

Univerzita Palackého v Olomouci  
Přírodovědecká fakulta  
Katedra experimentální fyziky

## **BAKALÁŘSKÁ PRÁCE**

**Řízení programovatelného napájecího zdroje – aplikační  
testy rozhraní RS-232, USB, GPIB a LXI**



Autor:	Lukáš Kouřil
Studijní program:	B1701 Fyzika
Studijní obor:	Přístrojová fyzika
Forma studia:	Prezenční
Vedoucí práce:	RNDr. Jiří Pechoušek, Ph.D.
Termín odevzdání práce:	10.5.2012

Prohlašuji, že jsem předloženou bakalářskou práci vypracoval samostatně pod vedením RNDr. Jiřího Pechouška, Ph.D. a že jsem použil zdrojů, které cituji a uvádím v seznamu použitých zdrojů.

V Olomouci .....

.....

## Bibliografická identifikace

Jméno a příjmení autora	Lukáš Kouřil
Název práce	Řízení programovatelného napájecího zdroje – aplikační testy rozhraní RS-232, USB, GPIB a LXI
Typ práce	Bakalářská
Pracoviště	Katedra experimentální fyziky
Vedoucí práce	RNDr. Jiří Pechoušek, Ph.D.
Rok obhajoby práce	2012
Klíčová slova	Programovatelný napájecí zdroj, přístrojová rozhraní, LabVIEW
Počet stran	62
Počet příloh	3
Jazyk	Český

### Abstrakt

Předkládaná bakalářská práce se zabývá řízením programovatelného napájecího zdroje Sorensen XEL 30-3DP. Pro veškerou komunikaci se zdrojem a všechna měření byly použity programy (VIs), které byly vytvořeny ve vývojovém programovacím prostředí LabVIEW. Hlavní náplní práce je porovnání jednotlivých přístrojových rozhraní zdroje z hlediska jejich přenosových rychlostí (odezev na různé příkazy). Druhá část práce se věnuje krátkodobému a dlouhodobému měření stability zdroje za pomoci digitálního multimetru Agilent 34401A.

## Bibliographical identification

Autor's first name and surname	Lukáš Kouřil
Title	Control of programmable power supply – application tests with RS-232, USB, GPIB and LXI interfaces
Type of thesis	Bachelor
Department	Department of Experimental Physics
Supervisor	RNDr. Jiří Pechoušek, Ph.D.
The year of presentation	2012
Keywords	Programmable power supply, instrument interfaces, LabVIEW
Number of pages	62
Number of appendices	3
Language	Czech

### Abstract

The proposed bachelor thesis deals with control of the programmable power supply Sorensen XEL 30-3DP. Programs (VIs), which were created in a development programming environment LabVIEW, were used for all communication with the power supply and all measurements. The main object of the thesis is to compare individual instrument interfaces in terms of baud rates (responses for different commands). The second part of the thesis is devoted to short-term and long-term measurements stability of power supply using the digital multimeter Agilent 34401A.

# Obsah

Úvod .....	6
<b>1 Číslicový měřicí systém .....</b>	<b>7</b>
1.1 Sériové rozhraní RS232 .....	7
1.2 Rozhraní USB .....	9
1.3 Sběrnice IEEE 488 (GPIB-General Purpose Interface Board) .....	12
1.3.1 Standard IEEE 488.1 .....	12
1.3.2 Standard IEEE 488.2 .....	15
1.4 Rozhraní LXI .....	16
<b>2 Technické prostředky pro realizaci práce .....</b>	<b>21</b>
2.1 Napájecí zdroj Sorensen XEL 30-3DP .....	21
2.1.1 Základní vlastnosti zdroje .....	21
2.1.2 Provoz vzdáleného ovládání .....	22
2.1.3 Vzdálené příkazy .....	25
2.2 Digitální multimetr Agilent 34401A .....	26
<b>3 Vývojové programovací prostředí LabVIEW .....</b>	<b>28</b>
3.1 Společné prvky programů .....	28
3.1.1 VISA název zdroje .....	29
3.1.2 Inicializace sériového portu .....	29
3.1.3 Zadání příkazu .....	30
3.1.4 Odpovědi přístroje .....	31
3.1.5 Vyčištění bufferu .....	32
3.1.6 Uzavření VISA .....	32
3.1.7 Struktura VISA řetězce .....	32
3.2 Programy pro měření času odezvy přístroje .....	33
3.3 Programy pro měření doby inicializace sériového portu .....	37
3.4 Program pro porovnání hodnot ze zdroje a hodnot naměřených multimetrem .....	37
3.5 Program pro měření dlouhodobé stability napájecího zdroje .....	39
<b>4 Výsledky měření a jejich analýza .....</b>	<b>40</b>
4.1 Měření komunikačních parametrů .....	40
4.1.1 Měření času odezvy při dotazu na napětí .....	41
4.1.2 Měření času nastavení napětí .....	43
4.1.3 Měření času odezvy při nastavení i dotazu na napětí .....	44
4.1.4 Porovnání naměřených a spočítaných hodnot pomocí přenosové rychlosti .....	46
4.1.5 Měření pro charakterizaci LXI .....	47
4.2 Měření doby inicializace sériového portu .....	51
4.3 Porovnání hodnot ze zdroje a hodnot naměřených digitálním multimetrem .....	51
4.3.1 Měření při rozsahu 0-1 V .....	52
4.3.2 Měření při rozsahu 0-5 V .....	53
4.3.3 Měření při rozsahu 10-20 V .....	54
4.3.4 Měření při rozsahu 0-30 V .....	55
4.4 Měření stability programovatelného napájecího zdroje .....	56
4.4.1 Stabilita programovatelného napájecího zdroje při napětí 5 V .....	56
4.4.2 Stabilita programovatelného napájecího zdroje při napětí 30 V .....	57
<b>Závěr .....</b>	<b>58</b>
<b>Seznam použitých zdrojů .....</b>	<b>59</b>
<b>Příloha 1 .....</b>	<b>60</b>
<b>Příloha 2 .....</b>	<b>61</b>
<b>Příloha 3 .....</b>	<b>62</b>

## Úvod

Hlavní podstatou této bakalářské práce je seznámení se s obsluhou programovatelného zdroje Sorensen XEL 30-3DP a následným vytvořením aplikací, nejen pro řízení zdroje pomocí čtyř různých rozhraní (RS232, USB, GPIB a LXI), ale také pro měření provozních parametrů zdroje pomocí digitálního multimetru. Trendem dnešní doby je měření co nejvíce zjednodušovat a automatizovat, aby člověk ušetřil čas a musel co nejméně asistovat u měření. Jedním způsobem je realizovat měření pomocí virtuální instrumentace. Bezpochyby největší výhodou této metody je, že uživatel vytvoří danou aplikaci spustitelnou pomocí jednoho tlačítka, která bez další asistence provádí jednotlivé kroky měření. Mezi výhody virtuální instrumentace určitě také patří případná rychlá opakovatelnost měření, snadnější ukládání a zpracování naměřených dat.

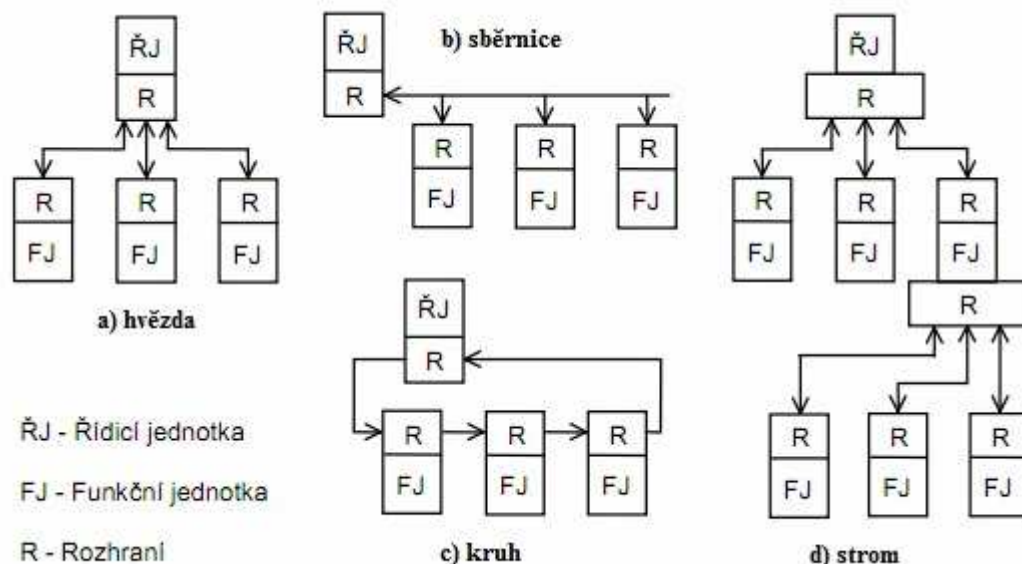
Dalším z cílů práce je proměření vlastností jednotlivých rozhraní, především rychlosti komunikace mezi zdrojem a počítačem. Všechny tyto programy jsou implementovány ve vývojovém grafickém prostředí LabVIEW, které umožňuje efektivní a rychlé vyvíjení programů pro tuto bakalářskou práci.

První část se věnuje popisu jednotlivých rozhraní zdroje, především jejich technické realizaci. Druhá kapitola je věnována oběma měřicím komponentům. Podrobně je také popsán vývoj programů ve vývojovém prostředí LabVIEW a analýza naměřených dat. V závěru je provedeno zhodnocení získaných výsledků a jejich porovnání s teoretickými hodnotami.

# 1 Číslicový měřicí systém

Číslicový měřicí systém je soustava různých vzájemně propojených zařízení, které zajišťují průběh celého měření včetně plné automatizace a samozřejmě i řešení dané úlohy.

Nejčastěji využívané topologie zapojení různých měřicích zařízení v systému jsou zobrazeny na Obr. 1. Pro zadané úlohy bylo aplikováno zapojení do hvězdy, které ovšem není moc výhodné z hlediska omezeného počtu zapojitelných přístrojů. Druhou nevýhodou může být menší rychlost přenosu dat. V praxi se proto více používá sběrnice, kde je možné připojit až 15 přístrojů (v případě paralelní sběrnice IEEE 488 jinak též GPIB). Pomocí tohoto typu zapojení lze rovněž dosáhnout vyšší přenosové rychlosti oproti sériového rozhraní RS232.



**Obr. 1** Topologie číslicových měřicích systémů; upraveno z [1].

Řídicí jednotkou je myšlen ve většině případů počítač a funkční jednotky jsou pak přístroje, které využívají nejrůznější druhy rozhraní pro komunikaci s tímto počítačem. Takto sestavené číslicové měřicí systémy jsou nejčastěji využívány přímo ve výzkumných laboratořích, ale také ve firmách, kde mohou řídit výrobu apod. V následujících podkapitolách jsou popsána rozhraní, kterými je vybaven zdroj Sorensen XEL 30-3DP.

## 1.1 Sériové rozhraní RS232

Toto rozhraní bylo původně navrženo pro telekomunikační techniku. Později se však objevilo i v osobních počítačích IBM a našlo uplatnění i v měřicí technice. Jeho hlavními

nedostatky jsou především nízká přenosová rychlost, spojení bod-bod a nízká odolnost proti rušení. Přesto se i v dnešní době pořád poměrně často využívá, především u levnějších měřicích přístrojů. Při komunikaci jsou data posílána jako posloupnosti jednotlivých bitů, přičemž v jednom časovém okamžiku je přenášen vždy jeden jediný bit. Většina nových počítačů (především notebooků) již není vybavena sériovým rozhraním.



Existuje několik druhů konektorů, ale používají se hlavně konektory s 25 a 9 piny (kontakty). U počítačů a měřicích přístrojů, které toto rozhraní podporují, se používá především konektor s 9 piny (viz Obr. 2). V Tab. 1 je pak význam těchto pinů podrobněji popsán.

**Obr. 2** Konektor RS232 s 9 piny.

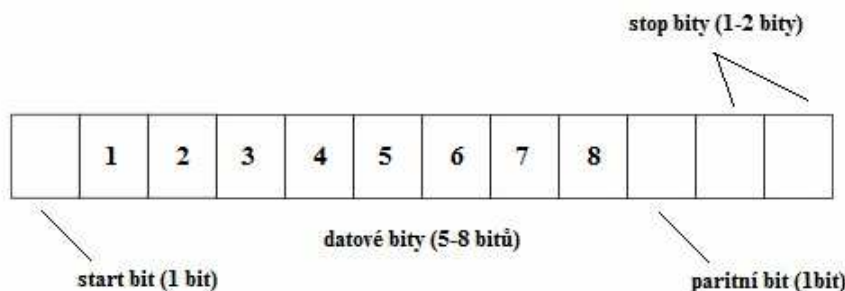
**Tab. 1:** Popis jednotlivých pinů pro sériové rozhraní RS232; upraveno z [1].

