy majú možnosť prepínať jednosmernú a striedavú väzbu. Rozsah je prepínateľný, $\pm$1V a $\pm$0.2V. Vstupy ďalej obsahujú prepínateľný hornopriepustný filter (medzná frekvencia 1Hz a 100Hz) a laditeľný dolnopriepustný filter s medznou frekvenciou nastaviteľnou v rozsahu 1 až 64kHz. Rozlíšenie vstupov je 12bitov. Maximálna vzorkovacia frekvencia je približne 34kHz (pre každý kanál). Chyba rozsahu $\pm$1V je približne 2%. Pri rozsahu $\pm$0.2V sa ukázalo, že nie je presne nastavené zosilnenie, takže určovať chybu rozsahu nemá význam. Ďalej je možnosť pripojenia nulového potenciálu na vstup, čo umožňuje napr. kompenzovanie nuly v ADC.

K modulu som vytvoril obslužnú knižnicu v jazyku C#. Táto umožňuje využiť plnú funkcionalitu modulu a je postavená tak, aby ju bolo možné využiť v prostredí LabVIEW. Túto možnosť demonštruje vytvorený ukážkový program.

Možných vylepšení je viacero, či už po stránke hardware, firmware alebo software. Úpravou hardware by sa mohla dosiahnuť vyššia presnosť analógových vstupov mo-

dulu, hlavne použitím presnejších súčiastok v deliči obvodu napäťovej referencie, ale hlavne v obvode vstupných zosilňovačov, keďže kvôli zmene v rozsahoch spomínanej skôr v práci boli v zosilňovači použité súčiastky, aké boli zrovna k dispozícii a dali by sa nahradiť presnejšími variantami (hlavne na rozsahu ±0.2V). Pri analógových výstupoch by sa zmenou rekonštrukčného filtra dala teoreticky zvýšiť strmosť zmeny výstupu a tým pádom aj maximálna frekvencia výstupného signálu. Vo firmware je možnosť pridaní viac funkcionalít, napr. viac možností triggerovania vstupu, implementovať komunikačný stack pre sériové linky v subsystéme digitálnych I/O, automatické posielanie zmeny stavu digitálnych I/O, kontinuálne nahrávanie dát pre analógové výstupy a lepšej optimalizácie už funkčných častí. S tým úzko súvisia aj možnosti vylepšenia obslužnej knižnice. Ďalej by sa dala zvýšiť použiteľnosť knižnice mimo LabVIEW upravením už vytvorených metód, ktoré sú obmedzené možnosťami prepojenia LabVIEW s knižnicou, a doplnením metód vyhodnocujúcich a spracovávajúcich získaný signál z analógových vstupov a metód na generovanie signálov pre analógové výstupy. Všetky tieto vylepšenia sa vzťahujú na už vytvorený hardware. Pri tvorbe nového hardware by sa dali vylepšiť rozsahy a celkovo možnosti analógových vstupov a výstupov. A existuje veľa možností vylepšenia subsystému digitálnych I/O (rôzne typy výstupov, možnosti konfigurácie, ...).

LITERATÚRA

[1] NATIONAL INSTRUMENTS. *NI myDAQ: Specifications* [online]. 2014-07 [cit. 2015-01-03]. Dostupné z URL: <http://www.ni.com/pdf/manuals/373061f.pdf>.

[2] NATIONAL INSTRUMENTS. *NI USB-6008/6009: User Guide and Specifications* [online]. 2012-02 [cit. 2015-01-03]. Dostupné z URL: <http://www.ni.com/pdf/manuals/371303m.pdf>.

[3] HALVORSEN, Hans-Petter *Introduction to LabVIEW* [online]. Porsgrunn, Norway: Faculty of Technology, Telemark University College, 2014-03-07, [cit. 2015-04-26]. Dostupné z URL: <http://home.hit.no/~hansha/documents/labview/training/Introduction%20to%20LabVIEW/Introduction%20to%20LabVIEW.pdf>.

[4] KUČERA, S. *LabVIEW – Učebná príručka* [online]. 2010 [cit. 2015-04-26]. Dostupné z URL: <http://www.spspp.sk/userfiles/file/ELE%20-%20modernizácia%20učebných%20procesov/labview.doc>.

[5] NATIONAL INSTRUMENTS. *National Instruments VISA: What Is VISA?* [online]. 2014 [cit. 2015-01-02]. Dostupné z URL: <http://www.ni.com/visa/>.

[6] NATIONAL INSTRUMENTS. *USB Instrument Control Tutorial* [online]. 2014-01-06 [cit. 2015-01-02]. Dostupné z URL: <http://www.ni.com/white-paper/4478/en/pdf>.

[7] NATIONAL INSTRUMENTS. *GPIB Hardware and Software Specifications* [online]. 2008-04-30 [cit. 2015-01-02]. Dostupné z URL: <http://www.ni.com/white-paper/3388/en/>.

[8] TEXAS INSTRUMENTS. *Analysis of the Sallen-Key Architecture* [online]. 1999-07, 2002-09 [cit. 2015-01-02]. Dostupné z URL: <http://www.ti.com/lit/an/sloa024b/sloa024b.pdf>.

[9] MANCINI, Ron. TEXAS INSTRUMENTS. *Op Amps For Everyone* [online]. 2002-08 [cit. 2015-01-02]. Dostupné z URL: <http://e2e.ti.com/cfs-file/__key/telligent-evolution-components-attachments/00-18-01-00-01-20-34-88/op-amps-for-everyone.pdf>.

[10] PATOČKA, M., VOREL P. *Řídicí elektronika - Pasivní obvody* [online]. Vysoké učení technické Brně, Fakulta elektroniky a komunikačních technologií,

2002 [cit. 2015-01-02]. Dostupné z URL: <https://www.vutbr.cz/www_base/priloha.php?dpid=22329>.

[11] TEXAS INSTRUMETS. *Programmer's Guide: MSP430 USB API & Descriptor Tool* [online] 2014-09 [cit. 2015-05-04]. Dostupné z URL: <http://www.ti.com/tool/msp430usbdevpack>.

ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

DAC	Digital-to-Analog Converter – Číslicovo analógový prevodník
ADC	Číslicovo-analógový prevodník
MCU	Microcontroller unit – Mikrokontrolér
I2C	Inter-Integrated Circuit – multi-masterová sériová zbernica
SPI	Serial Peripheral Interface – sériové periférne rozhranie
UART	Universal Asynchronous Receiver/Transmitter – Univerzálny Asynchrónny Prijímač/Vysielač
USB	Universal Serial Bus – univerzálna sériová zbernica
DAQ	Data Acquisition – získavanie dát
NI	National Instruments
VISA	Virtual Instrument Software Architecture
LV	LabVIEW
AI	Analógové vstupy
AO	Analógové výstupy
TI	Texas Instruments
AD	Analog Devices
OZ	Operačný zosilňovač

ZOZNAM PRÍLOH

A Špecifikácia modulu **85**

B Obsah priloženého CD **88**

C Schémy zapojenia **89**

D Doska plošných spojov **95**

E Zoznam súčiastok **100**

A ŠPECIFIKÁCIA MODULU

SPECIFICATION

UniModule

Analog Inputs

Number of channels: ..... 2 single ended

ADC resolution:..... 12bit

Maximum sampling rate:..... 34kS/s

Converter type:..... Succesive aproximation

Buffer size:..... 1024 samples

Timing resolution: ..... 3.2µs (312500Hz timebase)

