

České vysoké učení technické v Praze
Fakulta elektrotechnická



Bakalářská práce

**Diagnostické zařízení pro poloautomatické zkoušení
elektronických jednotek**

Košarko Martin

Vedoucí práce: Ing. Michal Sojka

Studijní program: Elektrotechnika a informatika strukturovaný bakalářský

Obor: Kybernetika a měření

září 2008

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra řídicí techniky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Martin Košarko**

Studijní program: Elektrotechnika a informatika (bakalářský), strukturovaný
Obor: Kybernetika a měření

Název tématu: **Diagnostické zařízení pro poloautomatické testování elektronických jednotek**

Pokyny pro vypracování:


1. Navrhněte a zrealizujte aplikaci pro podporu diagnostiky elektronických jednotek pomocí karty I/O NI USB-6008 připojené k PC.
2. Navrhněte řešení (SW, HW) a konzultujte jej se zadavatelem.
3. Řešení implementujte a otestujte.
4. Vytvořte dokumentaci navrženého řešení.

Seznam odborné literatury:

Dodá vedoucí práce

Vedoucí: Ing. Michal Sojka

Platnost zadání: do konce zimního semestru 2008/2009


prof. Ing. Michael Šebek, DrSc.
vedoucí katedry




doc. Ing. Boris Šimák, CSc.
děkan

V Praze dne 29. 2. 2008

Poděkování

Na tomto místě bych rád poděkoval vedoucímu bakalářské práce Ing. Michalu Sojkovi zejména za trpělivost, kterou prokázal, když se mi věnoval. Dále bych chtěl poděkovat své rodině a kolegům, kteří se podíleli na vytvoření příjemného prostředí, v němž jsem pracoval.

Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Praze, dne 11.7.2008


.....
podpis

Abstrakt

Tato bakalářská práce se zabývá možnostmi využití měřicí karty NI USB-6008 pro použití poloautomatického měření zejména stejnosměrných veličin a ověřování parametrů elektronických jednotek v trakčním regulačním provozu. Práce obsahuje stručný popis měřicího modulu, popis vytvořené knihovny funkcí zajišťujících sběr a zpracování dat a popis aplikace využívající tuto knihovnu. Knihovna a aplikace jsou vyvinuty v prostředí LabWindows/CVI.

Klíčová slova: NI USB-6008, CAE, měření, poloautomatické

Summary

This bachelor work deals with possibilities of measurement device NI USB-6008 as semi-automatic testing device for measuring direct signals and verification of function parameters on electronic units in traction regulation systems. This work contains description of measuring module, description of created library and description of application, which is using this library. Library and application are written in LabWindows/CVI.

Keywords: NI USB-6008, CAE, measurement, semi-automatic

Obsah

1 Úvod	2
2 Analýza zadání	3
2.1 Cíl.....	3
2.2 Volba zkoušené jednotky	3
2.3 Měřicí pracoviště	4
3 Popis měřicího modulu NI USB-6008	5
3.1 Obecně.....	5
3.2 Popis součástí modulu	5
3.3 Popis svorkovnice.....	6
3.4 Blokové schéma modulu	8
3.5 Technické údaje	8
3.5.1 Analogový vstup.....	8
3.5.2 Analogový výstup	9
3.5.3 Digitální linky.....	10
3.5.4 Čítač	10
3.6 Ověření přesnosti	11
4 Základní popis prostředí LabWindows/CVI	12
4.1 Obecně.....	12
4.2 Tvorba uživatelského rozhraní.....	12
4.3 Sběr dat	13
5 Specifikace datových souborů	17
5.1 Obecně.....	17
5.2 Specifikace souboru config.xml	17
5.3 Specifikace výstupních souborů	18
5.4 Specifikace souborů „Jednotka“ .xml	18
6 Popis knihovny myLib	22
6.1 Obecně.....	22
6.2 Popis částí	22
6.2.1 Popis definic	22
6.2.2 Popis globálních proměnných	22
6.2.3 Popis funkcí	22
7 Ověření na zvolené jednotce	28
7.1 Měřicí přípravek	28
7.2 Vlastní pracoviště	28
8 Závěr	30
9 Literatura	31
Přílohy	32
Schéma měřicího přípravku	32
Ukázka výstupního souboru.....	33
Ukázka konfiguračního souboru pro měření	34

1 Úvod

Tato bakalářská práce se zabývá možnostmi využití měřicí karty NI USB-6008 pro použití poloautomatického měření zejména stejnosměrných veličin a ověřováním parametrů elektronických jednotek v trakčním regulačním provozu.

Elektronická zařízení využíváme každodenně. Před expedicí každého zařízení je nutno provést kontrolu technických parametrů. Nedílnou součástí těchto kontrol je ověřování funkčních charakteristik jednotlivých modulů, z nichž jsou finální zařízení kompletována. Mnohé z těchto zkoušek jsou ovšem při provádění klasickými metodami (např. přímá kontrola v měřicím bodě voltmetrem) časově velmi náročné. Důsledkem této náročnosti často vznikají požadavky na nalezení náhradních řešení, která by zachovala (nebo v některých případech i předčila) kvalitu operací prováděných zkušebním technikem. Dalšími požadavky, které jsou posléze kladeny na tato řešení jsou zejména zrychlení a zjednodušení prováděné činnosti a její větší bezpečnost.

Jednou z možností, jak přistoupit ke zkvalitnění kusových zkoušek, je tvorba nejrůznějších mechanických přípravků, obsahujících zejména rozličné přepínače, propojovací konektory, komparátory a signalizaci pomocí LED, případně akustické signalizace.

Další možností je využití počítače pro alespoň poloautomatická, ještě lépe však plně automatická měření, která po odladění v konkrétní aplikaci plně eliminují chybovost lidského faktoru.

V kapitole 2 je uveden rozbor zadání. Následně se seznámíme s měřicím modulem NI USB-6008 (kapitola 3) a vývojovým prostředím LabWindows/CVI, zejména s kroky potřebnými pro tvorbu uživatelského rozhraní a sběr dat (kapitola 4). Jádrem práce je obsaženo v kapitolách 5, kde jsou specifikovány formáty souborů s nimiž se bude pracovat, v kapitole 6, kde je popsána vytvořená knihovna obsahující funkce a moduly pro práci se soubory a realizující jednotlivá měření a kapitole 7, kde je ověřena funkčnost. V závěrečné kapitole jsou zhodnoceny výsledky a plánované kroky do budoucna.

2 Analýza zadání

2.1 Cíl

Cílem tohoto projektu je vytvoření měřicího pracoviště pro poloautomatické měření elektronických jednotek. Jednotlivé kroky projektu by měly být následující:

- Návrh SW
 - Moduly realizující jednotlivé měřicí funkce zejména
 - Měření napětí
 - Měření proudu
 - Měření frekvence
 - Ovládání digitálních kanálů
 - Moduly pro práci s vstupními a výstupními soubory
 - Vyspecifikovat formáty vstupních a výstupních souborů (viz sekce 5 Specifikace datových souborů)
 - Vytvořit jednoduchou aplikaci realizující měření (viz sekce 7 Ověření na zvolené jednotce)
- Návrh HW
 - Specifikace konkrétní elektronické jednotky, pro kterou bude navržen HW interface (viz kapitola **Chyba! Nenalezen zdroj odkazů. Chyba! Nenalezen zdroj odkazů.**)
 - Sestavení vlastního měřicího pracoviště

2.2 Volba zkoušené jednotky

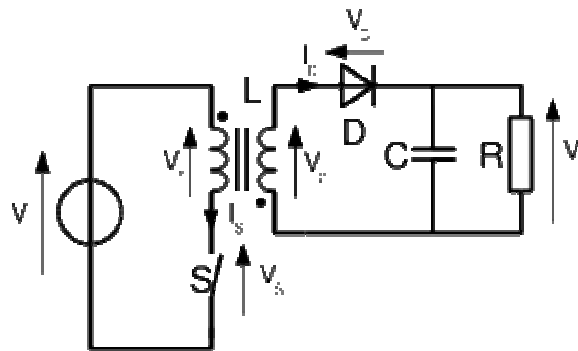
Ověření splnění cíle (funkčnost aplikace a HW rozhraní) proběhne na zvoleném typu jednotky. Podkladem pro volbu jednotky byla analýza zkušebních předpisů a tabulek požadovaných hodnot, což jsou dva dokumenty příslušné kusovým zkouškám ke každé jednotce.

- Zkušební předpis je souhrnný postup, který udává pořadí zkoušek, označuje měřicí body a případně odkazuje na další dokumenty, které se mohou týkat kusové zkoušky jednotky či zařízení (například postupy pro aktualizaci software na dané jednotce).
- Tabulka požadovaných hodnot udává nominální hodnoty veličin a povolené tolerance, které musí naměřené hodnoty splňovat, aby jednotka mohla být označena za shodnou a tím vyhověla kusové zkoušce.

Z analýzy vyplynulo, že jednotkou, na které se vyzkouší možnosti poloautomatického zkoušení s měřicím modulem NI USB-6008 bude typ TPG. Jedná se o spínaný zdroj typu Flyback (viz Obr. 2.1 Principiální schéma obvodu Flyback), se zpětnou vazbou, pracující na frekvenci 105kHz. Tato jednotka dále obsahuje ochrany proti přetížení a zkratu. Detailnější popis zařízení přesahuje rámec tohoto textu a pro tuto práci není podstatný.

Hlavní důvody pro volbu této jednotky byly:

- Velká náročnost zkoušení této jednotky klasickými metodami, která plyne z potřeby častého přepojování vodičů, přístrojů a externích rezistorů pro jednotlivé body zkoušek.
- V provozu se vyskytuje šest druhů těchto jednotek, lišících se vzájemně svými výstupními parametry (viz Tab. 2.1 Výstupní parametry zkoušených jednotek). Vzhledem k tomu, že desky plošných spojů a měřicí body jsou ve všech případech stejné, je možné navrhnout jeden univerzální hardwarový interface.
- Velké měšiční série, v nichž je potřeba provádět kusové zkoušky těchto jednotek.



