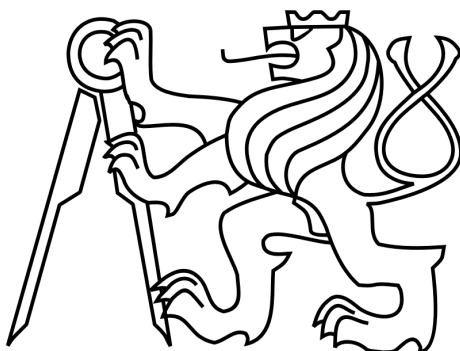


ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA ELEKTROTECHNICKÁ

KATEDRA MĚŘENÍ



DIPLOMOVÁ PRÁCE

Multifunkční deska sběru dat pro testovací účely

Autor: Bc. Jan Kříž

Vedoucí práce: doc. Ing. Jan Fischer, CSc.

Praha, 2016



ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: **Bc. Jan Kříž**

Studijní program: **Kybernetika a robotika**
Obor: **Senzory a přístrojová technika**

Název tématu česky: **Multifunkční deska sběru dat pro testovací účely**

Název tématu anglicky: **Multifunctional Data Acquisition Card for Testing Purposes**

Pokyny pro vypracování:

Navrhněte obvodové řešení multifunkční desky sběru dat, která bude určena pro testovací účely při vývoji a výrobě lékařských přístrojů. Analyzujte možnosti realizace takového testeru s využitím mikrořadičů řady STM32F4xx s orientací na testování vlastností vyráběných přístrojových bloků. Navrhněte a realizujte programy potřebné pro spolupráci mikrořadiče s PC a LabVIEW. Určete vlastnosti a meze použitelnosti navrženého testeru.

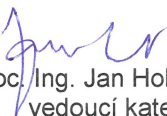
Seznam odborné literatury:

- [1] Yiu, J.: Definitive Guide to ARM CortexR-M3 and CortexR-M4 Processors
- [2] STMicroelectronics: RM0090 Reference manual
- [3] STMicroelectronics: DM00071990, STM32F429 Data

Vedoucí diplomové práce: doc. Ing. Jan Fischer, CSc.

Datum zadání diplomové práce: 14. prosince 2015

Platnost zadání do¹: 30. září 2017


Doc. Ing. Jan Holub, Ph.D.
vedoucí katedry



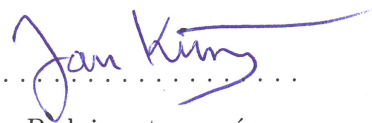

Prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 14. 12. 2015

¹ Platnost zadání je omezena na dobu tří následujících semestrů.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 27.5.2016



Podpis autora práce

Název práce: Multifunkční deska sběru dat pro testovací účely

Abstrakt

Tato práce řeší hardwarový a softwarový návrh multifunkční měřicí desky s mikrokontrolérem STM32F429, včetně realizace komunikace desky s PC a knihovných funkcí pro ovládání měření pomocí programu LabVIEW.

Toho bylo dosaženo návrhem vhodného protokolu pro komunikaci mikrokontroléru s PC po sériové lince a snadno rozšiřitelného systému obsluhy měření s kontrolou měřících prostředků.

Koncepce byla ověřena implementací 6 typů měřících funkcí (jednoduché měření napětí, měření napětí s nastavitelným vzorkováním, nastavitelný zdroj napětí, generátor PWM, ovládání digitálních vstupů a výstupů), a k nim příslušných knihoven pro program LabVIEW. S ohledem na implementované měřicí úlohy bylo navrženo doporučené hardwarové řešení měřicí desky včetně periférií mikrokontroléru nutných pro jeho bezchybný běh.

Na základě navrženého software a hardware je možné realizovat univerzální a funkčně snadno rozšiřitelnou měřicí desku postavenou na mikrokontroléru STM32F429, schopnou komunikovat s PC pomocí sériové linky s možností ovládání měření pomocí terminálu této linky, nebo navržených knihovných funkcí do programu LabVIEW

Title: Multifunction Data Acquisition Card for Testing Purposes

Abstract

This thesis deals with the software and hardware proposal of a multifunctional acquisition card built on the microcontroller STM32F429. Proposal includes design of the acquisition card, PC communication problematics and library functions for the measurement control with the LabVIEW program.

The goal was achieved by the design of a suitable protocol for the communication between microcontroller and PC via serial line and easily expandable system of measurement resources handling and control.

The concept was verified by implementation of six different types of measurement functions (simple voltage measurement, voltage measurement with adjustable sampling rate, adjustable voltage source, PWM generator, digital input and output control) and their relevant libraries for the LabVIEW program. Regarding the implemented measurement, the hardware solution of the acquisition card was designed including the microcontroller peripherals necessary for a faultless run.

On the basis of the designed software and hardware it is possible to realize universal and simply expendable acquisition card built on the microcontroller STM32F429 which is capable of communication with a PC via serial line. The measurement can be controlled by the terminal of the serial line or by the designed library functions for the LabVIEW program.

Rád bych na tomto místě poděkoval doc. Ing. Janu Fischerovi, CSc. za odborné vedení práce, velmi cenné rady, náměty a vstřícnost během celé doby spolupráce. Také děkuji své rodině a přátelům za neocenitelnou podporu během studia.

Obsah

Obsah	b
1 Úvod	1
2 Současný stav řešené problematiky a cíle práce	2
2.1 Měření a testování ve firmě BTL	2
2.2 Požadavky na vývoj měřící desky a cíle práce	3
2.3 MCU STM32F429II	4
2.3.1 Vnitřní struktura STM32F429	4
2.3.2 Vývojový kit	6
2.4 Práce s virtuálními multifunkčními kartami pomocí programu LabVIEW	6
2.5 Posouzení vhodnosti užití MCU STM32F429II pro realizaci měřící desky a stanovení strategie	8
3 Softwarová koncepce multifunkční měřící desky a návrh jejích funkcí	10
3.1 Výběr komunikačního rozhraní	10
3.2 Návrh komunikačního protokolu pro obsluhu desky	11
3.2.1 Obsluha komunikačního protokolu	14
3.2.2 Systém kontroly prostředků karty	16
3.3 Čtení digitální vstup a zápis na digitální výstupy měřící karty	17
3.4 Generování pulsně šířkové modulace (PWM) pomocí čítačů	18
3.5 Měření napětí integrovaným A/D převodníkem	19
3.6 Zdroj stejnosměrného napětí pomocí integrovaného D/A převodníku	23
3.7 Uchování kalibračních dat	24
3.8 Čtení identifikačního čísla MCU	24
4 Knihovna funkcí pro program LabVIEW	25
4.1 Integrace navrženého komunikačního protokolu do prostředí LabVIEW	25
4.1.1 Posílání příkazů MD a příjem odpovědi	26
4.1.2 Kontrola chyb MD a komunikace	26
4.2 Návrh obsluhy měřící úlohy	28
4.2.1 Vytvoření měřící úlohy a její propagace programem	28
4.2.2 Práce s více piny (kanály) v jedné úloze	29
4.2.3 Ukončení měřící úlohy	30
5 Hardwarový návrh měřící desky	31
5.1 Návrh obvodů pro předzpracování signálu navržených funkcí MCU a jeho ochranu	31
5.1.1 Obvod pro zdroj stejnosměrného napětí	31
5.1.2 Vstupní obvod pro měření stejnosměrného napětí	32
5.1.3 Vstupní obvod pro vzorkované měření napětí	33
5.2 Napájení MD	34

6	Zhodnocení dosažených výsledků	37
6.1	Měření stejnosměrného napětí	37
6.2	Kontinuální měření napětí s definovaným vzorkováním	38
6.3	Generování pulsně šířkové modulace (PWM)	40
6.4	Zdroj stejnosměrného napětí	41
6.5	Řízení digitálních vstupů a výstupů	43
6.6	Obecné funkce měřicí karty a její další možná rozšíření	43
6.7	Paleta LabVIEW	44
7	Závěr	46
	Literatura	48
	Seznam obrázků	50
	Seznam tabulek	51
	Seznam zkratk	52
	Příloha A Obsah příloženého CD	I
	Příloha B Strom komunikačních příkazu pro MD	II
	Příloha C Rozvržení funkcionalit pro jednotlivé piny MD	IV
	Příloha D Definované chybové kódy MD	VIII
	Příloha F Kompletní obvodové schéma MD	IX

1 Úvod

Společnost BTL Zdravotnická technika a.s. (dále jen BTL), která vyrábí lékařské zařízení, naráží na neustálou potřebu testování. Toho je potřeba už ve fázi vývoje nového přístroje, kdy výsledky testů pomáhají vývojářům při jeho návrhu. Velkým tématem je pak testování při fázi výroby finálního zařízení, kde jsou kladeny velké požadavky na kvalitu a spolehlivost výsledných výrobků jdoucích k zákazníkům. Častým typem tohoto testování je automatizované měření parametrů desek plošných spojů výrobků při jejich oživování. Tato měření v dnešní době dávají díky novým technologiím informaci nejen o chybách výroby, nýbrž i podklad pro dlouhodobé statistické sledování parametrů vyrobených přístrojů.

Z důvodu potřeby standardizace firemních procesů jsou pro velkou různorodost vyvíjených zařízení, a s tím spojenými rozličnými nároky na testování, využívány tzv. multifunkční měřicí desky (někdy také nazývané jako univerzální). Výhodou těchto desek je možnost vykonávat mnoho různých typů měření na velkém množství vstupů a výstupů s velkým rozsahem hodnot vstupních i výstupních veličin. Ovládání multifunkčních desek a řízení testování zpravidla provádí specializovaný software, kterým je v případě BTL program LabVIEW společnosti National Instruments.

Systém měření a testování v BTL prochází postupným vývojem. Během něj se ukázalo, že tyto multifunkční karty, i přes své výborné parametry, jejichž důvodem je právě jejich univerzálnost, jsou mnohdy nasazovány neefektivně. Důvodem je fakt, že prováděné testy z větší části spočívají v jednodušších měřících operacích, na větším množství měřících vstupů a výstupů (např. měření stejnosměrného napětí, generování pulsně šířkové modulace nebo nastavování a čtení logických úrovní vstupů a výstupů). Vzniká tedy poptávka po funkčně specifičtější tzv. „low-endové“ multifunkční měřicí kartě s větším počtem měřících vstupů a výstupů zvládajících základní měřící operace, a s kterou by se dalo měřit pomocí LabVIEW.

Možné řešení se naskýtá v nasazení běžně dostupných mikrokontrolérů (MCU) pro vestavné systémy, které jsou standardně vybaveny širokou paletou periférií (ADC, DAC, čítačové jednotky, SPI, I2C, USB atd.) v jednom pouzdře integrovaného obvodu. To přináší nové možnosti jejich nasazení, například pro základ multifunkční měřicí desky. Dalším možným budoucím využitím by mohla být možnost přímého osazení MCU na desku plošného spoje vyráběného přístroje, a to ve funkci bloku pro jeho interní diagnostiku.

Cílem této práce je proto analyzovat vhodnost MCU STM32F429II (vybraného společností BTL), pro vytvoření multifunkční měřicí karty se stanovenými požadavky, které pak pomocí vývojové sady pro daný MCU koncepčně realizovat.

2 Současný stav řešené problematiky a cíle práce

2.1 Měření a testování ve firmě BTL



Obrázek 2.1: Příklad přístroje BTL Vanquish[1]

V úvodu bylo zmíněno, že se firma BTL zabývá výrobou zdravotnických přístrojů. Jejich zaměření je zejména na fyzioterapii, kardiologii, spirometrii, a v neposlední řadě také na přístroje pro estetickou medicínu. Příkladem takového přístroje je vysokofrekvenční systém pro bezkontaktní neinvazivní odstranění tuku BTL Vanquish viz obrázek 2.1.

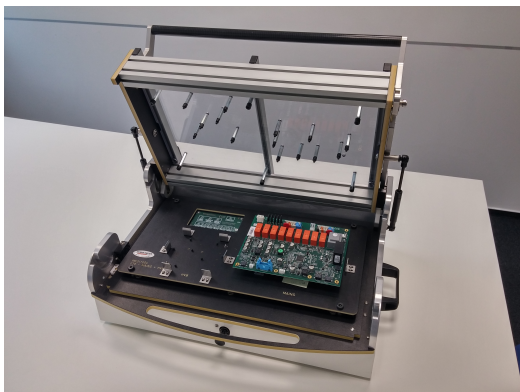
Všechny navrhované přístroje je firma schopna kompletně vyrobit pomocí sítě vlastní výroby. Aby toho mohla docílit, musí mimo jiné disponovat systémem kontroly kvality. Jednou z řady takových kontrol je testování vlastností a výstupních parametrů osazených desek plošných spojů, které musí splňovat v oblasti zdravotní techniky přísná kritéria. Pro tento typ testování výroba využívá tzv. *fixture* (viz obrázek 2.2).

Do těchto *fixtur* jsou dělníky usazovány jed-

notlivé desky plošných spojů, které jsou po uzavření *fixtury* automaticky testovány pomocí multifunkčních měřících desek a programu LabVIEW.

Multifunkční měřící desky (MD) používané v BTL spadají do kategorie tzv. virtuálních měřících přístrojů. Důvodem je absence jakýchkoliv řídicích či zobrazovacích prvků na měřící kartě, které jsou suplovány softwarově.

Typickým příkladem běžně užívané multifunkční měřící karty v BTL je NI 6008 společnosti National Instruments, jež je v tzv. OEM verzi na obrázku 2.3. Základní parametry této karty jsou v tabulce 2.1. Pro naprogramování této virtuální karty se využívá grafického programovacího jazyka „G“ a programu LabVIEW 2.6.



Obrázek 2.2: Fixtura pro testování osazených desek plošných spojů



Obrázek 2.3: Univerzální měřící deska NI 6008 ve verzi se šroubovacími konektory a OEM verzi [2]

Funkcionalita	Parametr	Hodnota
Analogové vstupy	Počet	4diferenciální nebo 8 jednoduchých
	Rozlišení	12bit jednoduchý, 13bit diferenciální vstup
	Vstupní rozsah	$\pm 10V$, $\pm 5V$, $\pm 4V$, $\pm 2.5V$, $\pm 2V$, $\pm 1.25V$, $\pm 1V$ (programově nastavitelný)
	Rychlost vzorkování	10K S/s
Analogové výstupy	Počet	2
	Rozlišení	12bit
	Výstupní rozsah	0 – 5V
Digitální vstupy a výstupy	Počet	12
Čítače	Počet	1
	Generování PWM	Ne
	Měření frekvence	Ano (do 5MHz, minimální šířka pulsu 5ns)

Tabulka 2.1: Stručné shrnutí parametrů měřící karty NI 6008[3]

2.2 Požadavky na vývoj měřící desky a cíle práce

Jak již bylo zmíněno v úvodu této práce, cílem je analyzovat vhodnost MCU STM32F429II společnosti ST Microelectronics pro vývoj vlastní multifunkční desky navržením demonstračního zařízení. To by mělo dle společnosti BTL splňovat požadavky a specifikace uvedené v tabulce (2.2).

Oblast	Požadavky
Analogové výstupy	▷ Zdroj stejnosměrného napětí (DC) v rozsahu 0V až 3V s výstupním proudem nejméně 10mA

Oblast	Požadavky
Hardwarový návrh	<ul style="list-style-type: none"> ▷ Použití MCU STM32F429II ▷ Využití v maximální možné míře firemního výběru součástek ▷ Napájení pomocí USB. ▷ Při komunikaci pomocí USB využívat maximálně jeho standard 2.0 full-speed (z důvodu přípravy na galvanické oddělení desky)
Obecné požadavky	<ul style="list-style-type: none"> ▷ V případě daných funkcí využít maximálně možnosti MCU - nalezení kompromisu mezi kvalitou, rychlostí a použitím co nejvíce výstupů a vstupů MCU. ▷ Nalezení možnosti uchovávání kalibračních dat uvnitř MCU ▷ Zajištění systému unikátní identifikace jednotlivých měřících desek
Analogové vstupy	<ul style="list-style-type: none"> ▷ Měření stejnosměrného napětí (DC) v rozsahu alespoň 0V až 3V, na co nejvíce kanálech ▷ Měření napětí se vzorkováním alespoň 10kHz v rozsahu alespoň -1V až 1V
Digitální vstupy a výstupy	<ul style="list-style-type: none"> ▷ Generování pulsně šířkové modulace (PWM) s možností měnit frekvenci modulace a střidu ▷ Možnost číst logické úrovně vstupů a nastavovat logické úrovně výstupů v rozsahu napětí TTL logiky
Komunikace	<ul style="list-style-type: none"> ▷ Pro komunikaci s multifunkční kartou aplikovat vhodný protokol kompatibilní s programem LabVIEW ▷ Možnost pracovat s deskou přes terminál

Tabulka 2.2: Požadavky společnosti BTL na novou multifunkční měří kartu

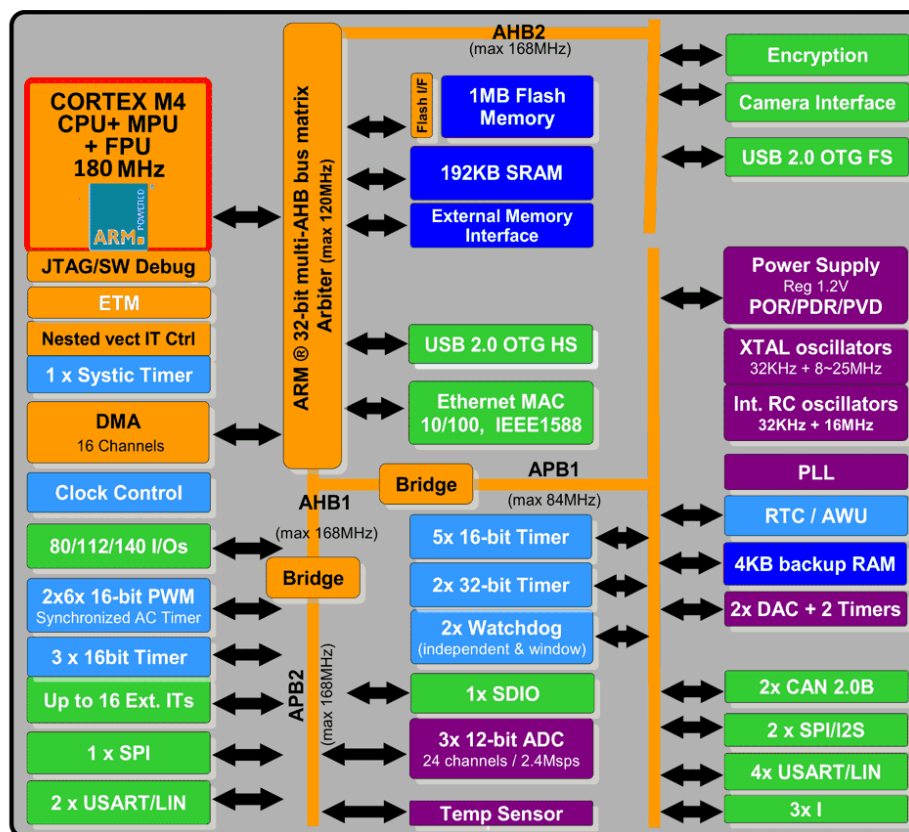
2.3 MCU STM32F429II

Jedná se o model MCU z rodiny Cortex-M, jenž využívá 32bitový procesor architektury ARMv7-M [4]. Ve společnosti BTL byl zaveden jakožto velmi výkonný a bohatě vybavený MCU pro práci s displeji. Pro ně má MCU integrovaný řadič a z tohoto důvodu bylo také zvoleno pouzdro s nadprůměrným počtem 176 vývodů (pinů), což je pro vývoj univerzální měřící desky společně s jeho výkonem a výbavou klíčový prvek.

2.3.1 Vnitřní struktura STM32F429

Charakteristickým rysem STM32F429 je bohatá paleta periférií, což je patrné na blokovém schématu 2.4. Aby bylo možné s těmito perifériemi fyzicky pracovat, obsahuje MCU sběrnice (AHB1, AHB2, APB1, APB2), které přeměrovávají programově zvolenou periférii k vybranému pinu - ten už je fyzicky vyveden z pouzdra MCU. I přes velikou variabilitu nelze toto přeměrování dělat libovolně, nýbrž každá periferie má seznam pinů, na kterých může být provozována.

Shrnutí důležitých parametrů a vlastností STM32F429II je uveden v tabulce 2.3.



Obrázek 2.4: Vnitřní struktura STM32F429[6]

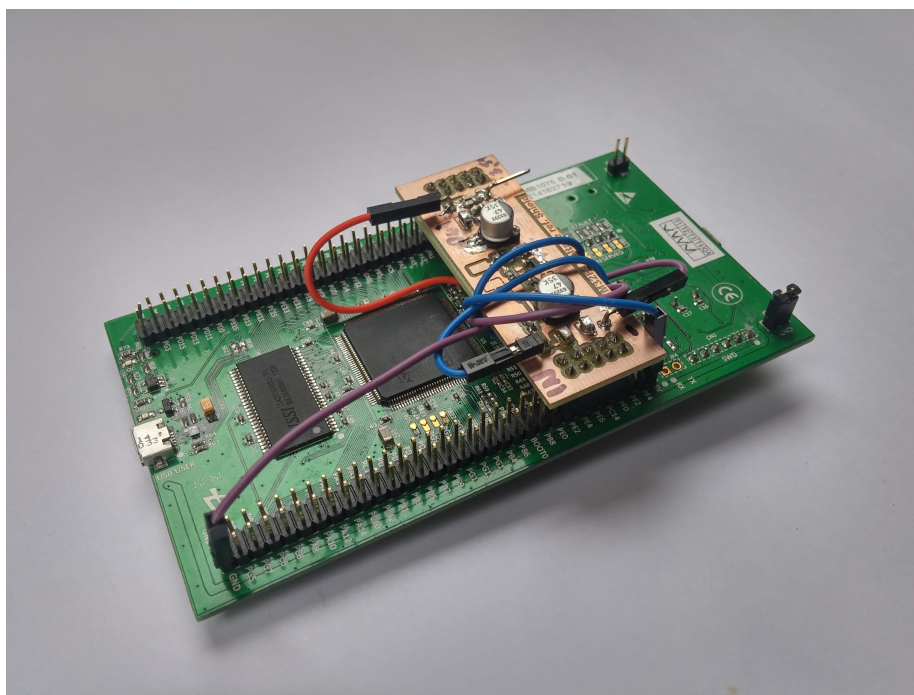
Funkcionalita	Vlastnosti
Analogově digitální převodník (A/D)	<ul style="list-style-type: none"> ▷ 3 nezávislé převodníky s rozlišením 12bit ▷ Rychlostí vzorkování až 2,4MS/s (jeden převodník) ▷ Rychlostí vzorkování až 7,4MS/s (3 převodníky) ▷ Až 24 měřících kanálů
Digitálně analogový převodník (D/A)	<ul style="list-style-type: none"> ▷ rozlišení až 12bitů ▷ 2 kanály
Čítače (<i>Counters</i>)	<ul style="list-style-type: none"> ▷ 14 čítačů s rozlišením až 32-bit ▷ Rychlost časové základny až 84MHz ▷ Až 4 vstupně výstupní komparační jednotky na čítač (<i>I/O capture/compare unit</i>) s možností generování PWM, čítání pulsů, měření délky pulsu, čtení kvadraturního enkodéru
Digitální vstupy a výstupy	<ul style="list-style-type: none"> ▷ 168 vstupů a výstupů, z toho 166 5V tolerantních ▷ Rychlost až 90MHz
Komunikační periferie	<ul style="list-style-type: none"> ▷ 4xUART s rychlostí až 11.25Mbit/s, podpora LIN a IrDA ▷ 3x I²Cs podporou SMBus/PMBus ▷ 6x SPI s rychlostí až 45Mbit/s ▷ Řadič USB 2.0 Full-speed (12Mbit/s), High-speed (480Mbit/s) pouze s podporou externího budiče ▷ Ethernet 10/100 pouze s externím budičem ▷ CAN (2.0B) pouze s externím budičem

Funkcionalita	Vlastnosti
Další funkce a vlastnosti	<ul style="list-style-type: none"> ▷ Unikátní ID v každém MCU ▷ DMA kontrolér pro většinu periférií ▷ Frekvence procesoru až 180 MHz ▷ podpora aritmetiky v plovoucí čárce ▷ Integrovaný teplotní senzor

Tabulka 2.3: Výbava MCU STM32F429II[5]

2.3.2 Vývojový kit

Pro usnadnění vývoje byl využit tzv. *DISCOVERY KIT* společnosti ST Microelectronics (viz obrázek (2.5)). Ten je osazen stejným typem MCU, což je výhodné při testování potřebných vlastností a implementovaných funkcí s jistotou jejich neměnnosti při finální realizaci MD. I z tohoto důvodu bylo rozhodnuto upravit napěťovou referenci, ovlivňující výsledky analogových měření, dle finálního návrhu napěťové reference v bodě(5.2).