Input range:..... ±1V, ±0.2V

Input Impedance:..... 1MΩ

Overvoltage protection:..... ±5V

Relative accouracy:

±1V range:..... 2%

±0.2V range:..... not specified

Coupling:..... DC, AC (0.16Hz, first order)

High-pass filter:..... none, 1Hz, 100Hz (2nd order)

Low-pass filter:..... none, adjustable 1kHz to 64kHz (10th order)

Analog Output

Number of channels:..... 2

DAC resolution:..... 12bit

Maximum update rate: ..... 1.81MHz, software-timed

Output range:..... 0 to +3.3V

Power-on state:..... 0V

Short circuit current:..... 150mA

Absolute typical accuracy (no load):..... 2mV

Digital I/O

Digital I/O lines:..... 8 lines

Direction control:..... 4 and 4 lines programable as input or output

Voltage level:..... 1.2 to 5V

Low-level treshold:..... 1/3* Voltage level

High-level treshold:..... 2/3* Voltage level

Power-on state:..... output, low

Bus Interface

USB specification:..... USB 2.0 Full-speed

USB bus speed..... 12Mbit/s

Max. current counsumption:..... 500mA

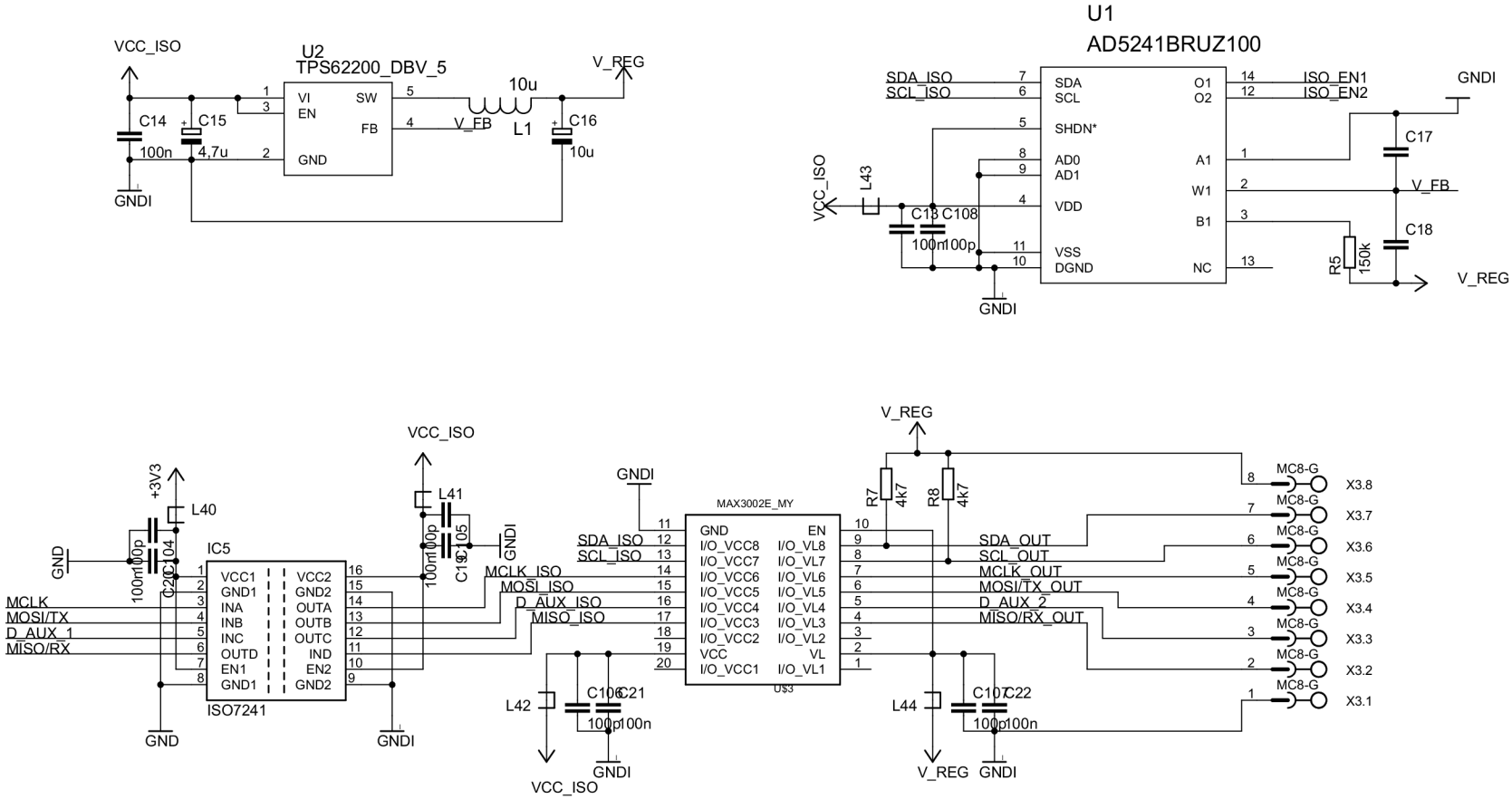
B OBSAH PRILOŽENÉHO CD

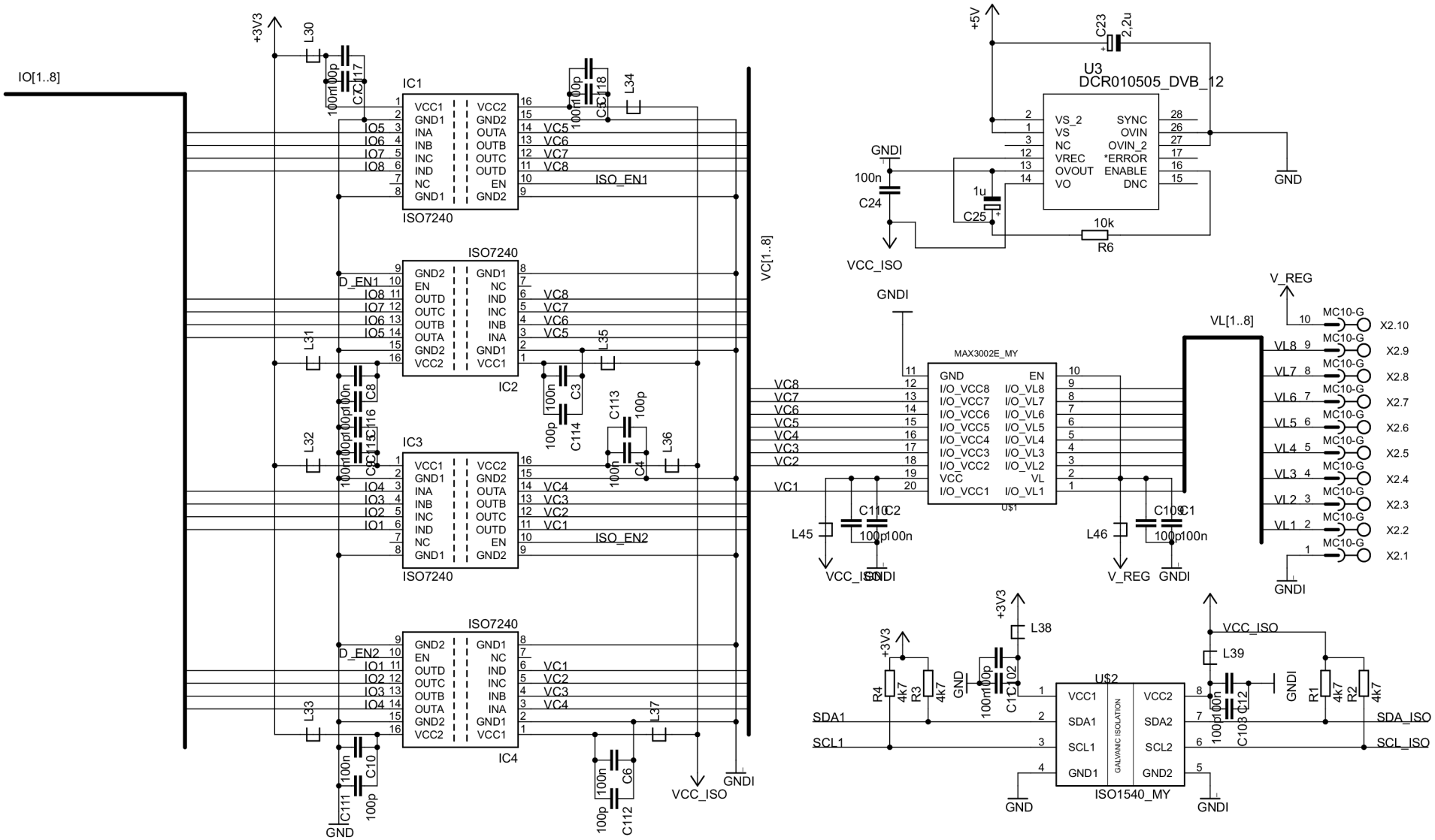
- Datasheety – datasheety aktuálne v čase návrhu obvodu
- Descriptors – USB deskriptory a identifikačný .inf súbor
- Eagle – tento priečinok obsahuje pôvodnú a opravenú verziu návrhu obvodu a DPS
- Firmware – obsahuje firmware pre modul (CCS v5)
- Sims – simulácie blokov meracieho kanálu (MS12 a MS13)
- Software – obsahuje software k modulu
  - UniModulue – projekt samotnej knižnice (VS2013)
  - DPv1_0_1 - benchmark – ukážkový program bez použitia knižnice doplnený o benchmark (VS2013)
  - DPv1_0_2 – ukážkový program využívajúci knižnicu (VS2013)
  - LV – ukážkový program v LabVIEW so SubVI (LV2014)
- Tabulky – rôzne tabuľky - výsledky meraní, prepočty napäťovej úrovne DI/O, dolnopriepustného filtra, operačné kódy, ...
- Text – zdrojové súbory práce v LaTEXu + samotná práca
- USB Developer package – USB vývojový balík od TI aktuálny v čase tvorby firmware