Číslo pinu	Označení	Název	Popis
1	DCD	Data Carrier Detected	signál vysílá funkční jednotka, sděluje řídicí jednotce, že byl detekován signál
2	RX	Received Data	data přijatá do řídicí jednotky
3	TX	Transmitted Data	data odeslaná z řídicí jednotky
4	DTR	Data Terminal Ready	signál vysílá řídicí jednotka informuje funkční jednotku, že je připravena
5	GND	Signal Ground	signálový zemní vodič
6	DSR	Data Set Ready	signál vysílá funkční jednotka, informuje, že je připravena a schopna komunikovat
7	RTS	Request To Send	signál vysílá řídicí jednotka, informuje funkční jednotku o tom, že je schopna přijímat data
8	CTS	Clear To Send	signál vysílá funkční jednotka, informuje řídicí jednotku o tom, že je připravena přijímat a odesílat data
9	RI	Ring Indicator	signál vysílá funkční jednotka, žádost o spojení



Pojmem řídicí jednotka je v této tabulce myšlen počítač nebo terminál, který se také někdy označuje zkratkou DTE (*Data Terminal Equipment*). Funkční jednotka, označovaná také zkratkou DCE (*Data Communication Equipment*), je pak měřicí přístroj nebo modem.

Při asynchronním sériovém přenosu je nutné nastavit parametry sériového portu. Těmito parametry jsou přenosová rychlost, počet start bitů, počet datových bitů, počet stop bitů a parita. Podle typu zařízení je možno nastavit přenosovou rychlost na 110, 300, 600, 1200, 2400, 4800, 9600 nebo 19200 bit/s [1]. Zbytek parametrů jsou součástí tzv. formátu přenosu dat, který je zobrazen na Obr. 3.



**Obr. 3** Formát přenosu dat pro sériové rozhraní RS232.

K řízení přenosu dat se většinou používá metoda programového řízení, která se označuje jako XON/XOFF. Při této metodě přístroj, který přijímá data, ovládá komunikaci vysláním speciálních znaků. Signál XON vyšle přístroj pokud je na příjem dat připraven, naopak signál XOFF, když není připraven, nebo nestíhá zpracovávat data. Tento cyklus se může opakovat libovolně dlouho. Nevýhodou této metody je dlouhá doba reakce přístroje a také omezení na data kódovaná v ASCII.

Kódování v měřicí technice je většinou v případě RS232 pomocí ASCII (7 bitový formát s paritou), binární (8 bitový formát bez parity) nebo Hex (každý byte v podobě dvou znaků reprezentujících hexadecimální číslice v 7 bitovém formátu s paritou).

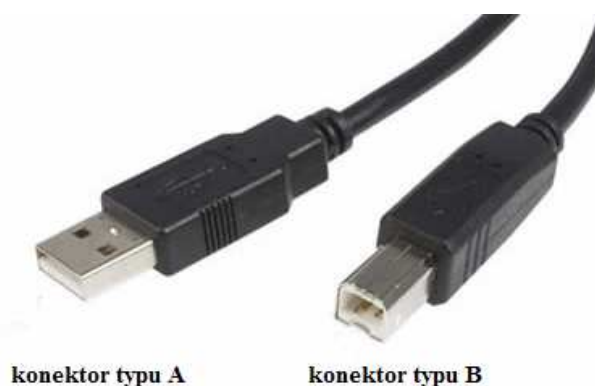
## **1.2 Rozhraní USB**

Rozhraní USB (*Universal Serial Bus*) bylo původně navrženo pro připojení počítačů do datové telefonní sítě, ale také pro připojení běžných komerčních periférií k počítači. Ačkoliv nebylo původní myšlenkou používat rozhraní USB v měřicí technice, existuje v dnešní době velké množství přístrojů, které umožňují použít toto rozhraní i pro měřicí aplikace. Tento typ rozhraní se stal „náhradou“ za sériové rozhraní RS232 a to hlavně díky jeho větší přenosové rychlosti a také většímu počtu zapojitelných přístrojů (až 127 pomocí rozbočovačů - hubů). Jeho přenosová rychlost (viz níže) už konkuruje rychlosti sběrnice

GPIB. Na rozdíl od GPIB toto rozhraní však nepotřebuje žádnou speciální zásuvnou desku rozhraní, jelikož je jím v dnešní době vybaven každý osobní počítač.

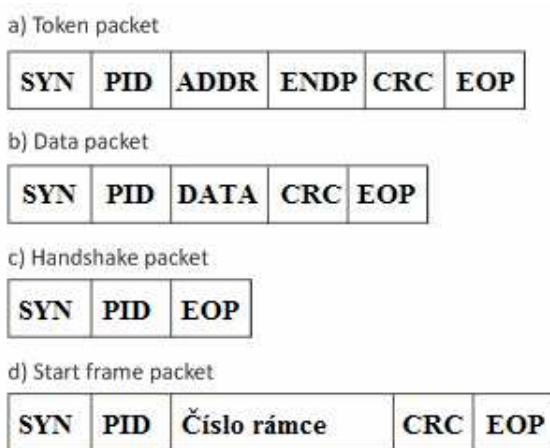
V dnešní době již existují tři verze USB. V první verzi USB 1.1 z roku 1995 se rozlišují dva typy – zařízení pomalá (*low-speed*) s přenosovou rychlostí až 1,5 Mb/s a rychlá zařízení (*high-speed*) s přenosovou rychlostí až 12 Mb/s. V roce 2000 přišla nová verze USB 2.0, která je v dnešní době standardním vybavením všech osobních počítačů. Tato verze disponuje v režimu *high-speed* přenosové rychlosti až 480 Mb/s. Samozřejmě je také kompatibilní s verzí 1.1 [1].

Data se v případě komunikace přes USB přenášejí podobně jako u RS232 bit po bitu, ale jsou přenášeny na stejných linkách oběma směry (u RS232 odděleně). Existují dva druhy konektorů (konektor typu A a konektor typu B), které jsou znázorněny na Obr. 4.



Obr. 4 Konektory USB.

Kabel USB obsahuje čtyři vodiče. První z nich (Vbus) slouží jako napájení +5 V. Další dva (D- a D+) jsou určeny pro přenos signálu a poslední (GND) je zemní vodič.



Komunikace mezi počítačem a funkční jednotkou probíhá za pomoci čtyř typů paketů (viz Obr. 5, zkratky z něj viz Tab. 2). Prvním z nich je tzv. *token packet*, který je odeslán na počátku každé datové výměny. Tento paket obsahuje popis typu a směru výměny dat, adresu zařízení připojeného přes USB a číslo koncové jednotky.

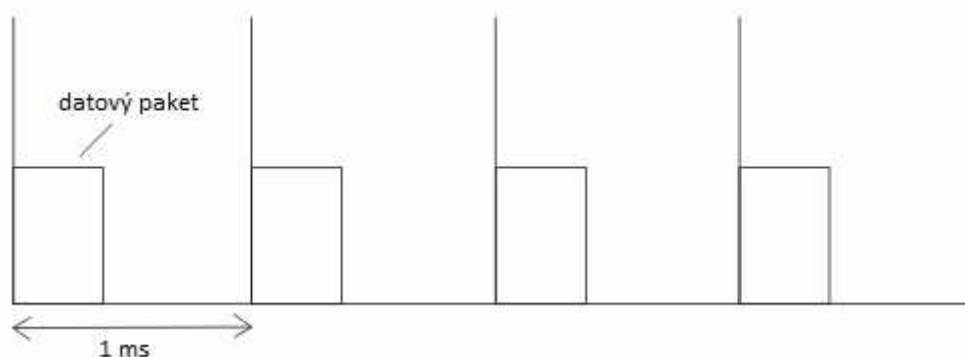
Obr. 5 Formáty paketů pro USB; upraveno z [2].

Po odeslání tohoto typu paketu zařízení, které má vyslat data, vyšle datový paket (jen v případě, že data jsou k dispozici). Posledním typem paketu je tzv. *handshake packet*, který odesílá zařízení přijímací data. Tento paket složí jako informace, že přenos dat proběhl v pořádku. Speciálním typem paketu je tzv. *start of frame packet*, který vysílá počítač v intervalu s periodou cca 1 ms. Tento paket definuje datový rámec u USB 2.0 navíc i mikrorámec, který odpovídá 0,125 ms [2].

**Tab. 2:** Význam zkratk z popisu formátu paketů pro USB [2].

Zkratka	Název	Počet bitů	Popis
<b>SYN</b>	Synchronizační pole	USB 1.1 8 bitů USB 2.0 32 bitů	synchronizuje hodiny přijímače
<b>PID</b>	Paket ID	8 bitů	identifikuje poslaný paket
<b>ADDR</b>	Adresa	7 bitů	specifikuje adresu zařízení, kterému je paket určen
<b>ENDP</b>	Pole koncového bodu	4 bity	povoluje 16 možných koncových bodů
<b>CRC</b>	Kontrolní součet	token packet 5 bitů frame packet 5 bitů data packet 16 bitů	provede se kontrolní součet na datech
<b>EOP</b>	Konec paketu	asi 2 bity	signalizuje ukončení paketu

Maximální možný náklad dat (viz Obr. 5b, zkratka DATA) se liší podle přenosové rychlosti. Pro režim *low-speed* (1,5 Mb/s) je to 8 bitů a režim *full-speed* (12 Mb/s) 64 bitů. USB 2.0 (*high-speed* režim, 480 Mb/s) má maximální datový náklad dokonce až 1024 bitů. Data se tedy vysílají a přijímají v krátkých paketech o délce 8 bytů a nebo v delších paketech o délce 256 bytů. Tyto pakety se mohou během přenosu libovolně střídat. Veškerý přenos se uskutečňuje v rámcích (*frames*), které trvají přesně 1 ms [3]. Umístění dat v těchto rámcích je znázorněno na Obr. 6.



**Obr. 6** Umístění datového paketu v rámcích; upraveno z [3].

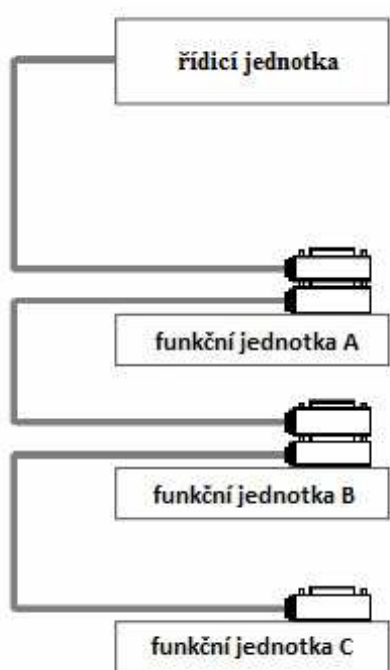
Pro samotný přenos dat jsou definovány dva způsoby, kterými může probíhat. Jedním z nich je tok dat (*stream*). Tok dat nemá přesně definovanou strukturu a využívá izochronní přenos dat v reálném čase. Druhým způsobem je zpráva (*message*), která má definovanou strukturu a využívá asynchronního přenosu dat.

Další výhodou USB je také to, že dokáže napájet některá periferní zařízení. Díky této výhodě dokáže napájet nejen periférie typu klávesnice a myš, ale také malé měřicí přístroje. Při komunikaci přes USB je vždy nutné nainstalovat příslušný ovladač. Tyto ovladače většinou dodává výrobce zařízení pokud jde o měřicí přístroj. V případě, kdy se jedná o periférii typu klávesnice nebo myš jej může obsahovat i operační systém počítače typu Windows.

### 1.3 Sběrnice IEEE 488 (GPIB – General Purpose Interface Board)

Propojení měřicích systémů pomocí sběrnice IEEE 488 patří v laboratorním prostředí určitě k těm nejvíce rozšířeným. Její historie sahá až do konce šedesátých let, kdy byla navržena americkou firmou Hewlett Packard. Americkou standardizační společností IEEE (*Institute of Electrical and Electronics Engineers*) však byla publikována až v roce 1975 jako standard IEEE 488. V roce 1987 byli již publikovány dva standardy (IEEE 488.1 a IEEE 488.2). Standard IEEE 488.1 byl vlastně klasický IEEE 488 z roku 1985, ale IEEE 488.2 byl již zcela nový standard.

#### 1.3.1 Standard IEEE 488.1

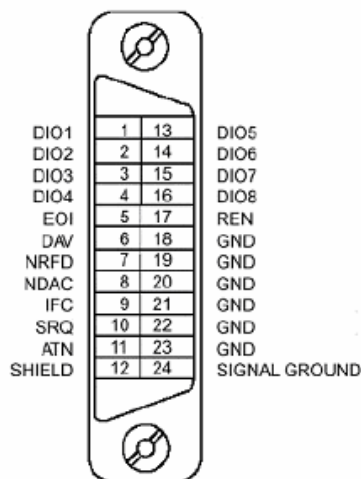


Měřicí systém řízený počítačem pomocí této sběrnice většinou potřebuje zásuvnou desku rozhraní (existují i redukce GPIB-USB). Řídicí počítač tedy řídí proces měření, ale také provádí zpracování, prezentaci a uložení naměřených dat. Připojení funkčních jednotek k řídicí jednotce je většinou realizováno jako sběrnice (viz Obr. 7). Tento druh zapojení umožňují speciální konektory, které mají z jedné strany „zástrčku“ a z druhé „zásuvku“. Maximální délka kabelů je omezena na 20 m a vzdálenost mezi dvěma funkčními jednotkami nesmí přesáhnout 2 m. Omezený je také maximální počet zapojitelných funkčních jednotek, který nesmí přesahovat 15 jednotek [4].

Obr. 7 Topologie sběrnice GPIB.

Funkční jednotky v Obr. 7 mohou být nejen měřicí přístroje, ale také jiné členy měřících obvodů a nebo periférie počítače např. tiskárna. Samozřejmě každá funkční jednotka z těchto příkladů musí být vybavena rozhraním GPIB.