Obr. 2.1 Principiální schéma obvodu Flyback

	TPG-1	TPG-2	TPG-3	TPG-4	TPG-5	TPG-6
Výstupní napětí [V]	5.2	12	14.8	18	24	21
Max.výstupní proud[A]	2	0.8	0.65	0.55	0.4	0.45

Tab. 2.1 Výstupní parametry zkoušených jednotek

2.3 Měřicí pracoviště

Měřicí pracoviště bude umístěno v klimatizované místnosti se stálou teplotou 22°C. U některých zkoušek nelze měřicí body vyvést na zkušební přípravek, tyto pak na zkoušené jednotce budou kontaktovány jehlovým polem.

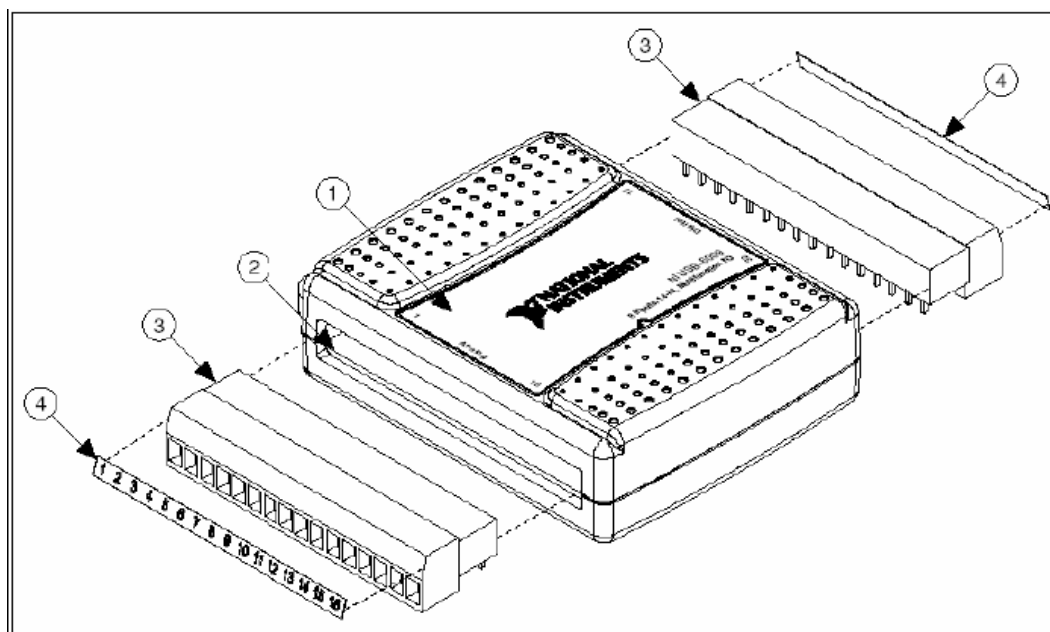
3 Popis měřicího modulu NI USB-6008

3.1 Obecně

Měřicí modul NI USB-6008 je zařízení poskytující jednoduchou funkcionalitu pro aplikace jako je zaznamenávání dat, jednoduchá snadno přenosná měření, experimenty, řízení nenáročných technologických procesů nebo zkoušení. Vzhledem ke své ceně (v současnosti přibližně 3000 Kč) je snadno dostupný i pro studenty, ale umožňuje použití v sofistikovanějších měřicích aplikacích. S modulem je dodáván software VI-Logger, s jehož použitím je možné, díky přehlednému průvodci, nastavit parametry měření, a tak mít první aplikace pro sběr dat a ukládání dat během několika málo minut. Dále jsou k dispozici ovladače NI-DAQmx (v současnosti ve verzi 8.0), které následně umožňují komunikaci s vyššími programovacími jazyky, jakými jsou například LabView, LabWindows/CVI, Measurement Studio for Visual Studio.NET nebo MatLab.

Ze strany firmy National Instruments je k dispozici velká podpora v manuálu k jednotce [1], popřípadě na internetových stránkách, kde je udržována obsáhlá KnowledgeBase [2], stejně jako velké množství příkladů, popřípadě řízené kvalitně vedené diskusní fórum.

3.2 Popis součástí modulu



Obr. 3.1 Části modulu NI USB-6008

- Originální štítek výrobce (1)
- Zásuvky pro svorkovnice (2)
- Šroubovací svorkovnice (3)

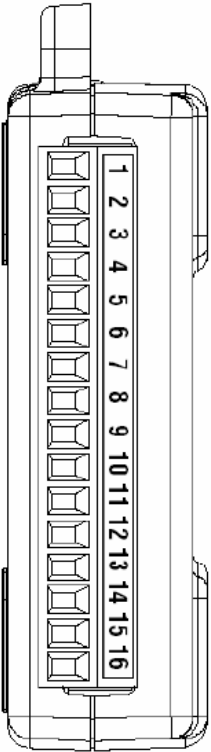
- Štítky (4)

Fyzické rozměry modulu s připojenými konektory

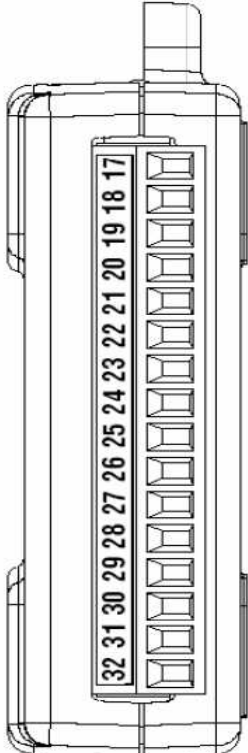
(v x š x h) 2,31cm x 8,18cm x 8,51cm

3.3 Popis svorkovnice

Pro přehlednost je zde uveden popis svorkovnice a význam signálů v jednotlivých zapojeních, v dalším textu je odkazováno na význam signálu a nikoliv na číslo svorky. Svorkovnice umožňuje k měřicímu modulu připojit vodiče do průměru až 1,29 mm.

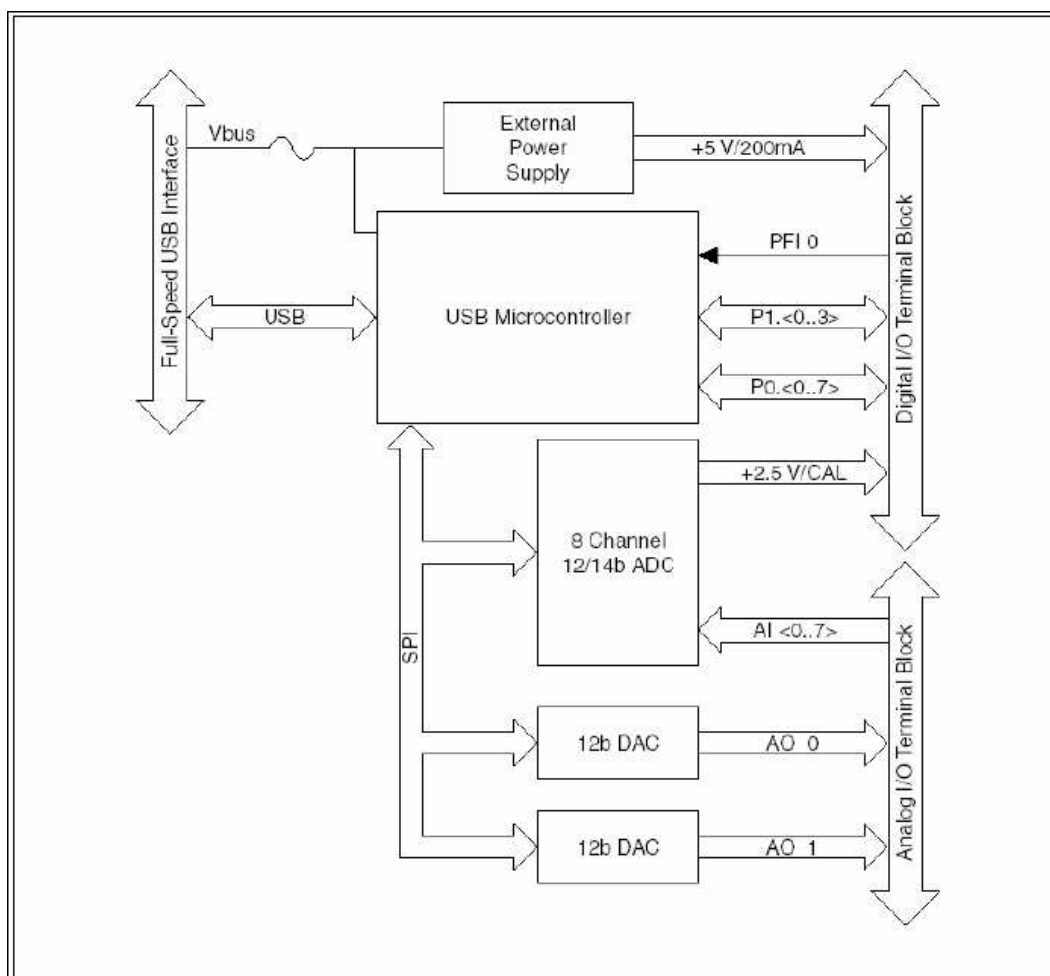
MODUL	SVORKA	SIGNÁL	SIGNÁL
		ZAPOJENÍ NA SPOL. ZEM	DIFERENCIÁLNÍ ZAP.
	1	GND	GND
	2	AI 0	AI 0+
	3	AI 4	AI 0-
	4	GND	GND
	5	AI 1	AI 1+
	6	AI 5	AI 1-
	7	GND	GND
	8	AI 2	AI 2+
	9	AI 6	AI 2-
	10	GND	GND
	11	AI 3	AI 3+
	12	AI 7	AI 3-
	13	GND	GND
	14	AO 0	AO 0
	15	AO 1	AO 1
	16	GND	GND

Obr. 3.2 Popis Analogových signálů zařízení

MODUL	SVORKA	SIGNÁL
	17	P0.0
	18	P0.1
	19	P0.2
	20	P0.3
	21	P0.4
	22	P0.5
	23	P0.6
	24	P0.7
	25	P1.0
	26	P1.1
	27	P1.2
	28	P1.3
	29	PFI 0
	30	+2.5 V
	31	+5 V
	32	GND

Obr. 3.3 Popis Digitálních signálů zařízení

3.4 Blokové schéma modulu



Obr. 3.4 Blokové schéma modulu

Uvedené schéma je totožné pro moduly NI USB-6008 a NI USB-6009. Odlišnost těchto dvou modulů spočívá v bloku 8 Channel 12/14b ADC, kdy v zařízení NI USB-6008 je dodáván 12bitový převodník.