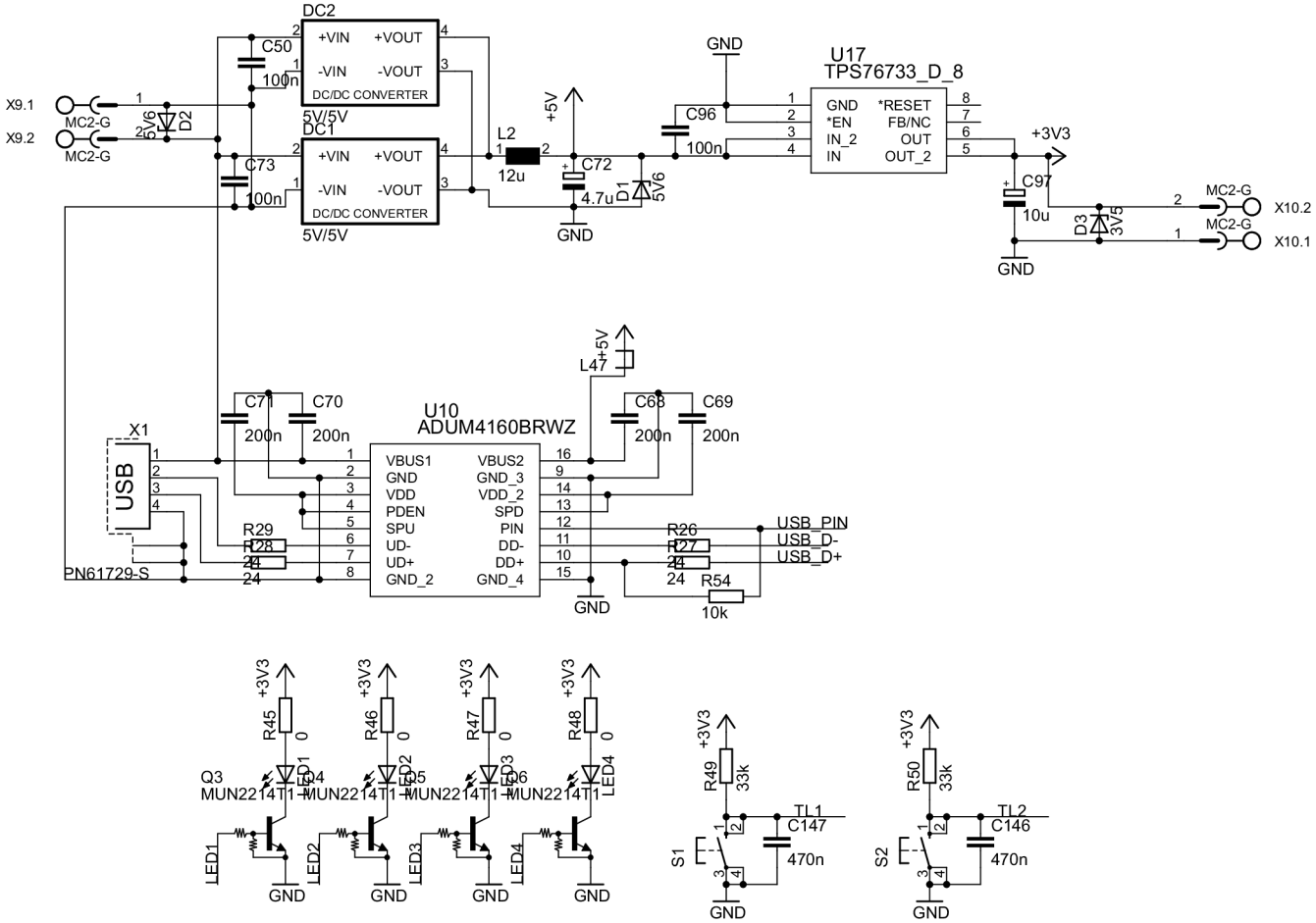
C SCHÉMY ZAPOJENIA

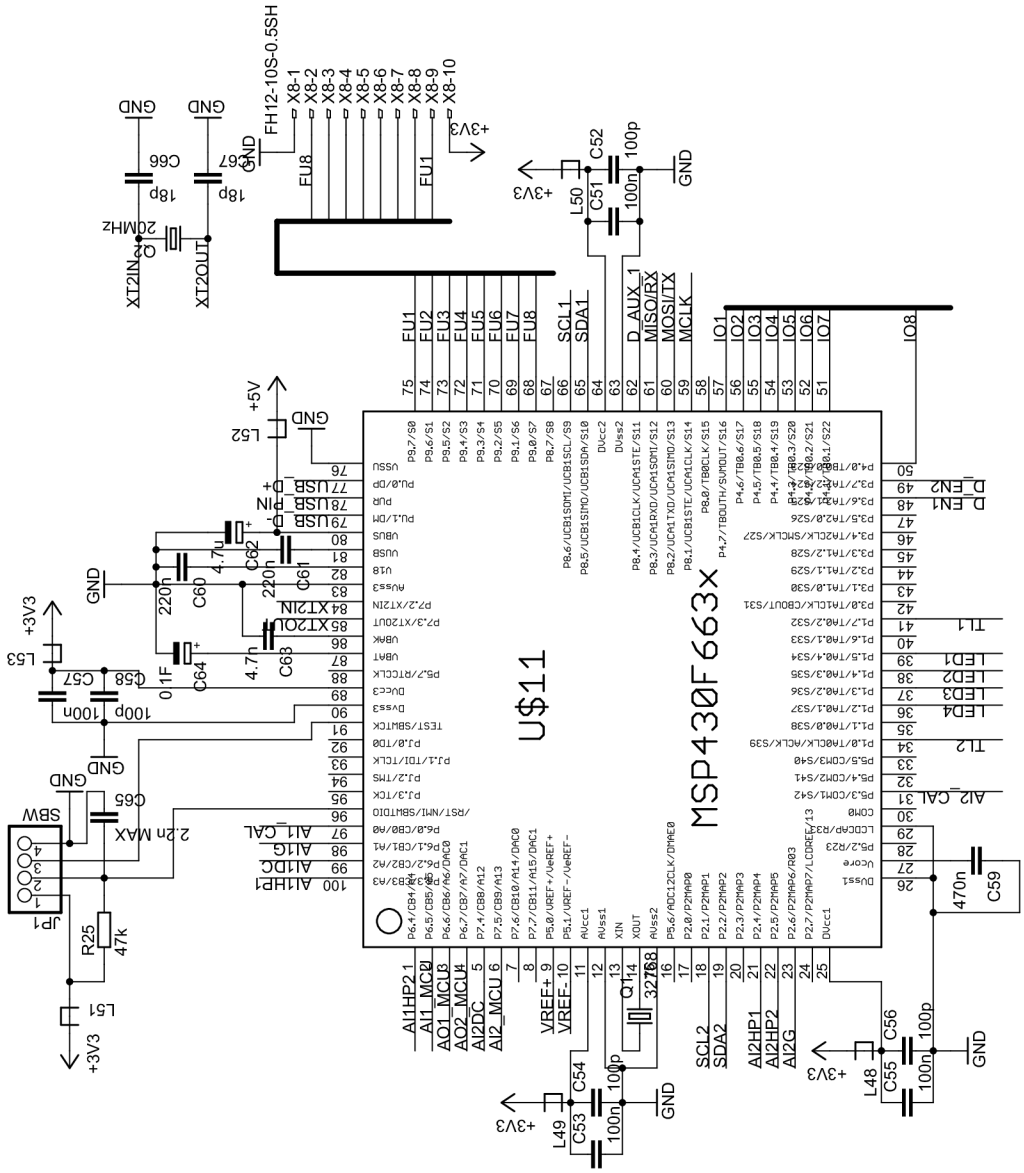
Nasledujú schémy zapojenia v nasledovnom poradí:

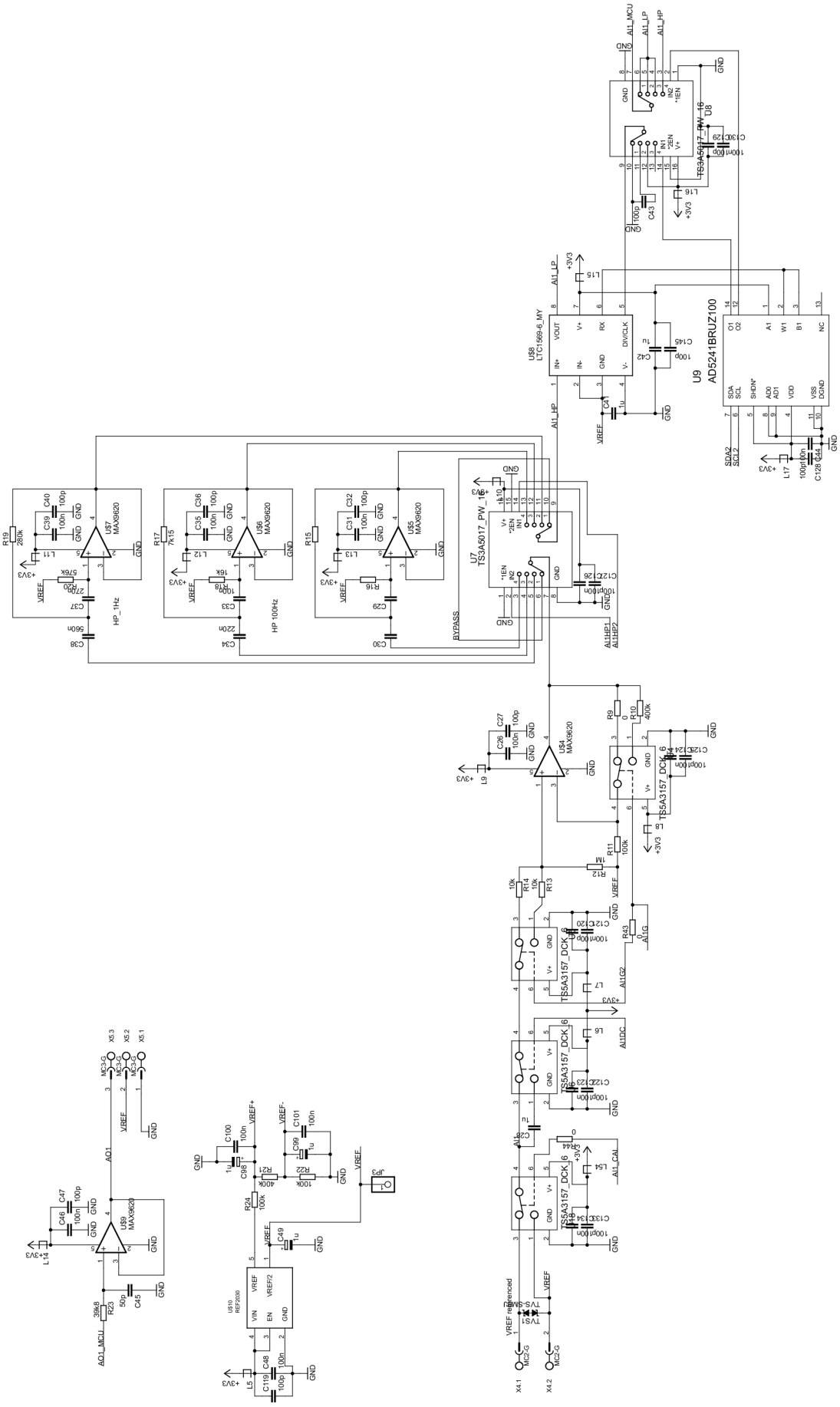
- Nastaviteľný menič na zmenu napäťovej úrovne, oddelenie zbernice SPI a obvody zmeny úrovne pre sériové zbernice (str. 90)
- Zapojenie oddelenia digitálnych vstupov a výstupov so zmenou úrovne, oddelenie zbernice I2C, zapojenie izolujúceho meniča napájacieho modul digitálnych vstupov a výstupov (str. 91)
- Zapojenie napájania, oddelenia USB, zapojenie signalizačných LED a ovládacích tlačidiel (str. 92)
- Zapojenie MCU (str. 93)
- Zapojenie obvodu napäťovej referencie, analógového výstupu a analógového vstupu (str. 94)