Obrázek 2.5: Discovery kit s MCU STM32F429 s realizovanou externí napěťovou referencí

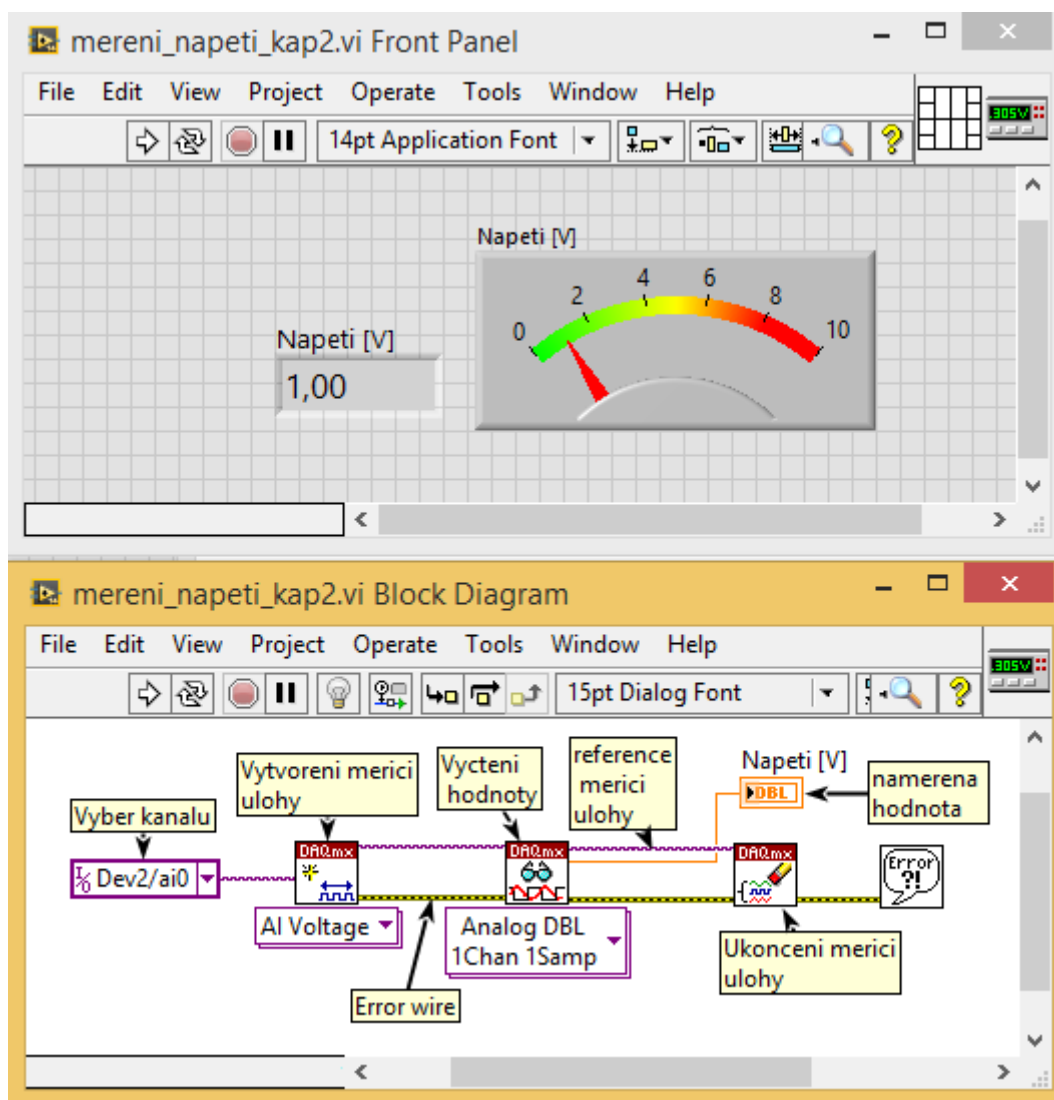
2.4 Práce s virtuálními multifunkčními kartami pomocí programu LabVIEW

LabVIEW je vývojové prostředí pro grafický programovací jazyk „G“ vyvinutý společností National Instruments primárně k měření pomocí virtuálních přístrojů a zpracování naměřených dat.

Díky mnoha knihovním funkcím, intuitivnímu prostředí, vysoké míře modularity programů

2.4 Práce s virtuálními multifunkčními kartami pomocí programu LabVIEW

a snadnému přístupu, velmi usnadňuje práci vývojářům, kteří do té doby při potřebě měření mnohokrát museli vyvíjet vlastní měřicí nástroje a systém pro jeho režii či správu.



Obrázek 2.6: Program pro odběr vzorku napětí pomocí měřicí karty a standardu DAQ v programu LabVIEW

Kromě programu LabVIEW se v nabídce National Instruments objevuje velké množství virtuálních multifunkčních měřících přístrojů, které se mnohdy velmi liší svým posláním. Všechny tyto přístroje mají jeden společný rys, a tím je obsluha pomocí LabVIEW skrze uzavřený standard *National Instruments DAQmx* a jeho knihovní funkce. Zákazník tak má jistotu, že se všechny typy měření, se všemi multifunkčními měřicími přístroji od National Instruments ovládají dle stejného schématu.

Ukázka měření v programu LabVIEW je na obrázku 2.6. Zde je vyobrazen tzv. *block diagram* (dolní okno aplikace) obsahující naprogramovaný kód měření. Tzv. *front panel*, který slouží jako ovládací rozhraní pro uživatele a jako zobrazovač naměřených hodnot (horní okno aplikace).

Obrázkem 2.6 zobrazený program nastiňuje zmíněné schéma práce se standardem DAQmx.

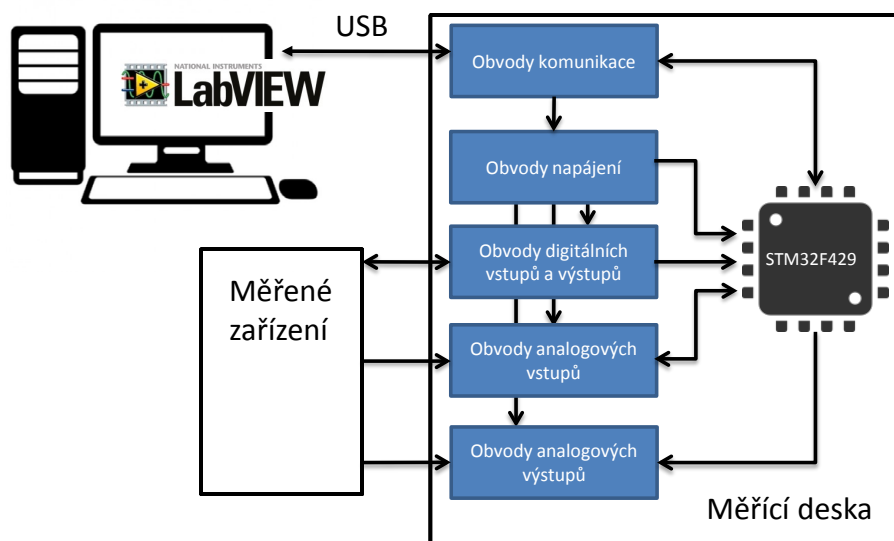
Účel tohoto programu je změření jedné hodnoty napětí na jednom napěťovém vstupu multifunkční měřicí desky. Plynutí programu určují naznačené „kabely“ sloužící k předávání parametrů a hodnot mezi jednotlivými bloky tzv. VI (*virtual instrument file*) a určují tak kauzalitu programu.

V případě jakékoliv chyby, která v programu nastane (například přerušení spojení s měřicí kartou) je tato informace předávána pomocí tzv. *error wire*, který je na zmiňovaném obrázku naznačen žlutozeleně. Jeho funkce je informovat jednotlivá VI programu o nastalém problému a upozornit na něj po skončení programu uživatele.

2.5 Posouzení vhodnosti užití MCU STM32F429II pro realizaci měřicí desky a stanovení strategie

Vize návrhu nové multifunkční měřicí desky (MD) je vyobrazena jednoduchým blokovým schématem na obrázku (2.7). Tato vize vychází z předpokladu, že pro měření nebude využito jiných specializovaných součástek, ale pouze MCU, které bude jádrem celé implementace MD.

Srovnáním zadaných požadavků v bodě 2.2 a funkcionalit MCU v bodě 2.3 bylo zjištěno, že je MCU STM32F429II je schopný dostát všech požadavků zadání.



Obrázek 2.7: Jednoduché blokové schéma návrhu měřicí desky

K dosažení zadaných cílů bude třeba najít spolehlivý způsob komunikace MCU s počítačem, dále vhodný protokol pro ovládání měřicí karty a přenos naměřených dat. Tento protokol by měl být snadno rozšiřovatelný a implementovatelný do programu LabVIEW.

Po těchto krocích bude možné implementovat jednotlivé měřicí funkce. Dokončením návrhu softwaru budou známy parametry jednotlivých funkcí, které se promítnou na následný HW návrh MD.

2.5 Posouzení vhodnosti užití MCU STM32F429II pro realizaci měřící desky a stanovení strategie

Na závěr celé práce bude potřeba vytvořit potřebné knihovny pro spolupráci MD s programem LabVIEW.

3 Softwarová koncepce multifunkční měřicí desky a návrh jejích funkcí

3.1 Výběr komunikačního rozhraní

Volba správné fyzické vrstvy pro komunikaci ovlivňuje zejména rychlost přenosu dat do a z měřicí karty, a tím i její výsledný výkon. Dále pak spolehlivost tohoto přenosu, na který je brán u měření velký důraz (odeslaná data musí být druhou stranou přijata a interpretována bez chyb). V úvahu musí být brán i požadavek z bodu (2.2), aby bylo možné MD snadno ovládat člověkem pochopitelnými příkazy a syntaxí, ideálně tedy skrze textový terminál.

Jak bylo zmíněno v bodě(2.3), STM32F429 podporuje více komunikačních standardů, mezi kterými lze brát v úvahu pro komunikaci s počítačem USB, Ethernet a UART, jejichž výhody a nevýhody shrnuje tabulka (3.1). Z této tabulky je patrné, že ideálním standardem je USB, které lze provozovat spolehlivě (bez chybných dat) a dnes jím disponuje každý počítač. Navíc podporuje tzv. třídu CDC (*communications device class*), což je abstraktní vrstva, která mimo jiné dokáže emulovat chování protokolu UART (jinak také Virtual COM port).

Periferie	Výhody	Nevýhody
USB	<ul style="list-style-type: none"> - Lze skrze něj napájet zařízení, - Dokáže emulovat sériovou linku (USB VCP) - Existuje pro něj speciální komunikační třída TMC (<i>Test and Measure Class</i>) - Spolehlivý 	<ul style="list-style-type: none"> -Komplikované galvanické oddělení
Ethernet	<ul style="list-style-type: none"> - Galvanicky oddělený - Zařízení lze provozovat na libovolném místě v síti - Spolehlivý 	<ul style="list-style-type: none"> - Potřeba externího budiče periferie - Chybí napájení (v podnikové síti) - Složitá implementace
USART	<ul style="list-style-type: none"> - Jednoduchý - Přenáší nativně znaky bez dalšího protokolu 	<ul style="list-style-type: none"> - Potřeba budiče nebo převodníku pro komunikaci s PC - Nezajišťuje kontrolu chyb

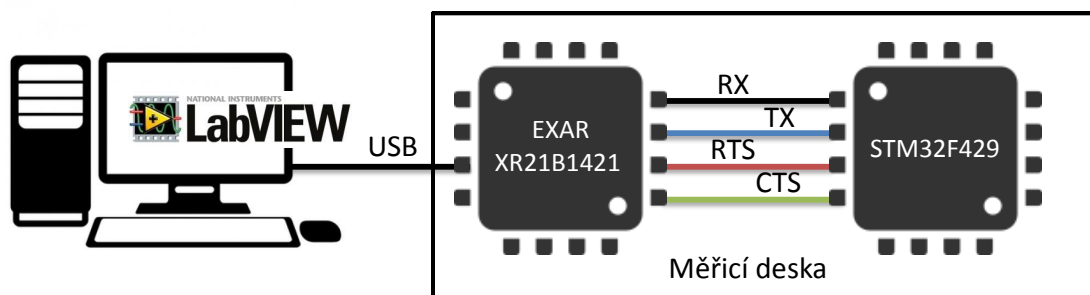
Tabulka 3.1: Uvažovaná komunikační rozhraní MCU s PC

V praxi se však ukázalo, že v přímé implementaci na STM32F4 může USB vykazovat

Parametr	Hodnota
Počet datových bitů	8
Parita	1 bit, lichá
Handshaking	CTS/RTS
Baudrate	11,25
Počet stopbitů	1

Tabulka 3.2: Parametry nastavení komunikace UART

nepředvídatelnou činnost vlivem komplikované vnitřní obsluhy, a tím také velkého záboru výkonu MCU. Z těchto důvodů byl pro snížení zatížení MCU a celkové zjednodušení zvolen kompromis - použití externího převodníku UART na USB, konkrétně XR21B1421 firmy Exar. Schéma výsledné fyzické konfigurace pro komunikaci je na obrázku (3.1).

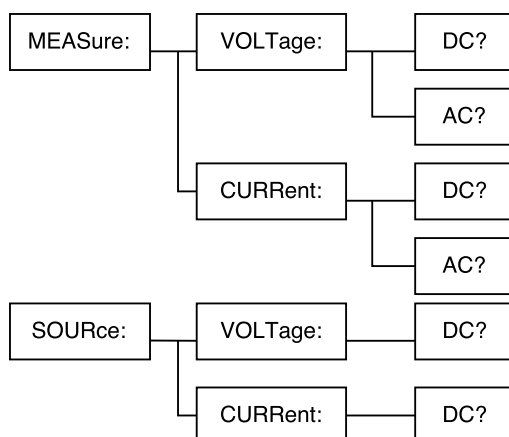


Obrázek 3.1: Schéma navrhované komunikace MD s PC

Rozhraní UART je převodníkem od firmy EXAR možné realizovat s maximální rychlostí 12 *MBaud/s* [7]. V případě STM32F429 je to pouze 11,25 *Mbaud/s*. Nevýhoda rozhraní UART je obtížná detekce chyb přenosu dat, která je realizována pouze pomocí kontroly struktury datového rámce. Pravděpodobnost detekce možné chyby je možné zvýšit použitím paritního bitu a užitím tzv. *handshakingu*. Ten je zajištěn užitím dalšího páru vodičů, kterými si obě strany komunikačního rozhraní ověřují, zda mohou posílat další data, čímž se dále snižuje pravděpodobnost chyby, která může na fyzické vrstvě vzniknout. Kompletní aplikované nastavení pro rozhraní UART je uvedeno v tabulce (3.2).

3.2 Návrh komunikačního protokolu pro obsluhu desky

Komunikaci zařízení přes rozhraní UART nedefinuje žádný protokol, přenáší pouze znaky ze zařízení A do zařízení B. V případě komunikace dvou zařízení pomocí rozhraní UART se z hlediska efektivity přenosu často využívají číselné příkazy (například znak „1“ může znamenat zapni se!, znak „0“ vypni se!, „2“ změř hodnotu! apod.). Výhodou je snadná implementace takového protokolu a pro zařízení snadné vyhodnocení takových příkazů, ke kterému stačí pouze porovnání čísel s databází definovaných příkazů. Nevýhodou je neintuitivní manipulace s příkazy pro člověka. Řešením je inspirace standardem SCPI (*Standard Commands for Programmable Instruments*), který vznikl v roce 1990 jako přídatná vrstva



Obrázek 3.2: Stromová hierarchie SCPI

standardu IEEE-488 (GPIB), aby usnadnil tuto problematiku [8]. Přesto, že se jedná o poměrně starý standard, podporuje jej v dnešní době velké množství měřicích přístrojů, je tedy velmi známý a rozšířený, což je dobrý důvod pro inspiraci se jím.

Základním rysem tohoto standardu je stromové větvení hierarchie příkazů jejíž příklad je na obrázku (3.2). Příkladem SCPI příkazu pak může být `MEASure:VOLTage:DC?` pro změření stejnosměrného napětí.

Ovládání MMD Jelikož MMD není navrhována dle standardu IEEE-488, a nepodporuje tedy některé funkcionality, které jsou v standardu SCPI definovány jako povinné [8], jde tedy pouze o inspiraci tímto standardem, nikoli jeho plnou implementaci, která nebyla vyžadována. Aby se výsledný protokol co nejvíce blížil standardu SCPI, byla použita následující pravidla tvorbu SCPI příkazů:

- Příkazy mají víceúrovňovou hierarchii, která se může v jednotlivých úrovních větvit. Jednotlivé úrovně jsou odděleny dvojtečkou.

– Příklad: Náměr stejnosměrného napětí

* `AI:DC:ReaD`

- Každou úroveň příkazu může být reprezentována libovolným kódovým slovem sestávající se z malých a velkých písmen, bez diakritiky. Plná forma kódového slova je volitelná, všechny příkazy však musí podporovat tzv. zkrácenou formu. Jako mnemotechnická pomůcka nalezení zkrácené formy od formy plné slouží velká písmena, systém však není striktně citlivý na velikost písmen (z důvodu potlačení možných překlepů).

– Příklady příkazů pro přečtení hodnoty digitálního vstupu, které budou vyhodnoceny stejně:

* `DIO:DigitalInput:ReaD`

* `DIO:DI:RD`

* DIO:DIGITALINPUT:Read

- Protokol podporuje zadání parametrů, které jsou odděleny od kódového slova znakem mezery (hex: 0x20). Parametry mohou být:
 - číselné s podporou desetinné tečky. Příklad pro nastavení zdroje napětí na velikost 2,2V:
 - * AO:DC:WRite 2.2
 - znakové. Příklad pro odeslání dat po sběrnici SPI
 - * COM:SPI:SenD text
- Příkazy musí být ukončeny znakem nové řádky „\n“ (hex: 0x0A)
- Protokol podporuje tzv. *query commands*. Rozlišuje, zda je za kódovým slovem znak otazníku „?“ (bez předcházející mezery). Tuto funkcionalitu lze využít pro odlišení nastavení, nebo získání parametrů funkce. Příklad pro nastavení frekvence PWM 1000Hz, resp. jejího zjištění:
 - CNT:PWM:FRQ 1000
 - CNT:PWM:FRQ?
- Protokol podporuje zadání tzv. *common commads*, které jsou ve standardu IEEE 488 definovány jako povinné. Ty jsou uvozeny znakem hvězdičky „*“. Příklad pro zjištění identifikace přístroje
 - *IDN?
- Maximální délka příkazu je limitována 100 znaky včetně parametru. Lze však libovolně změnit do velikosti 32-bitového ne-znaménkového celého čísla (`uint32_t`) úpravou parametru makra „#define MAX_STRLEN“ v hlavičkovém souboru „md_settings.h“ zdrojového MD.

Kompletní stromový seznam implementovaných příkazů je uveden v příloze B.

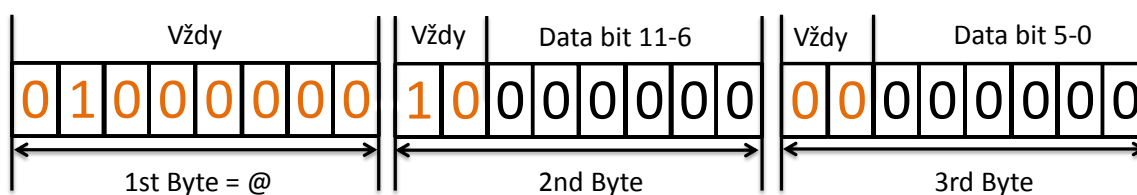
Odpovědi měřící desky Při návrhu systému odpovědí MD na příkazy, a odesílání dat směrem do počítače, bylo myšleno na robustnost, a dále na spolupráci s LabVIEW. Aby prostředí LabVIEW bylo schopné rozlišit typ odpovědi (chyba, číselná odpověď, textová odpověď, apod.), byl zvolen systém prefixů odchozí zprávy do počítače. Jejich výčet a význam je uveden v tabulce(3.3).

Takto navržený systém je vhodný pro odesílání dílčích odpovědí, které by chtěl vidět operátor skrz komunikační terminál počítače například bez využití LabVIEW. V případě přenosu rychlého toku dat (například kontinuálního měření), je takto navržená komunikace značně neefektivní, jelikož jsou jednotlivé čísla odesílána jako znaky, což není nutná podmínka pro přenos dat.

Bylo proto rozhodnuto navrhnout druhotný protokol, který by dokázal zvýšit propustnost rozhraní UART pro odesílání číselných hodnot měření, ovšem za cenu nečitelnosti dat pro člověka v počítačovém terminálu.