Konektory tohoto rozhraní obsahují 24 pinů. Piny (viz Obr. 8) označované jako DIO1-DIO8, tedy celkem 8 vodičů jsou využívány pro přenos dat. Dalších 8 vodičů je využíváno pro jednovodičové řídicí signály a zbylých 8 vodičů je zemnicích.



**Obr. 8** Konektor GPIB; upraveno z [4].

**DIO1-DIO8** jsou datové vodiče. Přenášejí data po jednotlivých bytech, mohou mít informační význam (naměřené hodnoty) nebo to mohou být adresy.

**DAV** (*Data valid*) určuje platnost jednoho bytu na výstupu, během jeho trvání se signály na datových vodičích nesmí měnit (vysílá řídicí jednotka).

**NRFD** (*Not Ready For Data*) informuje o tom, že jednotka není připravena pro příjem dalšího bytu (vysílá funkční jednotka).

**NDAC** (*Not Data ACcepted*) informuje o tom, že ještě jednotka nepřijala vysílaný byt (vysílá funkční jednotka).

**ATN** (*ATteNtion*) rozlišuje jestli se na datových vodičích přenášejí data nebo adresy, pokud je logická jednička - adresy, pokud logická nula - data.

**IFC** (*InterFace Clear*) uvádí všechny funkční jednotky do klidového (předurčeného) stavu (vysílá řídicí jednotka).

**SRQ** (*Service ReQuest*) je přerušovací signál. Když řídicí jednotka přijme tento signál, postupně se zeptá všech jednotek zda-li požádali o přerušení.

**REN** (*Remote ENable*) přepojuje ovládání funkční jednotky z místního na dálkové (vysílá řídicí jednotka).

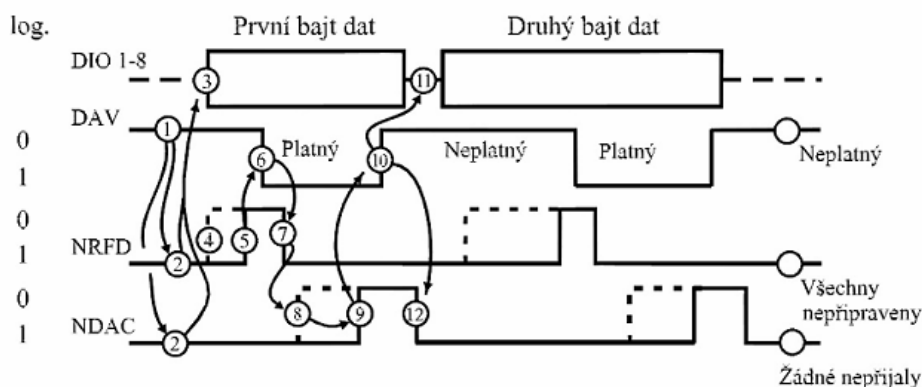
**EOI** (*End Or Identity*) je závislý na ATN. Pokud je na ATN logická nula potom určuje poslední byt přenosu, pokud je logická jednička tak je signál chápán jako výzva k zaslání stavového bytu.

V systému se sběrnici GPIB rozlišujeme čtyři typy přístrojů:

- a) **Typ A** - přístroj, který jako řidič, mluvčí i posluchač (příklad - počítač).
- b) **Typ B** - přístroj, který pracuje jako mluvčí i posluchač (příklad - číslicový voltmetr).
- c) **Typ C** - přístroj, který pracuje jen jako posluchač (příklad - tiskárna).
- d) **Typ D** - přístroj, který pracuje jen jako mluvčí (příklad - snímač děrné pásky).

Na Obr. 9 je znázorněn diagram přenosového cyklu, který je níže v bodech popsán:

- 1) Začíná start celého cyklu. Vysílající jednotka DAV= log. 0.
- 2) Proběhne kontrola signálů NRFD a NDAC.
- 3) Když NRFD= log. 1 a NDAC= log. 1 tak vysílající jednotka pošle data na sběrnici.
- 4) Příjemací jednotky postupně signalizují připravenost přijmout data.
- 5) Stav NRFD= log. 0 a NDAC= log. 1 znamenají, že všechny jednotky jsou připraveny přijmout data.
- 6) Když vysílající jednotka zjistí, že NRFD= log. 0 tak nastaví vodič DAV= log. 1, čímž potvrdí, že data na datových vodičích jsou platná.
- 7) Příjemací jednotky reagují na bod 6) změnou NRFD=log. 0 na NRFD= log. 1.
- 8) Příjemací jednotky indikují převzetí některých dat ze sběrnice vodičem NDAC=log. 0.
- 9) Indikace převzetí všech dat.
- 10) Vysílací jednotka reaguje na změnu stavu NDAC= log. 0 tím, že změní DAV na log. 0.
- 11) Data na datových vodičích po bodu 10) už tedy nejsou platná.
- 12) Změna stavu vodiče na NDAC= log. 1.



**Obr. 9** Přenosový cyklus GPIB; převzato z [4].

Výše uvedený cyklus je poměrně složitý, ale zajišťuje spolehlivý asynchronní přenos dat mezi funkčními jednotkami v systému. Přenosová rychlost se vždy musí přizpůsobit té nejpomalejší jednotce v systému. Tento cyklus umožňuje maximální přenosovou rychlost “jen“ 1 MB/s, typicky však nižší 250-500 kB/s [1].

Právě tato rychlost byla považována za velkou nevýhodu, zejména při přenosu velkých bloků dat. V návaznosti na tuto skutečnost byl později firmou National Instruments (NI) vyvinut rychlejší přenosový cyklus HS488 s přenosovou rychlostí až 8 MB/s. Skutečná rychlost ovšem závisí na délce použitého kabelu a na délce odesílaného bloku dat. Pokud je odesílaný blok dat menší než 5kB, je zrychlení oproti původnímu systému zanedbatelné (více o HS488 v [1] nebo [4]).

### 1.3.2 Standard IEEE 488.2

Je zcela novým standardem, který definuje funkční a operační vlastnosti přístrojů v systému s rozhraním IEEE 488.1. Všechny tyto pravidla pro činnost systému jsou:

- a) minimální sestava funkcí rozhraní dle IEEE 488.1
- b) formát dat a syntaxe zpráv
- c) rozšířený stavový model
- d) soubor obecných příkazů, které musí podporovat všechny přístroje
- e) protokol přístrojových zpráv

Formáty dat jsou rozděleny na dva druhy (pro příjem a vysílání). V podstatě se liší jen nepatrně. Přijímat lze “libovolný“ tvar a vysílat lze pouze přesně definovaný tvar. V původním standardu IEEE 488.1 byly data přenášeny po jednotlivých znacích v ASCII kódu (někdy se označuje ISO7) s tím, že signál DIO8 může být použit jako paritní bit, ale většinou se tak neděje. Novější verze standardu IEEE 488.2 umožňuje kódování dat pomocí tří kódů. Prvním z nich je již zmíněný ASCII kód a signál DIO8 se ignoruje. Druhý možný kód je tzv. celý osmibitový, kde jsou čísla vyjádřena binárně. Obsahuje tolik bytů, kolik je třeba. Nejvyšší váhu v bytu má bit DIO8. Nejvýznamnější byt se vysílá jako první. Konečně poslední je binární kód s pohyblivou čárkou. Jedná se o exponenciální tvar.

Standard IEEE 488.2 rovněž definuje syntaxe vysílaných a přijímaných zpráv (viz a-g) a rovněž povinné obecné příkazy (viz níže Tab. 5)

- a) oddělovač příkazů na dvou rozdílných úrovních pomocí dvojtečky
- b) oddělovač příkazů na stejné úrovni ve vícenásobném příkazu pomocí středníku

- c) oddělovač významné části příkazu od jeho parametrů pomocí mezery
- d) oddělovač jednotlivých parametrů zprávy nebo příkazu pomocí čárky
- e) ukončovací znak dotazu pomocí otazníku
- f) ukončovací znak řetězce při příjmu pomocí LF, LF i EOI nebo jen EOI
- g) ukončovací znak řetězce při vysílání pomocí LF a EOI současně

**Tab. 5:** Příklady některých povinných obecných příkazů pro IEEE 488.2; upraveno z [1].

<b>Příkaz</b>	<b>Název</b>	<b>Popis</b>
*RST	Reset command	vynuluje přístroj (ukončí operace, které přístroj provádí) a vrátí ho do klidového stavu, obsah stavových registrů zůstane nezměněn
*CLS	Clear status command	vynuluje stavové registry
*IDN?	Identification query	dotazuje se na specifikaci přístroje
*OPC	Operation complete	aktivuje hlášení o ukončení operace
*OPC?	Operation complete query	dotazuje se na ukončení operace
*WAI	Wait to continue	čeká na dokončení předchozích operací
*TST	Self test query	spustí vnitřní test přístroje

Samozřejmě, že těchto příkazů existuje více (všechny příkazy v [1]). Tab. 5 obsahuje jen pouhý výtah celého soupisu obecných příkazů. Většina ostatních příkazů jsou ovšem dotazy na obsahy různých registrů, které nejsou pro tuto práci nijak přínosné.

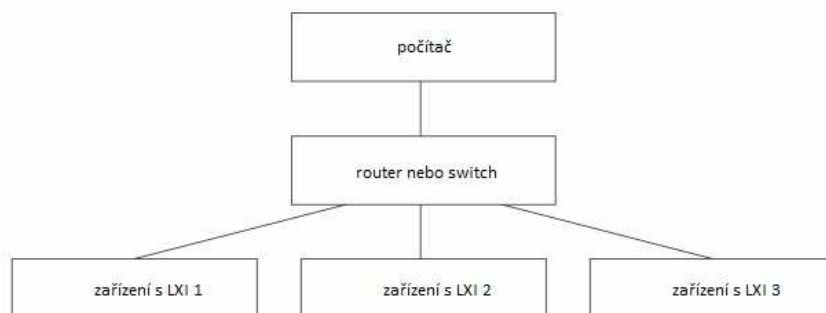
## **1.4 Rozhraní LXI**

V dnešní době je internet všude a proto se také rozšiřuje použití rozhraní LAN (*Local Area Network*) i do oblasti měřicí techniky. Největší výhodou je určitě fakt, že tímto rozhraním je v dnešní době vybaven snad každý počítač, což o jiných rozhraní typu RS232 nebo GPIB říct nelze. LXI (*Lan eXtension for Instrumentation*) je vlastně rozšířením rozhraní LAN do oblasti měřicí techniky. Toto rozšíření je nejnovějším vývojem v komunikaci s přístroji testovací a měřicí techniky. LXI má výhody Ethernetu, které umožňují šetřit čas v oblastech jako je programování a podporuje unikátní aplikace. Je to poměrně mladé rozhraní, které bylo představeno v roce 2005 firmami Hewlett Packard a VXI Technology. Kombinuje to nejlepší z GPIB přístrojů, VXI a PXI modulů. Díky standardizaci LAN, LXI nabízí nové možnosti při návrhu systémů jako jsou např. místní, dálkové nebo distribuované systémy. Dnes již existuje několik verzí LXI (1.1-1.4), v této práci bude popsána pouze LXI verze 1.2, kterou podporuje zdroj Sorensen XEL 30-3DP.

Konsorcium LXI očekává, že zařízení vybavená tímto rozhraním najdou využití v široké škále systémů, často i v kombinaci s těmi, které LXI nepodporují. Tato zařízení

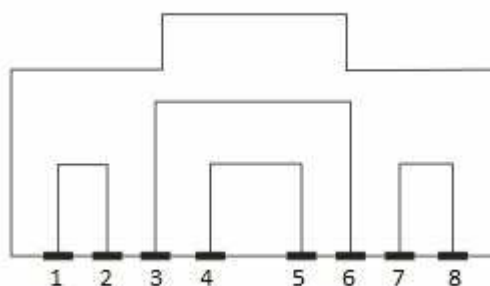


mohou podporovat např. GPIB, PXI, VXI a další rozhraní. Klíčovou vlastností LXI je použití standardizovaného LAN rozhraní, které podporuje internet na bázi rozhraní a programového řízení. Připojení může být jak fyzické pomocí UTP (*Unshielded Twisted Pair*) kabelu (topologie viz Obr. 11), tak i bezdrátové přes wi-fi. Rozhraní podporuje také provoz “peer to peer“.



**Obr. 11** Topologie zapojení zařízení s LXI.

Pro zapojení různých zařízení do systému se používají konektory typu RJ-45, který je zobrazen na Obr. 12. Piny jsou očíslovány a následně popsány v Tab. 6.



**Obr. 12** Konektor RJ-45.

**Tab. 6:** Popis pinů konektoru RJ-45.

Číslo pinu	Označení	Název	Popis
1	TD+	Transmit Data	přenáší data
2	TD-	Transmit Data	přenáší data
3	RD+	Receive Data	přijímá data
4	NC	No Connection	nezapojeno
5	NC	No Connection	nezapojeno
6	RD-	Receive Data	přijímá data
7	NC	No Connection	nezapojeno
8	NC	No Connection	nezapojeno

Standard LXI definuje tři základní funkční třídy přístrojů (údaje o třídách z [5]):

**a) Funkční třída C**

Základní funkční třída, která ale patří k nejrozšířenějším mezi přístroji (zahrnuje i programovatelný napájecí zdroj Sorensen XEL 30-3DP). Zařízení z této skupiny poskytují standardizované LAN rozhraní, internetový prohlížeč (konformní se standardem LXI), VXI-11 pro nalezení zařízení a IVI (*Interchangeable Virtual Instrument*) ovladače. Tato zařízení nemusí podporovat IEEE 1588 (PTP - *Precious Time Protocol*). Přístroje z této třídy jsou vhodné zejména pro aplikace, ve kterých jsou použity i přístroje bez podpory LXI. Tato třída může rovněž zahrnovat i malé produkty (např. senzory), které jsou napájeny pomocí baterií nebo využívají PoE (*Power of Ethernet*).