U2
TPS62200_DBV_5
VCC_ISO
V_REG
VI
EN
SW
FB
GND
C14
100n
C15
4,7u
10u
L1
V_FB
C16
10u
GNDI
U1
AD5241BRUZ100
SDA_ISO
SCL_ISO
SDA
SCL
SHDN*
AD0
AD1
VDD
VSS
DGND
O1
O2
A1
W1
B1
NC
ISO_EN1
ISO_EN2
GNDI
C17
V_FB
C18
R5
150k
V_REG
VCC_ISO
L43
C13
C108
100n
100p
GNDI
+3V3
L40
GND
100n
100p
C20
C104
IC5
VCC1
GND1
INA
INB
INC
OUTD
EN1
GND1
VCC2
GND2
OUTA
OUTB
OUTC
IND
EN2
GND2
ISO7241
MCLK
MOSI/TX
D_AUX_1
MISO/RX
GND
GNDI
VCC_ISO
L41
100n
100p
C19
C105
GNDI
MCLK_ISO
MOSI_ISO
D_AUX_ISO
MISO_ISO
SDA_ISO
SCL_ISO
GNDI
MAX3002E_MY
GND
EN
I/O_VCC8
I/O_VCC7
I/O_VCC6
I/O_VCC5
I/O_VCC4
I/O_VCC3
I/O_VCC2
VCC
I/O_VCC1
I/O_VL8
I/O_VL7
I/O_VL6
I/O_VL5
I/O_VL4
I/O_VL3
I/O_VL2
VL
I/O_VL1
US3
L42
C106
C21
100p
100n
GNDI
VCC_ISO
V_REG
R7
4k7
R8
4k7
SDA_OUT
SCL_OUT
MCLK_OUT
MOSI/TX_OUT
D_AUX_2
MISO/RX_OUT
L44
C107
C22
100p
100n
V_REG
GNDI
MC8-G
X3.8
X3.7
X3.6
X3.5
X3.4
X3.3
X3.2
X3.1


IO[1..8]
+3V3
L30
C17
100n
100p
IC1
ISO7240
VCC1
GND1
INA
INB
INC
IND
NC
GND1
VCC2
GND2
OUTA
OUTB
OUTC
OUTD
EN
GND2
IO5
IO6
IO7
IO8
VC5
VC6
VC7
VC8
ISO_EN1
C18
L34
IC2
D_EN1
L31
C8
C116
C115
L35
C3
C114
C113
L36
C4
IC3
IO4
IO3
IO2
IO1
VC4
VC3
VC2
VC1
ISO_EN2
L32
IC4
D_EN2
L33
C10
C111
GND
C6
C112
L37
VCC_ISO
GNDI
VC[1..8]
+5V
C23
2,2u
U3
DCR010505_DVB_12
VS_2
VS
NC
VREC
OVOUT
VO
SYNC
OVIN
OVIN_2
*ERROR
ENABLE
DNC
C24
C25
R6
10k
MAX3002E_MY
GND
I/O_VCC8
I/O_VCC7
I/O_VCC6
I/O_VCC5
I/O_VCC4
I/O_VCC3
I/O_VCC2
VCC
I/O_VCC1
EN
I/O_VL8
I/O_VL7
I/O_VL6
I/O_VL5
I/O_VL4
I/O_VL3
I/O_VL2
VL
I/O_VL1
U$1
C110
C2
C109
C1
L45
L46
V_REG
VL[1..8]
MC10-G
X2.10
X2.9
X2.8
X2.7
X2.6
X2.5
X2.4
X2.3
X2.2
X2.1
R4
4k7
R3
L38
C102
U$2
GALVANIC ISOLATION
VCC1
SDA1
SCL1
GND1
VCC2
SDA2
SCL2
GND2
ISO1540_MY
SDA_ISO
SCL_ISO
L39
C12
C103
R1
R2


DC2
+VIN
+VOUT
-VIN
-VOUT
DC/DC CONVERTER
5V/5V
DC1
C50
100n
C73
100n
X9.1
X9.2
MC2-G
MC2-G
5V6
D2
L2
12u
+5V
C72
4.7u
D1
5V6
GND
C96
100n
U17
TPS76733_D_8
GND
*EN
IN_2
IN
*RESET
FB/NC
OUT
OUT_2
+3V3
C97
10u
D3
3V5
X10.2
X10.1
X1
USB
PN61729-S
C71
200n
C70
200n
U10
ADUM4160BRWZ
VBUS1
GND
VDD
PDEN
SPU
UD-
UD+
GND_2
VBUS2
GND_3
VDD_2
SPD
PIN
DD-
DD+
GND_4
L47
C68
200n
C69
200n
R29
R28
24
24
R26
R27
24
24
R54
10k
USB_PIN
USB_D-
USB_D+
R45
R46
R47
R48
0
Q3
MUN2214T1
LED1
LED2
LED3
LED4
R49
33k
TL1
C147
470n
S1
R50
33k
TL2
C146
470n
S2


U$11
MSP430F663x
FH12-10S-0.5SH
X8-1
X8-2
X8-3
X8-4
X8-5
X8-6
X8-7
X8-8
X8-9
X8-10
FU1
FU8
XT2IN
XT2OUT
20MHz
Q2
18p
C66
C67
GND
+3V3
+5V
L50
L51
L52
L53
C51
C52
100n
100p
R25
47k
JP1
SBW
C65
2.2n MAX
C57
C58
C60
C61
C62
C63
C64
220n
4.7u
0.1F
4.7n
XT2IN
XT2OUT
AI1_CAL
AI1G
AI1DC
AI1HP1
AI1HP2
AI1_MCU
AO1_MCU
AO2_MCU
AI2DC
AI2_MCU
VREF+
VREF-
Q1
32768
C53
C54
L49
SCL2
SDA2
AI2HP1
AI2HP2
AI2G
L48
C55
C56
470n
C59
SCL1
SDA1
D_AUX_1
MISO/RX
MOSI/TX
MCLK
IO1
IO2
IO3
IO4
IO5
IO6
IO7
IO8
D_EN2
D_EN1
TL1
LED1
LED2
LED3
LED4
AI2_CAL
TL2