Prefix	Význam	Příklad odpovědi MD
\$	Potvrzení provedení příkazu. Je odesláno z MD v případě, kdy není ze strany počítače očekávána konkrétní odpověď. Po neobdržení tohoto znaku je možné opakovat příkaz, nebo nahlásit chybu zařízení či komunikace	\$
#	Chybové hlášení doplněné číselným kódem chyby, které MD odesílá vždy jako odpověď na příkaz v případě chyby, nikdy ne samovolně.	#5002 No ADC available!
@	Příznak pro odeslání numerické hodnoty	@2.2
&	Příznak pro odeslání textové odpovědi	&MD version 1.0

Tabulka 3.3: Tabulka prefixů odpovědí MD



Obrázek 3.3: Komunikační protokol pro zvýšení datové propustnosti komunikace

Aby bylo sníženo „plýtvání“ jednotlivými znaky, hodnota je do znaků zakomponována binárně. Ve výsledku je používáno pro odesílání 12-bitového čísla (což je bitová velikost čísla převedeného A/D převodníkem) pouze třech znaků (24 bitů), přičemž první znak je symbol @ určený pro synchronizaci, zbylé dva přenášejí hodnotu, přičemž pro kontrolu jsou první dva bity druhého znaku vždy „10“ a třetího vždy „00“. Systém názorně vysvětluje obrázek (3.3). Díky tomuto systému je při *baudrate* 11,25 *MBd* MD schopno odeslat data vzorkované frekvencí až 340 *kHz*.

3.2.1 Obsluha komunikačního protokolu

Komunikaci mezi MD a počítačem začíná vždy počítač, ze kterého musí požadavek ve formě příkazu, který byl popsán v bodě (3.2). MD postupně přijímá v přerušení jednotlivé znaky přicházející po rozhraní UART, které ukládá do příjmového bufferu. Tento proces probíhá tak dlouho, dokud se neobjeví znak „\n“ označující konec příkazu, nebo dokud nevyprší maximální časový interval pro příjem zprávy, který se začíná počítat od příchodu prvního znaku a je nastaven na 1s (v tomto případě je příkaz „zahozen“ - zřejmě byl odeslán příkaz bez „\n“, který by mohl blokovat další komunikaci).

Po úspěšném skončení příjmu příkazu je zavolána metoda „*SCPI_parser*“, jejímž parametrem je přijatý příkaz ve formě textového řetězce. Tato metoda rozdělí tento příkaz pomocí „:“ na jednotlivá kódová slova, která pak vrací jako pole.

Následuje vyhodnocení příkazu. Tento proces obstarává rutina *SCPI_handler*, která přebírá předešle získané pole kódových slov a snaží se najít pro první prvek pole ekvivalent v stromově strukturované „databázi“ možných příkazů. Pokud je tento ekvivalent nalezen, přesouvá se vyhledávání do dané větve stromové struktury, kde toto vyhledávání probíhá znovu.

Tento proces pokračuje do doby, kdy *SCPI_handler* nedorazí k poslednímu prvku pole. Po nalezení této poslední shody MD vykoná příkazem žádaný úkon a vrací počítači potvrzení ve formě „\$“, nebo odpověď s prefixem „@“ s daty.

Pokud některý z prvků není nalezen v „databázi“, odesílá *SCPI_handler* počítači nazpět chybovou hlášku s prefixem „#“ a kódem chyby, který je schopno vyhodnotit LabVIEW. Tyto chybové hlášky jsou shrnuty příloze D.

Aby bylo možné efektivně vyhodnotit jednotlivé příkazy, jsou jednotlivá kódová slova po příjmu přeložena tzv. hashovací funkcí, která má za úkol přiřadit každému kódovému slovu unikátní číselnou hodnotu. Tento postup je výhodný jelikož snižuje zatížení MCU, které místo textových řetězců porovnává čísla. Porovnání dvou čísel trvá řádově rychleji než porovnání řetězců [4], čímž je urychlena doba celé obsluhy jednotlivých příkazů.

Jako hashovací funkce byla vybrán algoritmus *SDBM*, který je hojně využíván pro porovnávání textů v databázích, je jednoduše implementovatelný a má velmi nízký počet kolizí jednotlivých hashů[9]. Implementace byla realizována následující funkcí [10]

```
uint32_t Hash(char *text, uint16_t delka_textu) {
    unsigned int hash = 0;           //definice promenne hash
    uint16_t i = 0;                  //definice iteratoru
    for(size_t i = 0; i < len; ++i) //iterace pres kazde pismenko
        hash = 65599 * hash + string[i]; //generovani hashe
    return hash ^ (hash >> 16); }    //hash jako navratova hodnota fce
```

Jelikož jsou jednotlivá kódová slova v „databázi“ dopředu známa a uložená v programové paměti MCU, mohou být s výhodou „zahashována“ už během překladačného kódu pro MCU preprocesorem dle následujícího makra [10]

```
#define H1(s,i,x)    (x*65599u+(uint8_t)s[(i)<strlen(s) \
                    ?strlen(s)-1-(i):strlen(s)])
#define H4(s,i,x)    H1(s,i,H1(s,i+1,H1(s,i+2,H1(s,i+3,x))))
#define H16(s,i,x)   H4(s,i,H4(s,i+4,H4(s,i+8,H4(s,i+12,x))))
#define H64(s,i,x)   H16(s,i,H16(s,i+16,H16(s,i+32,H16(s,i+48,x))))
#define H256(s,i,x)  H64(s,i,H64(s,i+64,H64(s,i+128,H64(s,i+192,x))))
#define HASH(s)      ((uint32_t)(H256(s,0,0)^(H256(s,0,0)>>16)))
```

3.2.2 Systém kontroly prostředků karty

Jak bylo v bodě (2.3) naznačeno, vnitřní struktura MCU dovoluje, aby mohly být jeho jednotlivé periferie (A/D převodníky, čítače, apod) požívány na více než jednom vývodu z pouzdra MCU. Nutně to znamená, že je pro zajištění spolehlivosti MD potřeba předcházet snaze definovat periferii s jinými parametry na různých vstupech nebo výstupech. Z tohoto důvodu byly zavedeny uměle navržené „registry“ a navržený systém inicializace a de-inicializace měřících funkcí.

Prvním typ navrženého registru je tzv. *pin_reg* (pinový registr). Jedná se o definované pole s počtem prvků rovnému počtu funkčních pinů MCU a je strukturou složenou z mnoha proměnných a dalších sub-struktur. Jeho význam je následující

- Definuje jaké měřící funkce na něm mohou být inicializovány. Touto informací se předchází možné definici funkce na pin, který ji nepodporuje.
- Definuje inicializační parametry pro funkce, které může pin vykonávat (například jaký převodník či čítač může pin využít). Odpadá tedy nutnost vytváření složitých konstrukcí vybírajících parametr periferie pro konkrétní pin u každé inicializační procedury k měřící funkci, které jsou předány pouze potřebné parametry přímo z *pin_reg*.
- Udržuje informaci o tom zda pin nevyužívá některá z funkcí. Touto informací se předchází využívání pinů funkcemi, které nejsou v danou chvíli na pinu inicializovány.

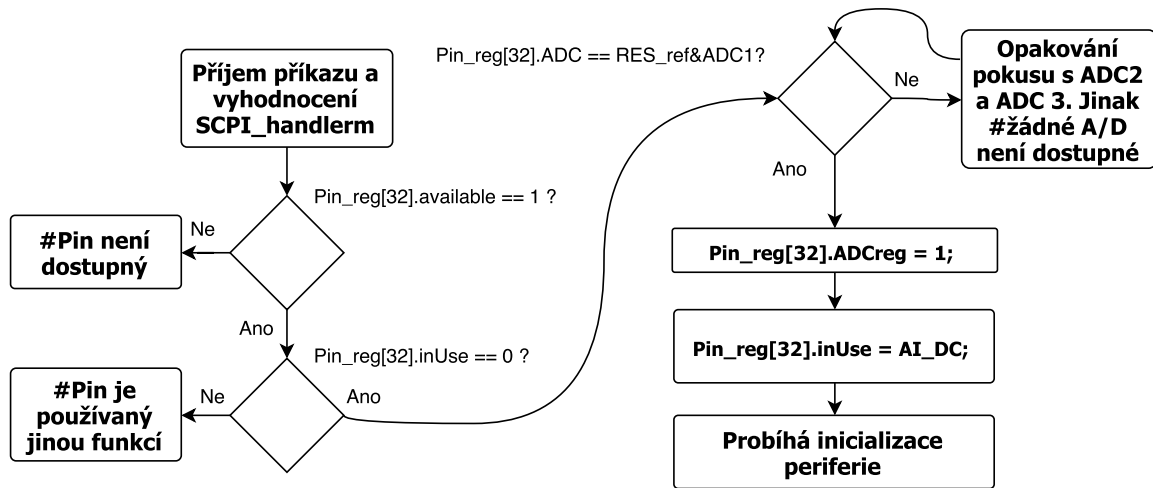
Ze zmíněných 19-ti prvků 15 nese některou z inicializačních informací. Zbylé 4 prvky jsou proměnné, do kterých je během běhu programu zapisováno, nebo naopak je z nich čteno. Tabulka (3.4) shrnuje jejich význam.

Proměnná prvku <i>pin_reg</i>	Význam
<i>available</i>	Nabývá binární hodnoty 0 a 1. Dává programátorovi MD možnost vyloučit pin z užívání, například pokud je pin fixně používán pro komunikaci
<i>in_use</i>	Každý konkrétní bit této proměnné je přiřazen implementované funkci a je nastaven do log 1, pokud jej daná funkce využívá. S výhodou může MD zjistit využívanost tohoto pinu porovnáním jej s 0
<i>AFreg</i>	Konkrétní bity určují podporované funkce pinem
<i>ADC reg</i>	Proměnná identifikující používaný A/D převodník pinem. Byla zavedena z důvodu možnosti inicializace pinu na všech třech převodnících. Při de-inicializaci funkce s A/D převodníky pak pin „ví“, který převodník/y má de-inicializovat

Tabulka 3.4: Proměnné prvku registrového pole *pin_reg*

Druhým typem zavedeného registru je *RES_reg*. Jedná se o celočíselnou proměnnou jejíž každý bit reprezentuje využití některé z periférií MCU, čímž brání tomu a by byla inicializována více piny ve stejný čas. Celkem takto může uchovávat stav využití až 64 periférií.

Názorným příkladem využití celého systému je příkaz *AI:DC:PIN 32:INIT\n*, který žádá o



Obrázek 3.4: Diagram procesu inicializace pinu

inicializaci funkce měření stejnosměrného napětí na pinu číslo 32 (viz obrázek 3.4). Důvod proč není inicializován pin 32 jako „C0“, což je kódové označení tohoto pinu na MCU, je jeho snadná adresace v poli *pin_reg*. Dokumentace této „re-adresace“ je uvedena příloze C.

3.3 Čtení digitální vstup a zápis na digitální výstupy měřící karty

Funkce digitální a vstupu a výstupu je základním typem periferie MCU, která se oproti ostatním liší tím, že jí disponuje každý vstupně-výstupní pin pouzdra.

Vnitřní struktura STM32F429 řadí piny do tzv. portů. Tyto porty mají mnoho registrů z nichž část přenastavuje vnitřní strukturu pinu a určují tak, zda je pin provozován jako digitální vstup, digitální výstup, analogový pin, nebo je k němu přivedena jiná periferie (např. čítač).

Kromě těchto registrů má port k dispozici také registry, které operují s logickými vstupy a výstupy. Pro vyčtení logické hodnoty konkrétního pinu (pinu musí být nastaven jako digitální vstup), je třeba vyčíst dolních 16bitů z registru *GPIO_ODR* (každý bit je pro jeden pin v portu).

Obdobně se postupuje při nastavení logické hodnoty konkrétních pinů portu. Zde je zapotřebí zapsat logické jedničky na konkrétní bity horních 16-ti bitů registru *GPIO_BSSR* pro získání logické 0 (nutno opět zapsat všech 16-bitů naráz). Stejným postupem pak pro získání logické 1 na výstupech zapsáním na konkrétní bity dolních 16-bitů stejného registru.

Rozlišení jednotlivých logických úrovní určuje velikost vstupního napájení STM32F429 (V_{CC}). Jako logická nula je signál vyhodnocen, pokud se pohybuje v rozmezí $0V$ až $0,35 \cdot V_{DD}$, jako logická jednička v rozmezí $0,45 \cdot V_{DD} + 0,3V$ až $5V$. Pásmo mezi těmito dvěma pásmy je definováno jako zakázaný stav.

Pro implementovanou funkci čtení digitální hodnoty je pro definici úrovně vstupní hodnoty přednastaven interní *pull-up* rezistor. Pro nastavování hodnot logického výstupu je využíván

Název čítače	Rozlišení [bit]	Rychlost časové základny [MHz]	Počet možných kanálů (CCC)
TIM1	16	42	4
TIM2	32	42	4
TIM3	16	42	4
TIM4	16	42	4
TIM5	32	42	4
TIM8	16	84	4
TIM9	16	84	2
TIM10	16	84	1
TIM11	16	84	1
TIM12	16	42	2
TIM13	16	42	1
TIM14	16	42	1

Tabulka 3.5: Parametry jednotlivých čítačů STM32F429

vnitřní struktura s *push-pull* tranzistory.

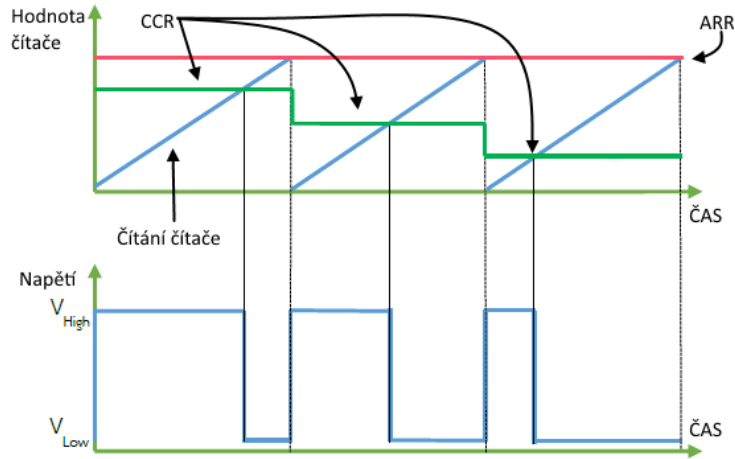
3.4 Generování pulsně šířkové modulace (PWM) pomocí čítačů

STM32F429 disponuje celkem 12 čítači schopnými generovat pulsně šířkovou modulaci (PWM). Jednotlivé čítače se liší svými parametry (velikost čítacího registru (rozlišení), rychlost časové základny, počet tzv. *capture/compare channels* (CCC), které PWM realizují). Tyto parametry jsou shrnuty v tabulce (3.5). Samotný výstupní signál PWM z CCC je možné vnitřní strukturou STM32F4 přeměrovat až na 3 výstupní piny.

Po uvození čítače do režimu generování PWM, určují její výsledné parametry (frekvenci a střídu) tři následující registry

- Prescaler register (PSC) - Jedná se o 16-bitové číslo, podle kterého je dělena vstupní časová základna, a podle které následně čítá čítač. K výslednému děliteli je potřeba přičítat číslo 1, jelikož $PSC = 0$ dělí číslem 1.
- Auto-reload register (ARR) - Složí jako jakási „zarážka“ nastavující hodnotu, do které má čítač čítat (má stejnou velikost jako čítač). Pokud se čítač dočítá k hodnotě uložené v ARR, je vygenerována událost, na kterou je v případě generování PWM reagováno vynulováním hodnoty čítače a přepnutím výstupního pinu do stavu logické jedničky.
- Capture/Compare register (CCRx) - Slouží pro určení střídy PWM (x značí číslo kanálu CCC). Registr ARR určuje resetováním čítače výslednou frekvenci PWM. CCR registr pracuje podobně jako ARR s tím rozdílem, že po dočítání čítače k jeho hodnotě je změněna logická hodnota výstupu do logické nuly. V kombinaci s přepínáním výstupu do logické jedničky v případě registru ARR, takto vzniká výsledná modulace PWM.

Práci těchto registrů při generování PWM s různými střídami shrnuje přehledně obrá-



Obrázek 3.5: Princip generování PWM pomocí čítače STM32F429

zek(3.5).

Pro nastavení zmíněných registrů pro zadanou a frekvenci a střídu slouží následující vzorce

$$PSC = \frac{\text{frekvence časové základny [Hz]}}{\text{zadaná frekvence [Hz]}}, \quad (3.1)$$

$$ARR = \frac{\text{frekvence časové základny [Hz]}}{\text{zadaná frekvence [Hz]}}, \quad (3.2)$$

$$CCR = ARR \cdot \frac{\text{střída [\%]}}{100}. \quad (3.3)$$

Z těchto vzorců je patrné, že při každém přepočtu frekvence je třeba znovu přepočítat registr CCR pro střídu, a že hledaná frekvence je pouze aproximací k frekvenci zadané. Vzniká tedy různě veliká odchylka od zadání. Aby bylo možné zjistit jaká frekvence a velikost střídy PWM byla finálně v MD nastavena, mohou být tyto hodnoty pomocí patřičných příkazů vyžádány. K zjištění aktuálně nastavené frekvence a střídy slouží následující vzorce

$$\text{frekvence [Hz]} = \frac{\text{frekvence časové základny [Hz]}}{\frac{PSC+1}{ARR}}, \quad (3.4)$$

$$\text{střída [\%]} = \frac{CCR \cdot 100}{ARR}. \quad (3.5)$$

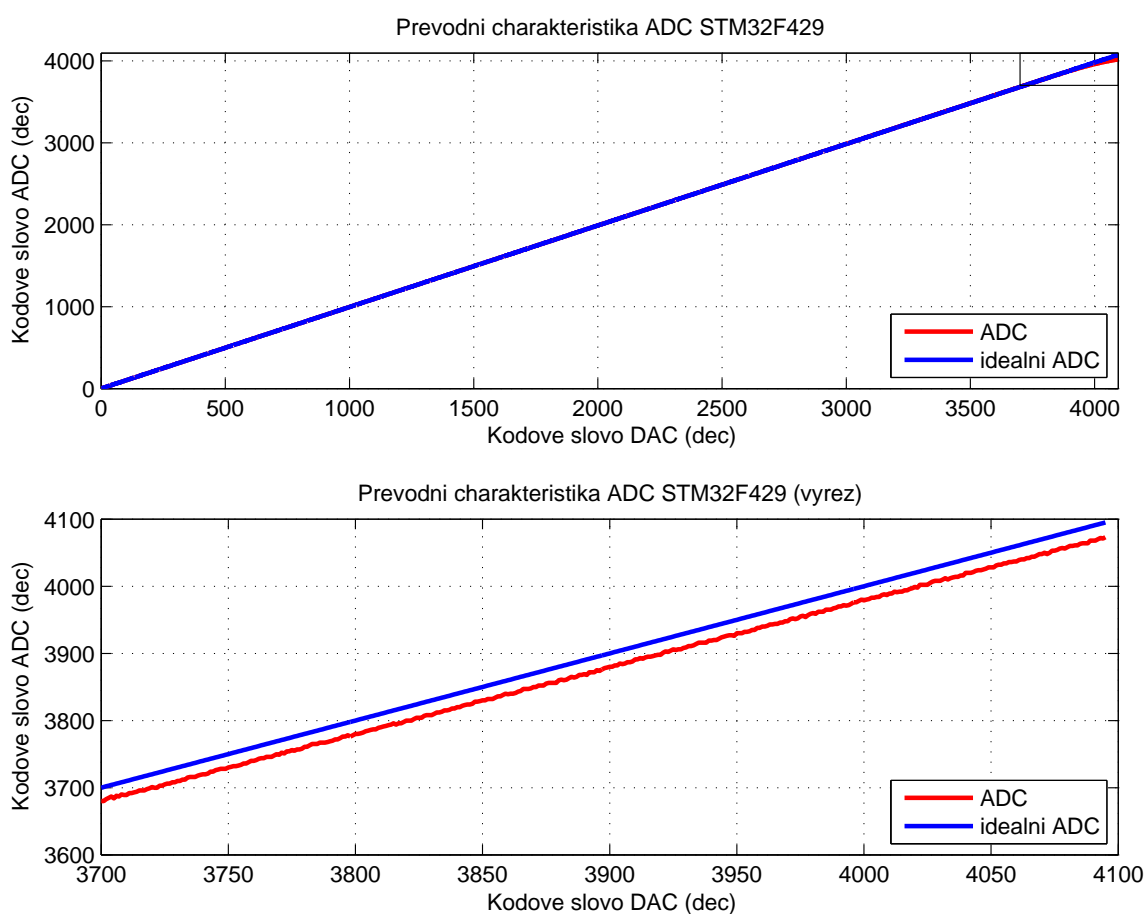
3.5 Měření napětí integrovaným A/D převodníkem

Dle požadavků z bodu(2.2) jsou požadovány dva typy měření. Tím prvním je jednoduché měření napětí, tím druhým vzorkované měření napětí. Pro tyto účely STM32F429 disponuje celkem třemi nezávislými 12-bitovými A/D převodníky, které je možné provozovat celkem na 24 různých vstupních pinech MCU.

3 Softwarová koncepce multifunkční měřicí desky a návrh jejích funkcí

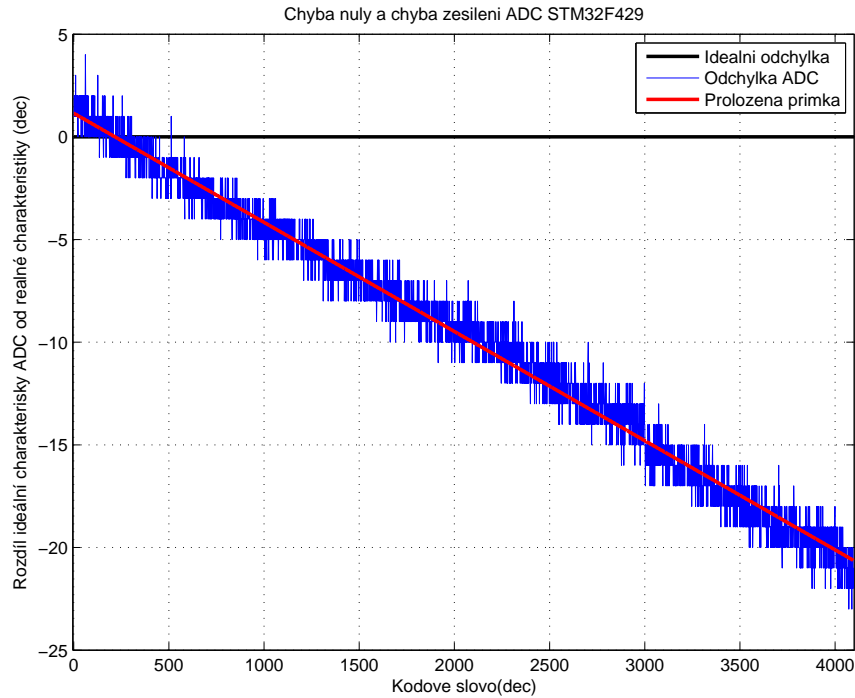
Aby bylo zjištěno reálné chování převodníku proti laboratorním měřením, byl jeden z převodníků podroben testování pomocí akviziční karty NI 6216 firmy National Instruments, která disponuje kvalitními 16-bitovými A/D a D/A převodníky, je tudíž pro měření 12-bitových převodníků dostačující.