3.5 Technické údaje

3.5.1 Analogový vstup

- **Počet vstupů** (přepínané softwarově)
 - 8 při zapojení na společnou zem
 - 4 při diferenciálním zapojení
- **Typ převodníku**
 - S postupnou aproximací
- **Rozlišení převodníku**
 - 12 bitů
 - 11 bitů

při diferenciálním zapojení
při zapojení na společnou zem

- Max. vzorkovací frekvence 10 kSa/s
- **Vstupní rozsahy**
 - ± 10 V při zapojení na společnou zem
 - až ± 20 V při diferenciálním zapojení
 - **min ± 1 V** při diferenciálním zapojení
- Přepětová ochrana ± 35 V
- Vstupní impedance 144 k Ω

Firma National Instruments garantuje následující přesnosti měření při různých nastaveních maximálního rozsahu, doporučený pracovní rozsah teplot je (0 – 55°C).

Rozsah	Obvykle při 25 °C [mV]	Maximálně při jiné teplotě [mV]
± 20 V	14,7	138
± 10 V	7,73	84,8
± 5 V	4,28	58,4
± 4 V	3,59	53,1
$\pm 2,5$ V	2,56	45,1
± 2 V	2,21	42,5
$\pm 1,25$ V	1,7	38,9
± 1 V	1,53	37,5

Tab. 3.1 Absolutní přesnost v závislosti na rozsahu při diferenciálním zapojení

3.5.2 Analogový výstup

- Počet výstupů 2
- Typ převodníku
 - S postupnou aproximací
- Rozlišení 12 bitů
- Výstupní rozsah 0 – 5V
- Výstupní impedance 50 Ω
- Výstupní proud 5mA
- Zkratový proud 50mA
- Max. obnovovací frekvence 150 Hz, softwarové časování
- Doba přeběhu 1V/ μ s

3.5.3 Digitální linky

- Počet digitálních linek 12
- Nastavení linek
 - Každá linka je programově nastavitelná jako vstup či výstup
- Typ výstupu Open collector
- Výstupní logika TTL, LVTTTL, CMOS
- Pull-up rezistor 4.7 k Ω na 5 V

Rozsah	Min	Max	Jednotka
Vstupní napětí nízká úroveň	-0,3	0,8	V
Vstupní napětí vysoká úroveň	2	5,8	V
Vstupní proud	-	50	mA
Výstupní napětí nízká úroveň (I=8,5mA)	-	0,8	V
Výstupní napětí vysoká úroveň:			
Active drive (push-pull), I=-8,5mA	2	3,5	V
Open collector (open-drain), I=- 0,6mA, nominální	2	5	V
Open collector (open-drain), I=- 8,5mA, s externím pull-up rezistorem	2	-	V

Tab. 3.2 Specifikace logických úrovní

Po zapojení modulu jsou všechny digitální linky nakonfigurovány jako logické vstupy. Po přeprogramování digitální linky na výstup je v úrovni logická 1. Na tento jev je potřeba brát ohled při tvorbě hardwarových rozhraní!

3.5.4 Čítač

- Počet čítačů 1
- Rozlišení 32 bitů
- Funkce čítače Čítání spádových hran
- Pull-up rezistor 4.7 k Ω na 5 V
- Maximální vstupní frekvence 5 MHz
- Minimální šířka pulzu vysoké úrovně 100 ns

- Minimální šířka pulzu nízké úrovně 100 ns
- Vstupní napětí vysoké úrovně 2 V
- Vstupní napětí nízké úrovně 0.8 V

3.6 Ověření přesnosti

Přestože výrobce garantuje přesnost měřicích kanálů, bylo nutné vzhledem ke stáří modulu ověřit, zda je možno tyto hodnoty považovat za relevantní i po uplynutí kalibrační lhůty, či zda bude nutné modul nechat recalibrovat. Měření se ověřily vlastnosti diferenciálního zapojení pro všechny 4 kanály. Výsledky shrnuje následující tabulka

Informativní zdroj	Napětí [V]								
	1	2	3	4	5	10	12	15	20
Kalibrovaný přístroj	1,006	2,001	2,996	4,00	5,00	10,00	12,00	15,00	19,97
NI-USB	1,002	1,998	2,983	3,995	4,996	9,993	11,991	14,988	19,983

Tab. 3.3 Ověření přesnosti měřicího modulu

Jako kalibrovaný měřicí přístroj byl použit digitální multimetr Summit 45 s automatickým přepínáním měřicích rozsahů, první tři měření proběhla na rozsahu 4V, další pak na rozsahu 40V.

Z naměřených hodnot je vidět, že přesnost měřicího modulu nebyla ovlivněna a že je dostatečná pro vývojové použití v naší aplikaci. Je však samozřejmé, že před finálním nasazením do procesu zkoušení bude modul kalibrován.

4 Základní popis prostředí LabWindows/CVI

4.1 Obecně

LabWindows/CVI je vývojové prostředí pro podporu programování v jazyce C. Poskytuje velké množství knihovných funkcí a srozumitelných nástrojů pro tvorbu aplikací určených pro sběr dat (přes rozhraní USB, RS-232, VXI či GPIB), jejich analýzu, popřípadě následnou vizualizaci.

V prostředí LabWindows/CVI je možné editovat, kompilovat či linkovat jakýkoliv ANSI C program. Dále je možné používat kompilované moduly i dynamické knihovny při tvorbě vlastního programu. Pro tuto práci bylo použito prostředí LabWindows/CVI ve verzi 8.0.1.

Typická tvorba programu v LabWindows/CVI zahrnuje následující kroky:

- Tvorba uživatelského rozhraní
- Sběr dat
- Analýza dat
- Řízení programu

4.2 Tvorba uživatelského rozhraní

Uživatelské rozhraní je nejnáze vytvořeno za pomoci interaktivního drag&drop editoru, který obsahuje řadu prvků od panelů, přes příkazová tlačítka, rolovací menu či grafy. Každému prvku lze sérií dialogových oken nastavit různé vlastnosti od velikosti a barev přes datové typy, které se zobrazují či čekají na vstup od uživatele.

Dvě nejdůležitější vlastnosti, které každému prvku přísluší a je potřeba nastavit, jsou :

- „jméno konstanty“, což je jméno proměnné, přes který je následně možno s daným prvkem pracovat, implicitní pojmenování se skládá z typu a pořadového čísla s nímž je prvek vložen
- „callback function“, což je jméno funkce, která se zavolá při každém výskytu události nad daným prvkem. Událostí může být například: kliknutí na tlačítko, aktualizace grafu, volba z menu, změna posuvníku, přepnutí tabulky, získání či ztráta focusu, uplynutí určitého času (pouze pro prvek typu Timer) atd.

Pokud mají všechny prvky přiřazeny jména a callback funkce, je možno nastavit, které události se budou zpracovávat a následně nechat automaticky vygenerovat kostru kódu.

Uložení uživatelského rozhraní vytvoří dva soubory:

- `gui.uir` obsahuje vlastní rozložení prvků
- `gui.h` obsahuje definice konstant a prototypy callback funkcí

4.3 Sběr dat

Aby bylo možno provést sběr dat, je potřeba mít nakonfigurován alespoň jeden Task (zde je dodržen původní název). Task je kolekce virtuálních kanálů s nastavením vzorkovací frekvence, počtu odebíraných vzorků, spouštěním, maximálních a minimálních hodnot, typu zapojení atd. Při tvorbě nového Tasku je vhodné mít připojené dané měřicí zařízení, poněvadž vývojové prostředí pak umožňuje jednoduchým nastavením v několika dialogových oknech vygenerovat celý kód obsluhující tvorbu nového měření automaticky (jednotlivé kroky průvodce pro nastavení měření napětí jsou zobrazeny na Obr. 4.1 Volba měřené veličiny, Obr. 4.2 Volba měřicího kanálu, Obr. 4.3 Nastavení měřicího kanálu). Tento průvodce při nastavování také kontroluje kompatibilitu jednotlivých vlastností s příslušným zařízením, v případě nesouladu s parametry měřicího modulu (např. pokus o nastavení čítání náběžných hran s modulem NI USB-6008, či nastavení maximální hodnoty na víc jak 20V) je uživatel upozorněn.