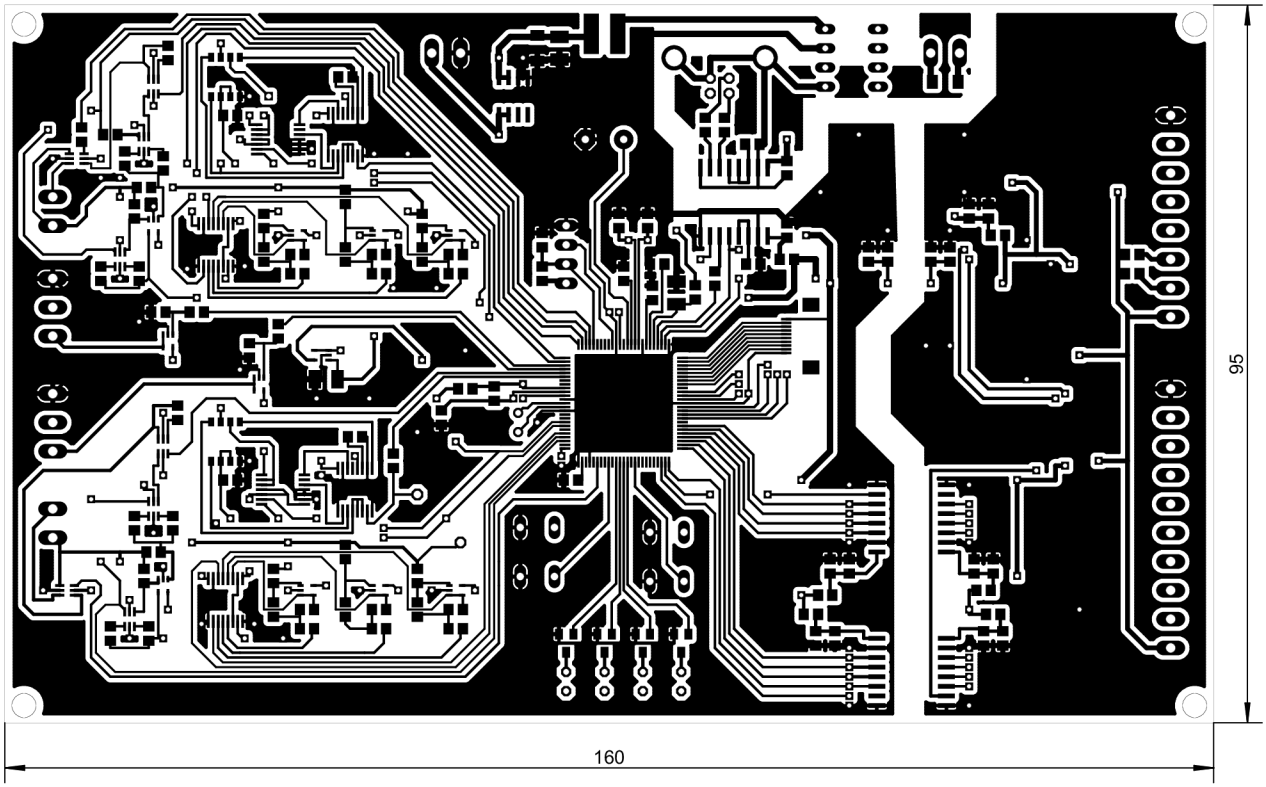

U9
AD5241BRUZ100
U58
LTC1569-6_MY
U7
TS3A5017_PW_16
U8
TS3A5017_PW_16
U54
MAX9620
U55
MAX9620
U56
MAX9620
U57
MAX9620
U59
MAX9620
U10
REF2030
TS5A3157_DCK_6
HP_1Hz
HP 100Hz
BYPASS
AI1_MCU
AI1_LP
AI1_HP
AI1G
AI1G2
AI1DC
AI1_CAL
AO1
AO1_MCU
VREF
VREF-
VREF+
VREF referenced
TVS-SMBJ
GND
+3V3
+3V0
SDA2
SCL2
X4.1
X4.2
X5.1
X5.2
X5.3
MC2-G
MC3-G
JP3