Výchozím měřením bylo měření přechodové charakteristiky A/D převodníku. To je možné vidět na obrázku (3.6), kde je na konci rozsahu převodníku patrná tzv, chyba zesílení[12]. Při podrobnějším zkoumání odečtem jednotlivých vzorků od ideální charakteristiky, bylo zjištěno, že tato chyba je lineární, což je možno vidět na obrázku(3.7). Kromě chyby zesílení byla zjištěna také chyba nuly. Obě tyto chyby tedy mohou být patřičně softwarově kompenzovány. Chyba integrální a diferenciální nelinearity byla rovněž analyzována a spadá do tabulkových hodnot. Výsledky měření statických vlastností převodníku v porovnání s tabulkovými hodnotami jsou uvedeny v tabulce 3.6.



Obrázek 3.6: Naměřená převodní charakteristika převodníku STM32F429

3.5 Měření napětí integrovaným A/D převodníkem



Obrázek 3.7: Chyba nuly a zesílení A/D převodníku STM32F429

Parametr	Naměřená hodnota [LSB]	Tabulková hodnota [LSB]
Chyba nuly	2	± 3
Chyba zesílení	21	± 6
Chyba integrální nelinearity	4	± 6
Chyba diferenciální nelinearity	3	± 3

Tabulka 3.6: Statické parametry A/D převodníku STM32F429[5]

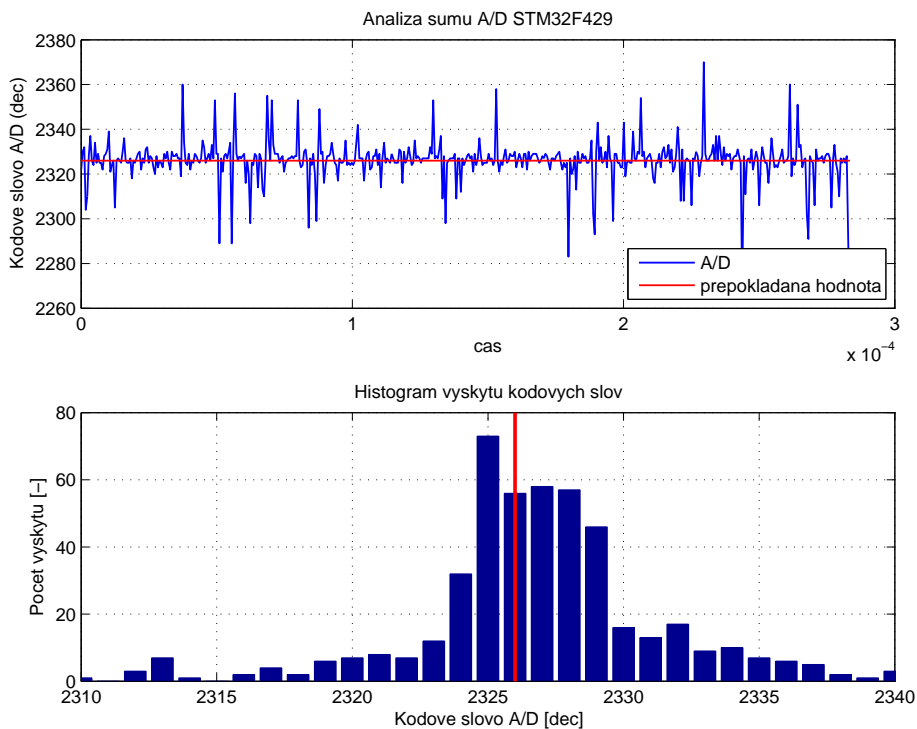
V dalším kroku byl otestován statický šum převodníku náměrem 500 hodnot zdrojového napětí o hodnotě $2V$, což odpovídalo hodnotě přibližně v půlce rozsahu převodníku. Vzorkovací kmitočet byl zvolen $1,764 MHz$, což je maximální frekvence, kterou může být A/D převodník vzorkován pomocí interního čítače [5] (v kontinuálním režimu je ho možné provozovat až se vzorkovací frekvencí $2,4MHz$ [5]).

Z výsledku měření na obrázku 3.8 je patrný poměrně velký rozptyl naměřených hodnot, který je roven $\pm 9 LSB$.

Aby byla tato poměrně velká nepřesnost zmírněna, bylo rozhodnuto průměrovat jistý počet po sobě jdoucích vzorků. Pro případ jednoduchého měření statického napětí byla zvolena hodnota 100 vzorků. Zároveň byla rychlost vzorkování snížena na $5kHz$ tak, aby zmiňovaných 100 vzorků pokrylo jednu periodu síťového napětí $50Hz$, a tím se jeho možný rušivý vliv na měření částečně průměrováním eliminoval. Výsledný rozptyl $0,9 LSB$ pro statické měření $2V$ je možné vidět na obrázku 3.9

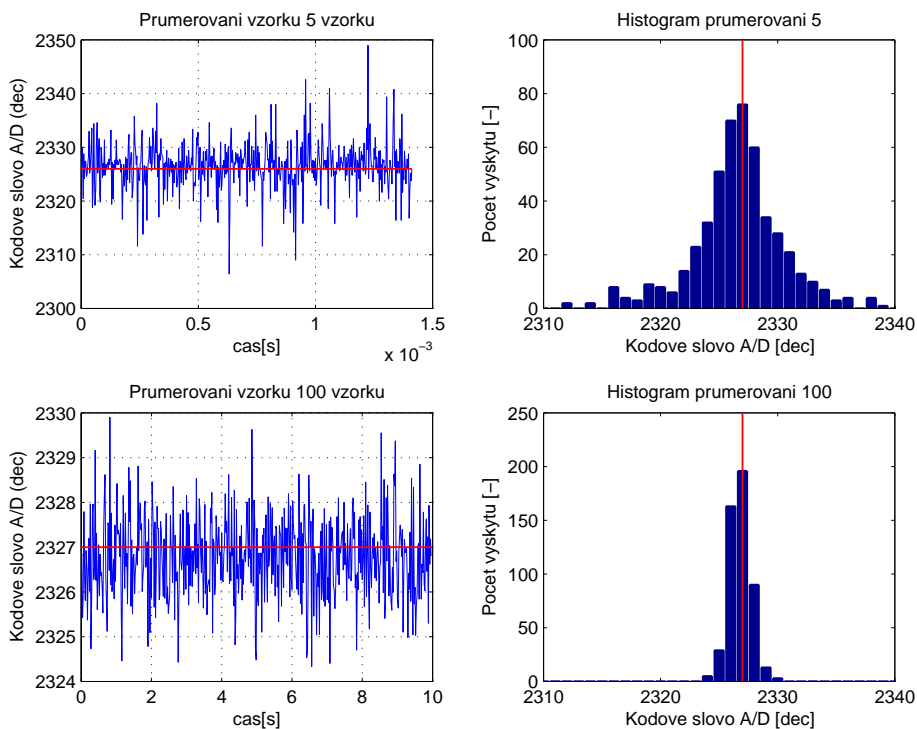
V případě vzorkovaného měření se počet průměrovaných vzorků odvíjí od maximální propustnosti, v bodě 3.2 navrženým, komunikačním protokolem pro přenos dat, který je scho-

3 Softwarová koncepce multifunkční měřicí desky a návrh jejích funkcí



Obrázek 3.8: Šum A/D převodníku STM32F429

pen přenést maximálně 340000 vzorků za sekundu. Vzhledem k výše zmíněné maximální možné vzorkovací frekvenci pomocí čítače, bylo je možné průměrovat pouze 5 po sobě jdoucích vzorků. Výsledný rozptyl 4 LSB na 500 vzorcích s průměrem pěti je možné vidět a histogramu obrázku 3.9.

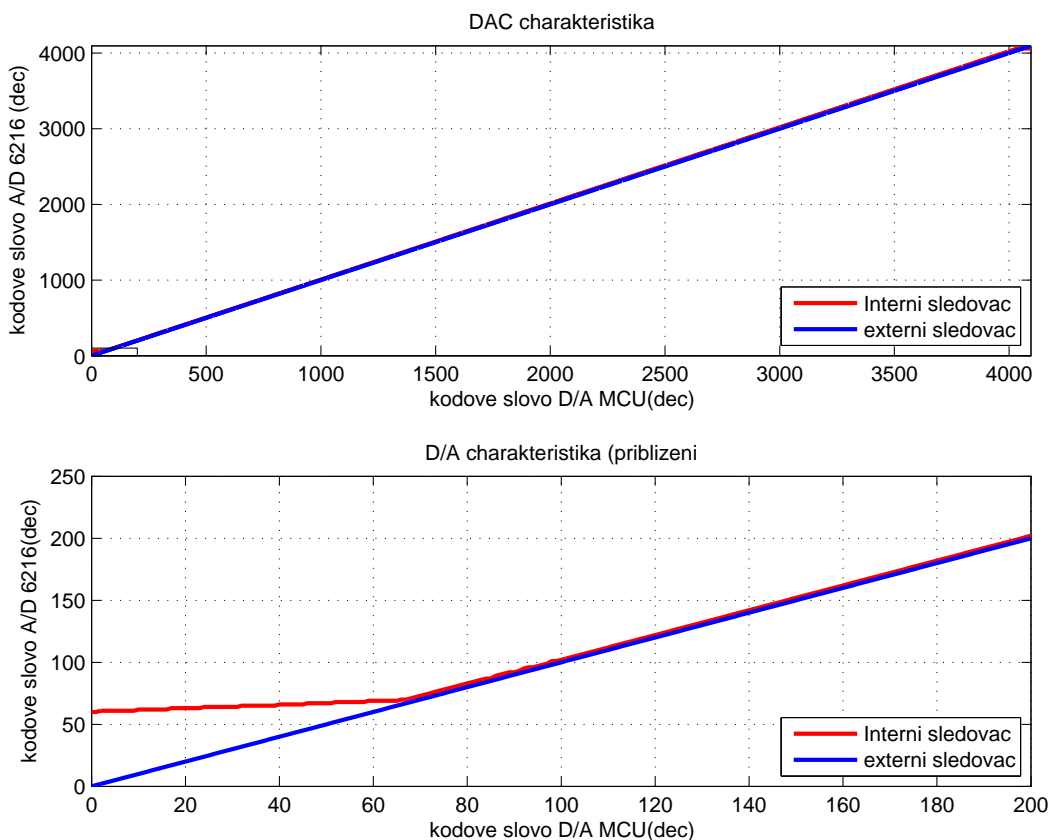


Obrázek 3.9: Šum průměrování

Jelikož je třeba provést průměrovací výpočet během pokračujícího měření, je využito tzv. DMA (Direct Memory Access) kontroléru. Ten zajišťuje ukládání jednotlivých vzorků do předem zvolené proměnné. Zároveň je DMA kontroléru předána hodnota o počtu vzorků, které má přenést než upozorní přerušením procesor o tom, že se tak stalo. Procesor tak má jistý čas mezi každým dalším přerušením od DMA, ve kterém si převezme nově naměřené vzorky, na jejich průměrování a odeslání pomocí komunikačního rozhraní.

3.6 Zdroj stejnosměrného napětí pomocí integrovaného D/A převodníku

Obdobně jako u A/D převodníku v bodě 3.5 byl otestován interní 12-bitový dvoukanálový D/A převodník. Jeho převodní charakteristika je na obrázku 3.10, na kterém je vidět, mezi kódovými slovy 0 až 64, které odpovídají 0V až 0,053V, pokřivení této charakteristiky vlivem fyzikálních možností interního napěťového sledovače, který je zařazen za D/A převodník viz . 5.1.1. Bylo proto nutné charakteristiku změřit znovu avšak s pomocí externího sledovače napětí, který byl napájen duálně (tedy i zápornou napájecí větví), a který danou chybu nezpůsobuje, čímž bylo dokázáno, že se nejedná o chybu převodníku. Výsledky naměřených parametrů jsou spolu s tabulkovými hodnotami uvedeny v tabulce (3.7).



Obrázek 3.10: Přechodová charakteristika

Parametr	Naměřená hodnota [LSB]	Tabulková hodnota [LSB]
Chyba integrální nelinearity	3	± 2
Chyba diferenciální nelinearity	5	± 4
Chyba zesílení	2	± 4
Chyba nuly	1	± 4

Tabulka 3.7: Parametry D/A převodníku STM32F429[5]

3.7 Uchování kalibračních dat

V případě MD je třeba uchovávat hodnotu externí napěťové reference, pomocí které je přepočítávána poměrová hodnota naměřená A/D převodníkem na reálnou hodnotu napětí, nebo naopak je podle ní vypočteno kódové slovo pro D/A převodník ze zadané hodnoty napětí příkazem.

STM32F429 pro tyto účely obsahuje paměť typu *flash*. Ta obsahuje sektory nazývané *OTP* (*One-time programmable memory*). Tato část paměti počínaje adresou `0x1FFF7800` obsahuje celkem 16 paměťových bloků po 16 bajtech. Do nich je možné zapsat hodnotu, která je posléze zamknuta zápisem hodnoty `0xFF` na pro každý blok určené místo 17-bloku. Nevýhoda tohoto postupu uchovávání dat je nutnost zapisovat data vždy do nového bloku, a tím i omezení, jelikož lze tuto paměť takto využít pouze $16x$ [11].

3.8 Čtení identifikačního čísla MCU

Problematiku sériových čísel, na rozdíl od problematiky kalibračních dat řešené v bodě 3.7, není nutné řešit obdobně, nýbrž je možné využít unikátního 12-bajtového čísla uloženého na paměťové adrese `0xE0042000` MCU, a které je pro každý MCU unikátní [11].

4 Knihovna funkcí pro program LabVIEW

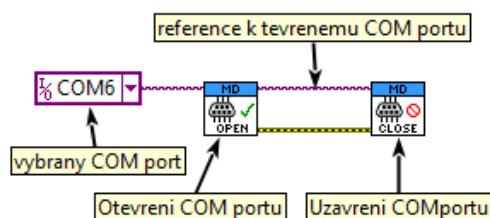
V bodě 2.4 počáteční analýzy je zmíněn systém měření s multifunkčními měřicími deskami pomocí standardu DAQmx, který je používán pro univerzální měřicí desky firmy National Instruments. Při návrhu softwaru MD v bodě 3 bylo co nejvíce dbáno na to, aby knihovna pro použití MD v prostředí LabVIEW mohla co nejvíce kopírovat schéma užití standardu DAQmx. V ideálním případě by měly být jednotlivé obslužné VI pro práci s DAQmx vyměnitelné kus za kus s obslužnými funkcemi navrženými pro práci s MD (například při rozhodnutí nevyužít měřicí desku National Instruments, nýbrž navrhovanou MD, bez nutnosti přepracovávat celý měřicí program).

Po analýze možného řešení knihovny pro MD v LabVIEW se ukázalo, že výsledné řešení bude muset obsahovat koncepční výjimku. Touto výjimkou je nutnost otevření komunikačního rozhraní COM (UART) v počítači, kterou je možné provést právě a jen jednou při začátku práce s MD, na rozdíl od standardu DAQmx, který správu komunikace, řeší interně. Rozdíl mezi systémem práce s použitím knihovny DAQmx a knihovny pro MD je již s navrženými VI na obrázku 4.9.

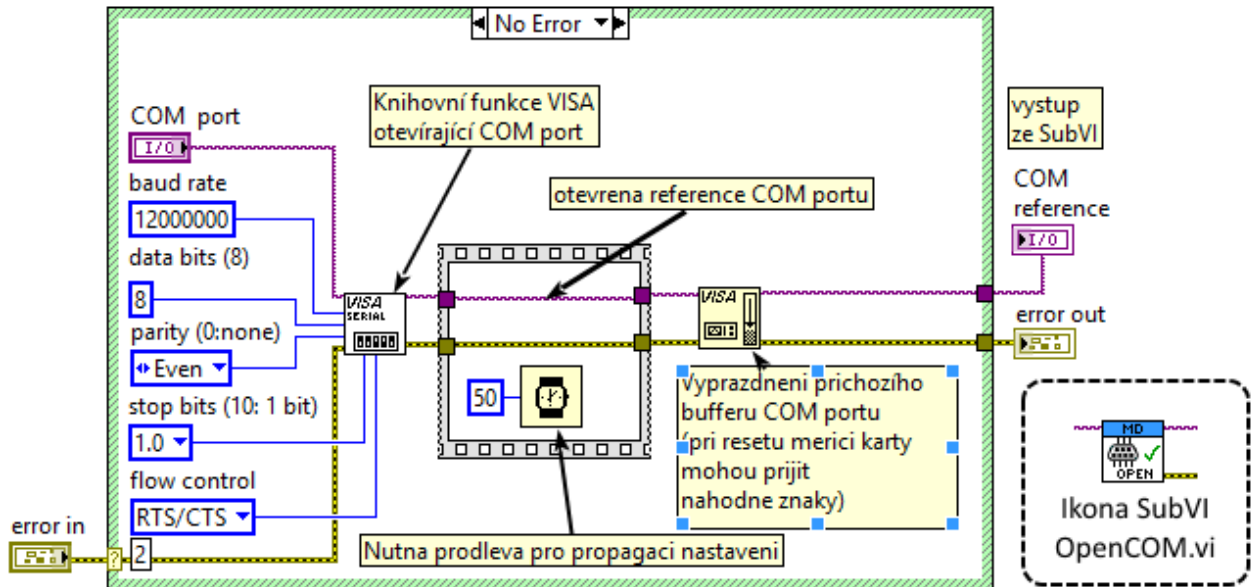
4.1 Integrace navrženého komunikačního protokolu do prostředí LabVIEW

Komunikaci MD s počítačem probíhá přes rozhraní UART (viz bod 3.1). Jako mezivrstvou pro práci s tímto rozhraním využívá LabVIEW standardizovaný komunikační standard VISA, pro který má vlastní knihovní funkce. Těchto funkcí je využito pro otevření a uzavření komunikace po rozhraní COM, dále pak pro posílání příkazů kartě a příjem odpovědí.

Jednoduchý program, který otevře port COM pomocí navrženého VI z obrázku (4.2) a opětně ho uzavře je uveden na obrázku (4.1).

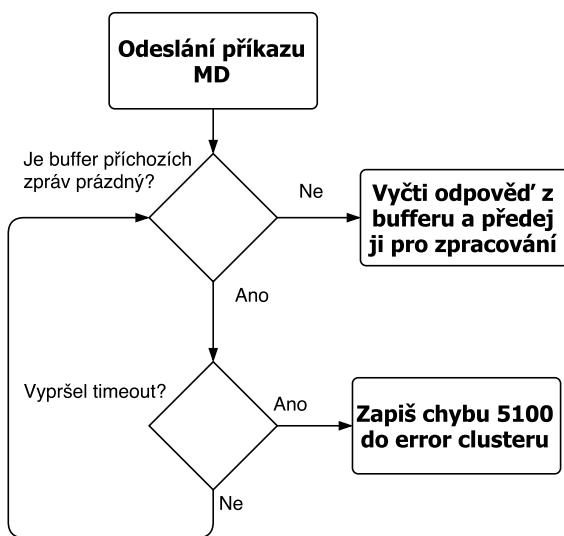


Obrázek 4.1: Ukázka programu pro otevření a uzavření komunikace COM pomocí knihovních funkcí MD



Obrázek 4.2: Program VI OpenCOM.vi

4.1.1 Posílání příkazů MD a příjem odpovědi



Obrázek 4.3: Diagram funkce

Pokud proběhne otevření portu COM bez problémů, je na něj vytvořena reference. V ten moment je možné začít s kartou komunikovat. K tomu slouží VI *MD_Write_Command*, jehož funkce je znázorněna digramem na obrázku (4.3). V principu je odeslán příkaz, poté se cyklicky kontroluje zda nebyla přijata odpověď od MD do přichozího bufferu portu COM. Tato kontrola probíhá do příchodu odpovědi, nebo vypršení doby stanovené volitelným parametrem *timeout*, která dává MD šanci provést měření a odeslat zprávu (bez udání tohoto parametru je jeho hodnota nastavena na *1ms*).

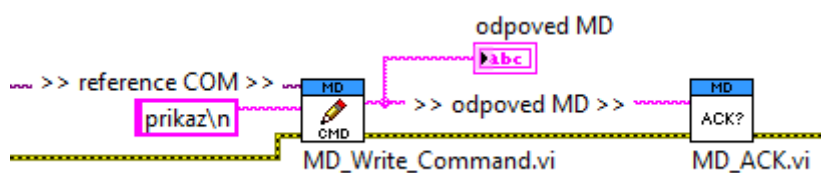
4.1.2 Kontrola chyb MD a komunikace

Pro systém práce s MD bylo navrženo několik kontrolních mechanismů pro detekci chyb. Primárním cílem bylo navrhnout a otestovat funkce MD, tak aby chyby nenastávaly, nebo byly detekovány a informace o nich byla propagována směrem k uživateli.

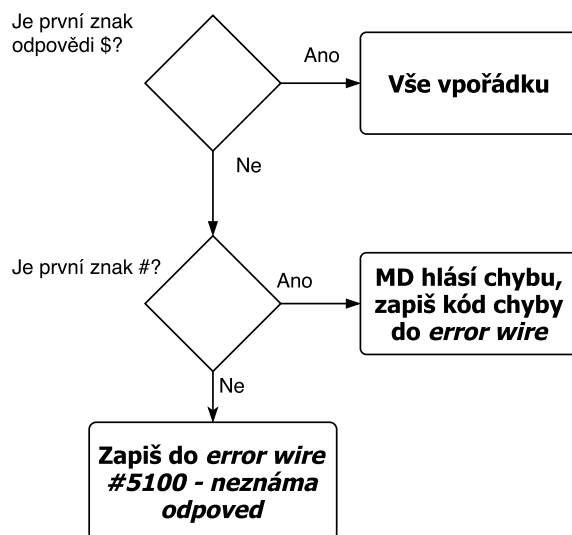
Obecně chyby mohou vznikat následujícími způsoby:

1. Ztrátou komunikace s MD -
 - a) Pokud chyba nastane před odesláním příkazu směrem k MD, pak vyhlásí chybu samotný ovladač komunikačního rozhraní počítače a předá jí LabVIEW, které informuje uživatele.
 - b) Pokud chyba nastane, po odeslání příkazu, pak není v definovaném čase obdržena odpověď, VI *MD_Write_Command* zapíše chybu do *error clusteru*.
2. Chybou komunikace (například vlivem zarušení komunikačních vodičů) -
 - a) Pokud chyba nastane během přenosu příkazu MD, pak tomuto příkazu deska neporozumí a odpoví odpovědí s příznakem # a chybovou hláškou o problému s komunikací
 - b) Pokud chyba nastane po odeslání odpovědi kartou, pak jí počítač může detekovat dvěma způsoby
 - i. chybou parity, pak ovladač portu COM předá tuto chybovou hlášku prostředí LabVIEW
 - ii. kontrolou konzistence dat - každá odpověď musí mít prefix, a má pevně stanovenou délku pro případy, kdy striktně záleží na jejím obsahu.
3. Vlivem špatného příkazu kartě - karta tento fakt vyhodnotí a odpoví chybovou hláškou o neporozumění příkazu
4. Interním problémem MD. Příkladem může být nedostatek prostředků, nebo inicializace měřicího kanálu na již inicializovaný pin

Pro rozlišení typu odpovědi MD bylo naprogramováno VI *MD_ACK* jehož princip práce vysvětluje diagram na obrázku 4.5, ukázka užití ve spolupráci s komunikačním VI *MD_Write Command* je na obrázku 4.4.