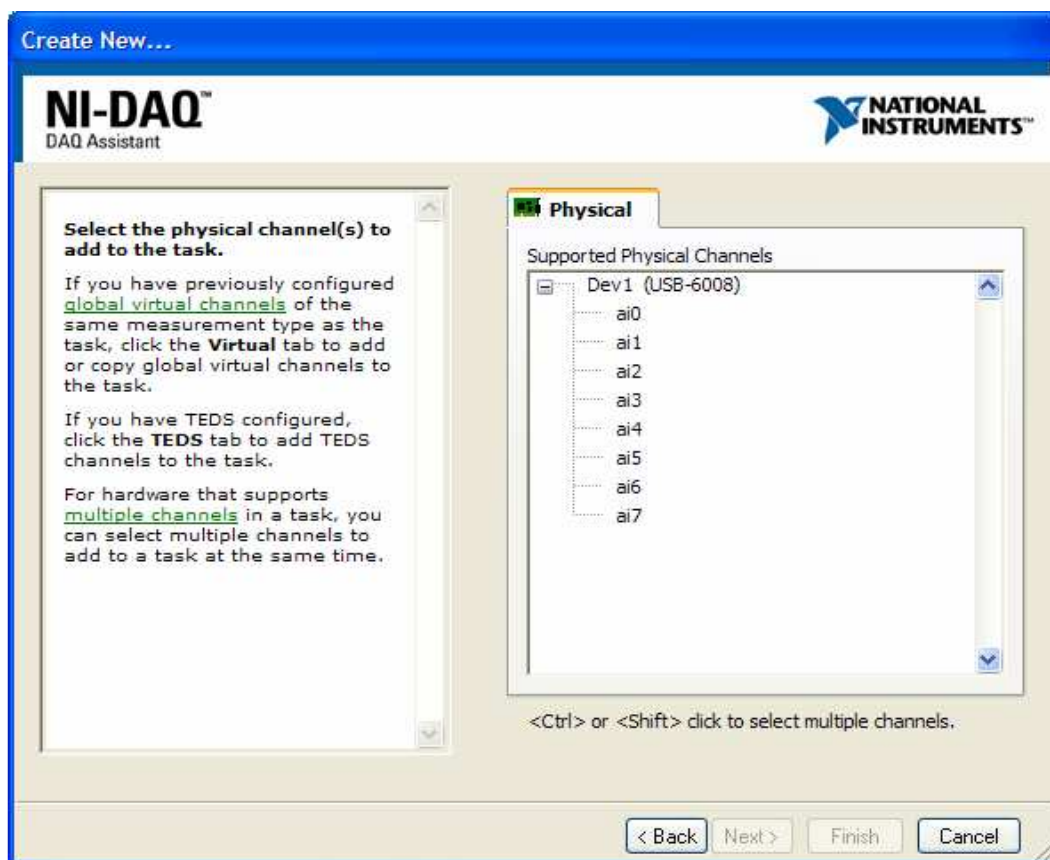
Potvrzením nastavení se do projektu přidají tři generované soubory.

- `TaskName.h` – hlavička funkce
- `TaskName.c` – vlastní kód funkce, obsluhující tvorbu nového kanálu a přiřazení nastavených hodnot
- `TaskName.mxb` – jehož otevřením ve vývojovém prostředí se načte průvodce nastavením příslušného Tasku.

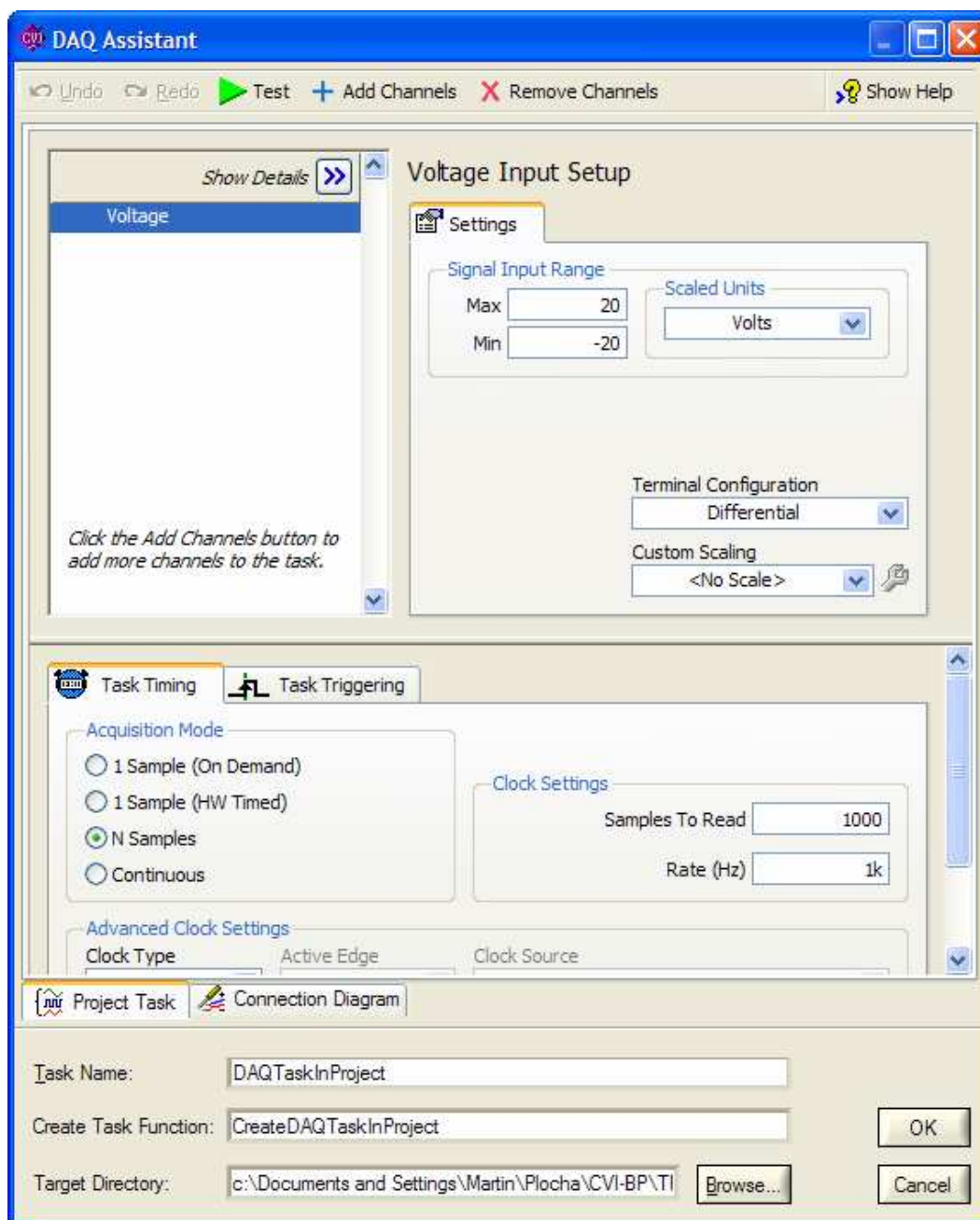
Ve vlastní aplikaci je pak potřeba použít příslušný hlavičkový soubor. Následným zavoláním obsažené `CreateTaskName` funkce je možné právě nastavený Task používat.



Obr. 4.1 Volba měřené veličiny



Obr. 4.2 Volba měřicího kanálu



Obr. 4.3 Nastavení měřicího kanálu

Takto vytvořené Tasky mohou obsluhovat najednou více kanálů, například současně vyčíst všechny analogové vstupy. Každý vstup může mít nastavené rozdílné maximální a minimální hodnoty měřených veličin, vlastnosti jako je spouštění, nebo počet odebraných vzorků z každého kanálu však zůstávají společné. S knihovnou NIDAQmx 8 a vyšší je možno vyčíst i různé typy veličin, pokud toto umožňuje měřicí modul.

Pokud chceme měřit na jednom vstupu různé napěťové hladiny narážíme pak na problém. Buď je při počátečním nastavení zvolen maximální rozsah, čímž

je posléze omezeno rozlišení a tím ovlivněna přesnost nebo se při předpokládané změně rozsahu vstupních hodnot musí přeprogramovat vlastnosti kanálu.

Vzhledem k tomu, že aplikace není časově kritická, je použita tvorba nového měřicího Tasku pro každé měření, místo kontroly a případného přeprogramování atributů před každým bodem. Automaticky vygenerované soubory tvorby proto byly upraveny, tak aby při tvorbě nového Tasku byly specifikovány jednotlivé parametry programově a nikoliv v návrhu při tvorbě programu.

5 Specifikace datových souborů

5.1 Obecně

Aplikace má pracovat se dvěma druhy vstupních konfiguračních souborů, jeden druh pro nastavení obecných informací (viz sekce 5.2 Specifikace souboru config.xml), druhý typ je pro konfiguraci vlastního měření (viz sekce 5.4 Specifikace souborů „Jednotka“.xml).

Vstupní konfigurační soubory jsou ve formátu xml příponou *.xml. Tento formát byl zvolen pro jednoduchou možnost editace bez nutnosti zásahu do zdrojového kódu a tudíž snadné přidávání či ubírání zkoušených jednotek, popřípadě přidání, ubrání či změn zkoušek, jednotlivých hodnot nebo zapojení do měřicího modulu v závislosti na změnách revize zkoušené jednotky. Dalším důvodem této volby je možnost přímé editace bez nutnosti dalšího software, který by pracoval se soubory binárními. Také vzhledem k existenci parseru v LabWindows/CVI se soubor xml snadněji kontroluje, zda obsahuje všechna požadovaná nastavení či hodnoty. Na druhou stranu právě vzhledem k tomu, že je soubor snadno upravitelný a obsahuje důležité informace, které mohou zásadně ovlivnit výsledky kusových zkoušek, bylo potřeba přijmout některá opatření proti neoprávněným změnám těchto souborů. Jako nejjednodušší a dostatečně spolehlivá se ukázala metoda systémového omezení vytvořením úzké administrátorské skupiny (vedoucí zkušebny a servisu, technolog výroby...), která má právo tyto soubory modifikovat, pro ostatní uživatele je přístup k těmto souborům nastaven na jen pro čtení.

Výstupní soubory jsou textové s příponou *.csv, respektive text oddělený středníky. Tento formát byl zvolen pro možnost převodu do formátu *.xls a následné eventuální zpracování výsledků.