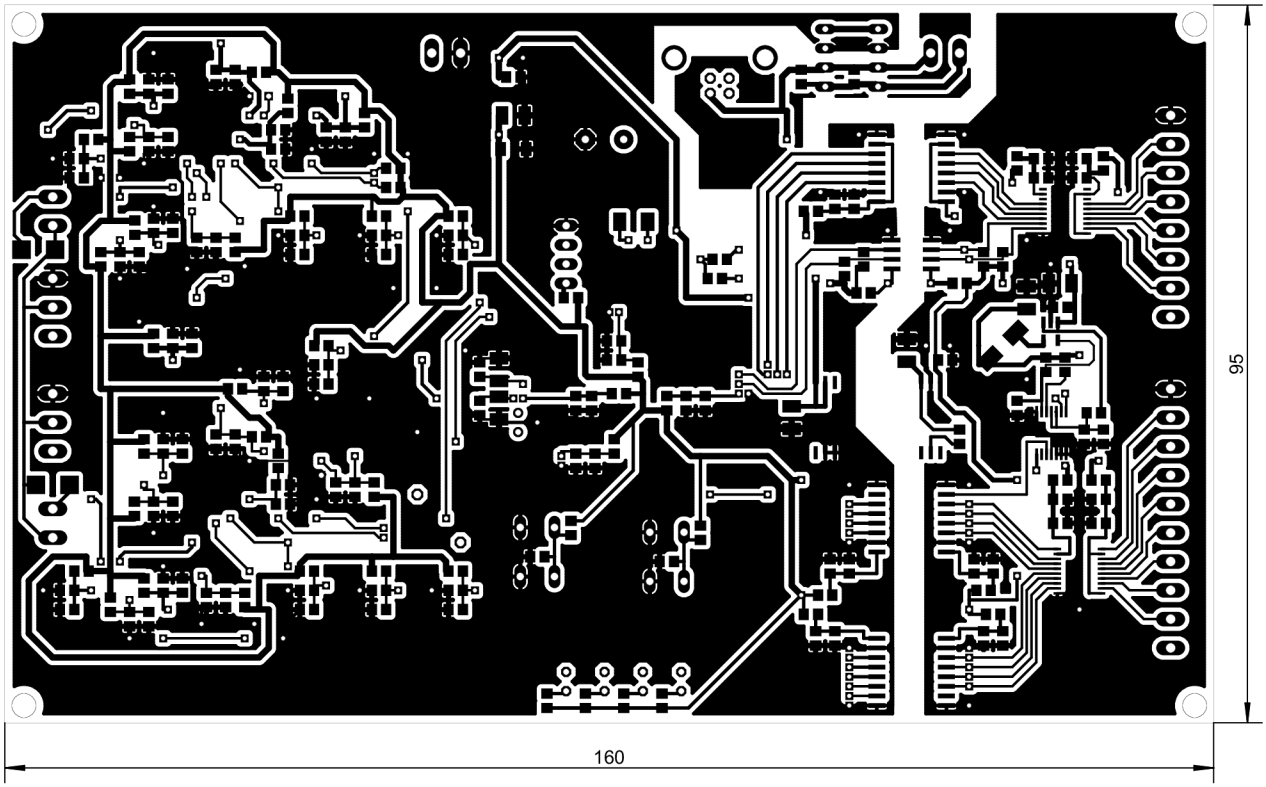
D DOSKA PLOŠNÝCH SPOJOV

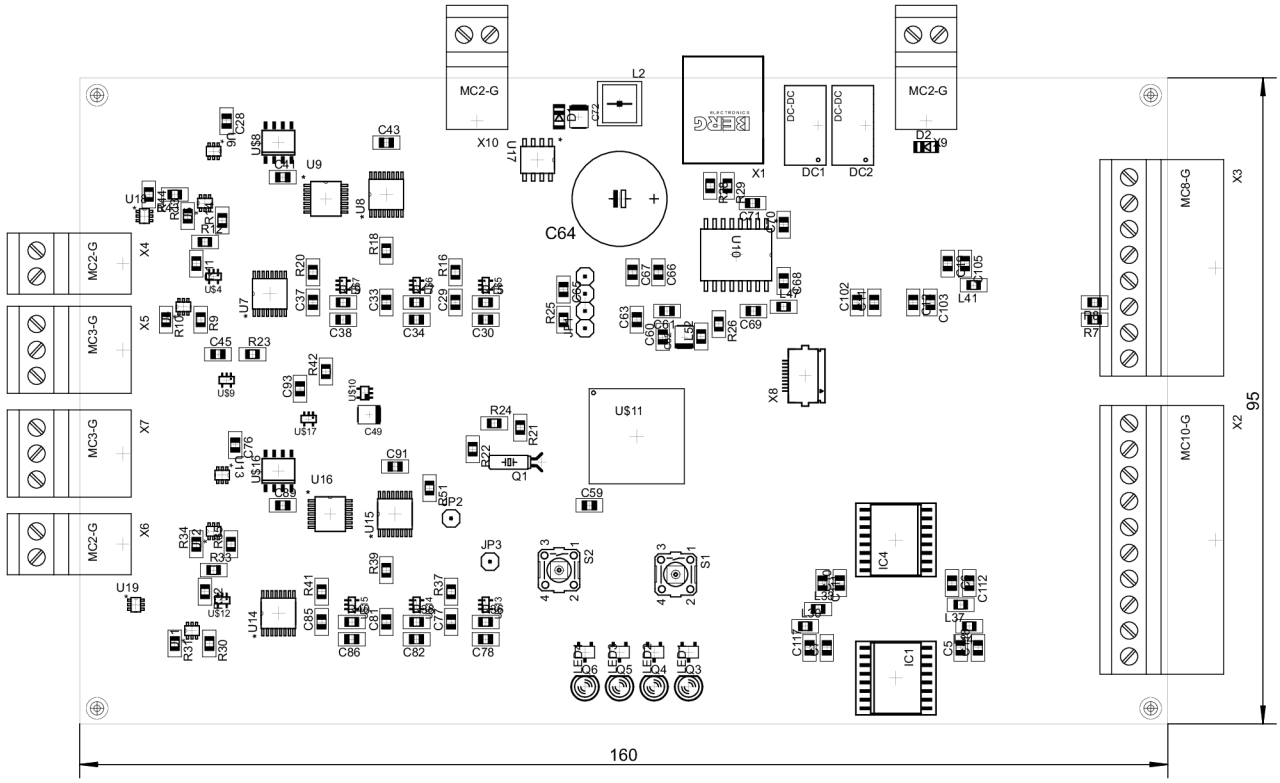
Nasledujú schémy zapojenia v nasledovnom poradí:

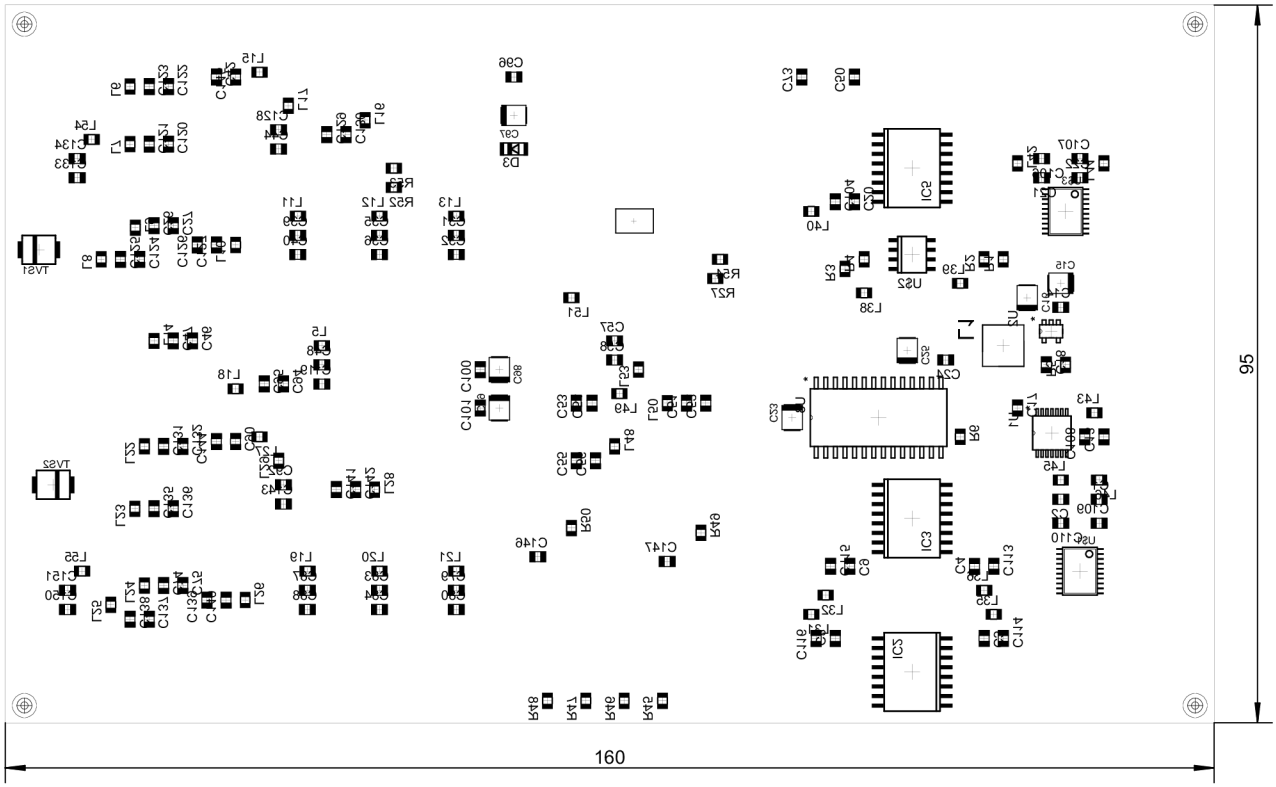
- Návrh plošného spoja - vrchná vrstva (str. 96)
- Návrh plošného spoja - spodná vrstva (str. 97)
- Rozloženie súčiastok - vrchná vrstva (str. 98)
- Rozloženie súčiastok - spodná vrstva (str. 99)

Rozmery DPS sú 160x95mm.