Obrázek 4.4: Spolupráce VI *MD_ACK.vi* s VI *MD_Write_Command.vi*



Obrázek 4.5: Diagram práce VI *MD_ACK*

4.2 Návrh obsluhy měřicí úlohy

4.2.1 Vytvoření měřicí úlohy a její propagace programem

V bodě 3.2.1 bylo popsáno zavedení procesu inicializace prostředků MD. Pro tento proces bylo určeno klíčové slovo `INIT` uváděné na konci příkazu, pro každý typ měření. Tyto příkazy jsou využity i při práci s MD v LabVIEW při vytváření tzv. měřících úloh (*meassure task*).

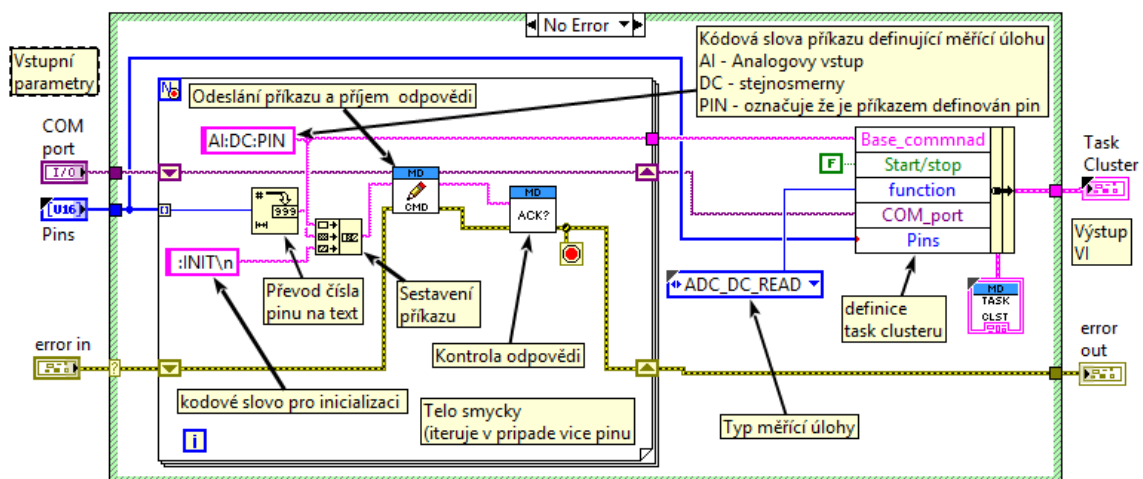
Jednotlivé měřicí úlohy jsou reprezentovány k tomuto účelu přizpůsobeným datovým klastrem (*meassure task cluster*). Cílem tohoto, pro všechny typy úloh univerzálního klastru, je usnadnit správu měření udržováním důležitých informací vázajících se ke konkrétnímu měření. Příkladem takové informace je reference na využívaný COM port pro případ práce s více MD v jeden okamžik, nebo například výčet pinů, s kterými se v rámci dané měřicí úlohy pracuje. Celkový výčet proměnných spravovaných *meassure task* klastrem je uveden v tabulce 4.1.

Název proměnné	Význam
Base_command	Pokud je úloha vytvořena, pak jsou pro její správu důležitý začátek příkazu, který je vždy stejný (proto base command) a dále klíčová slova za poslední oddělovací „:“.
start/stop	Pokud je nastavena hodnota „pravda“ - TRUE, pak umí s touto měřicí úlohou pracovat VI, které mohou spustit nebo pozastavit měřicí úlohu (například generování PWM). Pokud je nastaveno False, tyto VI při jejich použití negenerují, žádné příkazy

Název proměnné	Význam
function	Nese informaci o typu úlohy. To je důležité v případě nesprávného užití akčních VI. Příkladem může být pokus nastavení logické úrovně na pin s úlohou měření napětí
Com_port	Obsahuje referenci na otevřený COM port. Tím je jasně definováno jakou MD měřící úloha využívá
pins	Jedná se o výčet identifikačních čísel pinů, na kterých je inicializována měřící úloha
sampling_frq	Jedná se o pomocnou proměnnou využívanou měřícími úlohami pracujícími s tokenem dat. Při vytvoření úlohy z MD vyčtená informace o vzorkovací frekvenci je využívána pro vytvoření časových značek pro každý vzorek.

Tabulka 4.1: Výčet proměnných spravovaných *task clusterem*

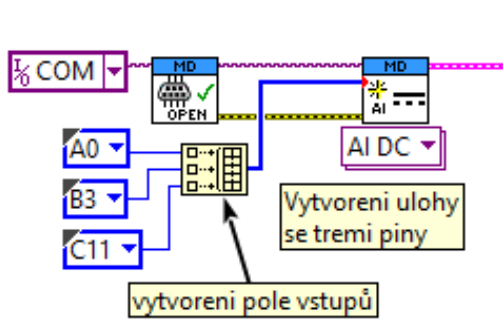
Na obrázku 4.6 je na příkladu vytvoření úlohy měření stejnosměrného napětí znázorněno, jak je *meassure task* klastr zadefinován. Z důvodu rozdílných proměnných musí mít každý typ měřící úlohy vlastní inicializační VI.



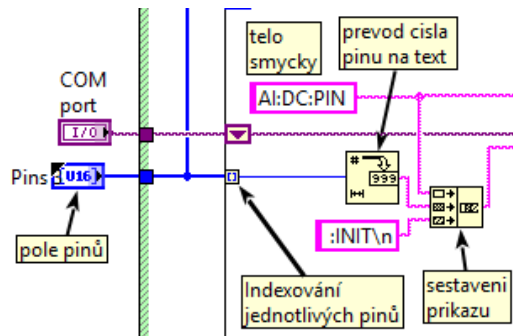
Obrázek 4.6: Vytvoření úlohy měření stejnosměrného napětí

4.2.2 Práce s více piny (kanály) v jedné úloze

Velkou výhodou navržené struktury měřících úloh je možnost zadefinovat a pracovat s více piny (kanály) zároveň v jedné měřící úloze. Toho bylo docíleno ukládáním referenčních čísel pinů do datového pole při inicializaci měřící úlohy (viz obrázek 4.7). Každé navržené VI pracující s *measure task* klastrem pak využívá datové pole s referencemi na čísla pinů tak, že toto pole postupně prochází, a při každé iteraci využije, při stavbě příkazu, referenci na pin uložený v aktuálně indexovaném prvku pole, což je nastíněno na obrázku 4.8.



Obrázek 4.7: Vytváření měřící úlohy s více piny

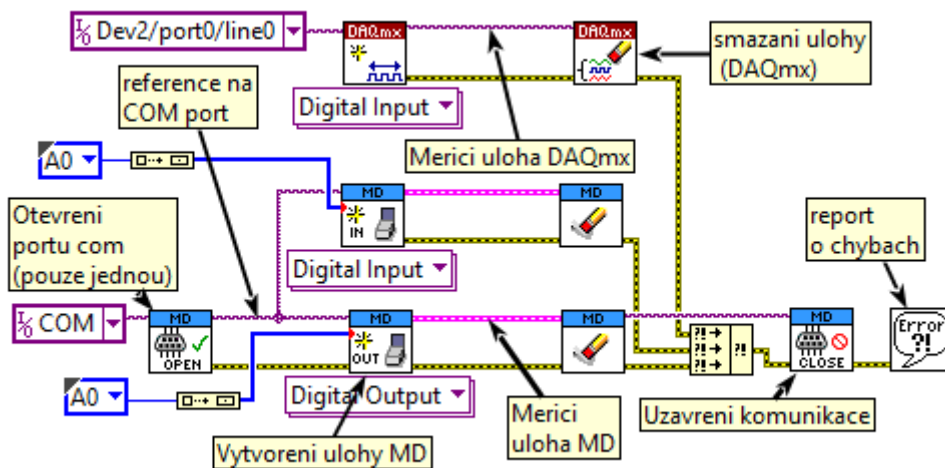


Obrázek 4.8: Programová práce s více piny uvnitř VI podporující tuto funkci

4.2.3 Ukončení měřící úlohy

Ukončení práce s měřící úlohou je realizováno univerzálním VI pro všechny typy měřících úloh. De-inicializační VI využívá základní příkaz (base command) z *measure task* klastru, ke kterému připojí klíčové slovo DEINit (DEIN), které má po sestavení a odeslání příkazu MD de-inicializuje prostředky v MD, které mohou být následně využity jinou měřící úlohou.

Ukázka práce s jednou MD, inicializací několika měřících úloh a jejich následnou de-inicializací, je uvedena na obrázku 4.9 společně s se stejnou ukázkou za použití knihovny DAQmx pro multifunkční měřící karty od National Instruments.



Obrázek 4.9: Porovnání přístupu otevření a uzavření měřící úlohy systému DAQmx a navrženého MD

5 Hardwarový návrh měřicí desky

Pro návrh obvodového schématu pro fyzickou realizaci MD byl použit software Altium Designer. Aby byla zjednodušena orientace v jednotlivých částech schématu, byl celý návrh rozdělen několika funkčních celků (napájení, ochrana digitální vstupů a výstupů atd.), dle prvotního koncepčního návrhu v bodě 2.5 na obrázku 2.7. Obecným cílem tohoto návrhu bylo

1. Zajištění kvalitního napájení pro jednotlivé součástky
2. Předzpracování signálu pro navržené funkce MD, které jej vyžadují
3. Rozšířit ochranu jednotlivých vstupů a výstupů MCU s cílem zvýšit životnost MD při možném neopatrném zacházení
4. Využívat v co největší míře součástky standardně využívané ve firmě BTL.

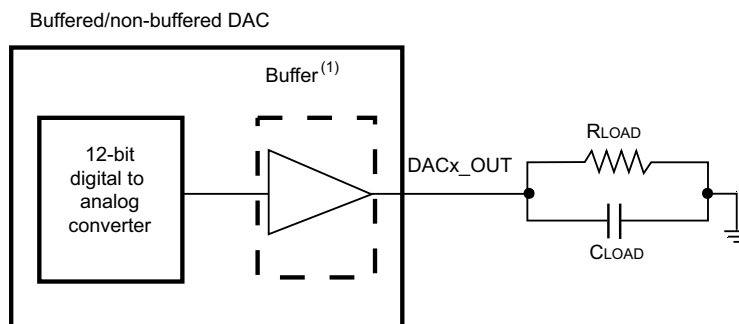
Jelikož každá funkce vyžaduje vlastní hardwarovou periférii, musí být během obvodového návrhu rozhodnuto, jaké funkce budou přiřazeny konkrétní piny (mnohdy je možné směřovat více funkcí na jeden pin). Tabulka tohoto přiřazení je uvedena pro svoji velikost v příloze C. Kompletní obvodové schéma je pro přehlednost uvedeno v příloze E.

5.1 Návrh obvodů pro předzpracování signálu navržených funkcí MCU a jeho ochranu

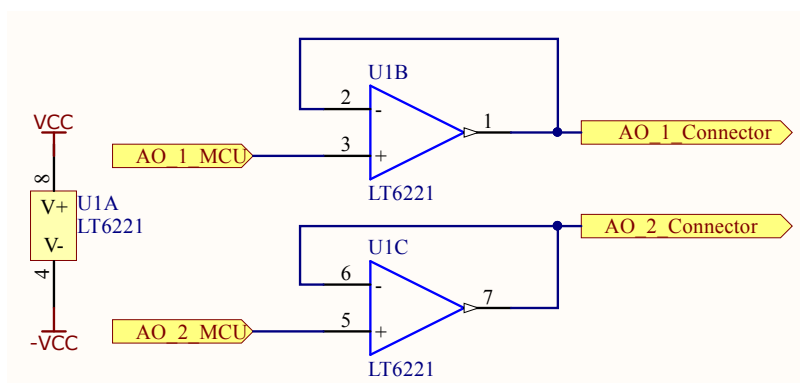
Následující obvody vyžadují jednotlivé funkce ke svému správnému provozu. S výhodou jich lze využít k ochraně samotného MCU, kterému hrozí při nevhodné manipulaci zničení. Typickým příkladem je překročení maximálního vstupního napětí na pinu MCU, což může MCU nevratně zničit.

5.1.1 Obvod pro zdroj stejnosměrného napětí

V bodě 3.6 byl navržen zdroj stejnosměrného napětí. Ten je realizován pomocí D/A převodníku, jehož převodní charakteristika je vyobrazena na obrázku 3.10. Na ní je možné vidět nelinearitu zhruba v oblasti kódových slov 0 až 64, které v případě použité napěťové reference $V_{ref} = 3,4886V$ odpovídají rozsahu $0V - 53mV$. Tato nelinearita je způsobena konstrukčním omezením interního napěťového sledovače, který má za úkol snížit vnitřní odpor D/A převodníku, aby byl umožněn jeho větší proudový odběr (viz obrázek (5.1)). Omezení spočívá ve faktu, že operační zesilovač (OZ), kterým je tento sledovač realizován nezvládá sledovat napětí až k hranicím svého napájení, což je důvod pokřivení zmiňované charakteristiky.



Obrázek 5.1: Vnitřní uspořádání D/A převodníku v STM32F429 [5]



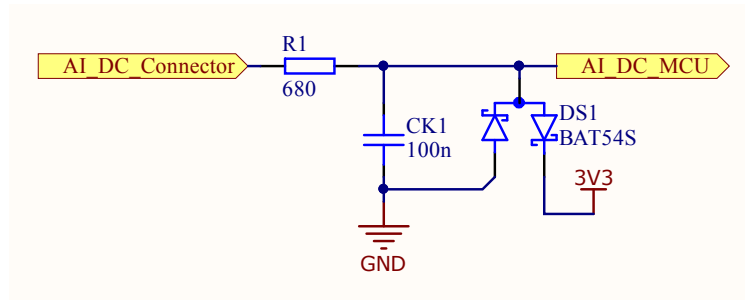
Obrázek 5.2: Schéma s napěťový sledovači pro výstup D/A převodníku

Aby bylo možné zmiňovaný jev eliminovat, byl vnitřní sledovač softwarově vypnut a jako náhrada použit externí OZ. K realizaci byl vybrán obvod LT6221 s dvěma OZ v jednom pouzdře (s výhodou každý pro jeden kanál D/A převodníku), a který je schopen každým OZ dodávat na svůj výstup až 20mA [13]. Tyto OZ mohou být napájeny tzv. duálně, tedy i zápornou napájecí větví. Tím je docíleno fungování napěťového sledovače v oblasti nulového výstupního napětí. Obvodové schéma je uvedeno na obrázku (5.2).

5.1.2 Vstupní obvod pro měření stejnosměrného napětí

A/D převodník je velmi citlivá snadno zničitelná součást MCU. Je proto vhodné chránit jej před nežádoucími vlivy, například špatnou manipulací s MD, kterou může být například měření napětí mimo stanovený rozsah převodníku.

K tomuto účelu slouží dvojice ochranných Schottkyho diod BAT54 v zapojení na obrázku (5.3). Těmi začíná téci proud v případě, že je napětí na vstupním pinu mimo vstupní rozsah převodníku a tím ho chrání. Tento proud se zvyšujícím napětím roste až do úrovně 100mA při napětí $0,8\text{V}$ nad, nebo pod rozsahem A/D, kdy se tato dioda nevratně zničí [15]. Aby byl tento rozsah zvětšen, byl před vstupní konektor zařazen rezistor $R_1 = 220\Omega$. Ten posouvá hranici, kdy Schottkyho diodou bude protékat proud více jak 100mA o 22V . S výhodou je tento rezistor doplněn o kondenzátor C_{K1} s hodnotou 100nF . Tyto dva prvky společně tvoří dolnoproustní filtr prvního řádu se zlomovou frekvencí $f_c = 7234\text{Hz}$, která vychází



Obrázek 5.3: Obvodový návrh pro piny měřící stejnosměrné napětí

ze vztahu 5.1[12].

$$f_c = \frac{1}{2\pi R_1 \cdot C_{K1}} \quad (5.1)$$

Pro odstup signál-šum (SNR) 74dB 12-bitového A/D převodníku STM32F429, který se spočítá jako

$$SNR = 6,02 \cdot n + 1,76, \quad (5.2)$$

kde n udává počet bitů převodníku je tento filtr užitečný zejména pro potlačení vysokofrekvenčních šumů, které se pohybují nad hodnotou 36MHz [12].

5.1.3 Vstupní obvod pro vzorkované měření napětí

V bodě 3.5 byla navržena funkce kontinuálního snímání napětí A/D převodníkem s vzorkovací frekvencí $f_s = 340kS/s$. Aby bylo zaručeno, že převodník bude digitalizovat pouze správný signál, je třeba dodržovat vzorkovací teorém

$$f_s \geq 2f_{max}, \quad (5.3)$$

kde f_{max} značí maximální frekvenci vzorkovaného signálu [12]. Aby bylo tohoto teorému dosaženo pro celý vstupní rozsah převodníku, který je 74dB (viz(5.2)), je před převodník nutné předřadit vhodný dolnoproputní filtr. Aby se tímto krokem snížila šířka frekvenčního pásma měření co možná nejméně, byl zvolen filtr 5-tého řádu, který je na schématu na obrázku (5.4) reprezentován dvěma aktivními filtry druhého řádu typu *Sallen-Key* [12] ($OZ U_3$ a U_4) a jedním pasivním filtrem, který reprezentuje RC článek složený z rezistoru R_1 a kondenzátoru C_1 pro kladnou vstupní větev, a dvojice R_2 a C_3 pro zápornou větev vstupního přístrojového zesilovače U_1 (AD623), kterého bylo z výhodou užito pro umožnění měření napětí v rozsahu $< \frac{-V_{refADC}}{2} ; \frac{V_{refADC}}{2} >$. Toho je u zmiňovaného přístrojového zesilovače docíleno posunutím napěťové reference o $\frac{V_{refADC}}{2}$ [14]. Tento posun realizuje $OZ U_2$ pomocí napěťového děliče složeného z rezistorů R_5 a R_6 .

Potřebná zlomová frekvence $f_c = 30940Hz$ vstupního filtru byla spočítána podle následujícího odvozeného vztahu

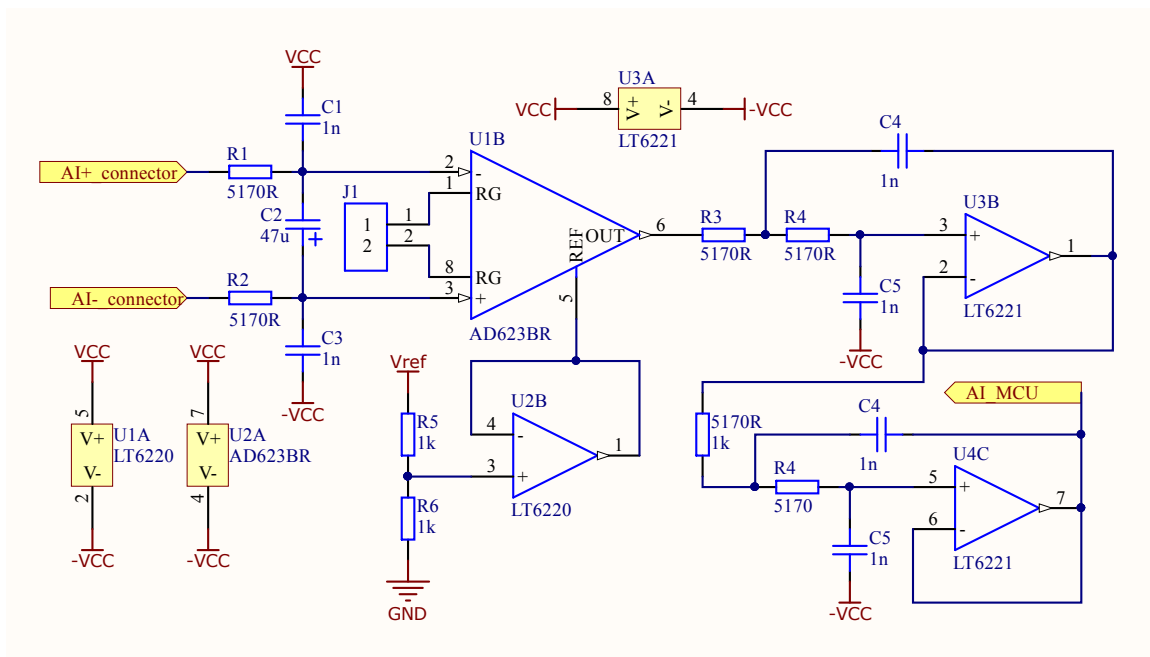
$$f_c = \frac{\frac{f_s}{2}}{10^{\frac{SNR_{ADC}}{20n}}}, \quad (5.4)$$

kde n značí řád použitého filtru. Tato zlomová frekvence zároveň určuje maximální frekvenci užitečného signálu, který je možné vzorkovat (výšší frekvence jsou tlumeny filtrem). Pro dosažení určené zlomové frekvence bylo nutné najít hodnoty rezistorů a kondenzátorů jednotlivých filtrů, které svazují vzorce 5.55.65.7[12].

$$f_c = \frac{1}{2\pi R_1 C_1} \quad (5.5)$$

$$f_c = \frac{1}{2\pi R_2 C_3} \quad (5.6)$$

$$f_c = \frac{1}{2\pi R_3 C_4 R_4 C_5} \quad (5.7)$$

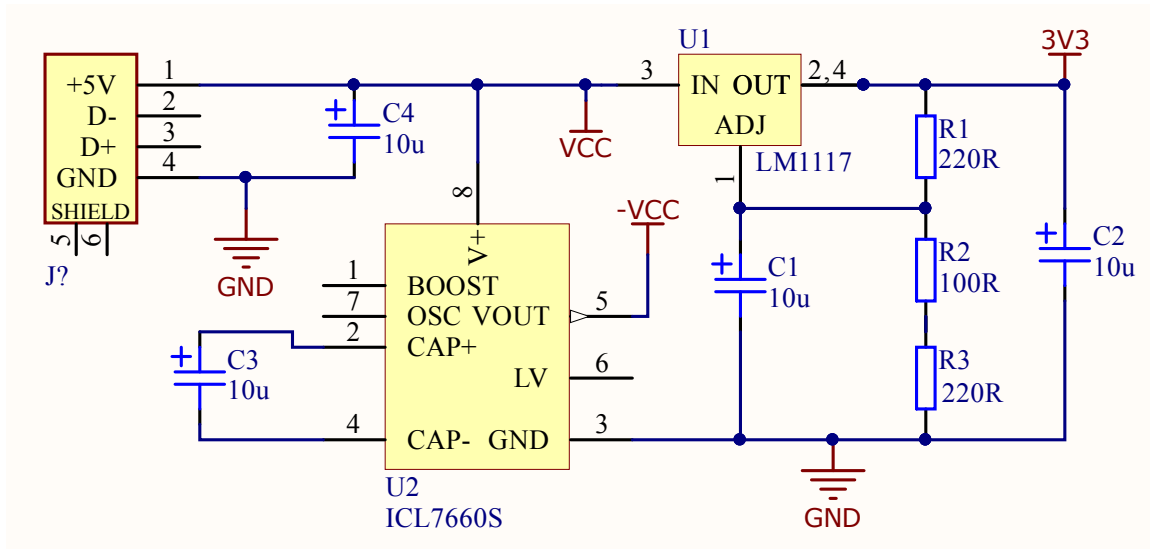


Obrázek 5.4: Schéma filtračního obvodu pro A/D převodník s vzorkovací frekvencí $340kS/s$

5.2 Napájení MD

Návrhy zdrojů napájení pro jednotlivé součásti MD jsou uvedeny na obrázku 5.5 a 5.6. Jako zdroj vstupního napětí slouží USB, používané pro komunikaci. To zaručuje dle specifikace $4,75V - 5,25V$ [16].