**b) Funkční třída B**

Již méně rozšířená funkční třída, která navazuje na třídu C. Přidává ke všem vlastnostem funkční třídy C navíc podporu IEEE 1588 a synchronizaci API (Application Programming Interface).

**c) Funkční třída A**

Snad nejméně rozšířená funkční třída přístrojů. Navazuje na třídu B, ale přidává k ní navíc ještě synchronizační a spouštěcí signály tzv. triggerů.

LXI standard tedy vyžaduje, aby každé zařízení s LXI mělo IVI ovladač, který je nutný pro podporu VISA (*Virtual Instrument Software Architecture*) funkcí.

Ke každému přístroji je nutné umožnit přístup z internetového prohlížeče, který poskytuje základní informace o přístroji (např. typ přístroje, funkční třídu...). Internetové rozhraní je povinné poskytovat IP konfigurační stránku (každý přístroj musí mít nutně svou vlastní IP adresu) a také synchronizační stránku (pouze u přístrojů funkční třídy B nebo A). Přístup k webovým stránkám přístroje může být chráněn heslem.

Všechna zařízení LXI musí podporovat Ethernet. Přenosová rychlost se tedy odvíjí podle použitého typu Ethernetu (viz Tab. 7).

**Tab. 7: Příkladné typy Ethernetů pro LXI; údaje z [6].**

Standard	Název	Přenosová rychlost	Druh připojení
IEEE 802.3u	Fast Ethernet (100Base-T)	100 Mb/s	UTP kabel
IEEE 802.3ab	Gigabit Ethernet (1000Base-T)	1 Gb/s	UTP kabel
IEEE 802.11b	Wi-fi	11 Mb/s	bezdrátově

Jelikož standard uvádí podporu Ethernetů vyšších než 100Base-T [5], mohl by standard LXI podporovat i 10 Gigabitový Ethernet, který má být představen v půlce roku 2012. Tento Ethernet má dosahovat přenosové rychlosti až 10 Gb/s. Zatím je to ovšem pouze spekulace. Maximální délka kabelu UTP může dosahovat až 100 m [6]. Dosah Wi-fi je závislý na kvalitě použitého vysílače a přijímače.

Komunikace mezi moduly LXI systému probíhá pomocí zpráv. Tyto zprávy jsou nazývány datové pakety. Formát datového paketu je zobrazen na Obr. 14.

HW Detect	Domain	Event ID	Sequence	Epoch	Flags	Data Fields	0 (2 byty)
-----------	--------	----------	----------	-------	-------	-------------	------------

**Obr. 14** Formát datového paketu LXI 1.2c; upraveno z [5].

a) **HW Detect** (*detekce hardwaru*) je prvním polem datového paketu LXI. Jeho délka jsou tři byty (oktety). Používá se jako “magická hodnota“ pro identifikaci platných paketů. Obsah tohoto pole by měl být nastaven na hodnotu “LXI“, jinak může být paket ignorován.

b) **Domain** (*doména*) se považuje za neoznačený byt. Výchozí hodnota tohoto pole je nulová a délka je jeden byt.

c) **Event ID** (*identifikátor události*) je pole, které obsahuje prvních 16 bytů ASCII řetězce (názevu události). Řetězce, které jsou delší než 16 bytů (ASCII znaků), jsou zkráceny na prvních 16 bytů (ASCII znaků). Pokud je řetězec kratší než 16 bytů, nevyužité byty jsou nulové.

d) **Sequence** (*pořadí*) je pole o délce čtyř bytů. Obsahuje pořadové číslo paketu, které se zvyšuje každým odesláním paketu přesně o jednu hodnotu. Počáteční hodnota tohoto pole není definována standardem, ale je ponechána na dodavateli zařízení. Pokud jsou některé pakety přenášeny znovu pro zvýšení spolehlivosti, tyto pakety pak obsahují stejná pořadová čísla jako originální paket.

e) **Epoch** (*epocha*) je nastavena na hodnotu nula, jelikož funkční třída C nepodporuje IEEE 1588. Délka tohoto pole je stanovena na dva byty.

f) **Flags** (*příznaky*) je pole o délce dvou bytů, které obsahuje údaje o paketu.

g) **Data Fields** (*datové pole*) obsahuje vlastní odesílaná data. Délka datového pole není specifikována standardem. Délkově proměnné datové pole je navrženo tak, aby splnilo tyto dvě různé potřeby. Za prvé to dovolí konsorciu LXI definovat nová datová pole, která se mohou stát součástí specifikace LXI a za druhé umožňuje prodejci definovat jejich vlastní datová pole.

Datový paket končí, když se **nula** (2 byty) potká v délce dalšího pole nebo když je dosaženo maximálního limitu datového zatížení (údaje o polích datového paketu z [5]).

Zařízení LXI musí podporovat protokol TCP/IP, minimálně IPv4. Zařízení s tímto rozhraním lze ovládat a komunikovat s nimi pomocí libovolné vyšší úrovně protokolů (např. RPC - *Remote procedure call*). Nízkoúrovňové protokoly nižší než TCP/IP mohou být použity jen pro jiné aplikace.

## 2 Technické prostředky pro realizaci práce

Tato kapitola se věnuje dvěma přístrojům, které byli použity pro všechna měření v této práci. Prvním z nich je samotný napájecí zdroj Sorensen XEL 30-3DP a druhým je digitální multimetr Agilent 34401A.

### 2.1 Napájecí zdroj Sorensen XEL 30-3DP

Programovatelný napájecí zdroj je výrobkem americké firmy Ametek, která je předním výrobcem elektronických přístrojů a elektromechanických zařízení.

#### 2.1.1 Základní vlastnosti zdroje

Zdroj Sorensen XEL 30-3DP (viz Obr. 15) je duální, což znamená, že dva zdroje jsou vedle sebe spojeny v jeden fyzický. Rozsah napětí na každém ze zdrojů je 0-30 V, přičemž v případě ovládání z čelního panelu je minimální krok změny napětí 10 mV. Čtyřmístný displej zobrazuje nastavené hodnoty se stejnou přesností jako je minimální krok změny napětí. Jelikož je zdroj programovatelný, lze jej tedy dálkově ovládat pomocí různých přístrojových rozhraní (viz podkapitola 2.1.2), získává se tím také menší minimální krok změny napětí než je hodnota pro analogové ovládání. Při dálkovém ovládání je tento krok zmenšen na hodnotu 1 mV a při zobrazení této hodnoty na fyzickém displeji přístroje je tato hodnota zaokrouhlována (např. vzdáleně nastavená hodnota 21,345 V se na displeji přístroje zobrazí jako 21,35 V). Přesnost je stanovena na  $\pm 0,05\% + 5 \text{ mV}$  [7]. Samozřejmostí jsou na čelním panelu analogové výstupy napětí, které byly použity také k měření stability napětí na zdroji spolu v kombinaci s digitálním multimetrem Agilent 34401A (více viz podkapitola 2.2).



**Obr. 15** Čelní panel programovatelného napájecího zdroje Sorensen XEL 30-3DP.

Tento zdroj má svou přepěťovou a přeprroudovou ochranu (tzv. OVP a OCP), která je při analogovém napětí nastavena na 105% max. rozsahu přístroje. Mohou být ovšem nastaveny jen přes dálkové ovládání. Je nutné také zmínit, že ačkoliv je zdroj duální, každý výstup je nezávislý a izolovaný. Proto je lze ovládat každý zvlášť a vzdáleně i oba zároveň. Přístrojový IVI ovladač, který poskytuje podporu pro běžné aplikace jako je např. LabVIEW nebo LabWindows, je dodáván spolu se zdrojem.

## 2.1.2 Provoz vzdáleného ovládání

Přístroj lze vzdáleně ovládat přes RS232, USB, GPIB a LAN přístrojová rozhraní. Všechna tato rozhraní jsou umístěna na zadním panelu přístroje (viz Obr. 16). Přístroj je dodáván standardně s RS232, USB a LAN, GPIB je volitelné. Manuál přístroje [7] udává, že doba náběhu napětí na výstupy je menší než 15 ms.

Programovací rychlost napětí je stanovena následovně (hodnoty nezahrnují dobu zpracování příkazu). Při zvyšování napětí je přibližně rovna 45 ms při plném zatížení a bez zatížení 40 ms. Naopak při snižování napětí jsou tyto hodnoty rovny 20 ms při plném zatížení a 150 ms bez zatížení. Doba zpracování příkazu (od přijmutí příkazu přístrojem po dobu, kdy se začne napětí na přístroji měnit) je obvykle nižší než 25 ms.



**Obr. 16** Zadní panel programovatelného napájecího zdroje Sorensen XEL 30-3DP.

### Zamykání rozhraní

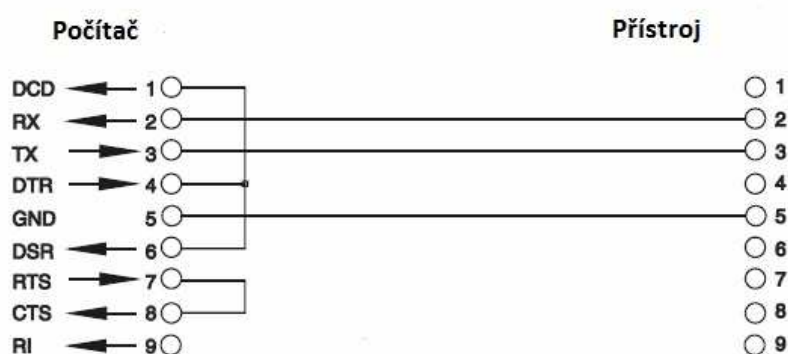
Všechna rozhraní jsou po celou dobu aktivní, což odstraňuje nutnost vybrání aktivního rozhraní. To je také jedním z požadavků standardu LXI. Je-li nutné snížit riziko neúmyslného ovládání přes dvě rozhraní najednou, existuje jednoduchý zámek. Přístup k rozhraní může být také omezen použitím přístupu přes webové rozhraní.

### Vzdálená/místní obsluha

Po zapnutí přístroje je zdroj v lokálním stavu (s *REMOTE* indikátorem vypnutým). V tomto stavu je možné provádět všechny operace na čelním panelu. Začne-li se přístroj řídit vzdáleně, tak v okamžiku, kdy je příkaz přijat, přepne se do vzdáleného stavu a indikátor *REMOTE* se zapne. V tomto stavu je čelní panel uzamčen a zdroj zpracovává jen vzdálené příkazy. Přístroj může být vrácen do lokálního stavu stisknutím tlačítka *LOCAL* (vliv této akce zůstane pouze po dobu, dokud není přístroj znovu osloven nebo nepřijme jiný znak z rozhraní), potom vstoupí opět do vzdáleného stavu. Návrat do lokálního stavu zachovává poslední vzdáleně nastavené hodnoty napětí (z [7]).

### Rozhraní RS232

Na zadním panelu zdroje (viz Obr. 16) je umístěn 9 pinový konektor typu D. Počítač může být připojen k přístroji pomocí plně obsazeného kabelu, nebo lze zapojit jen piny číslo 2, 3 a 5. V tomto případě zapojení je nutné provést další úpravy (viz Obr. 17).



**Obr. 17** Schéma propojení kabelu RS232 se zapojením pouze tří pinů; upraveno z [7].

Přenosová rychlost tohoto rozhraní je stanovena na 9600 b/s, další parametry jsou stanoveny následovně [7]:

Start bity: 1

Parita: žádná

Datové bity: 8

Stop bity: 1

Přístroj používá metodu programového řízení XON/XOFF (více o této metodě viz podkapitola 1.1). Jednotka vyšle signál XOFF když je počet zbývajících bytů 50 a naopak signál XON, když se tento počet zvýší na 100. Při této metodě je možné posílat data kódované jen v ASCII.

Jelikož používaný počítač (Acer Aspire 5920G), na kterém byla měření prováděna, již není vybaven sériovým rozhraním RS232, bylo nutné použít redukci RS232-USB. Tato redukce je znázorněna na Obr. 18.



**Obr. 18** Redukce RS232-USB.

### **Rozhraní USB**

Rozhraní USB 2.0 (režim *full-speed*) je virtuální port COM, který může být řízen počítačem a chová se jako by byl řízen přes rozhraní RS232. Virtuální port COM může běžet na Windows aplikacích přesně stejným způsobem jako standardní port COM, kromě toho, že je ignorováno nastavení přenosové rychlosti COM portu.

### **Sběrnice GPIB**

Před zahájením komunikace s přístrojem pomocí GPIB je nutné nejprve nastavit adresu přístroje. Nastavení adresy před první komunikací je popsáno v manuálu [7].

Rozhraní GPIB není standardním vybavením osobních počítačů, proto byla při komunikaci v experimentální soustavě opět použita redukce, tentokrát však GPIB-USB-HS od firmy National Instruments (viz Obr. 19). Tato redukce podporuje i rychlejší přenosový cyklus HS488.