5.2 Specifikace souboru config.xml

Soubor config.xml je uložen v kořenovém adresáři programu. Obsahuje základní informace pro inicializaci zkoušení. Soubor config.xml má následující Document Type Definition (dále DTD).

```
<!ELEMENT config (Operator+, Jednotka+)>
<!ELEMENT Operator (#PCDATA)>
<!ELEMENT Jednotka (Typ, Cesta)>
<!ELEMENT Typ (#PCDATA)>
<!ELEMENT Cesta (#PCDATA)>
```

- Operator – jméno zkoušejícího technika
- Typ – je jméno jednotky, které se zobrazí v programu při volbě
- Cesta je standardní cesta k souboru „Jednotka“.xml s informacemi o jednotce a zkouškách, které se na ní provádí, není-li implicitní soubor nalezen program se zeptá na jeho umístění.

5.3 Specifikace výstupních souborů

Výstupní soubory jsou ve formátu text oddělený středníky s příponou *.csv, každý řádek obsahuje naměřené hodnoty jedné kusové zkoušky příslušné jednotky. Formát řádku je následující:

```
číslo jednotky;  
revize jednotky;  
jméno zkoušejícího technika;  
datum a čas zahájení zkoušky;  
v následujících sloupcích jsou informace o naměřených  
hodnotách v průběhu kusové zkoušky jednotky v pořadí  
daném specifikací měřicího souboru;
```

Desetinná čísla jsou oddělována desetinnou čárkou a nikoliv tečkou, vzhledem k tomu, že MSExcel, který se nejpravděpodobněji bude používat pro případné zpracovávání výsledků, chápe čísla oddělaná tečkami jako datum a ne jako desetinné číslo.

5.4 Specifikace souborů „Jednotka“.xml

Soubor „Jednotka“.xml je unikátní pro každou jednotku. Předpokládané umístění je ve standardním adresáři výrobní dokumentace každé jednotky. Na základě analýzy zkušební předpisu a tabulky požadovaných hodnot obsahuje seznam kroků, které je po připojení na příslušný hardwarový interface nutné provést pro vyzkoušení dané elektronické jednotky. Obsahuje také nominální hodnoty a povolené tolerance jednotlivých bodů zkoušek. Soubor „jednotka“.xml má následující DTD.

```
<!ELEMENT Jednotka (Typ, Revize, Cislo, Log, Mereni?)>
<!ELEMENT Typ (#PCDATA)>
<!ELEMENT Revize (#PCDATA)>
<!ELEMENT Cislo (#PCDATA)>
<!ELEMENT Log (#PCDATA)>
<!ELEMENT Mereni (Voltage?, Current?, Switch?, Load?,
AnalogVystup?, Frequency?, RegNapeti?)>
<!ELEMENT Analogvystup (#PCDATA)>
<!ELEMENT Voltage (#PCDATA)>
<!ELEMENT Switch (#PCDATA)>
<!ELEMENT Load (#PCDATA)>
<!ELEMENT AnalogVystup (#PCDATA)>
<!ELEMENT Frequency (#PCDATA)>
<!ELEMENT Current (#PCDATA)>
<!ELEMENT RegNapeti (#PCDATA)>

<!ATTLIST Voltage
    zapojeni (RSE|Diff) "Diff"
    Kanal      (ai0|ai1|ai2|ai3|ai4|ai5|ai6|ai7)
#REQUIRED
    Nominalni CDATA #REQUIRED
    Tolerance CDATA #REQUIRED
    Resistor CDATA #IMPLIED>

<!ATTLIST Load
    zapojeni (RSE|Diff) "Diff"
    kanal      (ai0|ai1|ai2|ai3|ai4|ai5|ai6|ai7)
#REQUIRED
    Nominalni CDATA #REQUIRED
    Tolerance CDATA #REQUIRED
    Zesilovac CDATA #REQUIRED
    ovladani (ao0|ao1) #REQUIRED>

<!ATTLIST Current
    zapojeni (RSE|Diff) "Diff"
    Kanal      (ai0|ai1|ai2|ai3|ai4|ai5|ai6|ai7)
#REQUIRED
    Nominalni CDATA #REQUIRED
    Tolerance CDATA #REQUIRED
    Resistor CDATA #REQUIRED>

<!ATTLIST Analogvystup
    Kanal (ao0|ao1) #REQUIRED
    nominalni CDATA #REQUIRED>

<!ATTLIST Switch
    kanal (d0|d1|d2|d3|d4|d5|d6|d7) #REQUIRED
    nominalni CDATA #REQUIRED>
```



```

<!ATTLIST Frequency
    Nominalni CDATA #REQUIRED
    Tolerance CDATA #REQUIRED
    kanal (PF10|PF11) "PF10">

<!ATTLIST RegNapeti
    zapojeni (RSE|Diff) "Diff"
    kanal      (ai0|ai1|ai2|ai3|ai4|ai5|ai6|ai7)
#REQUIRED
    Nominalni CDATA #REQUIRED
    Tolerance CDATA #REQUIRED
    Zesilovac CDATA #REQUIRED
    ovladani (ao0|ao1) #REQUIRED>

```

- Typ - jméno jednotky
- Revize - aktuální číslo revize jednotky
- LOG - defaultní cesta k souboru, do něhož se ukládají výsledky kusových zkoušek
- Číslo - předpokládá se ve tvaru XXXX-YY
 - První čtyřčíslí je typ jednotky - určuje desku plošných spojů
 - Dvojčíslí je podtyp - určuje osazení součástek na desce a tím parametry jednotky
 - U většiny jednotek následuje troj- či čtyřčíslí, udávající pořadové číslo jednotky, vzhledem k tomu že je unikátní, tak se v konfiguračním souboru neuvádí. Jednotky bez pořadového čísla se nepoužívají samostatně, ale používají se jako moduly do dalších jednotek.
- Element mereni je začátek seznamu kroků měření. Jednotlivé kroky mohou být následující:
 - Voltage pro měření napětí
 - Current pro měření proudu
 - Load pro nastavení proudu zátěže
 - Frequency pro měření frekvence
 - Switch pro ovládání digitálních výstupů
 - RegNapeti pro regulaci napětí
- Atributy elementů jednotlivých kroků měření
 - Zapojeni (viz Obr. 3.2 Popis Analogových signálů zařízení)
 - Diff pro diferenciální zapojení
 - RSE pro zapojení se společnou zemí

- kanal měřicí kanál pro připojení vodičů
- nominal požadovanou správnou hodnotu
- tolerance toleranční pásmo v %
- resistor měřicí odpor pro měření proudů přímo a pro použití jako měření nastavované veličiny při regulaci
- zesilovac koncový stupeň hardwarového interface, pro přizpůsobení měřených hodnot větších než 20V, pokud není uveden počítá se s hodnotou 1.
- ovladani analogový výstup, kterým se nastavuje akční veličina

6 Popis knihovny myLib

6.1 Obecně

V této části se nachází dokumentace jednotlivých modulů vytvořených v rámci této bakalářské práce. Knihovna obsahuje moduly pro práci s vstupními a výstupními soubory a moduly pro jednotlivá měření. S použitím této knihovny byla následně vyvinuta příslušná aplikace realizující zkoušení. Při psaní této knihovny bylo hojně využíváno literatury [3],[4],[5].

6.2 Popis částí

6.2.1 Popis definic

- DATALEN Konstanta určující počet vzorků odebraných při měření
- BUFFLEN Konstanta určující délku pomocných řetězců
- FIRLEN Délka FIR filtru
- SAMPLING_FREQUENCY Vzorkovací frekvence
- LOWER_CUTOFF spodní hranice propustného pásma FIR filtru
- UPPER_CUTOFF horní hranice propustného pásma FIR filtru (0 pokud je použita dolní propust)
- INTERVAL doba po kterou se snaží nastavit požadovanou výstupní hodnotu
- MAX_ANALOG_OUTPUT maximální napětí analogových výstupů
- MEASURE_DELAY_TIME doba na ustálení přechodových jevu
- WINDUP_LIMIT limitace sumy odchylek
- TIMER_FREKVENCE doba po kterou měří frekvenci
- P_CONSTANT
- I_CONSTANT Konstanty regulátoru
- D_CONSTANT
- NOOP_DELAY_TIME doba čekání

6.2.2 Popis globálních proměnných

- TaskHandle digitaloutputs[8] obecně nastavené digitální výstupy
- TaskHandle analogoutputs[2] obecně nastavené analogové výstupy
- TaskHandle analoginputs[4] obecně nastavené analogové vstupy v diferenciálním zapojení
- int Lock mutex
- int nastaveno logická proměnná, nastavení regulace

6.2.3 Popis funkcí

6.2.3.1 Zápis na digitální výstup

- int zapisNaDigitalniVystup(int hodnota, int kam);
- parametry:

- hodnota – logická hodnota, která se má zapsat, nenulová hodnota parametru se zapíše jako logická 1
- kam – digitální linka, na kterou se má výstup zapsat (viz Obr. 3.3 Popis Digitálních signálů zařízení – P0.kam)
- vrací:
 - 0 pokud proběhne úspěšně
 - kladnou hodnotu, pokud dojde k varování
 - zápornou hodnotu, pokud dojde k chybě
 - číselné kódy návratových hodnot chyb a varování jsou definovány v knihovně NIDAQmx

6.2.3.2 Zápis na analogový výstup

- `int zapisNaAnalogovyVystup(float64 hodnota, int kam);`
- parametry:
 - hodnota – hodnota, která se zapíše na analogový výstup, hodnoty menší než 0 se zapíše jako 0, hodnoty větší než 5 se zapíše jako 5V
 - kam – analogový výstup, na který se má zapsat hodnota, nenulové číslo kanálu způsobí zápis na linku ao1 (viz Obr. 3.2 Popis Analogových signálů zařízení)
- vrací:
 - 0 pokud proběhne úspěšně
 - kladnou hodnotu, pokud dojde k varování
 - zápornou hodnotu, pokud dojde k chybě
 - číselné kódy návratových hodnot chyb a varování jsou definovány v knihovně NIDAQmx

6.2.3.3 Inicializace globálních proměnných

- `int vytvorKanaly(void);`
- tato funkce vytvoří nejobecněji nastavené měřicí kanály, pro analogové výstupy, analogové vstupy a digitální výstupy. Informace popisující tyto kanály jsou uloženy v globálních proměnných specifikovaných v souboru `myLib.h`
- vrací
 - 0 pokud proběhne úspěšně
 - kladnou hodnotu, pokud dojde k varování
 - zápornou hodnotu, pokud dojde k chybě
 - číselné kódy návratových hodnot chyb a varování jsou definovány v knihovně NIDAQmx

6.2.3.4 Zápis do souboru

- `int myWrite(char * buffer);`
- parametry
 - buffer – řetězec, který se má zapsat

- vrací
 - počet úspěšně zapsaných znaků
 - pokud dojde k chybě nastaví proměnnou errno na nenulovou hodnotu
- Funkce slouží k formátování dat zapisovaných výstupního měřicího souboru. Za ukládaný řetězec buffer přidá středník. Uložení řetězce se provede do souboru, který je specifikován v příslušném konfiguračním souboru zkoušené jednotky.