Pro běh MCU je vyžadováno napětí v rozsahu $1,7V - 3,6V$. Jako jeho zdroj byl vybrán lineární napěťový stabilizátor $LM1777-ADJ$, jehož výstupní napětí určuje vzorec 5.8[17]. Pomocí něj byly zvoleny velikosti rezistorů $R1$ a $R2$ s cílem co nejvíce se přiblížit napětí $3,3V$. Toto napětí se často vyskytuje v elektronice jako nominální napětí digitálních periférií.



Obrázek 5.5: Schéma návrhu zdrojů napětí pro realizaci MD

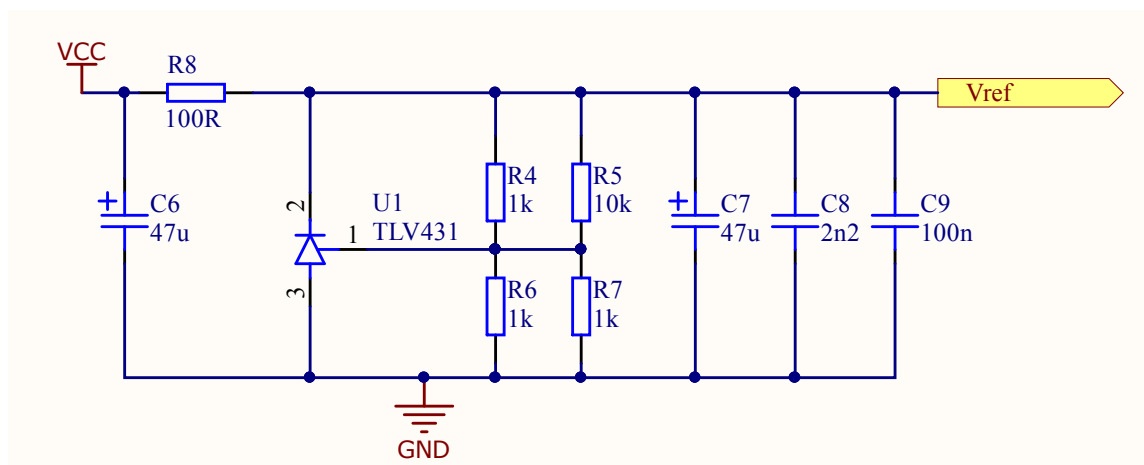
Zároveň musela být splněna podmínka minimálního proudového odběru ze stabilizátoru alespoň 5mA pro jeho bezproblémový chod [17]. Výsledně byly zvoleny hodnoty rezistory $R1 = 220\Omega$ a $R2 = 367\Omega$ (sériové zapojení rezistorů $100\Omega, 220\Omega$ a 47Ω) určující výstupním napětí $V_{out} = 3,33\text{V}$.

$$V_{out} = 1,25 \cdot \left(1 + \frac{R2}{R1}\right) \quad (5.8)$$

K zajištění co nejpřesnějších výsledků při práci s analogovými periferiemi (D/A, A/D převodníky), disponuje STM32F429II samostatným vstupem pro napájení jeho analogových periferií (V_{DDA}) a vstupem pro přesnou napěťovou referenci (V_{ref}) [5]. Hodnota napěťové reference určuje maximální rozsah pro analogové převodníky. Zároveň je nutné tuto hodnotu přesně změřit a uložit do paměti MD, jelikož jsou od ní odvozeny výsledky všech analogových měření. Vzhledem k malému proudovému odběru obou těchto vstupů ($2,3\text{mA}$) a jejich podobným nárokům na napěťovou stálost, bylo rozhodnuto použít jeden zdroj pro oba tyto vstupy [5]. Pro tento účel byl vybrán obvod *TLV431* pracující na principu *Band Gap*, jehož změna napětí při maximální změně vstupního proudu vstupů je s jistotou menší než napětí 1LSB převodníků [18].

Nastavení velikosti výstupního napětí probíhá obdobně jako u obvodu *LM1777*, tedy pomocí vhodně zvoleného dělicího poměru napěťového děliče přivedeného do referenčního vstupu *TLV431*. Cílem bylo dosáhnout výstupního napětí $3,5\text{V}$, které bylo vybráno z možného rozsahu $1,7 - 3,6\text{V}$ pro V_{DDA} a V_{REF} tak, aby by byl napěťový rozsah D/A a A/D převodníků, s jistou rezervou, co největší. Dle výpočtu (5.9) s vybranými hodnotami rezistorů $R4, R6, R7 = 1\text{k}\Omega$, $R5 = 10\text{k}\Omega$, bylo docíleno hodnoty výstupního napětí $V_{ref} = 3,4938\text{V}$ (při hodnotách $V_{refTLV} = 1,24\text{V}$ a $I_{refTLV} = 0,5\mu\text{A}$).

$$V_{ref} = V_{refTLV} \cdot \left(1 + \frac{R4||R5}{R6||R7}\right) - I_{refTLV} \cdot R4||R5 \quad (5.9)$$



Obrázek 5.6: Schéma návrhu přesné napěťové reference pro analogové obvody STM32F429

Jak byl v bodě 2.3.2 zmíněno, byl tento obvod realizován jako tzv. *shield* pro používaný discovery kit (viz obrázek 2.5). Pro stejné hodnoty rezistorů jako ve výpočtu (5.9) byla u této realizace naměřena hodnota $V_{ref} = 3,4886V$.

Zdrojem záporného napětí pro operační zesilovače je nábojová pumpa ICL7660. Ta je schopná pomocí střídavého dvoupólového přepínání kondenzátoru $C3$ dodávat na svůj výstup napětí opačné polaritě k napětí vstupnímu s maximálním výstupním proudem $20mA$. Tento proud je dostatečný pro plánovaný odběr všech operačních zesilovačů na MD.

Kondenzátory $C1 - C9$ jsou tzv. blokovací a jsou voleny dle datových listů jednotlivých součástek.

6 Zhodnocení dosažených výsledků

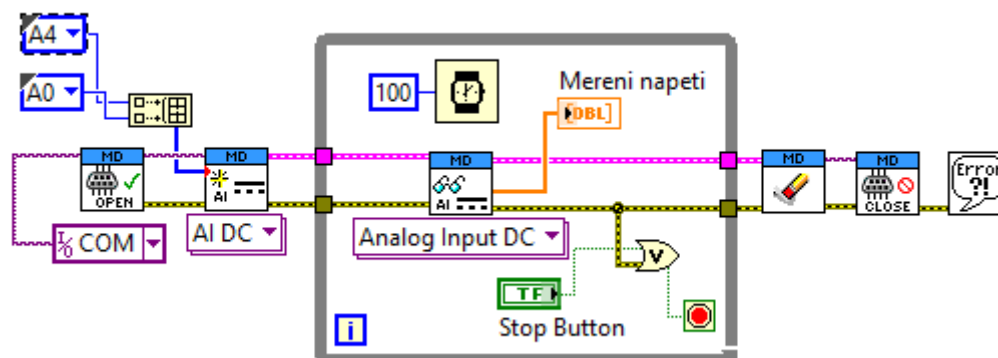
V této kapitole jsou diskutovány dosažené výsledky vývoje multifunkční měřicí karty, meze její použitelnosti a možnosti jejího budoucího rozšíření.

6.1 Měření stejnosměrného napětí

Tato funkce supluje funkci digitálního voltmetru s rozsahem $0V$ až $3,4938V$. Celkem byla realizována na 18 vstupech, limitně však mohou být v jeden čas inicializovány maximálně 3 (každý pro jeden převodník).

Přesnost $\pm 0.9 \text{ LSB}$ byla dosažena pomocí průměrování 100 vzorků, které jsou fixní vzorkovací frekvencí 5 kHz sejmuty během 0,02 sekundy. Výhoda tohoto postupu je potlačení rušení síťovým napětím, které má 50 Hz .

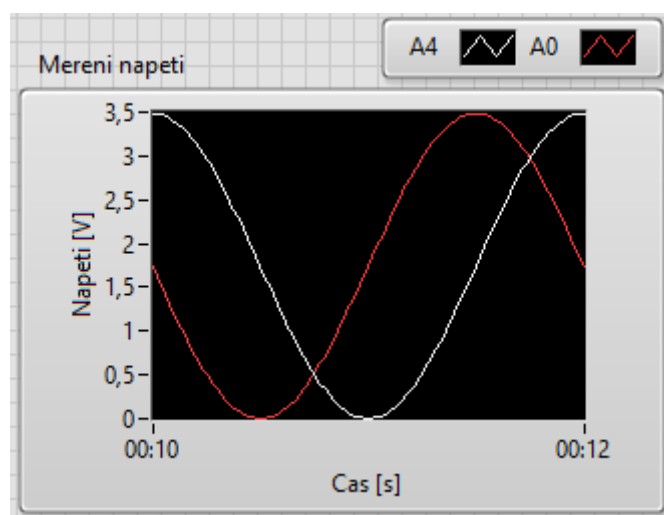
Možná realizace měření v programu LabVIEW je zobrazena na obrázku (6.1), kde je cyklicky snímána hodnota napětí dvou vstupů ve zvoleném intervalu 100 ms . Příkazy pro ovládání této funkce bez LabVIEW jsou uvedeny v tabulce 6.1.



Obrázek 6.1: Měření stejnosměrného napětí dvěma kanály MD v programu LabVIEW

Příkaz	Význam
AI:DC:PIN <číslo_pinu>:INIT	Alokace periferie MCU a inicializace funkce danému pinu
AI:DC:PIN <číslo_pinu>:Read	Příkaz pro sejmutí napětí
AI:DC:PIN <číslo_pinu>:DEINit	Uvolnění prostředků MCU

Tabulka 6.1: Příkazy pro ovládání měření stejnosměrného napětí pomocí MD



Obrázek 6.2: Výsledek měření z obrázku 6.1

Hardwarové uspořádání vstupů pro tuto funkci bylo navrženo tak, aby bylo chránilo A/D převodník proti napětí mimo jeho měřicí rozsah, a to až do mezích hodnot $-22V$ až $25V$. Zároveň byl na měřicí vstup aplikován dolnoproustní filtr, který filtruje šum s frekvencemi nad $36MHz$.

Prostor pro zdokonalení této funkce je zejména v potlačení šumu navýšením počtu průměrovaných vzorků. Dále lze funkci rozšířit o možnost synchronního odběru vzorku až z tří kanálů současně (díky třem nezávislým A/D převodníkům STM32F4), nebo o spouštění odměru externím signálem.

V neposlední řadě by bylo výhodné za vstupní konektor umístit multiplexor, který by přepínáním svých vstupů měnil odbočky napěťové děliče, čímž by bylo na úkor několika digitálních výstupů dosaženo přepínání vstupního rozsahu převodníku v mezích stanovených dělicím poměrem tohoto děliče.

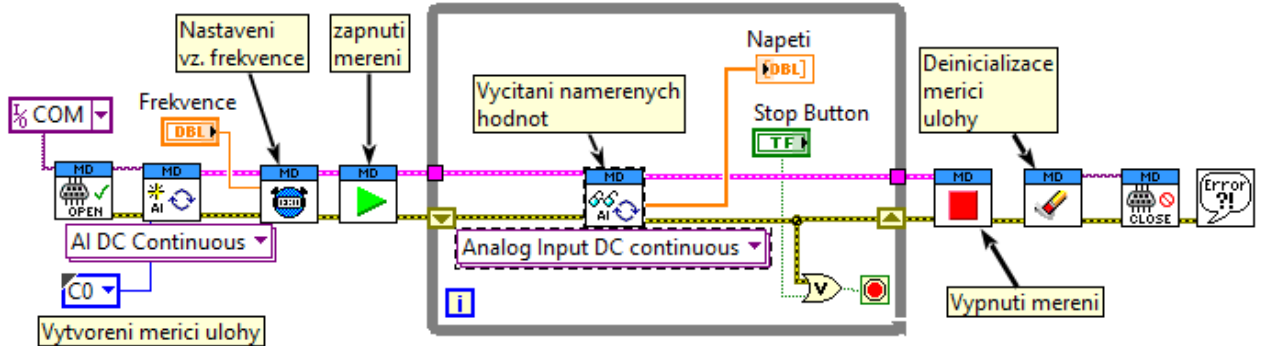
6.2 Kontinuální měření napětí s definovaným vzorkováním

Pomocí této funkce je možné vzorkovat jeden ze čtyř k tomuto účelu přizpůsobených napěťových vstupů vzorkovací frekvencí $340kHz$ s přesností $\pm 4LSB$. Tato hodnota je ve srovnání s měřením stejnosměrného napětí větší z důvodu průměrování pouhých pěti vzorků.

Navržený vstupní filtr dovoluje rozsah měření signálu až do frekvence $30kHz$. Kromě tohoto filtru byl použit přístrojový zesilovač s posunutou napěťovou referencí na úroveň poloviny referenčního napětí A/D převodníku. Umožňuje tak měření napětí v rozsahu $-1,7V$ až $1,7V$. Toto měření je zároveň diferenční, je tedy nutné měřený signál připojit na patričné vstupy, nikoli mezi napěťový vstup a zemi, jako tomu bylo v bodě (6.1).

Na obrázku 6.3 je zobrazeno doporučené uspořádání měření v programu LabVIEW. Vzorkování, a tím i měření, je nutné dle konceptu zapnout a vypnout příslušnými VI. Dále je možné programově měnit rychlost vzorkování (v rozsahu $0,0097Hz$ až $340kHz$). Funkce tak

může být využita například jednoduché měření napětí v terminálu sériové linky s definovanou frekvencí zobrazování hodnot. Pro tuto možnost slouží příkazy uvedené v tabulce 6.2.



Obrázek 6.3: Kontinuální, vzorkované měření napětí v programu LabVIEW pomocí MD

Příkaz	Význam
AI:CONT:PIN <číslo_pinu>:INIT	Alokace periferie MCU a inicializace funkce danému pinu
AI:CONT:PIN <číslo_pinu>:Start	Zapnutí měření (MCU začne odesílat hodnoty)
AI:CONT:PIN <číslo_pinu>:Stop	Vypnutí měření (MCU přestane odesílat hodnoty)
AI:CONT:PIN <číslo_pinu>:DEINit	Uvolnění prostředků MCU
AI:CONT:PIN <číslo_pinu>:FRQ <hodnota>	Nastavení frekvence vzorkování. Počáteční hodnota je nastavena na 340kS/s. Lze použít v kombinaci s „?“ pro zjištění vzorkovací frekvence

Tabulka 6.2: Příkazy pro ovládání kontinuálního měření napětí

Obdobně, jako v případě měření stejnosměrného napětí 6.1, by tato funkce šla vylepšit přidáním možnosti odstartovat záznam pomocí externího triggeru, přepínáním vstupního rozsahu, potlačení šumu průměrováním více vzorků, které by mohlo být realizováno zvýšením vzorkovací frekvence spoluprací všech tří A/D převodníků, které má STM32F4 k dispozici (a tuto možnost podporuje).

Dalším námětem na zlepšení funkce může být přepínatelné nastavení zesílení vstupního přístrojového zesilovače, nebo možnost ukládat vzorky do paměti. Tím by se otevřela možnost vzorkovat signály maximální možnou rychlostí A/D převodníků.

6.3 Generování pulsně šířkové modulace (PWM)

STM32F429 dovoluje generovat PWM na 34 výstupech. Amplitudu signálu určuje napájecí napětí MCU, tedy 3,3V určených v bodě 5.2. Z důvodu různých parametrů jednotlivých čítačů se pro jednotlivé výstupy mohou lišit rozsahy dosažitelných frekvencí. Vliv těchto parametrů na rozsahy popisuje tabulka 6.3.

Rozlišení čítače [bit]	Rychlost časové základny čítače [MHz] (pro 8MHz krystalový oscilátor)	Rozsah nastavitelných frekvencí
32	84	$2,98 \cdot 10^{-7} \text{ Hz}$ až 42 MHz
32	42	$1,49 \cdot 10^{-7} \text{ Hz}$ až 21 MHz
16	84	$0,019 \text{ Hz}$ až 42 MHz
16	42	$0,0097 \text{ Hz}$ až 21 MHz

Tabulka 6.3: Tabulka rozsahů generování PWM na STM32F4

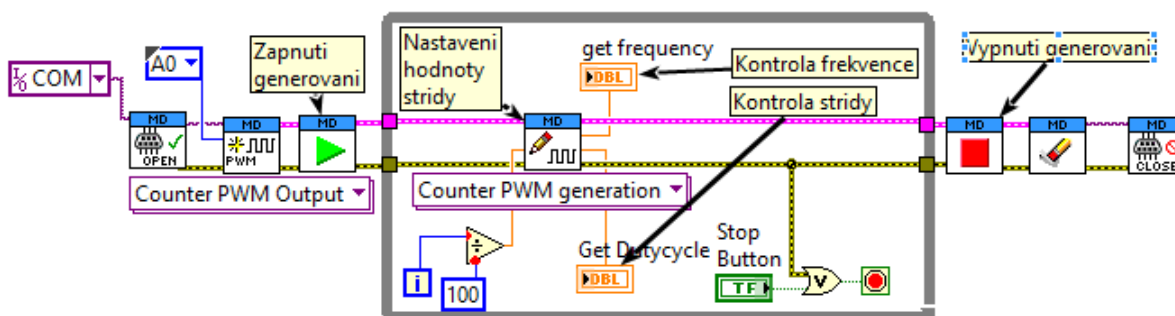
Příkazy uvedenými v tabulce 6.4 lze frekvenci a střihu dynamicky měnit během měření (po změně frekvence se automaticky přepočítávají registry MCU určující střihu, tak aby střiha zůstala nezměněna). Z principu jakým je PWM na MCU STM32F429 generována vyplývá, že dosažení požadovaných frekvencí může být pro některá zadání zatíženo chybou, jejíž velikost záleží právě na zvolené frekvenci (viz bod 3.4).

Příkaz	Význam
CNT:PWM:PIN <číslo_pinu>:INIT	Alokace periferie MCU a inicializace funkce danému pinu
CNT:PWM:PIN <číslo_pinu>:StArt	Zapnutí generování PWM
CNT:PWM:PIN <číslo_pinu>:StoP	Vypnutí generování PWM
CNT:PWM:PIN <číslo_pinu>:DEINit	Uvolnění prostředků MCU
CNT:PWM:PIN <číslo_pinu>:FRQ <hodnota>	Nastavení frekvence PWM [Hz]. Desetiny jsou odděleny tečkou. Lze použít v kombinaci s „?“ pro zjištění vzorkovací frekvence
CNT:PWM:PIN <číslo_pinu>:DuTy <hodnota>	Nastavení střihu PWM [0 až 1]. Desetiny jsou odděleny tečkou. Lze použít v kombinaci s „?“ pro zjištění vzorkovací frekvence

Tabulka 6.4: Příkazy pro ovládání generování PWM

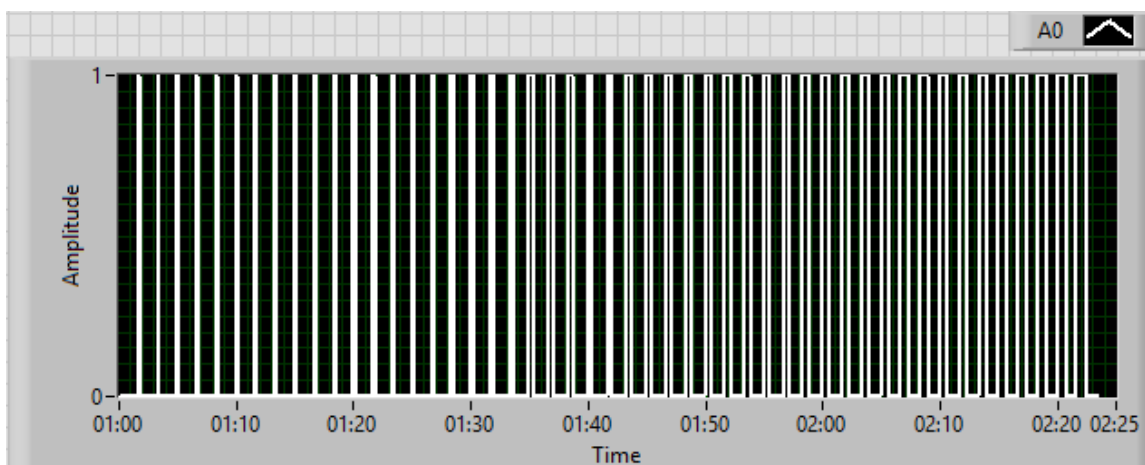
Při zadání požadované frekvence jsou vypočítány parametry pro řídicí registry čítače s ohledem na co největší přiblížení k zadané hodnotě. K zjištění této hodnoty složí příkazy s „?“ tedy :FRQency? a DuTy?

Ukázka generování PWM pomocí programu LabVIEW je na obrázku 6.4. Zde je v každé iteraci smyčky zvýšena střída o 1%. Opětovně je možné vidět, že lze generování pomocí příslušných VI vypnout a zapnout.



Obrázek 6.4: Ukázka generování PWM s proměnou frekvencí a střídou pomocí MD a programu LabVIEW

Kromě generování PWM umožňují čítače STM32F429 mnoho dalších funkcí, které mohou být námětem k rozšíření. Mezi ně patří možnost měření šířky pulsu, měření frekvence, příprava pro práci s kvadraturními enkodéry, nebo by bylo možné implementovat například funkce pro práci se servomotory.



Obrázek 6.5: Náměr hodnot generovaných kódem na obrázku 6.4 (pomocí měřicí karty NI 6216)

6.4 Zdroj stejnosměrného napětí

Zdrojem stejnosměrného napětí se stal 12-bitový D/A převodník. Ten je schopen generovat napětí od 0V do velikosti napěťové reference 3,48V, a to na dvou kanálech. Pro jeho správnou funkci musel být vypnut napěťový sledovač umístěný v MCU a nahrazen externím sledovačem s duálním napájením, který dovoluje provozovat D/A převodník už od napětí 0V.