95
160


95
160


MC2-G
X10
L2
C72
D1
U17
X1
DC-DC
DC1
DC2
D2
C64
U10
C43
U$8
C28
U9
U8
U1
R28
R27
R29
C71
C70
X4
MC2-G
R18
R20
C37
C38
C33
C34
C29
C30
R16
C67
C66
C68
C102
C103
L41
C105
R25
C65
C63
C60
L3
R26
C69
X5
MC3-G
R10
R9
U7
C45
R23
U$9
C93
R42
U$10
C49
U$17
X8
R7
R6
MC8-G
X3
95
X7
MC3-G
R24
R21
R22
Q1
U$11
C76
U13
U$16
C99
U16
C91
U15
BP2
C59
X2
MC10-G
X6
MC2-G
R34
R33
R32
U19
U$12
U14
R41
C85
C86
C81
C82
C77
C78
R39
R37
JP3
S2
S1
IC4
IC1
C112
L37
C5
C117
R31
R30
LED1
LED2
LED3
LED4
C6
C8
C7
C3
C4
C2
C1
C9
C10
C11
C12
C13
C14
C15
C16
C17
C18
C19
C20
C21
C22
C23
C24
C25
C26
C27
C31
C32
C35
C36
C39
C40
C41
C42
C44
C46
C47
C48
C50
C51
C52
C53
C54
C55
C56
C57
C58
C61
C62
C73
C74
C75
C79
C80
C83
C84
C87
C88
C89
C90
C94
C95
C96
C97
C98
C100
C101
C104
C106
C107
C108
C109
C110
C111
C113
C114
C115
C116
C118
C119
C120
C121
C122
C123
C124
C125
C126
C127
C128
C129
C130
C131
C132
C133
C134
C135
C136
C137
C138
C139
C140
C141
C142
C143
C144
C145
C146
C147
C148
C149
C150
C151
C152
C153
C154
C155
C156
C157


160
95

E ZOZNAM SÚČIASTOK

Qty	Name/Value	Package	Parts	Description
2		B3F-10XX	S1, S2	Switch
6		C0805	C17, C18, C29, C30, C77, C78	Capacitor
51		R0805	L5, L6, L7, L8, L9, L10, L11, L12, L13, L14, L15, L16, L17, L18, L19, L20, L21, L22, L23, L24, L25, L26, L27, L28, L29, L30, L31, L32, L33, L34, L35, L36, L37, L38, L39, L40, L41, L42, L43, L44, L45, L46, L47, L48, L49, L50, L51, L52, L53, L54, L55	Ferrite bead
4		LED3MM	LED1, LED2, LED3, LED4	LED
4		R0805	R15, R16, R36, R37	Resistor
8	0	R0805	R9, R30, R43, R44, R45, R46, R47, R48	Resistor
1	0.1F	E5-13	C64	SuperCapacitor
4	100k	R0805	R11, R22, R24, R32	Resistor

Qty	Name/Value	Package	Parts	Description
55	100n	C0805	C1, C2, C3, C4, C5, C6, C7, C8, C9, C10, C11, C12, C13, C14, C19, C20, C21, C22, C24, C26, C31, C33, C35, C39, C44, C46, C48, C50, C51, C53, C55, C57, C73, C74, C79, C81, C83, C87, C92, C94, C96, C100, C101, C121, C123, C124, C126, C130, C132, C134, C136, C138, C140, C142, C151	Capacitor
50	100p	C0805	C27, C32, C36, C40, C43, C47, C52, C54, C56, C58, C75, C80, C84, C88, C91, C95, C102, C103, C104, C105, C106, C107, C108, C109, C110, C111, C112, C113, C114, C115, C116, C117, C118, C119, C120, C122, C125, C127, C128, C129, C131, C133, C135, C137, C139, C141, C143, C144, C145, C150	Capacitor
6	10k	R0805	R6, R13, R14, R34, R35, R54	Resistor
2	10u	SMC_B	C16, C97	Polarized Capacitor
1	10u	IND_ELL5PS	L1	Choke Coils

Qty	Name/Value	Package	Parts	Description
1	12u	WE-TPC 4818/4828	L2	Shielded Tiny Power Inductor
1	150k	R0805	R5	Resistor
2	16k	R0805	R18, R39	Resistor
2	18p	C0805	C66, C67	Capacitor
6	1u	C0805	C28, C41, C42, C76, C89, C90	Capacitor
4	1u	SMC_B	C25, C49, C98, C99	Polarized Capacitor
1	2.2u	SMC_B	C23	Polarized Capacitor
1	2.2n MAX	C0805	C65	Capacitor
4	200n	C0805	C68, C69, C70, C71	Capacitor
1	20MHz	7A	Q2	Crystal
4	220n	C0805	C34, C60, C61, C82	Capacitor
4	24	R0805	R26, R27, R28, R29	Resistor
2	270n	C0805	C37, C85	Capacitor
2	280k	R0805	R19, R40	Resistor
1	32768	TC26H	Q1	Crystal
2	33k	R0805	R49, R50	Resistor
2	39k8	R0805	R23, R42	Resistor
1	3V5	SOD80C	D3	Zener Diode
1	4.7n	C0805	C63	Capacitor
3	4.7u	SMC_B	C15, C62, C72	Polarized Capacitor
3	400k	R0805	R10, R21, R31	Resistor
3	470n	C0805	C59, C146, C147	Capacitor
1	47k	R0805	R25	Resistor
8	4k7	R0805	R1, R2, R3, R4, R7, R8, R52, R53	Resistor
2	50p	C0805	C45, C93	Capacitor
2	560n	C0805	C38, C86	Capacitor
2	576k	R0805	R20, R41	Resistor
2	5V/5V	NME	DC1, DC2	DC-DC Converter

Qty	Name/Value	Package	Parts	Description
2	5V6	SOD80C	D1, D2	Zener Diode
2	1M	R0805	R12, R33	Resistor
2	7k15	R0805	R17, R38	Resistor
3	AD5241-100	BRUZ	U1, U9, U16	Digital Potenciometer, 100k
1	ADUM4160	BRWZ	U10	USB Isolator
1	DCR010505	DVB12-L	U3	5V/5V, 1W, DC/DC Converter, Regulated output
1	FH12-10S-0.5SH	FH12-10S-0.5SH	X8	0.5mm Pitch Connectors For FPC/FFC
1	ISO1540	SO-08	U$2	Low-Power Bidirectional I2C Isolators
4	ISO7240	SO-16DW	IC1, IC2, IC3, IC4	High Speed Quad Digital Isolators
1	ISO7241	SO-16DW	IC5	High Speed Quad Digital Isolators
2	LTC1569-6	S8	U$8, U$16	Linear Phase, DC Accurate, Low Power, 10th Order Lowpass Filter
2	MAX3002E	TSSOP20	U$1, U$3	+1.2V to +5.5V, Level Translator
10	MAX9620	SC70-5L	U$4, U$5, U$6, U$7, U$9, U$12, U$13, U$14, U$15, U$17	Zero-Drift, OA
1	MC10-G	MC10-G	X2	Mini-Combicon 3.81mm 10-pole

Qty	Name/Value	Package	Parts	Description
4	MC2-G	MC2-G	X4, X6, X9, X10	Mini-Combicon 3.81mm 2-pole
2	MC3-G	MC3-G	X5, X7	Mini-Combicon 3.81mm 3-pole
1	MC8-G	MC8-G	X3	Mini-Combicon 3.81mm 8-pole
1	MSP430F5638	PZ100	U$11	MSP430 Microcontroller
4	MUN2214T1	SC59-BEC	Q3, Q4, Q5, Q6	NPN Bias Resistor Transistor
1	PN61729-S	PN61729-S	X1	USB connector
1	REF2030	SC70	U$10	3V/1,5V Voltage reference
1	SBW	1X04	JP1	Pin Header
1	TPS62200	SOT-23	U2	Adjustable, 300-mA, 95% Efficient Step-Down Converter
1	TPS76733	SOIC	U17	Single Output LDO, 1.0A, Fixed(3.3V)
4	TS3A5017	TSSOP	U7, U8, U14, U15	14-Ohm Dual SP4T Analog Switch
8	TS5A3157	SC70	U4, U5, U6, U11, U12, U13, U18, U19	10-Ohm SPDT Analog Switch
2	TVS-SMBJ	SMBJ	TVS1, TVS2	TVS BI-Directional