6.2.3.5 Měření napětí

- float64* zmerNapeti(char *connectionType, char *lineConnected, double requestedValue);
- parametry
 - connectionType – řetězec obsahující informaci o typu zapojení, „RSE“ pro zapojení se společnou zemí, „Diff“ pro diferenciální zapojení
 - lineConnected – řetězec obsahující informaci o připojeném kanálu (viz Obr. 3.2 Popis Analogových signálů zařízení)
 - requestedValue – nominální hodnota, kterou očekáváme, slouží pro nastavení měřicího rozsahu při tvorbě kanálu
- vrací
 - pole naměřených hodnot délky DATALEN

6.2.3.6 Měření proudu

- float64* zmerProud(char *connectionType, char *lineConnected, double requestedValue, double resistor);
- parametry
 - connectionType – řetězec obsahující informaci o typu zapojení, „RSE“ pro zapojení se společnou zemí, „Diff“ pro diferenciální zapojení
 - lineConnected – řetězec obsahující informaci o připojeném kanálu (viz Obr. 3.2 Popis Analogových signálů zařízení)
 - requestedValue – nominální hodnota, kterou očekáváme, slouží pro nastavení měřicího rozsahu při tvorbě kanálu
 - resistor – hodnota odporu na němž se měří úbytek napětí jako obraz protékajícího proudu
- vrací
 - pole naměřených hodnot délky DATALEN

6.2.3.7 Vyhodnocení naměřených dat

- int vyhodnotData(float64 *data, double * nominal, double * max, double * min);
- parametry
 - data – pole naměřených hodnot
 - nominal – adresa, do níž se uloží střední hodnota
 - max – adresa, do níž se uloží maximální hodnota

- min – adresa, do níž se uloží minimální hodnota
- vrací
 - 0 pokud proběhne úspěšně
 - kladnou hodnotu, pokud dojde k varování
 - zápornou hodnotu, pokud dojde k chybě
 - číselné kódy návratových hodnot chyb a varování jsou definovány v knihovně analysis
- Funkce provede jednoduchou filtraci (digitální filtr je navržen jako dolní propust s parametry specifikovanými příslušnými konstantami v hlavičce)

6.2.3.8 Porovnání naměřených hodnot

- int porovnej(double value, double nominal, double tolerance);
- parametry
 - value – naměřená hodnota
 - nominal – správná hodnota
 - tolerance – šířka tolerančního pásma v procentech
- vrací
 - 0 – pokud je naměřená hodnota v tolerančním pásmu
 - -1 – jinak

6.2.3.9 Nastavení parametrů měření

- int nastavParametry(CVIXMLElement Element, char* connectionType, char* lineConnected, double * requestedValue, double * allowedTolerance, double * resistor, double * zesilovac, char * ovladani, char *jmenoZkousky);
- Parametry
 - Element – XML element prvku měření
 - connectionType – řetězec obsahující informaci o typu zapojení, „RSE“ pro zapojení se společnou zemí, „Diff“ pro diferenciální zapojení
 - lineConnected – řetězec obsahující informaci o připojeném kanálu (viz Obr. 3.2 Popis Analogových signálů zařízení)
 - requestedValue – nominální hodnota, kterou očekáváme, slouží pro nastavení měřicího rozsahu při tvorbě kanálu
 - resistor – hodnota odporu na němž se měří úbytek napětí jako obraz protékajícího proudu
 - zesilovac – hodnota zesilovače, který je použit pro přizpůsobení měřených veličin, není-li uveden počítá se s hodnotou 1
 - ovladani – analogový výstup, který je použit jako ovládací kanál pro regulaci
 - jmenoZkousky – text, který se zobrazí při provádění dané zkoušky
- Vrací
 - 0 pokud proběhne úspěšně
 - -1 pokud dojde k chybě

6.2.3.10 Měření frekvence

- `int zmerFrekvenci(char * lineConnected);`
- parametry
 - `lineConnected` řetězec obsahující informaci o připojeném kanálu (viz Obr. 3.3 Popis Digitálních signálů zařízení)
- vrací
 - počet naměřených pulsů
 - zápornou hodnotu pokud dojde k chybě

6.2.3.11 Zpoždění

- `void noop(void)`
- žádná operace po dobu `NOOP_DELAY_TIME`

6.2.3.12 Regulace proudu

- `void nastavZatez(char *connectionType, char* lineConnected, char* ovladani, double requestedValue, double allowedTolerance, double resistor);`
- Parametry
 - `connectionType` – řetězec obsahující informaci o typu zapojení, „RSE“ pro zapojení se společnou zemí, „Diff“ pro diferenciální zapojení
 - `lineConnected` – řetězec obsahující informaci o připojeném kanálu (viz Obr. 3.2 Popis Analogových signálů zařízení)
 - `ovladani` – jméno výstupního analogového kanálu, který slouží jako řídicí veličina
 - `requestedValue` – nominální hodnota, kterou očekáváme, slouží pro nastavení měřicího rozsahu při tvorbě kanálu
 - `resistor` – hodnota odporu na němž se měří úbytek napětí jako obraz protékajícího proudu
 - `ovladani` – analogový výstup, který je použit jako ovládací kanál pro regulaci
- Funkce se snaží nastavit na měřícím kanálu `lineConnected` hodnotu `requestedValue`. Pokud se toto úspěšně podaří před uplynutím doby `INTERVAL` změní globální proměnnou „nastaveno“ na hodnotu 1.

6.2.3.13 Regulace napětí

- `void nastavNapeti(char *connectionType, char* lineConnected, char* ovladani, double requestedValue, double allowedTolerance, double zesilovac);`
- Parametry
 - `connectionType` – řetězec obsahující informaci o typu zapojení, „RSE“ pro zapojení se společnou zemí, „Diff“ pro diferenciální zapojení
 - `lineConnected` – řetězec obsahující informaci o připojeném kanálu (viz Obr. 3.2 Popis Analogových signálů zařízení)
 - `ovladani` – jméno výstupního analogového kanálu, který slouží jako řídicí veličina

- requestedValue - nominální hodnota, kterou očekáváme, slouží pro nastavení měřicího rozsahu při tvorbě kanálu
- resistor - hodnota odporu na němž se měří úbytek napětí jako obraz protékajícího proudu
- ovladani - analogový výstup, který je použit jako ovládací kanál pro regulaci
- Funkce se snaží nastavit na měřícím kanálu lineConnected hodnotu requestedValue. Pokud se toto úspěšně podaří před uplynutím doby INTERVAL změní globální proměnnou „nastaveno“ na hodnotu 1.

7 Ověření na zvolené jednotce

Tato kapitola se zabývá finálním testováním vyvinuté aplikace a HW rozhraní pro zkoušení zvolené jednotky. Je zde popsáno ovládání zkušební aplikace.

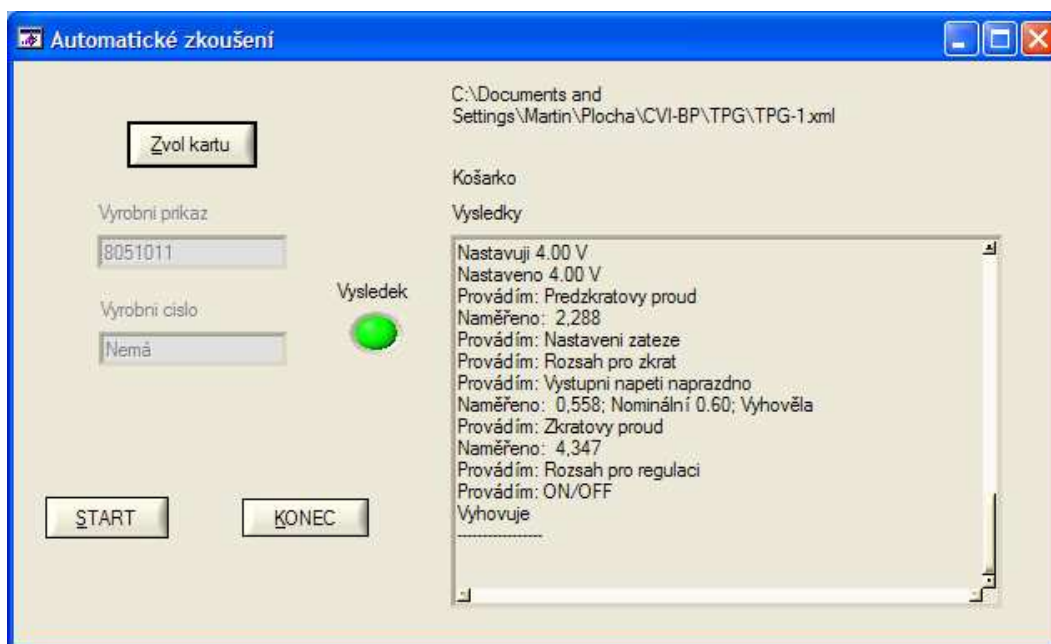
7.1 Měřicí přípravek

Měřicí přípravek (viz schéma příloha 1) tvoří hardwarové rozhraní mezi měřicím modulem NI USB-6008 a zkoušenou jednotkou. Je koncipován tak, aby kontaktoval co nejvíce měřicích bodů na příslušné jednotce. Body, které nelze vyvést takto přímo jsou kontaktovány jehlovým polem. Při tvorbě bylo myšleno na univerzálnost v rámci podobných typů zkoušených jednotek. Vzhledem k tomu, že původní tabulka požadovaných hodnot bylo napsána pro zkoušení se starým přípravkem, bylo potřeba po zavedení nové metody zkoušení s novým přípravkem tyto neplatné údaje opravit. Opravení tabulka je již zakomponována v konfiguračních souborech pro měření jednotlivých jednotek. Ověření proběhlo na zkušební sérii 20 kusů, během nichž se odladily některé nedostatky HW rozhraní a odhalené SW chyby.