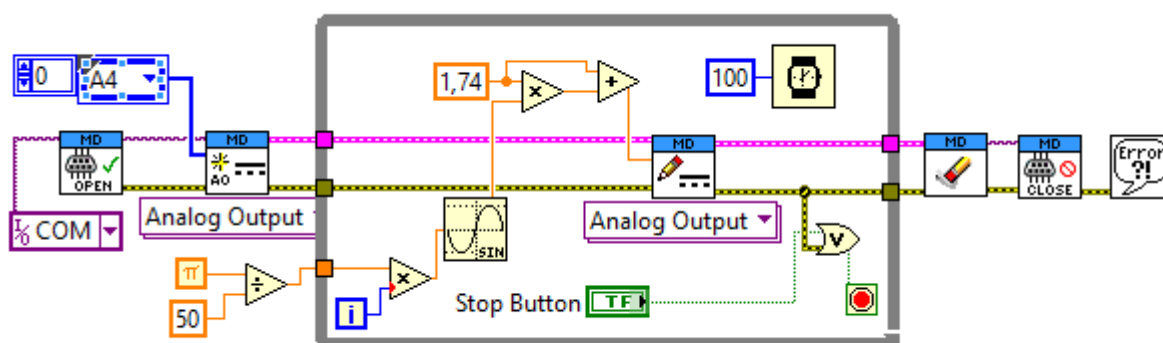
Ukázka programu v LabVIEW pracující s D/A převodníkem je na obrázku 6.6. Zde je

6 Zhodnocení dosažených výsledků

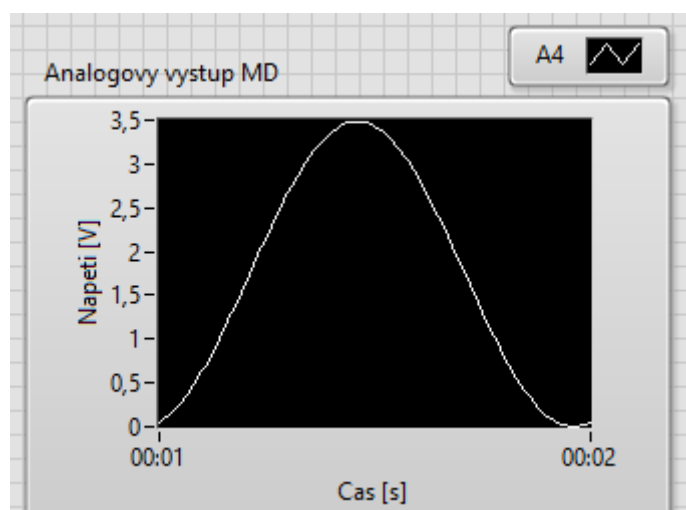
Příkaz	Význam
A0:DC:PIN<číslo_pinu>:INIT	Alokace periferie MCU a inicializace funkce danému pinu
A0:DC:PIN<číslo_pinu>:WRite <hodnota>	Příkaz pro změnu napětí na výstupu [V]
A0:DC:PIN<číslo_pinu>:DEINit	Uvolnění prostředků MCU

Tabulka 6.5: Příkazy pro práci s analogovým výstupem MD

jednotlivými vzorky generován sinusový signál. Ten je pro kontrolu měřen měřicí kartou NI 6216, a je jej možné vidět na obrázku 6.7.



Obrázek 6.6: Ukázka práce s D/A převodníkem MD v programu LabVIEW

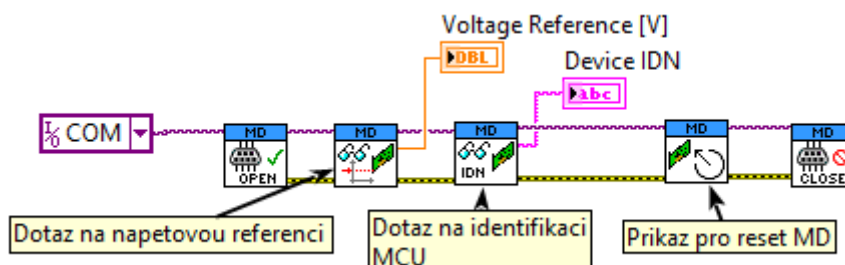


Obrázek 6.7: Výsledný signál generovaný úlohou z obrázku 6.6

Možným rozšířením funkcionalit na bázi D/A převodníku by mohly být funkce generující různé průběhy (pila, sinus, trojúhelník) pomocí „přehrávání“ vzorků z paměti, nebo pomocí digitální syntézy DDS.

Příkaz	Význam
*IDN?	Vyvolání sériového čísla MD (pro každou MD unikátní)
*RST	Vyvolání softwarového resetu MCU a tím navrácení desky do výchozího stavu
*VREF?	Vyvolání napěťové reference [V], kterou MD používá pro výpočet hodnot

Tabulka 6.8: Obecné příkazy pro MD pro její správu



Obrázek 6.9: Využití funkcí pro správu MD v LabVIEW

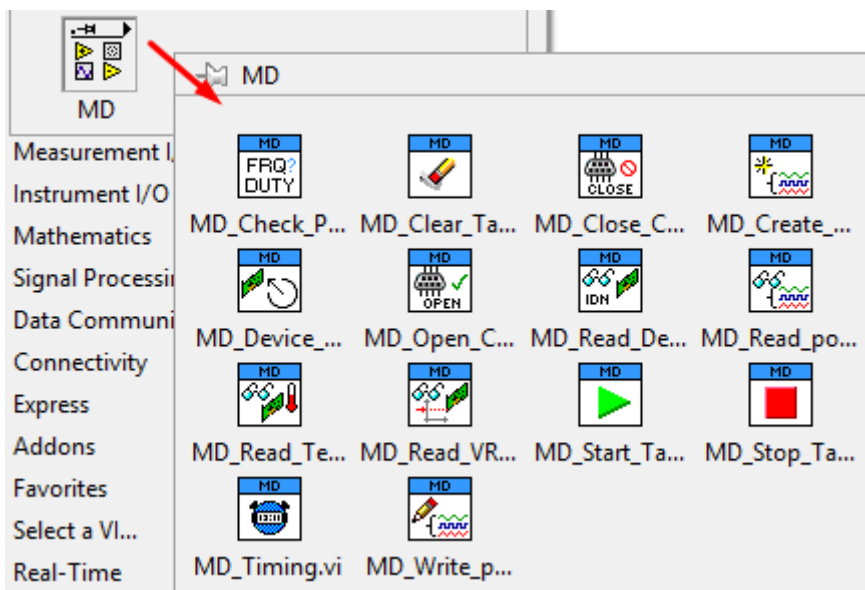
Aby bylo možné MD kalibrovat, byla implementována dvojice příkazů, které jsou ve spolupráci schopny změnit velikost napěťové reference. Prvním příkazem - `SYS:VREF <hodnota_napětí>` se předá MD nová velikost napěťové reference. Druhý příkaz `SYS:CONFIRM` uloží hodnotu do ne-volatilní paměti MCU. Tyto příkazy nejsou úmyslně implementovány v knihovně funkcí LabVIEW.

Dalšími směry rozvíjení funkcí MD by mohly být jiné fyzické vrstvy komunikace. STM32F429 disponuje periferií pro nativní komunikaci přes USB. To by mohlo poskytnout bezchybnou komunikaci a větší datovou propustnost. Další možnou komunikační periferií se stejnými výhodami je Ethernet. Díky němu by bylo možné používat MD kdekoliv v podnikové síti s možností obsluhy na dálku.

V neposlední řadě je možné MD využívat jako most pro komunikační periferie jako jsou SPI, RS232, RS485 nebo I2C.

6.7 Paleta LabVIEW

Pro snadnější práci s MD v programu LabVIEW pro něj byla navržena tzv. funkční paleta. Tu je možné vidět na následujícím obrázku



Obrázek 6.10: Paleta funkcí vytvořená pro snadnější práci s MD v LabVIEW

7 Závěr

Cílem práce bylo navržení softwaru a hardwaru multifunkční měřicí desky založené na MCU STM32F429, která by jako *low-endová* varianta doplnila ekosystém měřících, s programem LabVIEW kompatibilních, multifunkčních karet ve společnosti BTL.

Prvním krokem vývoje byl průzkum firemních požadavků na novou měřicí desku. Po něm proběhla analýza možností samotného MCU STM32F429, která ukázala, že jeho prostředky budou stačit pro splnění zadání.

Aby mohla být splněna podmínka integrace navržených měřících funkcí multifunkční desky do programu LabVIEW, bylo nutné vybrat vhodnou fyzickou vrstvu pro komunikaci, kterou se stalo rozhraní UART. Pro toto rozhraní byl dále navržen komunikační protokol založený na SCPI příkazech, jehož výhodou je možnost pracovat s navrženou měřicí deskou, v případě potřeby, bez programu LabVIEW skrze terminál sériové linky, čímž se navrhovaná měřicí deska stává dostupnějším měřícím nástrojem většímu počtu firemních vývojářů a testerů.

Nutným krokem, před implementací požadovaných měřících funkcí do MCU, bylo navrhnout robustní systém pro kontrolu měřících prostředků a správu samotného měření navázanou na navržený komunikační protokol. Tento bod se ukázal být stěžejním, neboť zajišťuje spolehlivost měření a umožňuje velice snadno rozšířit paletu měřících funkcí, a tím i multifunkčnost navrhované měřicí desky, což bylo cílem.

Pro snadnou práci s navrženou měřicí deskou byla předprogramována paleta funkcí pro program LabVIEW včetně ukázek doporučeného použití. Tímto krokem byla splněna původní vize začlenit navrhovanou měřicí desku do systému stávající měřících desek používaných ve společnosti BTL.

Poslední řadě bylo navrženo hardwarové zapojení pro realizaci měřicí desky. Toto schéma odpovídá požadavkům na zpracování signálu navržených funkcí a zajišťuje chod MCU včetně komunikace.

Diskuzi dosažených výsledků návrhu multifunkční měřicí desky, mezi její použitelnosti, příklady jejího užití ve spolupráci s programem LabVIEW, a jejího možného budoucího rozšíření shrnuje kapitola 6.

Realizace této práce ukázala, že úroveň funkcionalit a periférií, které jsou schopny propůjčit dnešní moderní mikrokontroléry, otevírá možnost jejich plnohodnotného nasazení vedle dražších a sofistikovanějších měřících zařízení.

Literatura

- [1] BTL Vanquish. In: BTL Aesthetics [online]. © 2016 Aesthetic Everything. [vid. 9.5.2016]. Dostupné z: http://aestheticeverything.com/wp-content/uploads/2015/04/BTL-Vanquish_unit-with-logo-2-1.jpg
- [2] NI USB-6008 OEM. In: *NI USB-6008 OEM* [online]. © 2016 National Instruments. [vid. 10.5.2016]. Dostupné z: <http://sine.ni.com/nips/cds/view/p/lang/cs/nid/202752>
- [3] National Instruments. *USB-6008 Bus-Powered Multifunction DAQ USB Device*. Dostupné z: <http://www.ni.com/pdf/manuals/375295a.pdf>.
- [4] YIU, Joseph. *The definitive guide to ARM CORTEX-M3 and CORTEX-M4 processors. 3rd ed.* Oxford: Elsevier, c2014, xxxv, 818 s. ISBN 978-0-12-408082-9.
- [5] ST Microelectronics. *STM32F429xx ARM® Cortex-M4 32b MCU Datasheet*, Zář 2015. Dostupné z: <http://www.st.com/web/en/resource/technical/document/datasheet/DM00071990.pdf>.
- [6] STM32F429. In: STM Documents Tips [online]. © 2016 Aesthetic Everything. [vid. 7.5.2016]. Dostupné z: <http://documents.tips/documents/154432059-stm32f4-technical-training.html>.
- [7] EXAR. *XR21B1421 Enhanced 1-Ch Full-Speed USB HID to UART Bridge*, Ř 2014. Dostupné z: <https://www.exar.com/content/document.ashx?id=21517>.
- [8] SCPI Consortium. *SCPI Syntax & Style*, 1999. Dostupné z: <http://www.ivifoundation.org/docs/scpi-99.pdf>.
- [9] MISHR, Sattyam Kishor; PANDE, Manish. *An Efficient Word Matching Algorithm For off Line Text*. Global Journal of Computer Science and Technology, 2010, 10.11.
- [10] C/C++ trick: static string hash generation. Lolengine.net [online]. 12.1.2012 [vid. 8.2.2016]. Dostupné z: <http://lolengine.net/blog/2011/12/20/cpp-constant-string-hash>.
- [11] ST Microelectronics. *STM32F4xxx advanced ARM® - based 32-bit MCUs Reference manual*, Ř 2014. Dostupné z: http://www.st.com/web/en/resource/technical/document/reference_manual/DM00031020.pdf.
- [12] VEDRAL, Josef a Jan FISCHER. *Elektronické obvody pro měřící techniku*. Vyd. 2. Praha: Vydavatelství ČVUT, 2004. ISBN 80-01-02966-2.
- [13] Linear Technology. *LT6221 Dual, Low Power, Rail-to-Rail In-*

Literatura

- put and Output Precision Op Amps*, Květen 2015. Dostupné z: <http://cds.linear.com/docs/en/datasheet/622012fc.pdf>.
- [14] Analog Devices. *AD623 Single-Supply, Rail-to-Rail, Low Cost Instrumentation Amplifier*, Červen 2008. Dostupné z: <http://www.analog.com/media/en/technical-documentation/data-sheets/AD623.pdf>.
- [15] NXP Semiconductors. *BAT 54 Surface Mount Schottky Barrier Diode*, Říjen 2012. Dostupné z: http://www.nxp.com/documents/data_sheet/BAT54_SER.pdf.
- [16] Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, Philips. *USB Universal Serial Bus Specification*, Duben 2000. Dostupné z: http://sdphca.ucsd.edu/lab_equip_manuals/usb_20.pdf.
- [17] Texas Instruments. *LM1117 800-mA Low-Dropout Linear Regulator*, Leden 2016. Dostupné z: <http://www.ti.com/lit/ds/symlink/lm1117.pdf>.
- [18] Texas Instruments. *TLV431x Low-Voltage Adjustable Precision Shunt Regulator*, Leden 2015. Dostupné z: <http://www.ti.com/lit/ds/symlink/tlv431a.pdf>.
- [19] Maxim. *ICL7660 Switched-Capacitor Voltage Converter*, Července 1997. Dostupné z: <https://datasheets.maximintegrated.com/en/ds/ICL7660-MAX1044.pdf>.

Seznam obrázků

2.1	Přístroj BTL Vanquish[1]	2
2.2	Fixtura pro testování osazených desek plošných spojů	3
2.3	Univerzální měřicí deska NI 6008 ve verzi se šroubovacími konektory a OEM verzi [2]	3
2.4	Vnitřní struktura STM32F429[6]	5
2.5	Discovery kit s MCU STM32F429 s realizovanou externí napěťovou referencí	6
2.6	Program pro odběr vzorku napětí pomocí měřicí karty a standardu DAQ v programu LabVIEW	7
2.7	Jednoduché blokové schéma návrhu měřicí desky	8
3.1	Schéma navržené komunikace MD s PC	11
3.2	Stromová hierarchie SCPI	12
3.3	Komunikační protokol pro zvýšení datové propustnosti komunikace	14
3.4	Diagram procesu inicializace pinu	17
3.5	Princip generování PWM pomocí čítače STM32F429	19
3.6	Naměřená převodní charakteristika převodníku STM32F429	20
3.7	Chyba nuly a zesílení A/D převodníku STM32F429	21
3.8	Šum A/D převodníku STM32F429	22
3.9	Šum průměrování	22
3.10	Přechodová charakteristika	23
4.1	Ukázka programu pro otevření a uzavření komunikace COM pomocí knihovných funkcí MD	25
4.2	Program VI OpenCOM.vi	26
4.3	Diagram funkce	26
4.4	Spolupráce VI <i>MD_ACK.vi</i> s VI <i>MD_Write_Command.vi</i>	27
4.5	Diagram práce VI <i>MD_ACK</i>	28
4.6	Vytvoření úlohy měření stejnosměrného napětí	29
4.7	Vytváření měřicí úlohy s více piny	30
4.8	Programová práce s více piny uvnitř VI podporující tuto funkci	30
4.9	Porovnání přístupu otevření a uzavření měřicí úlohy systému DAQmx a navrženého MD	30
5.1	Vnitřní uspořádání D/A převodníku v STM32F429 [5]	32
5.2	Schéma s napěťový sledovači pro výstup D/A převodníku	32
5.3	Obvodový návrh pro piny měřicí stejnosměrné napětí	33

Seznam obrázků

5.4	Schéma filtračního obvodu pro A/D převodník s vzorkovací frekvencí $340kS/s$	34
5.5	Schéma návrhu zdrojů napětí pro realizaci MD	35
5.6	Schéma návrhu přesné napěťové reference pro analogové obvody STM32F429	36
6.1	Měření stejnosměrného napětí dvěma kanály MD v programu LabVIEW .	37
6.2	Výsledek měření z obrázku 6.1	38
6.3	Kontinuální, vzorkované měření napětí v programu LabVIEW pomocí MD .	39
6.4	Ukázka generování PWM s proměnou frekvencí a střídou pomocí MD a programu LabVIEW	41
6.5	Náměr hodnot generovaných kódem na obrázku 6.4(pomocí měřicí karty NI 6216)	41
6.6	Ukázka práce s D/A převodníkem MD v programu LabVIEW	42
6.7	Výsledný signál generovaný úlohou z obrázku 6.6	42
6.8	Ukázka práce digitálními vstupy a výstupy MD v programu LabVIEW . . .	43
6.9	Využití funkcí pro správu MD v LabVIEW	44
6.10	Paleta funkcí vytvořená pro snadnější práci s MD v LabVIEW	45

Seznam tabulek

2.1	Stručné shrnutí parametrů měřicí karty NI 6008[3]	3
2.2	Požadavky společnosti BTL na novou multifunkční měřicí kartu	4
2.3	Výbava MCU STM32F429II[5]	6
3.1	Uvažovaná komunikační rozhraní MCU s PC	10
3.2	Parametry nastavení komunikace UART	11
3.3	Tabulka prefixů odpovědí MD	14
3.4	Proměnné prvku registrového pole <code>pin_reg</code>	16
3.5	Parametry jednotlivých čítačů STM32F429	18
3.6	Statické parametry A/D převodníku STM32F429[5]	21
3.7	Parametry D/A převodníku STM32F429[5]	24
4.1	Výčet proměnných spravovaných <i>task clusterem</i>	29
6.1	Příkazy pro ovládání měření stejnosměrného napětí pomocí MD	37
6.2	Příkazy pro ovládání kontinuálního měření napětí	39
6.3	Tabulka rozsahů generování PWM na STM32F4	40
6.4	Příkazy pro ovládání generování PWM	40
6.5	Příkazy pro práci s analogovým výstupem MD	42
6.6	Meze vstupních a výstupních napětí digitální logiky MD [5]	43
6.7	Tabulka příkazů pro práci s digitálními vstupy a výstupy MD	43
6.8	Obecné příkazy pro MD pro její správu	44

Seznam zkratek

ARM	Advanced RISC Machine
MD	Měřicí deska (nebo také multifunkční měřicí deska)
A/D	Analogově digitální (převodník)
D/A	Digitálně analogový (převodník)
ID	Identifikační číslo
DMA	Direct memory access (přímý přístup k paměti)
DAC	Digital-to-analog converter (Digitálně-analogový (převodník))
ADC	Analog-to-digital converter (Analogově-digitální (převodník))
<i>OTP</i>	One-time programmable memory
OZ	Operační zesilovač

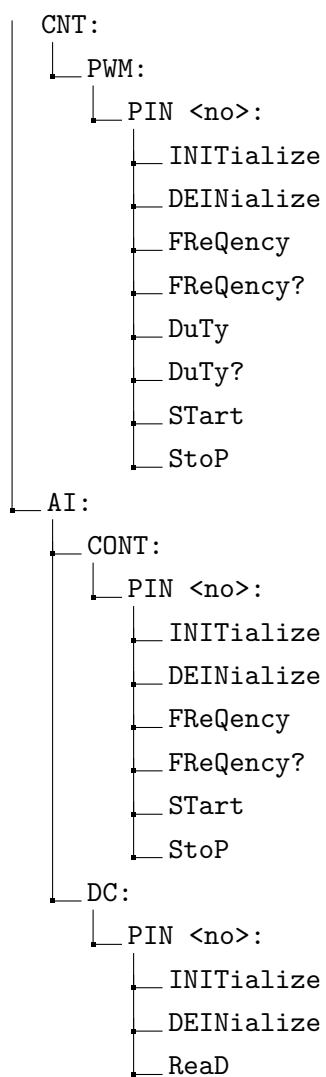
Příloha A

Obsah přiloženého CD

- CD
 - Text Dimplomové Práce (PDF)
 - Zdrojové kódy MCU
 - Zdrojové kódy LabVIEW
 - HW dokumentace
 - Datové listy použitých součástek

Příloha B

Strom komunikačních příkazů pro MD



pokračování

DIO:
DO:
PIN <no>:
INITialize
DEINialize
WRite <hodnota>
DI:
PIN <no>:
INITialize
DEINialize
ReaD
AO:
DC:
PIN <no>:
INITialize
DEINialize
WRite <hodnota>
SYS:
VREF?
VREF <hodnota>
CONFirm
*IDN?
*RST

Příloha C

Rozvržení funkcionalit pro jednotlivé piny

Číslo pinu (pro příkaz)	Označení na STM32F429II	Podporované funkce
0	A0	AI DC, DI, DO
1	A1	AI DC, DI, DO
2	A2	AI DC, DI, DO
3	A3	AI DC, DI, DO
4	A4	AO DC, DO
5	A5	AO DC, DO
6	A6	AI DC, DI, DO
7	A7	AI DC, DI, DO
8	A8	PWM
9	A9	PWM
10	A10	PWM
11	A11	PWM
12	A12	DI, DO
13	A13	No FNC (SWD)
14	A14	No FNC (SWD)
15	A15	PWM
16	B0	AI DC, DI, DO
17	B1	AI DC, DI, DO
18	B2	DI, DO
19	B3	PWM
20	B4	PWM
21	B5	PWM
22	B6	PWM
23	B7	PWM
24	B8	PWM
25	B9	PWM
26	B10	PWM
27	B11	PWM
28	B12	DI, DO
29	B13	DI, DO
30	B14	DI, DO

31	B15	DI, DO
32	C0	AI CONT
33	C1	AI CONT
34	C2	AI CONT
35	C3	AI CONT
36	C4	AI DC, DI, DO
37	C5	AI DC, DI, DO
38	C6	PWMDI, DI, DO
39	C7	PWM, DI, DO
40	C8	PWM, DI, DO
41	C9	PWM, DI, DO
42	C10	DI, DO
43	C11	DI, DO
44	C12	DI, DO
45	C13	DI, DO
46	C14	DI, DO
47	C15	DI, DO
48	D0	DI, DO
49	D1	DI, DO
50	D2	DI, DO
51	D3	DI, DO
52	D4	DI, DO
53	D5	DI, DO
54	D6	DI, DO
55	D7	DI, DO
56	D8	DI, DO
57	D9	DI, DO
58	D10	DI, DO
59	D11	DI, DO
60	D12	PWM, DI, DO
61	D13	PWM, DI, DO
62	D14	PWM, DI, DO
63	D15	PWM, DI, DO
64	E0	DI, DO
65	E1	DI, DO
66	E2	DI, DO
67	E3	DI, DO
68	E4	DI, DO
69	E5	PWM, DI, DO
70	E6	PWM, DI, DO
71	E7	DI, DO