**Obr. 19** Redukce GPIB-USB-HS.



## Rozhraní LAN

Rozhraní LAN je navrženo v souladu se standardní LXI verzí 1.2 funkční třídy C. Je nutné znát IP adresu přístroje. Pro zobrazení IP adresy je možné použít program *LXI Discovery Tool*, který je dodán na příloženém CD. Při prvním spuštění se zdroj pokusí získat nastavení přes DHCP, pokud je k dispozici. Pokud není a vyprší časový limit (30 sekund), tak přes funkci Auto-IP. Ve velmi nepravděpodobném případě, že funkce Auto-IP nemůže nalézt adresu, přiřadí se zdroji statická IP adresa 192.168.0.100. Pokud není přístroj připojen do sítě LAN, začne blikat na displeji hlášení *LAN Error*.

Dálkové ovládání pomocí rozhraní LAN je tedy prováděno pomocí protokolu TCP/IP. Tento přístroj také obsahuje základní webový server, který poskytuje informace o přístroji a umožňuje nakonfigurovat některé parametry rovnou z webového prohlížeče. Konfigurační stránka může být chráněna heslem o délce až 15 znaků, výchozí nastavení stránky je bez hesla. Uživatelské jméno by mělo být ponecháno prázdné. Stisknutím tlačítka *LAN RESET* na zadním panelu přístroje se změní všechny uživatelem nastavené parametry na tovární nastavení přístroje.

Dalším důležitým krokem je změna tzv. VISA názvu zdroje. Napájecímu zdroji byla v našem případě přiřazena IP adresa 158.194.68.30. Formát VISA názvu zdroje s touto IP adresou je obvykle ve tvaru "TCPIP::158.194.68.30::inst0::INSTR". Tento VISA název zdroje však musí být upraven do následujícího formátu "TCPIP::158.194.68.30::9221::SOCKET", kde "9221" je TCP port. Tento TCP port slouží k řízení a monitorování zdroje.

### 2.1.3 Vzdálené příkazy

Formát příkazů se nerozlišuje dle typu rozhraní, kdy všechna rozhraní mají stejný formát příkazu. Souhrn všech příkazů je uveden v manuálu [7] a každý příkaz musí být ukončen tzv. řádkovací konstantou (*Line Feed, LF*). V Tab. 8 jsou uvedeny příkazy, které byli použity v aplikacích pro měření v této práci.

**Tab. 8:** Použité příkazy.

Formát příkazu	Popis příkazu
V<n> <nrf>	nastavuje napětí (hodnota <nrf>) na výstupech (<n> 1 nebo 2)
V<n>?	dotaz na napětí na výstupech (<n> 1 nebo 2)
OPALL <nrf>	zapíná (<nrf> je 1) nebo vypíná (<nrf> je 0) napěťové výstupy

## 2.2 Digitální multimetr Agilent 34401A

Model digitálního multimetru Agilent 34401A (viz Obr. 20) je 6 ½ číselný, vysoce výkonný digitální multimetr. Jeho kombinace systémových funkcí dělá z tohoto multimetru univerzální řešení pro měření. Rozsah digitálního multimetru je možné nastavit při měření stejnosměrného napětí od 100 mV do 1 kV. Přesnost měření udává manuál [8] při zvoleném rozsahu do 100 V jako  $\pm 0,002\%$  (z přečtené hodnoty) + 0,0006% (z rozsahu).



Obr. 20 Čelní panel digitálního multimetru Agilent 34401A.

Tento digitální multimetr je vybaven dvěma přístrojovými rozhraními RS232 a GPIB (viz Obr. 21), pomocí nichž lze s přístrojem komunikovat. V níže uvedených aplikacích bylo s digitálním multimetrem komunikováno pomocí rozhraní GPIB. Čtecí rychlost dat z přístroje může dosahovat až 1000 hodnot za sekundu a přístroj si může uložit až 512 hodnot do své interní paměti.



Obr. 21 Zadní panel digitálního multimetru Agilent 34401A.

Obvykle je na vstupech digitálního multimetru nastaven vstupní odpor na hodnotu 10 M $\Omega$  pro všechna stejnosměrná napětí aby se minimalizoval šum. Za účelem snížení chyby

měření je možné nastavit větší vstupní hodnotu odporu na 10 GΩ. Tohoto zvýšení vstupního odporu lze dosáhnout za pomoci následujícího příkazu “*INP:IMP:AUTO {OFF/ON}*“ (hodnota *OFF* je výchozí hodnota pro 10 MΩ a hodnota *ON* slouží k nastavení vstupního odporu na 10 GΩ).

Dalším důležitým nastavením přístroje před zahájením měření je nastavení rozlišení a rozsahu měřeného napětí. Rozlišení je možno nastavit na 4, 5 nebo 6 celých čísel a ½ čísla, která může být jen “0“ nebo “1“. Pro zvýšení přesnosti měření a potlačení šumu je vhodné zvolit 6½ čísel. Pro zrychlení měření je pak vhodnější nastavení 4½ čísel. Při měřeních uváděných níže bylo nastaveno rozlišení 5½ čísla a rozsah do 100 V. Toto nastavení bylo plně dostačující pro uskutečnění měření. Příkaz, pomocí kterého bylo nastavení provedeno je “*CONF:VOLT:DC 100,0.001*“.

Před započítáním každého nového měření (např. nového cyklu měření) je vhodné vynulovat naměřené hodnoty v přístroji. K tomuto úkonu je příkaz “*CALC:FUNC NULL*“.

Pro vyčtení naměřených dat z bufferu (vyrovnávací paměti) existují dva druhy příkazů. Prvních z nich je příkaz “*MEAS?*“, jelikož při tomto druhu příkazu si digitální multimetr sám stanovuje rozlišení a rozsah nebyl tento příkaz použit. Druhým příkazem, který lze kombinovat s jinými typy příkazů, je příkaz “*READ?*“. Po odeslání tohoto příkazu jsou naměřené hodnoty okamžitě odeslány k výstupu bufferu a je nutné zadat čtení z bufferu bezprostředně hned po odeslání příkazu. Pokud se tak nestane, digitální multimetr může zastavit měření, jelikož bude buffer plný. Naměřené hodnoty se neukládají do interní paměti multimetru. Nejjednodušším způsobem jak lze získat naměřená data z digitálního multimetru je za pomoci příkazu “*MEAS?*“. Nicméně tento příkaz je méně flexibilní a pokud jsou kladeny vyšší nároky na naměřené hodnoty je výhodnější použít příkaz “*READ?*“.

## 3 Vývojové programovací prostředí LabVIEW

LabVIEW (*Laboratory Virtual Instrument Engineering Workbench*) je vyvíjeno firmou National Instruments, v původní verzi jen pro počítače od firmy Apple, až v nejnovější verzi LabVIEW 2011, která je dostupná pro klasické počítače IBM-PC. Zlomový byl rok 1992, ve kterém byla k dispozici první verze právě pro počítače IBM-PC.

Hlavní myšlenkou tohoto programovacího prostředí jsou tzv. virtuální přístroje (*Virtual Instruments*). Samotný VI se skládá ze tří základních částí:

- 1) Blokovaný diagram - v této části uživatel definuje svoje vlastní algoritmy pomocí uzlů (*nodes*) a spojů (*wires*). Uzly jsou ve většině případů funkce (*functions*), ale mohou to být i podprogramy (*subVIs*). Spojové se dají definovat jako proměnné nebo konstanty v jiných programovacích jazycích.
- 2) Čelní panel - uživatelské rozhraní daného VI. Obsahuje ovládací a indikační prvky pomocí nichž může uživatel nejen ovlivňovat parametry měření, ale také kontrolovat jeho správný průběh.
- 3) Konektory - piny, které slouží k volání podprogramů na nižších úrovních.

Celý systém prostředí je orientován graficky, proto se typ tohoto programovacího jazyka někdy označuje jako G-jazyk. Programování v tomto jazyce je hodně odlišné od klasického psaní zdrojového kódu.

*„Zásadní rozdíl je v tom, že program (VI) je řízen tokem dat a nikoli tokem příkazů. Předepsání určité posloupnosti provádění operací, je nutné použít v blokovém diagramu zvláštních konstrukcí (objektů). Pokud se tak neučiní, jsou operace aktivovány hned, jak mají k dispozici na všech svých vstupech data.“* převzato z[9]

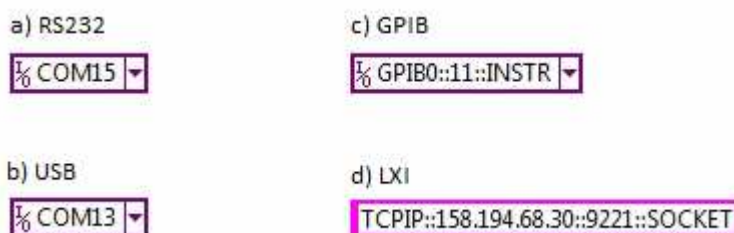
A jelikož lze pomocí LabVIEW snadno programovat systémy, analyzovat signály a řídit měřicí přístroje, bylo toto vývojové prostředí vybráno jako nejvhodnější k řízení napájecího zdroje Sorensen XEL 30-3DP.

### 3.1 Společné prvky programů

Všechny programy popsané v dalších kapitolách mají v blokovém diagramu poměrně hodně společných funkcí a prvků, které se týkají především správného chodu programu a ovládání zdroje pomocí příkazů. Jelikož by bylo zbytečné je popisovat v každé kapitole zvlášť, je tato kapitola shrnutím všech jejich společných vlastností.

### 3.1.1 VISA název zdroje

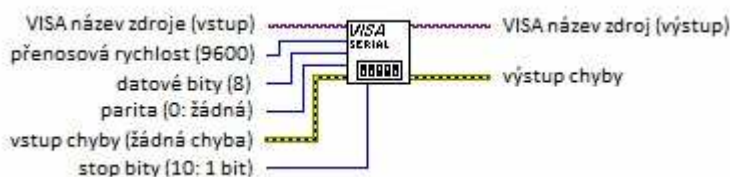
Nepochybně nejdůležitějším prvkem každého programu je parametr VISA název zdroje. Slouží k specifikaci zařízení, která budou v programu používány. VISA název zdroje (výstup) je kopie názvu zdroje, která je předávána do dalších funkcí. Předáváním názvu zdroje ven a do VISA funkcí, se dá zjednodušit programování toku dat řetězením VISA funkcí. K popsání VISA názvu zdroje lze také využít textový řetězec v F-formátu string. Jednotlivé VISA názvy zdrojů pro různá přístrojová rozhraní jsou na Obr. 22.



**Obr. 22** Názvy VISA zdrojů pro čtyři různá rozhraní.

### 3.1.2 Inicializace sériového portu

Inicializace sériového portu je nutná při komunikaci se zdrojem přes rozhraní RS232 a USB, kdy funkce nastaví sériový port dle názvu zdroje na specifické nastavení. Na Obr. 23 je znázorněna funkce s důležitými vstupy a výstupy, kterou lze najít v paletě funkcí pro sériovou komunikaci. Hodnoty v závorkách jsou standardně nastavené prostředím a jsou totožná s hodnotami uvedenými v manuále, proto je není potřeba měnit.

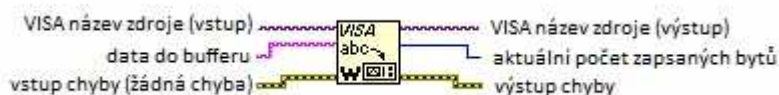


**Obr. 23** Inicializace sériového portu.

Funkce obvykle je volána jen jednou na počátku měření nebo před začátkem ovládání zařízení. Teoreticky by se však mohla provést při každé změně nastavení zařízení, ale v tomto případě by zbytečně zdržovala chod programu.

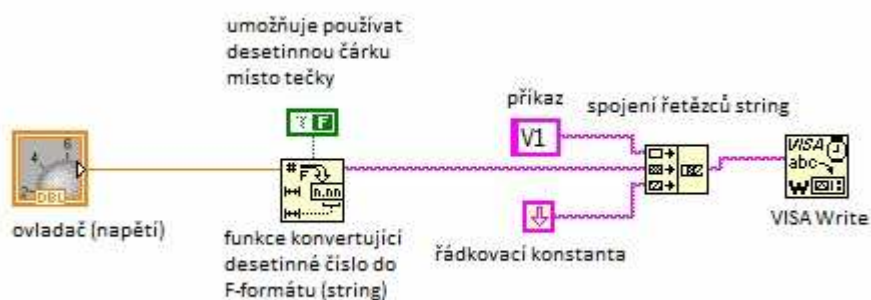
### 3.1.3 Zadání příkazu

Zasílání příkazů zdroji je prováděno pomocí funkce VISA Write z palety *VISA VIs and functions*. Všechny výstupy a vstupy této funkce jsou znázorněny na Obr. 24. Příkazy, pomocí nichž lze zdroj ovládat, jsou stejné pro všechny druhy rozhraní.



Obr. 24 Funkce VISA Write.

Datový typ vodiče pro příkazy je *string*. V podkapitole 2.1.3 Vzdálené příkazy zobrazené v Tab. 8 jsou rozlišovány 3 druhy příkazů. Pro každý druh příkazů je také jiná struktura příkazu. Jednotlivé struktury příkazů jsou znázorněny na Obr. 25-27.