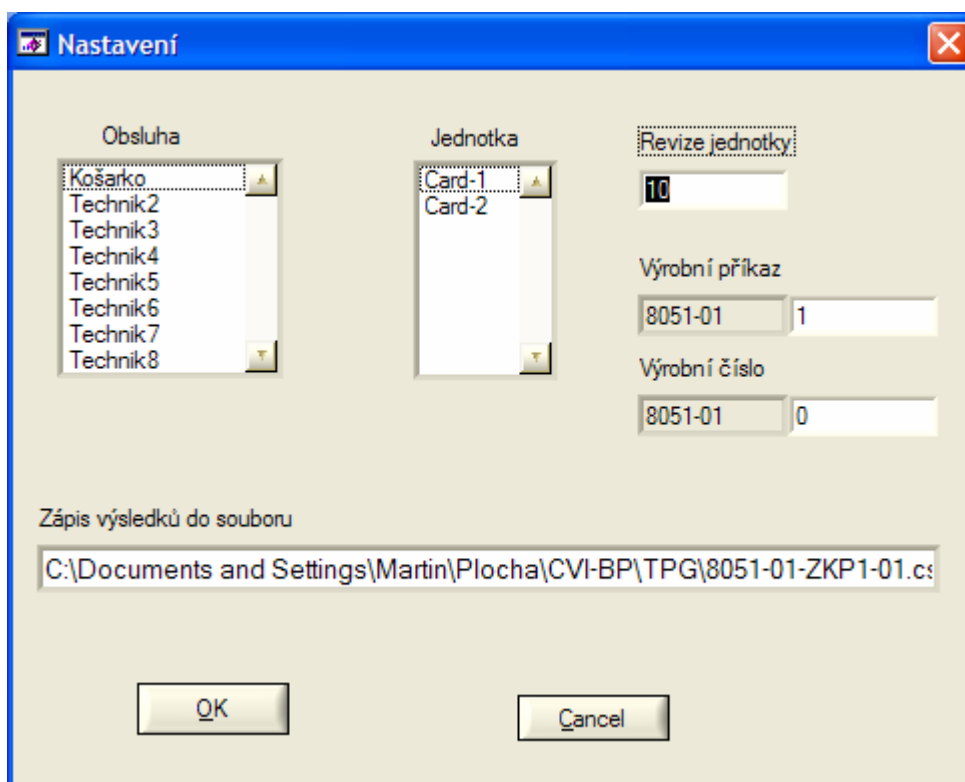
7.2 Vlastní pracoviště

Po připojení měřicího modulu k PC a připojení konektorů zapneme oba napájecí zdroje (pro zkoušení při 16,8V a 32V podle normy ČSN EN 50155 Drážní zařízení - Elektronická zařízení drážních vozidel) . Poté nastartujeme aplikaci AZK.exe. Spustí se hlavní okno programu (viz Obr. 7.1 Hlavní okno programu s ukončenou zkouškou). Po stisknutí tlačítka **ZVOL KARTU** se otevře modální okno s nastavením (viz Obr. 7.2 Nastavení pro zkoušení). Zde je nejprve přístupná pouze volba zkušební technika, který bude zařízení obsluhovat. Následně je zpřístupněna volba jednotek, do tohoto seznamu jsou načteny jednotky, které jsou specifikovány v nalezeném souboru config.xml. Po volbě jednotky se program pokusí nalézt defaultní xml soubor s informacemi o zkoušené jednotce. Pokud není soubor ve standardním umístění nalezen, program se zeptá na umístění. Dále je možno upravit nadefinovaný soubor pro ukládání výsledků, změnit číslo revize a doplnit čísla výrobního příkazu, či případně výrobní číslo zkoušené jednotky. Pokud jednotka nemá vlastní číslo (jako v našem případě), vloží se do pole 0. Po stisknutí **OK** se vrátíme do hlavního okna, kde je již aktivní tlačítko **START**. V hlavním okně se zobrazí cesta ke konfiguračnímu souboru zkoušené jednotky. Je zde také možné měnit výrobní číslo právě zkoušené jednotky. Po stisknutí **START** se v okně výsledky zobrazují aktuálně prováděné činnosti a naměřené hodnoty. LED v hlavním okně indikuje výsledek zkoušky:

- zelená - vyhověla
- červená - nevyhověla
- černá - zkouška probíhá



Obr. 7.1 Hlavní okno programu s ukončenou zkouškou



Obr. 7.2 Nastavení pro zkoušení

8 Závěr

S pomocí měřicího modulu NI USB-6008 a vývojového prostředí LabWindows/CVI byla vytvořena aplikace pro poloautomatické měřicí pracoviště. Spolu s měřicím přípravkem je počítáno, že se práce obsluhy zredukuje na výměnu měřené jednotky. Vzhledem k problémům, které vznikaly při klasickém zkoušení (například nastavování proudu zátěží, přepojování vodičů pro různé zkoušky) jsme ve vývojovém stádiu na hardwarovém rozhraní dosáhli časové úspory 90%, což je velmi uspokojivý výsledek.

Použití měřicího modulu NI USB-6008 ukázalo cestu, kterou je možno se nadále ubírat pro zkvalitnění a zrychlení kusových zkoušek. Mezi hlavní nedostatky tohoto konkrétního zařízení pro zvolenou aplikaci však patří malý počet analogových vstupů a nízká maximální měřitelná hladina jednotlivých veličin. Měřicí karta firmy National Instruments se navzdory své ceně ukázala jako velmi spolehlivý a přesný nástroj pro takováto měření.

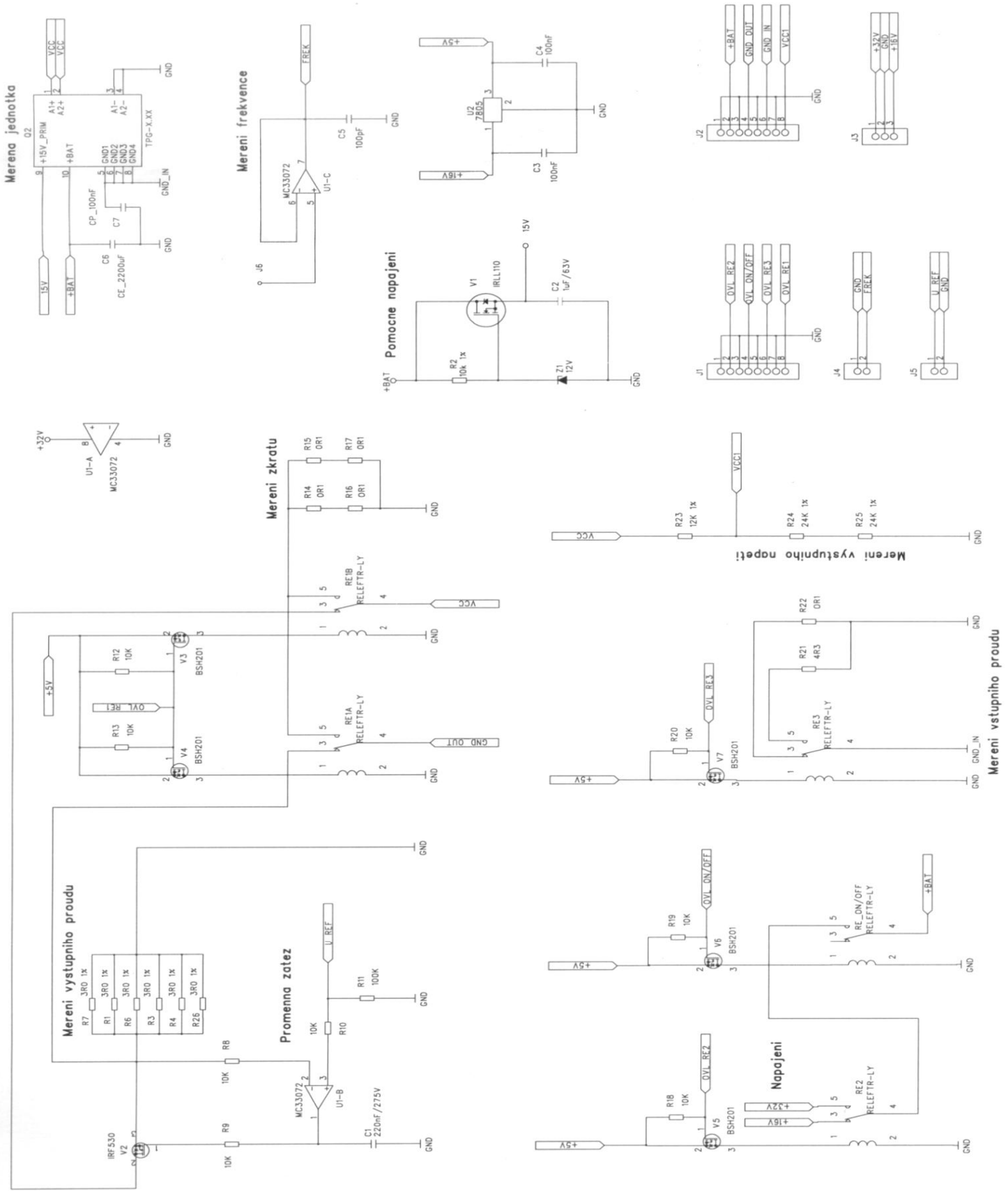
Jako další krok by bylo vhodné vyvinout aplikaci, která by s jednoduchým grafickým prostředím usnadňovala tvorbu a editaci měřicích XML dokumentů, toto však momentálně nebylo předmětem této práce. Dále bych zvážil nákup pokročilejšího zařízení ze sortimentu firmy National Instruments, které by eliminovalo nedostatky modulu NI USB-6008 ve zvolené aplikaci, pravděpodobně zařízení z kategorie M.