Seznam tabulek

72	E8	DI, DO
73	E9	DI, DO
74	E10	PWM, DI, DO
75	E11	PWM, DI, DO
76	E12	DI, DO
77	E13	DI, DO
78	E14	PWM, DI, DO
79	E15	PWM, DI, DO
80	F0	DI, DO
81	F1	DI, DO
82	F2	DI, DO
83	F3	AI DC, DI, DO
84	F4	AI DC, DI, DO
85	F5	AI DC, DI, DO
86	F6	AI DC, DI, DO
87	F7	AI DC, DI, DO
88	F8	AI DC, DI, DO
89	F9	AI DC, DI, DO
90	F10	AI DC, DI, DO
91	F11	DI, DO
92	F12	DI, DO
93	F13	DI, DO
94	F14	DI, DO
95	F15	DI, DO
96	G0	DI, DO
97	G1	DI, DO
98	G2	DI, DO
99	G3	DI, DO
100	G4	DI, DO
101	G5	DI, DO
102	G6	DI, DO
103	G7	DI, DO
104	G8	DI, DO
105	G9	No FNC (UART RX)
106	G10	DI, DO
107	G11	DI, DO
108	G12	No FNC (UART TX)
109	G13	No FNC (UART RTS)
110	G14	No FNC (UART CTS)
111	G15	DI, DO
112	H0	No FNC (Krystal)

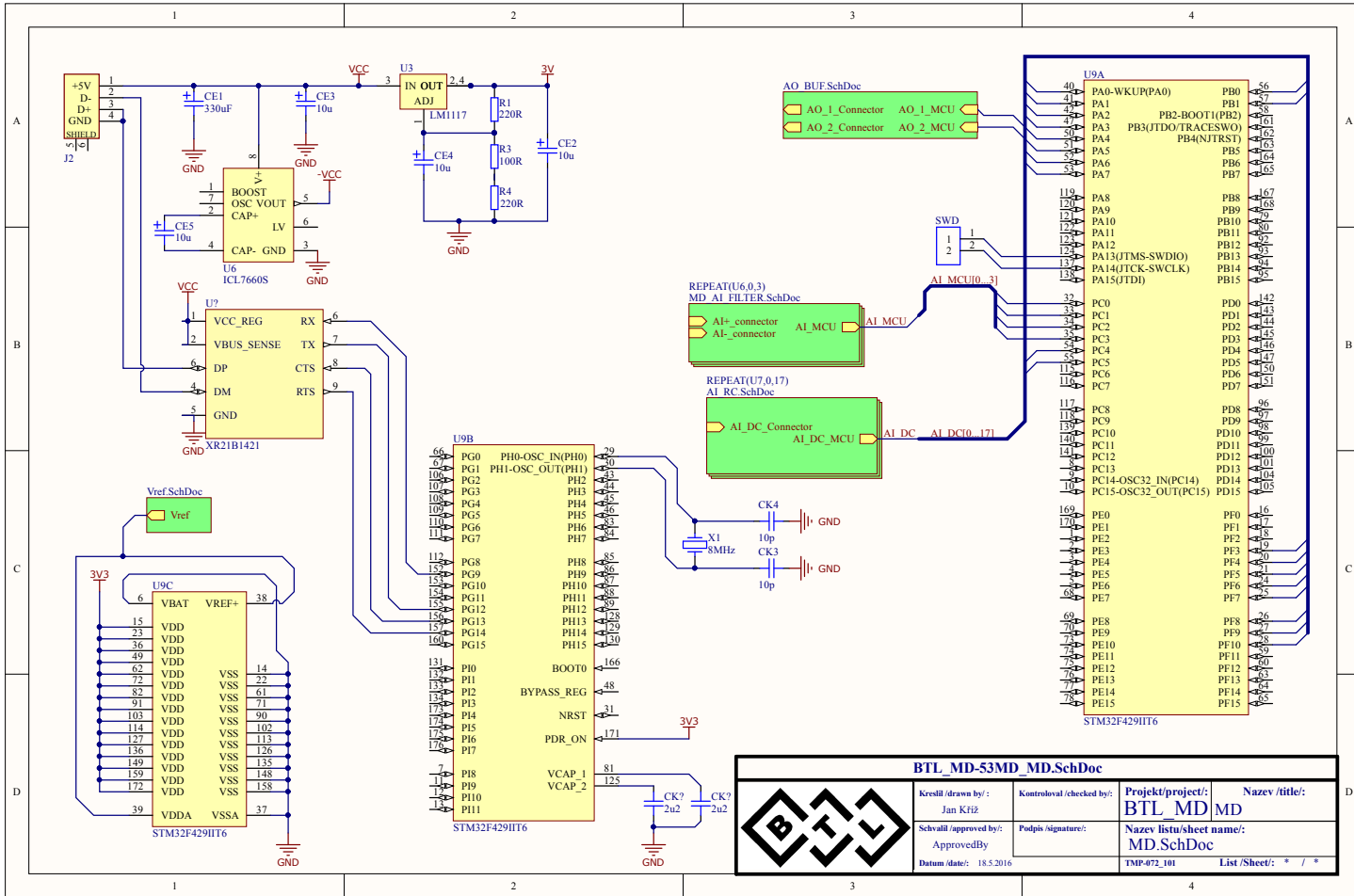
113	H1	No FNC (Krystal)
114	H2	DI, DO
115	H3	DI, DO
116	H4	DI, DO
117	H5	DI, DO
118	H6	PWM, DI, DO
119	H7	DI, DO
120	H8	DI, DO
121	H9	PWM, DI, DO
122	H10	PWM, DI, DO
123	H11	PWM, DI, DO
124	H12	PWM, DI, DO
125	H13	DI, DO
126	H14	DI, DO
127	H15	DI, DO
128	I0	PWM, DI, DO
129	I1	DI, DO
130	I2	DI, DO
131	I3	DI, DO
132	I4	DI, DO
133	I5	DI, DO
134	I6	DI, DO
135	I7	DI, DO
136	I8	DI, DO
137	I9	DI, DO
139	I10	DI, DO


Příloha D

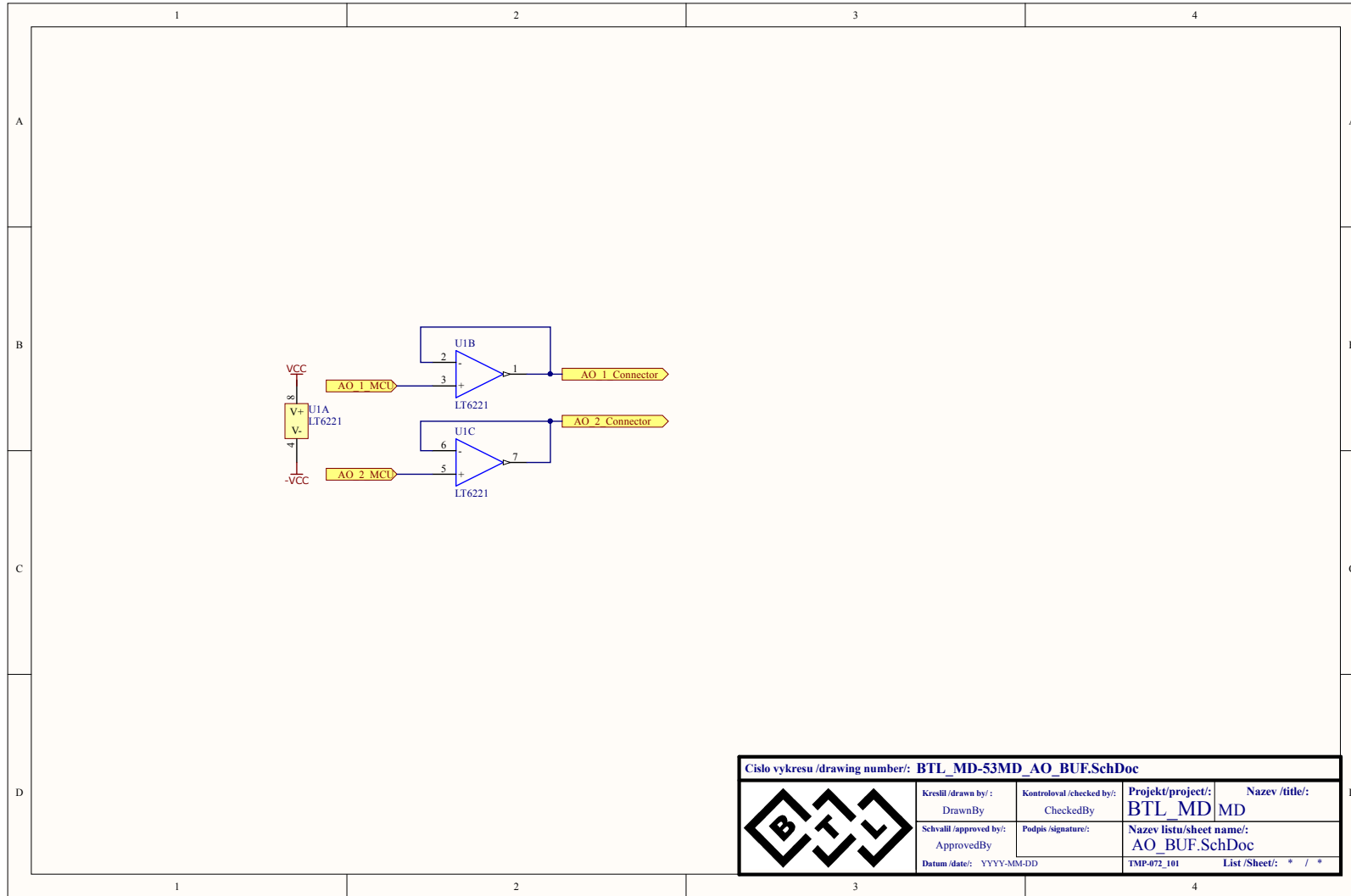
Definované chybové kódy MD


Kód chyby	Text chyby
#5000	Unknown error
#5001	MD parsing error
#5002	No ADC available!
#5003	This function is not initied for this pin!
#5004	Unknown command!
#5005	Empty cmd!
#5006	No pin selected!
#5007	This pin does not have ADC!
#5008	Pin is used by another function!
#5009	Pin is no available!
#5100	Timeout pro odpověď MD vypršel
#5101	Neznámá odpověď

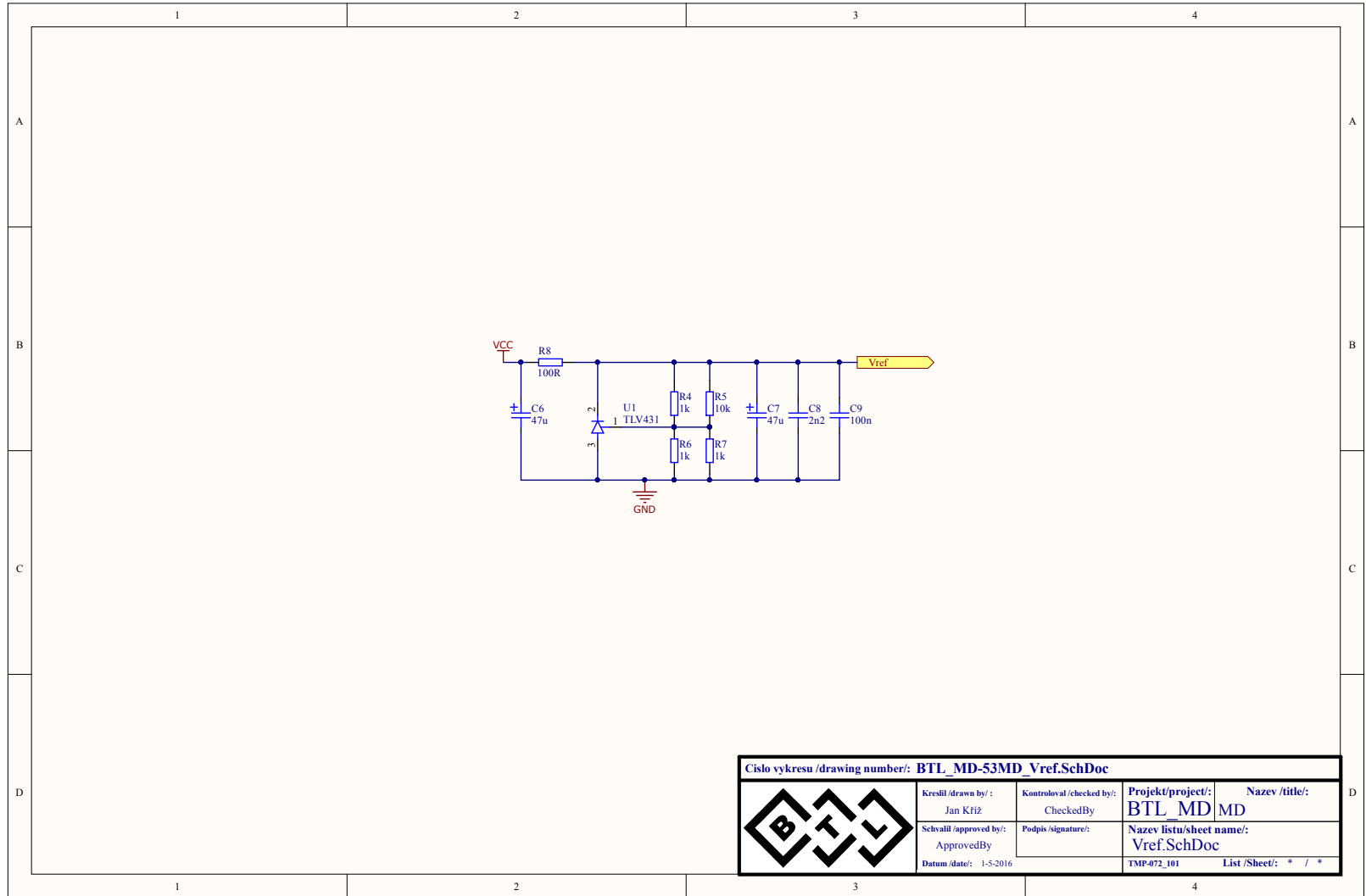
Příloha E - Kompletní obvodové schéma MD

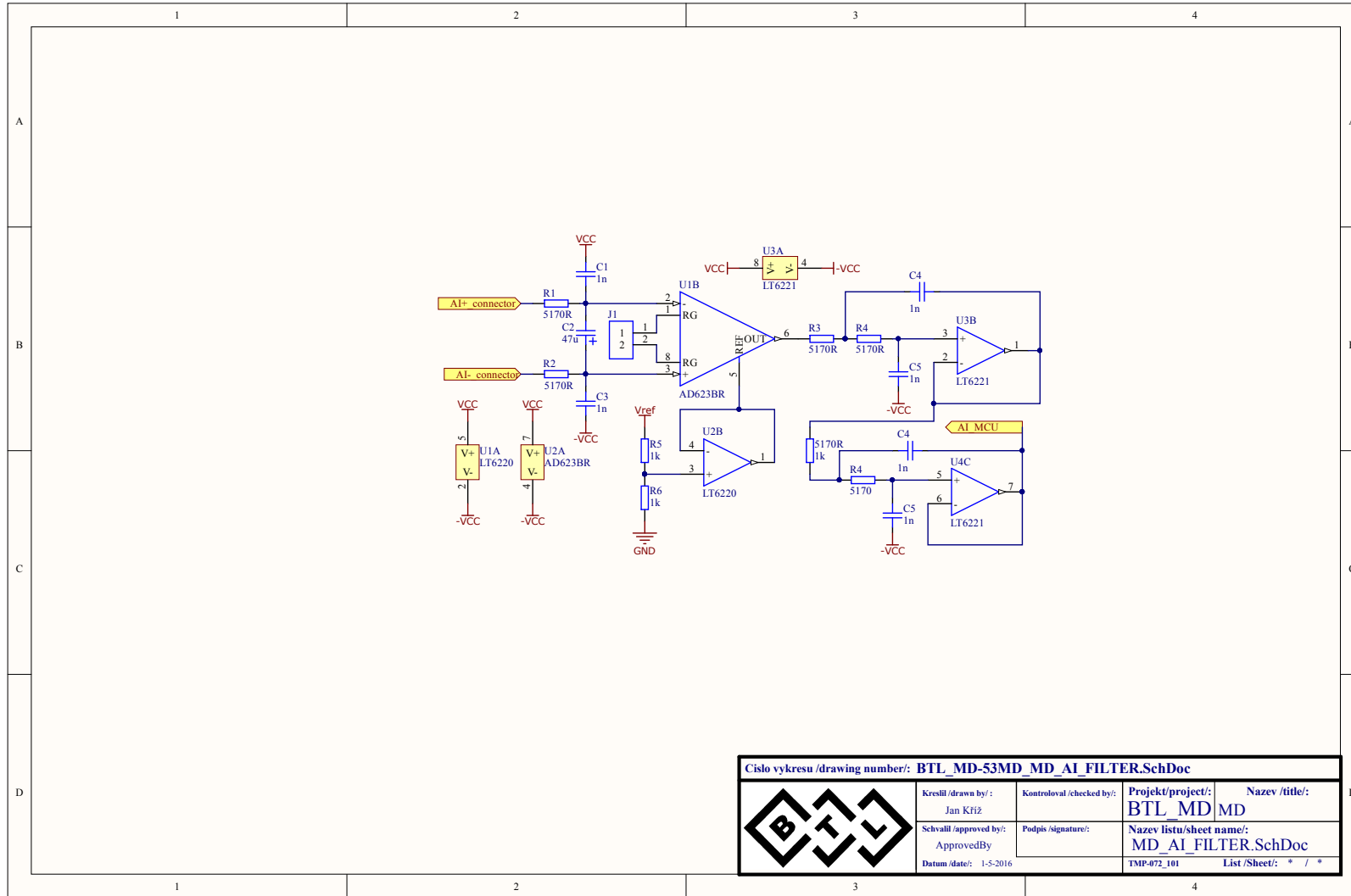



BTL_MD-53MD_MD.SchDoc			
	Kreslil / drawn by: Jan Kříž	Kontroloval / checked by:	Projekt/project: BTL_MD MD
	Schválil / approved by: ApprovedBy	Polpisil / signature:	Název listu/sheet name: MD.SchDoc
	Datum / date: 18.5.2016		List /Sheet: * / *

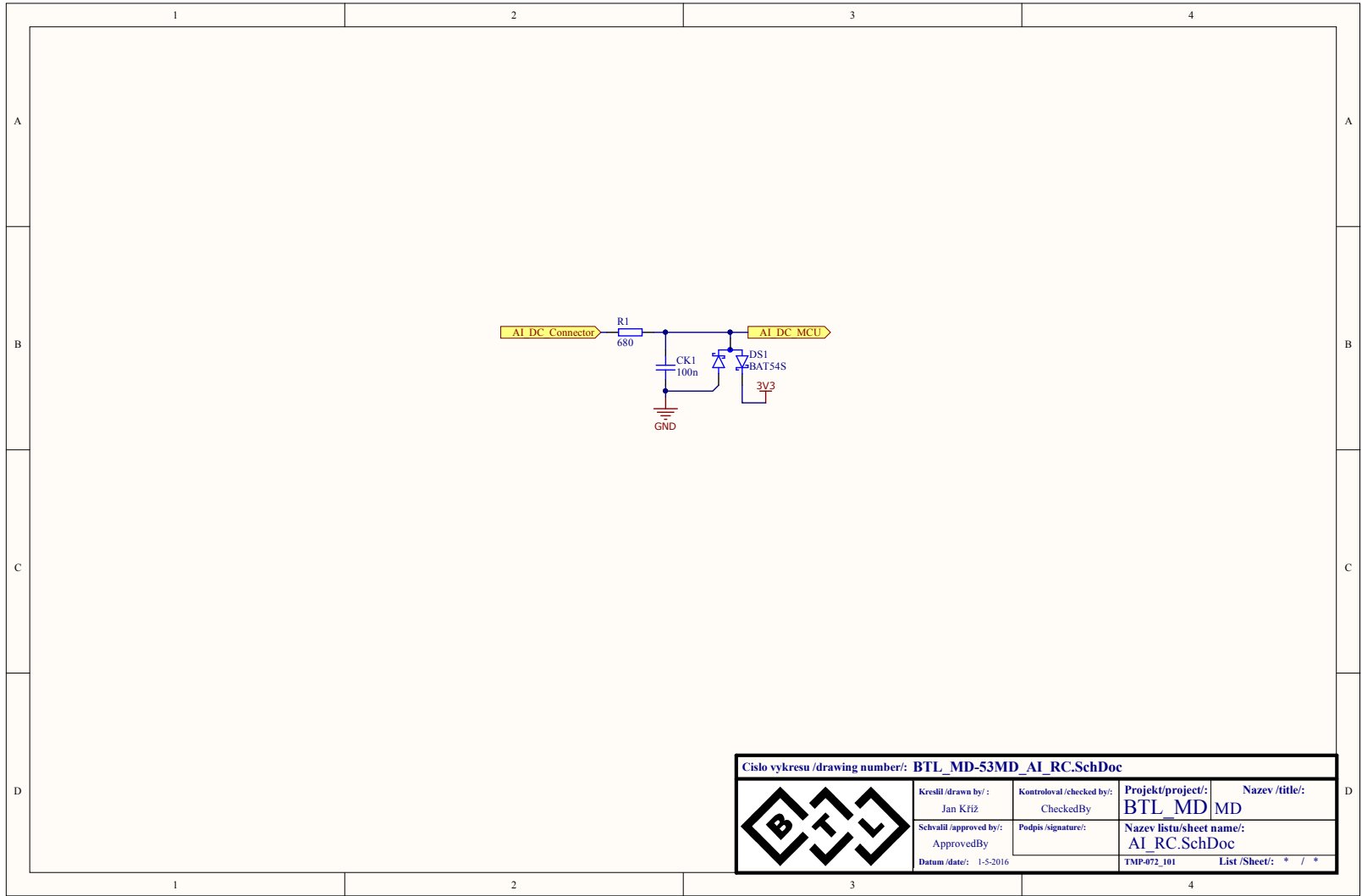



Cislo vykresu /drawing number/: BTL_MD-53MD_AO_BUF.SchDoc			
	Kresil /drawn by/: DrawnBy	Kontroloval /checked by/: CheckedBy	Projekt/project/: BTL_MD MD
	Schválil /approved by/: ApprovedBy	Podpis /signature/:	Nazev listu/sheet name/: AO_BUF.SchDoc
	Datum /date/: YYYY-MM-DD	TMP-072_101	List /Sheet/: * / *





Cislo vykresu /drawing number/: BTL_MD-53MD_MD_AI_FILTER.SchDoc				
	Kresil /drawn by/: Jan Kříž	Kontroloval /checked by/:	Projekt/project/: BTL_MD MD	
	Schválil /approved by/: ApprovedBy	Podpis /signature/:	Název listu/sheet name/: MD_AI_FILTER.SchDoc	
	Datum /date/: 1-5-2016		TMP-072_101	List /Sheet/: * / *



Cislo vykresu /drawing number/: BTL_MD-53MD_AI_RC.SchDoc			
	Kresil /drawn by/: Jan Kříž	Kontroloval /checked by/: CheckedBy	Projekt/project/: BTL_MD MD
	Schválil /approved by/: ApprovedBy	Podpis /signature/:	Nazev listu/sheet name/: AI_RC.SchDoc
	Datum /date/: 1-5-2016		TMP-072_101