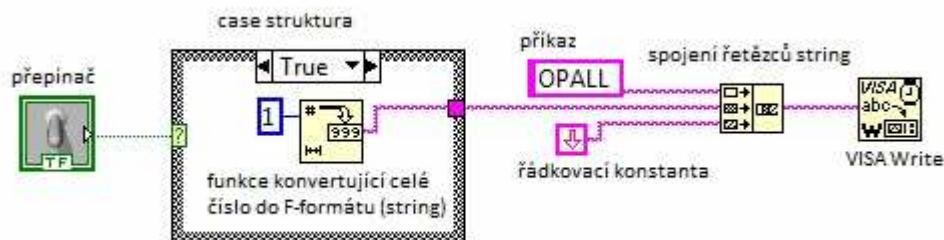
Obr. 25 Struktura příkazu pro nastavení napětí.

Takto složený příkaz (viz Obr. 25) pro nastavení napětí na zdroji master nemusí nutně nastavovat jen napětí, ale také rozsah proudu nebo jakoukoliv hodnotu, která má podobnou strukturu příkazu jako pro nastavení napětí. Samozřejmě se po tomto druhu příkazu neočekává odpověď, ale provedení změny hodnoty na zdroji.



Obr. 26 Struktura příkazu pro dotaz na napětí.

Po odeslání druhu příkazu z Obr. 26 uživatel očekává od zdroje odpověď. O odpovědích přístroje je uvedeno více v kapitole 3.1.4 Odpovědi přístroje.

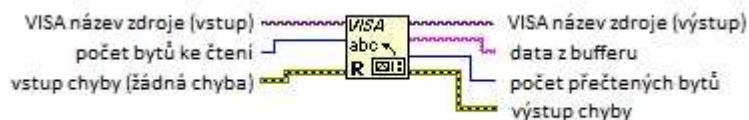


**Obr. 27** Struktura příkazu pro vypnutí/zapnutí napěťových výstupů zdroje.

Posledním druhem příkazů jsou ty, které pracují s logickými hodnotami 1 a 0 v souvislosti se zapnutím nebo vypnutím např. výstupů napětí zdroje. Příkaz na Obr. 27 funguje tak, že při změně polohy přepínače dojde ke změně rámce v Case struktuře (rozvětvení kódu), ve které je v případě rámce True nastavena logická jednička, čímž se oba výstupy napětí zapnou. V rámci False, který již není na obrázku znázorněn je logická nula a při jeho provedení se výstupy napětí vypnou. Alternativou takto strukturovaného příkazu může být samozřejmě i oddělené ovládání napěťových výstupů zdroje.

### 3.1.4 Odpovědi přístroje

Po zadání jakéhokoliv příkazu ve tvaru xxx? (např. V1?) se očekává odpověď od přístroje. Tato odpověď se vyčte z bufferu za pomoci funkce VISA Read z palety VISA VIs and functions znázorněné na Obr. 28.



**Obr. 28** Funkce VISA Read.

Požadovaná odpověď od přístroje se vyčte z bufferu v F-formátu (string). Ve většině případů je však nutné s těmito daty dále pracovat. Ať již ve stejném formátu pomocí funkcí z palety *String* a nebo převést tento formát na jiný, např. na číslo.

### 3.1.5 Vyčištění bufferu

Rozhraní LXI vyžaduje po každém provedení příkazu vyčistit buffer. Tento proces provádí funkce VISA Clear (viz Obr. 29).

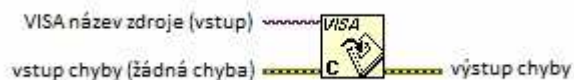


Obr. 29 Funkce VISA Clear.

Uvedenou funkci lze nalézt v paletě *VISA VIs and functions* a provádí vymazání vstupní a výstupní vyrovnávací paměti zařízení.

### 3.1.6 Uzavření VISA

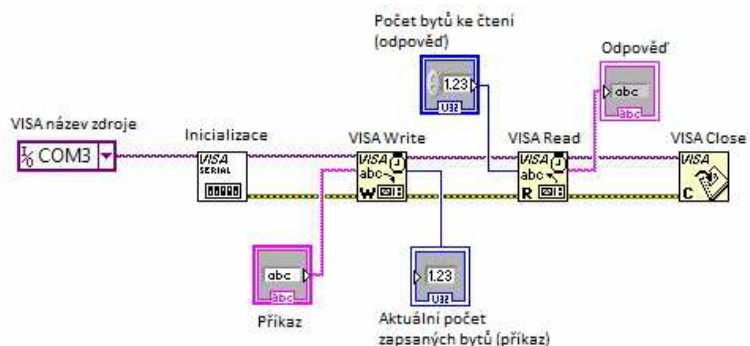
Po dokončení měření je při komunikaci přes rozhraní USB a RS232 nutné uzavřít používané zařízení specifikovaném VISA názvem zdroje. Funkce VISA Close (Obr. 30) je umístěna opět v paletě funkcí pro sériovou komunikaci.



Obr. 30 Funkce VISA Close.

### 3.1.7 Struktura VISA řetězce

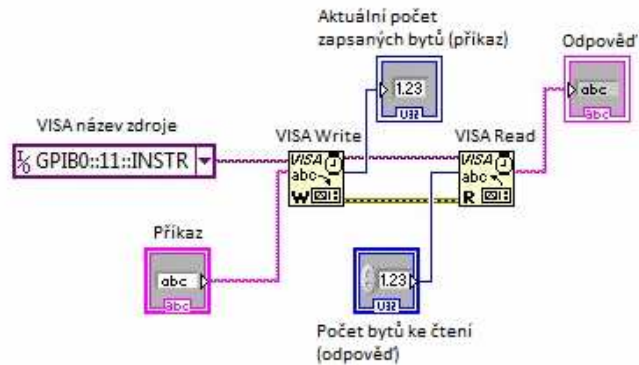
Tato část je shrnutím všech předchozích částí 3.1.1 - 3.1.6 do jedné, která popisuje pomocí Obr. 31-33 logicky správné uspořádání VISA funkcí v řetězci s upozorněním na menší rozdíly při komunikaci pomocí různých rozhraní. Uvedené řetězce jsou pouze ilustrační, za účelem nástinu struktury pro dané rozhraní, tudíž nejsou plně funkční.



Obr. 31 Struktura VISA řetězce pro komunikaci přes přístrojová rozhraní USB a RS232.

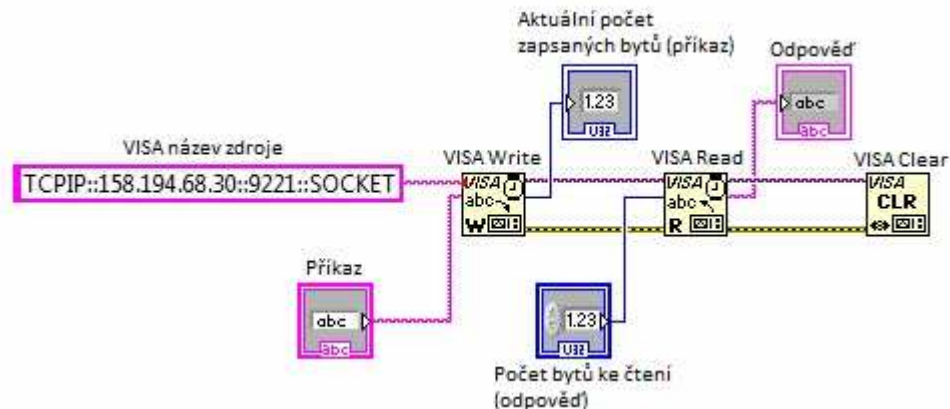


Tato struktura se vyskytuje v každém programu, který předpokládá komunikaci přes přístrojové rozhraní USB nebo RS232. Toho je dosaženo záměnou jediného prvku (VISA název zdroje). Přesné VISA názvy pro jednotlivá rozhraní jsou uvedena v části 3.1.1 VISA název zdroje.



**Obr. 32** Struktura VISA řetězce pro komunikaci přes přístrojové rozhraní GPIB.

A pro úplné srovnání je na Obr. 33 zobrazena struktura VISA řetězce pro LXI, jelikož má od všech ostatních rozhraní jednu další odlišnost, kterou je funkce VISA Clear.



**Obr. 33** Struktura VISA řetězce pro komunikaci přes přístrojové rozhraní LXI.

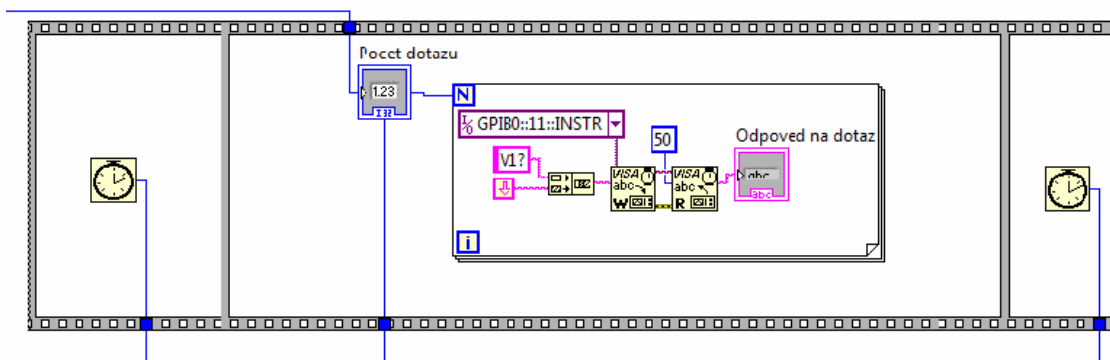
### 3.2 Programy pro měření času odezev přístroje

Jednotlivé programy byly vytvořeny za účelem proměření času odezev zdroje přes všechna rozhraní, pomocí nichž je zdroj schopen komunikovat s řídicím počítačem. Pro každé rozhraní (RS232, USB, GPIB a LXI) bylo vytvořeno 6 různých programů, které se od sebe liší v počtu a druhu příkazů a dále tím, jestli měření probíhá při ovládní pouze jednoho zdroje nebo obou zdrojů zároveň.

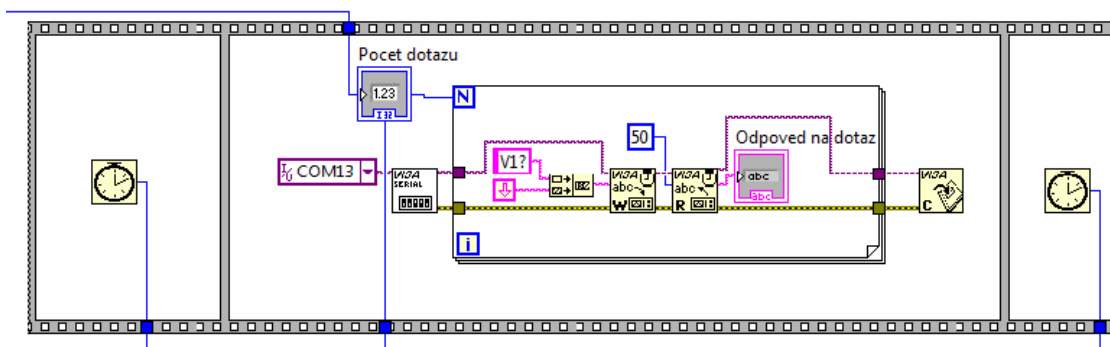
Střední část kódu je jediný článkem programu, který odlišuje všechny vytvořené programy. Pro každé rozhraní byly vytvořeny následující programy:

- program pro měření času odezvy na dotaz na aktuální hodnotu napětí,
- program pro měření času odezvy nastavení napětí,
- program pro měření času odezvy nastavení a dotazu na napětí.

Diskutovanou část kódu tvoří Flat struktura se třemi po sobě jdoucími rámci (Obr. 34). V prvním rámci je odečten počáteční čas měření za pomoci funkce Tick count. Druhý rámec obsahuje For smyčku, ve které je vnořen systém příkazů. Pokud měření bude probíhat přes GPIB nebo LXI, je v této smyčce obsažen také název zdroje. Pokud ovšem měření probíhá přes USB nebo RS232 název zdroje společně s inicializací musí být umístěn mimo tuto smyčku (viz Obr. 35). Tato smyčka je řízena jednorozměrným polem, které je umístěno mimo všechny struktury. Pole obsahuje celkem 5 členů o hodnotách 1, 10, 100, 1000 a 10000. Určuje tak počet provedených příkazů. Realizace každého z použitých příkazů a struktura VISA řetězce je podrobně popsána v podkapitole 3.1.3 Zadání příkazů. Třetí a zároveň i poslední rámec Flat struktury je obdobný jako první rámec s tím rozdílem, že je zde odečten konečný čas měření.

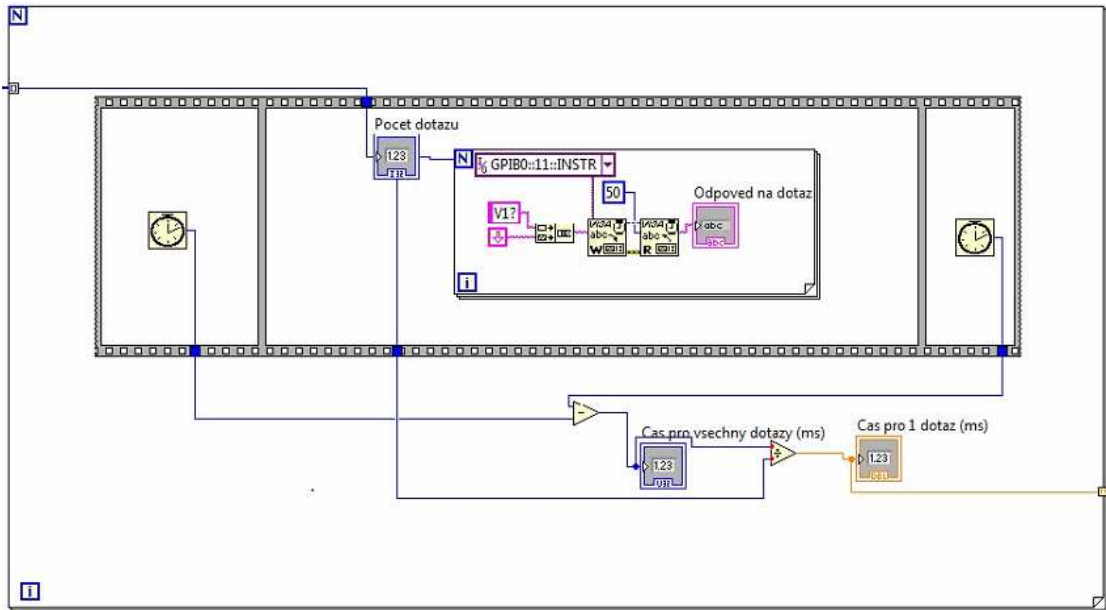


Obr. 34 Flat struktura pro měření času odezvy dotazu na napětí přes GPIB.