9 Literatura

- [1] National Instruments Corporation, USER GUIDE AND SPECIFICATIONS: SB-6008/6009, 2004-2005
- [2] LabWindows/CVI Support, product manual, Knowledge base dostupné na <http://www.ni.com/support/cvisupp.htm>
- [3] Essential C dostupné na <http://cslibrary.stanford.edu/>
- [4] Pointers and Memory dostupné na <http://cslibrary.stanford.edu/>
- [5] <http://www.cppreference.com>
- [6] ČSN EN 50155 ed. 2 (33 3555) Drážní zařízení – Elektronická zařízení drážních vozidel (vydání – srpen 2002)
- [7] Gene F. Franklin, J.David Powell, Abbas Emami-Naeini, Feedback Control of Dynamic Systems fifth edition, Pearson Prentice Hall, 2006
- [8] Jan Uhlíř, Pavel Sovka, Roman Čmejla, Úvod do číslicového zpracování signálů, ČVUT, 2003
- [9] Václav Hlaváč, Miloš Sedláček, Zpracování signálů a obrazů, ČVUT, 2000

Přílohy

Schéma měřicího přípravku



Ukázka výstupního souboru

V této ukázce je potlačen výpis jména operátora a data zkoušení z důvodu přizpůsobení šířky výstupu šířce stránky.

cislo	rev.	lin-16,8	Uout-16,8	linP-16,8	UoutP-16,8	frek-16,8	loutPK-16,8	UoutK-16,8	loutK-16,8	lin-32	Uout-32	linP-32	UoutP-32	frek-32	loutPK-32	UoutK-32	loutK-32
8051011	10	0,025	5,262	0,59	4,984	107481	2,167	0,476	3,547	0,015	5,262	0,313	5,005	107639	2,324	0,558	4,346
8051011	10	0,025	5,263	0,565	4,968	106337	2,135	0,474	3,52	0,015	5,263	0,309	5,001	106097	2,3	0,556	4,325
8051011	10	0,025	5,263	0,591	4,983	107516	2,147	0,475	3,538	0,015	5,263	0,311	5,007	107802	2,303	0,556	4,327
8051011	10	0,025	5,263	0,59	4,983	115344	2,145	0,474	3,53	0,015	5,264	0,311	5,003	108239	2,291	0,555	4,32
8051011	10	0,025	5,264	0,574	4,977	109468	2,121	0,473	3,524	0,015	5,264	0,312	5,002	107836	2,279	0,555	4,311
8051011	10	0,025	5,264	0,578	4,987	104768	2,124	0,473	3,527	0,015	5,264	0,312	5,002	108216	2,302	0,555	4,314
8051011	10	0,025	5,265	0,571	4,982	107631	2,124	0,473	3,521	0,015	5,264	0,312	4,999	108423	2,267	0,554	4,31
8051011	10	0,025	5,266	0,579	4,968	107763	2,097	0,47	3,496	0,015	5,265	0,312	4,991	114182	2,22	0,553	4,297
8051011	10	0,025	5,266	0,579	4,967	107770	2,095	0,47	3,49	0,015	5,266	0,31	4,99	114314	2,274	0,552	4,285
8051011	10	0,025	5,263	0,575	4,981	107801	2,139	0,474	3,534	0,015	5,263	0,312	4,992	108615	2,297	0,557	4,344
8051011	10	0,025	5,263	0,575	4,963	106117	2,123	0,471	3,511	0,015	5,264	0,312	4,987	107698	2,267	0,556	4,328
8051011	10	0,025	5,265	0,582	4,963	107728	2,113	0,472	3,506	0,015	5,265	0,311	4,988	108674	2,242	0,555	4,319
8051011	10	0,025	5,264	0,58	4,971	107526	2,117	0,473	3,521	0,015	5,264	0,312	4,992	108958	2,275	0,555	4,321
8051011	10	0,025	5,265	0,578	4,971	108048	2,102	0,471	3,506	0,015	5,266	0,309	4,989	109226	2,254	0,554	4,309
8051011	10	0,025	5,266	0,577	4,969	108315	2,09	0,471	3,5	0,015	5,266	0,312	4,989	107618	2,189	0,553	4,3
8051011	10	0,025	5,263	0,584	4,972	107695	2,145	0,474	3,535	0,016	5,263	0,312	4,991	108510	2,299	0,557	4,34

Ukázka konfiguračního souboru pro měření

```
<jednotka>
<typ>TPG-1</typ>
<revize>10</revize>
<cislo>8051-01</cislo>
<log>C:\Documents and Settings\Martin\Plocha\CVI-BP\TPG\8051-01-ZKP1-01.csv</log>
<mereni>
  <Switch nominalni="0" kanal="d0">ON/OFF</Switch>

  <Switch nominalni="0" kanal="d3">Prepnuti rozsahu vstupniho proudu</Switch>
  <Current nominalni="0.024" tolerance="10" kanal="ai1" zapojeni="Diff" resistor="4.3">zkouska vstupniho proudu naprazdno</Current>
  <Switch nominalni="1" kanal="d3">Prepnuti rozsahu vstupniho proudu</Switch>
  <Voltage nominalni="5.2" tolerance="5" kanal="ai0" zapojeni="Diff" zesilovac="0.77">vystupni napeti naprazdno</Voltage>
  <Load nominalni="1.2" tolerance="3" kanal="ai2" zapojeni="Diff" resistor="0.5" ovladani="ao0">Nastaveni zateze</Load>
  <Current nominalni="0.55" tolerance="12" kanal="ai1" zapojeni="Diff" resistor="0.1">Vstupni proud v pracovnim bode</Current>

  <Voltage nominalni="5" tolerance="7" kanal="ai0" zapojeni="Diff" zesilovac="0.77">vystupni napeti v jmenovitem bode</Voltage>
  <Frequency nominalni="105000" tolerance="10">Mereni frekvence</Frequency>

  <RegNapeti nominalni="4.0" kanal="ai0" ovladani="ao0" zapojeni="Diff" zesilovac="0.77" tolerance="5">Uout pro zkratovou ochranu</RegNapeti>

  <Current nominalni="2.0" tolerance="10" kanal="ai2" zapojeni="Diff" resistor="0.5">Predzkratovy proud</Current>
  <Load nominalni="0" tolerance="3" kanal="ai2" zapojeni="Diff" resistor="0.5" ovladani="ao0">Nastaveni zateze</Load>
  <Switch nominalni="0" kanal="d1">Rozsah pro zkrat</Switch>
  <Voltage nominalni="0.47" tolerance="10" kanal="ai0" zapojeni="Diff" zesilovac="0.77">vystupni napeti naprazdno</Voltage>
  <Current nominalni="3.5" tolerance="10" kanal="ai2" zapojeni="Diff" resistor="0.1">Zkratovy proud</Current>

  <Switch nominalni="1" kanal="d1">Rozsah pro regulaci</Switch>
  <Load nominalni="0" tolerance="3" kanal="ai2" zapojeni="Diff" resistor="0.5" ovladani="ao0">Odlehcen</Load>

  <Switch nominalni="0" kanal="d2">Prepnuti napajeciho napeti na 32V</Switch>

  <Switch nominalni="0" kanal="d3">Prepnuti rozsahu vstupniho proudu</Switch>
  <Current nominalni="0.015" tolerance="10" kanal="ai1" zapojeni="Diff" resistor="4.3">zkouska vstupniho proudu naprazdno</Current>
  <Switch nominalni="1" kanal="d3">Prepnuti rozsahu vstupniho proudu</Switch>
  <Voltage nominalni="5.2" tolerance="5" kanal="ai0" zapojeni="Diff" zesilovac="0.77">vystupni napeti naprazdno</Voltage>
  <Load nominalni="1.2" tolerance="3" kanal="ai2" zapojeni="Diff" resistor="0.5" ovladani="ao0">Nastaveni zateze</Load>
  <Current nominalni="0.29" tolerance="12" kanal="ai1" zapojeni="Diff" resistor="0.1">Vstupni proud v pracovnim bode</Current>

  <Voltage nominalni="5" tolerance="7" kanal="ai0" zapojeni="Diff" zesilovac="0.77">vystupni napeti v jmenovitem bode</Voltage>
  <Frequency nominalni="105000" tolerance="10">Mereni frekvence</Frequency>

  <RegNapeti nominalni="4.0" kanal="ai0" ovladani="ao0" zapojeni="Diff" zesilovac="0.77" tolerance="5">Uout pro zkratovou ochranu</RegNapeti>
  <Current nominalni="2.3" tolerance="10" kanal="ai2" zapojeni="Diff" resistor="0.5">Predzkratovy proud</Current>
  <Load nominalni="0" tolerance="3" kanal="ai2" zapojeni="Diff" resistor="0.5" ovladani="ao0">Nastaveni zateze</Load>
  <Switch nominalni="0" kanal="d1">Rozsah pro zkrat</Switch>
  <Voltage nominalni="0.6" tolerance="15" kanal="ai0" zapojeni="Diff" zesilovac="0.77">vystupni napeti naprazdno</Voltage>
  <Current nominalni="4.4" tolerance="10" kanal="ai2" zapojeni="Diff" resistor="0.1">Zkratovy proud</Current>
  <Switch nominalni="1" kanal="d1">Rozsah pro regulaci</Switch>

  <Switch nominalni="1" kanal="d0">ON/OFF</Switch>
</mereni>
</jednotka>
```