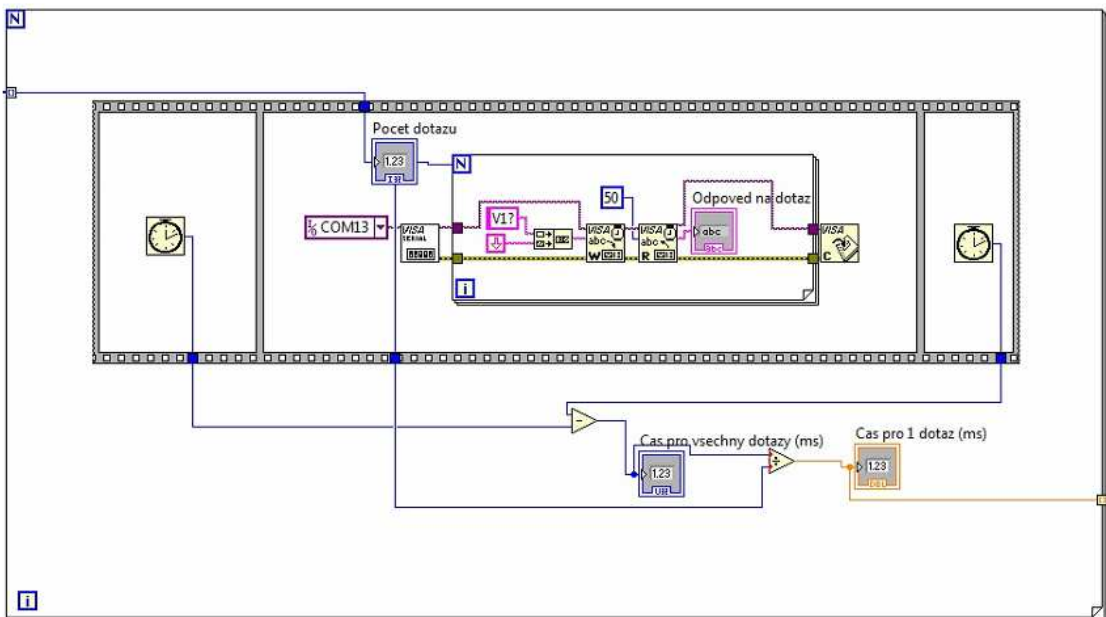


Obr. 35 Flat struktura pro měření času odezvy dotazu na napětí přes USB.

Aby celý program fungoval správně, musí být tato Flat struktura vnořena do další For smyčky, která nemá sice zadaný počet cyklů, ale provede se pro každý cyklus měření právě tolikrát, kolik je zadáno hodnot v jednorozměrném poli. Dále jsou v ní umístěny triviální matematické funkce pro snadnější zpracování naměřených dat (viz Obr. 36 a Obr. 37). Těmito funkcemi jsou odčítání, pro vypočtení rozdílu doby měření a následné dělení počtem příkazů či dotazů k získání průměrné hodnoty času odezvy.

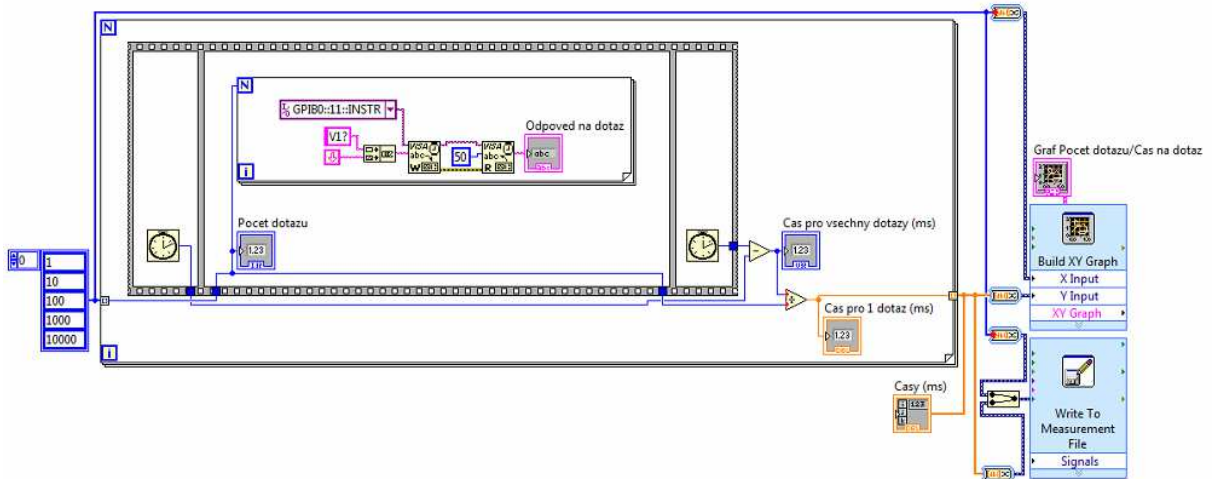


**Obr. 36** For smyčka pro měření času odezvy dotazu na napětí přes GPIB.

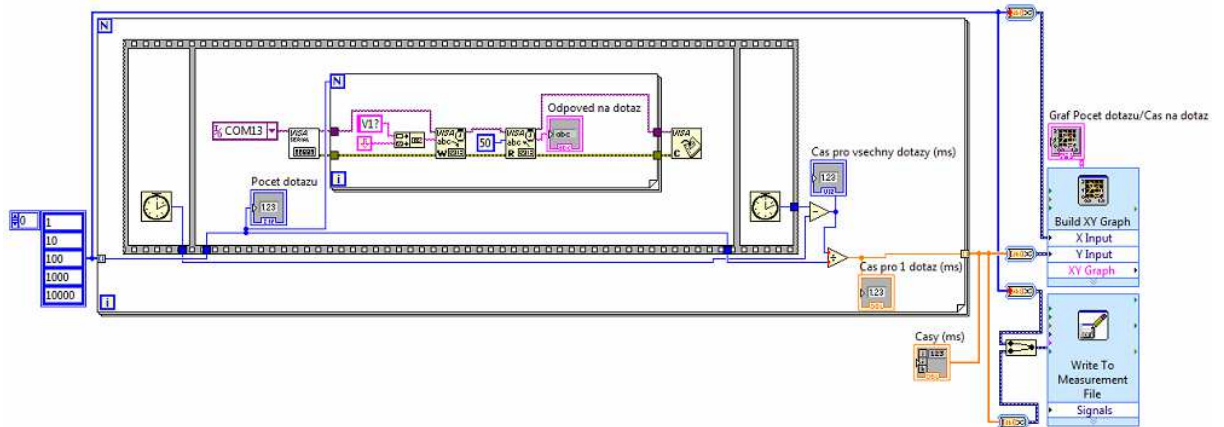


**Obr. 37** For smyčka pro měření času odezvy dotazu na napětí přes USB.

Ve vytvořeném programu se data nejen ukládají do souboru, který si uživatel sám zvolí po dokončení měření, ale data se také vykreslí do grafu a znázorní do jednorozměrného pole (až po provedení všech cyklů měření a matematických funkcí). Blokové diagramy již hotových programů jsou znázorněny na Obr. 38 a Obr. 39.



**Obr. 38** Blokový diagram programu pro měření času odezvy dotazu na napětí přes GPIB.

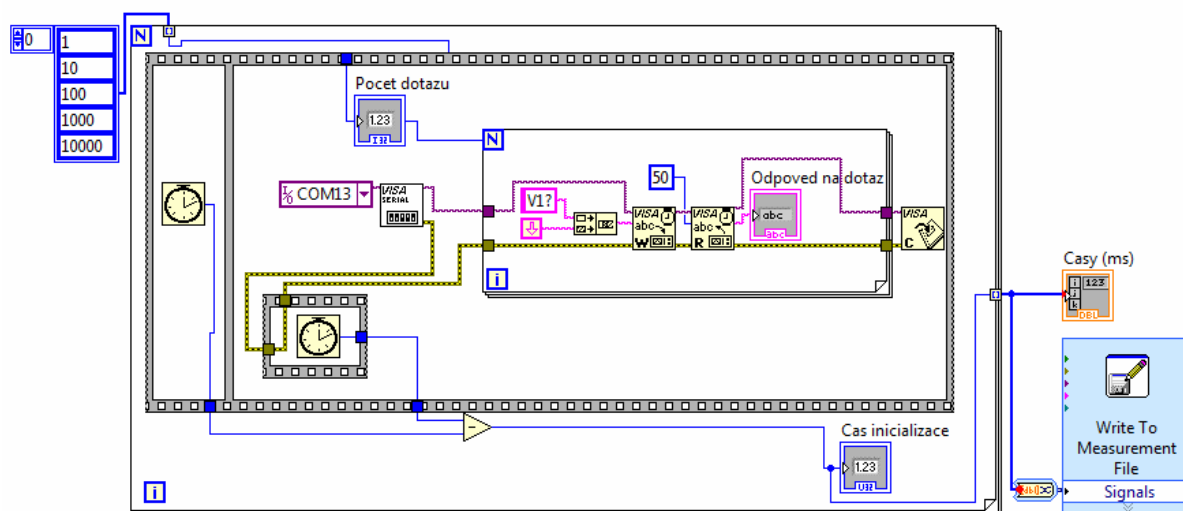


**Obr. 39** Blokový diagram programu pro měření času odezvy dotazu na napětí přes USB.

Pokud se změní v blokovém diagramu název zdroje z Obr. 38 (respektive Obr. 39) na název uvedený na Obr. 22d (respektive Obr. 22a), lze tyto programy použít pro měření vlastností přístrojových rozhraní LXI (respektive RS232). U rozhraní LXI je nutné přidat ještě funkci VISA Clear.

### 3.3 Programy pro měření doby inicializace sériového portu

Tyto programy se nijak v zásadě neliší od programů pro měření časů odezvy přístroje. V podstatě jsou to stejné blokové diagramy s několika malými změnami. Byla přidána navíc jedna Flat struktura s funkcí Tick count a bylo změněno pořadí provádění jednotlivých operací v kódu pomocí chybových spojů (*error*). Třetí rámeček původní Flat struktury byl vymazán spolu s vyčítáním naměřených hodnot do grafu. Na počátku měření se odečte čas, provede se inicializace sériového portu a vzápětí se opět provede odečet času. Potom se začnou provádět jednotlivé příkazy. Tento cyklus se opakuje dokud se neuskuteční požadovaný počet měření. V tomto případě je měřena inicializace celkem pětkrát a jelikož už není potřeba dělit počtem příkazů (inicializace se provede vždy jednou), byly zredukovány i matematické operace na jedinou - rozdíl naměřených časů. Tyto hodnoty se pak zapíší do jednorozměrného pole a po skončení měření jsou uloženy do uživatelem zvoleného souboru. Celý blokový diagram je znázorněn na Obr. 40.



Obr. 40 Blokový diagram programu pro měření doby inicializace sériového portu (USB).

### 3.4 Program pro porovnání hodnot ze zdroje a hodnot naměřených digitálním multimetrem

Pro účel této aplikace byl použit nejen napájecí zdroj Sorensen XEL 30-3DP, ale i digitální multimetr Agilent 34401A. Napájecí zdroj byl připojen k počítači pomocí USB a digitální multimetr pomocí sběrnice GPIB. V blokovém diagramu (celý blokový diagram viz Příloha 1) byly aplikovány dva různé VISA názvy zdroje. V prvním kroku je

provedena inicializace sériového portu pro zahájení komunikace se zdrojem. Následuje první While smyčka, ve které je umístěno tlačítko *Start* pro spuštění měření a tlačítko *Exit* pro ukončení aplikace. Tato While smyčka obsahuje Case strukturu, která je řízena již zmiňovaným tlačítkem *Start*. Před stisknutím tlačítka *Start* si však uživatel musí zvolit vhodné parametry pro měření (časování celkové a digitálního multimetru, počáteční hodnotu napětí, konečnou hodnotu napětí, inkrement napětí a také jestli chce ukládat naměřená data nebo ne).

V Case struktuře jsou rozlišeny dva rámce. První rámec je False, při kterém program nic neprovádí. Druhým rámcem je True, v kterém v první řadě proběhne přednastavení některých parametrů na zdroji a digitálním multimetru. K nastavení těchto parametrů byly použity struktury příkazů z podkapitoly 3.1.3. Pořadí, ve kterém byly příkazy vykonány, určují chybové spoje (*error*). Prvním z těchto příkazů je zapnutí napěťových výstupů zdroje, druhým příkazem je nastavení rozsahu měřeného napětí digitálního multimetru a poslední příkaz slouží k nulování displeje digitálního multimetru. Tyto příkazy se za celou dobu měření provedou pouze jednou.

Nejdůležitějším článkem celého programu je však druhá While smyčka umístěná rovněž v Case struktuře (rámec True). Tato smyčka zajišťuje postupné nastavování napětí na zdroji, vyčítání hodnot ze zdroje i multimetru a rovněž uložení dat. Jako první se provede nastavení počáteční hodnoty napětí zadané uživatelem. Následuje dotaz na napětí s vyčtením odpovědi (i do grafu). Po vyčtení odpovědi je nutné nějakou dobu vyčkat, aby digitální multimetr stihl zareagovat na tuto změnu. Z tohoto důvodu byla do While smyčky vložena Flat struktura s dvěma po sobě jdoucími rámci. V prvním z těchto rámců je umístěna funkce VISA Clear za účelem vyčištění bufferu a rovněž funkce Wait, která určuje právě již zmiňovanou dobu čekání (tuto hodnotu rovněž nastavuje uživatel). Druhý rámec Flat struktury obsahuje samotný příkaz pro vyčtení naměřené hodnoty digitálním multimetrem společně s odpovědí.

Uložení dat je realizováno pomocí Case struktury, která je umístěna rovněž vně této While smyčky. V rámci True (pokud uživatel zvolí možnost uložení dat) je umístěna funkce "Write to measurement file", která zapisuje během měření jednotlivé hodnoty do předem zvoleného souboru. Těmito hodnotami je čas měření, napětí udávané zdrojem, napětí naměřené digitálním multimetrem a rozdíl těchto dvou napětí. Rámec False zůstává prázdný.

Cyklus se opakuje dokud se hodnota konečného napětí nastavená uživatelem nebude rovnat hodnotě vyčítané ze zdroje, přičemž po proběhnutí jednoho cyklu se hodnota napětí

na zdroji zvýší o inkrement, který nastaví uživatel na počátku měření. Druhou možností, jak zastavit tento cyklus je použitím tlačítka *Stop*.

Po dokončení cyklu měření a stisknutím tlačítka *Exit* se data vykreslí do grafu XY, kde X jsou hodnoty vyčtené ze zdroje a Y hodnoty naměřené multimetrem. Rovněž je uzavřena komunikace se zdrojem za pomoci funkce VISA Close.

### **3.5 Program pro měření dlouhodobé stability napájecího zdroje**

V podstatě se liší jen velmi málo od programu pro porovnání napětí ze zdroje a naměřeného napětí digitálním multimetrem z předchozí kapitoly 3.4. První zásadní změnou je, že napětí zůstává konstantní během celé doby měření. Proto je žádoucí jej nastavit pouze jednou na počátku měření. V blokovém diagramu znázorněném v Příloze 2 je tedy umístěn příkaz pro nastavení napětí na hodnotu zvolenou uživatelem a umístěn do první Case struktury, u které se rámec True provede při stisknutí tlačítka *Start*.

Druhou a zároveň poslední odlišností v blokovém diagramu je změna časování. V případě, že by zůstalo původní časování při předčasném ukončení měření stisknutím tlačítka *Stop*, program by čekal nastavenou dobu časování (např. 10 minut) na ukončení měření. Z tohoto důvodu byla do blokového diagramu vložena další Case struktura, která je řízena porovnáním rozdílu časů odečtených pomocí přidaných funkcí Tick count a nastaveného časování uživatelem. Jednotlivé akce se provádějí v sekundových intervalech (případně program čeká, pokud se všechny akce provedou rychleji). Měření může být tedy zastaveno v jakémkoliv okamžiku stisknutím tlačítka *Stop* nebo se ukončí po čase, který je nutno nastavit uživatelem před zahájením měření. Program rovněž ukládá naměřené hodnoty v průběhu měření.

## 4 Výsledky měření a jejich analýza

Prvním z hlavních cílů této bakalářské práce bylo proměření komunikačních parametrů jednotlivých přístrojových rozhraní. Všechna tato měření popisují podkapitoly 4.1 a 4.2. Druhým cílem bylo porovnání hodnot napětí vyčtených ze zdroje s hodnotami naměřených pomocí multifunkční měřicí karty. Po přehodnocení situace byla multifunkční měřicí karta nahrazena digitálním multimetrem Agilent 34401A, který je plně dostačující pro měření tohoto typu. Výsledky těchto měření uvádí podkapitola 4.3. Konečně poslední měření uvedené v podkapitole 4.4 se zabývá sledováním stability napájecího zdroje Sorensen XEL 30-3DP, ve kterém byl opět použit výše zmíněný digitální multimetr.

Měření v podkapitolách 4.1-4.3 byla prováděna na osobním počítači Acer Aspire 5920G (s procesorem Intel Core Duo 2 GHz, operační paměť RAM 4 GB a 64 bitovým operačním systémem Windows 7).

### 4.1 Měření komunikačních parametrů

Jednotlivé časy odezev byly naměřeny za pomoci programů vytvořených dle kapitoly 3.2. Jelikož tento zdroj je “dvojitý,” každá série měření byla provedeno dvakrát - jednou pouze pro jeden zdroj a podruhé pro oba zdroje zároveň a vždy byl počet přístupů nastaven tak, aby rostl logaritmičticky v rozmezí od 1 do 10 000. Měření byla uskutečněna pro rozhraní USB, GPIB, RS232 a LXI.

Cílem těchto měření bylo porovnat jednotlivá rozhraní z hlediska jejich přenosové rychlosti a porovnat je také s hodnotami spočítanými z teoretických přenosových rychlostí. Uvedené hodnoty v následujících Tab. 9-16 jsou již přepočteny na průměr, což lze ovšem vyčíst i z blokových diagramů programů uvedených v kapitole 3.2. Funkce Tick count měří čas v milisekundách, proto bylo nutné výsledky ještě zaokrouhlit na celá čísla.

Pro výpočet aritmetického průměru (1), přesnosti měření (2) a střední kvadratické chyby (3) byli použity následující vzorce:

$$(1). \bar{t} = \frac{\sum_{i=1}^n t_i}{n}$$

$$(2). \Delta t = \frac{\sum_{i=1}^n |\bar{t} - t_i|}{n}$$

$$(3). \bar{s} = \sqrt{\frac{\sum_{i=1}^n (\bar{t} - t_i)^2}{n(n-1)}}$$



#### 4.1.1 Měření času odezvy při dotazu na napětí

V následujících Tab. 9 a Tab. 10 jsou uvedeny všechny naměřené hodnoty odezev pro dotaz na napětí při komunikaci přes různá přístrojová rozhraní.

**Tab. 9:** Čas odezvy dotazu na napětí na jednom zdroji pro různá rozhraní.

Počet dotazů	Odezva pro USB [ms]	Odezva pro GPIB [ms]	Odezva pro RS232 [ms]	Odezva pro LXI [ms]
1	58	14	424	7
10	14	14	60	7
100	10	14	25	7
1000	9	14	21	7
10000	9	14	20	7
<b>Aritmetický průměr</b>	-	14	-	7
<b>Přesnost měření</b>	-	1	-	1
<b>Střední kvad. chyba</b>	-	1	-	1

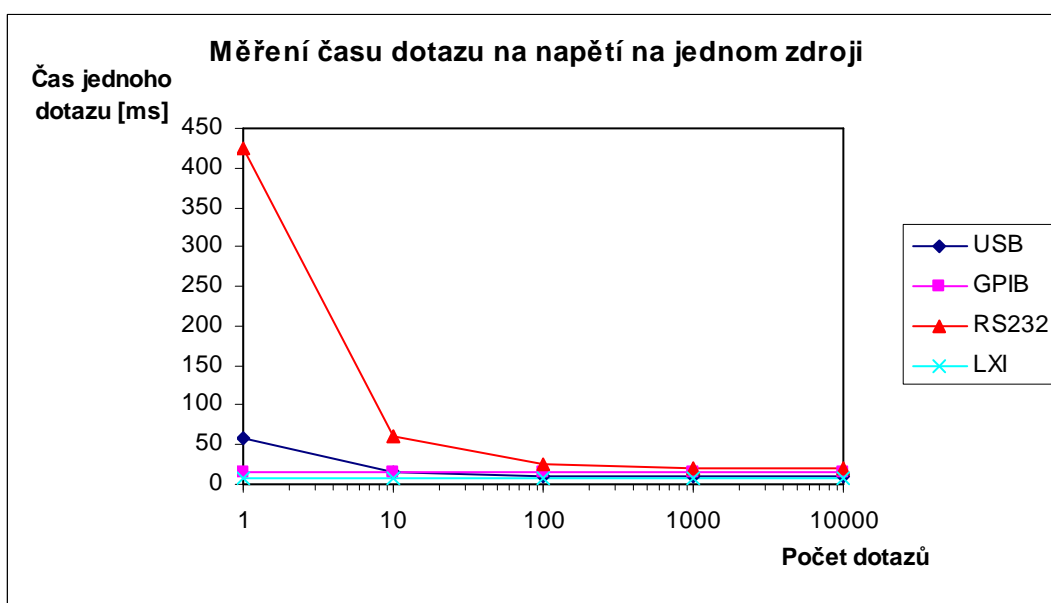
**Tab. 10:** Čas odezvy dotazu na napětí na obou zdrojích pro různá rozhraní.

Počet dotazů	Odezva pro USB [ms]	Odezva pro GPIB [ms]	Odezva pro RS232 [ms]	Odezva pro LXI [ms]
1	69	28	454	12
10	23	28	81	12
100	20	28	45	12
1000	19	28	41	12
10000	19	28	41	12
<b>Aritmetický průměr</b>	-	28	-	12
<b>Přesnost měření</b>	-	1	-	1
<b>Střední kvad. chyba</b>	-	1	-	1

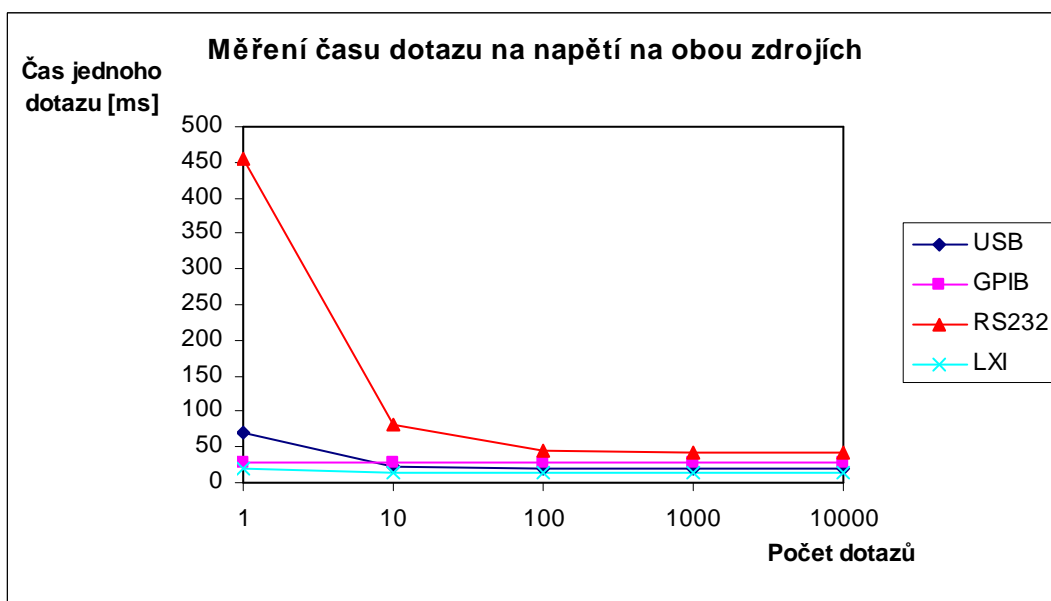
Z naměřených hodnot v Tab. 9 i v Tab. 10 se od sebe liší čas pro jeden dotaz na napětí od zbytku měření při zapojení zdroje přes USB a RS232. Důvodem tohoto rozdílu je, že v blokovém diagramu programu probíhá navíc inicializace sériového portu, která značně zpomalí celý proces. Tato inicializace portu se provede ovšem pouze jednou za celou sérii měření. Nejvýznamnější vliv má právě při jednom dotazu, protože v dalších sériích měření je již rozpočítána do průměru počtu měření. Čím vyšší počet měření tedy je, tím méně v něm hraje roli právě tato inicializace. Z důvodu inicializace sériového portu nebyl spočítán aritmetický průměr, přesnost měření ani střední kvadratická chyba u těchto dvou rozhraní (USB a RS232).

Méně zajímavý je již fakt, že většina hodnot z Tab. 9 je ve většině případů velmi blízká dvojnásobku hodnot z Tab. 10, což bylo očekáváno již před spuštěním měření.

Jednotlivé hodnoty byly zaznamenány do Grafů 1 a 2.



**Graf 1**



**Graf 2**

V Grafech 1 a 2 je o něco přehledněji vidět rozdíl času odezvy zaviněný inicializací sériového portu. O dalších naměřených hodnotách při počtu dotazů větším než 10 se již dá říct, že čas pro odezvu je téměř konstantní.

#### 4.1.2 Měření času nastavení napětí

V následujících Tab. 11 a Tab. 12 jsou uvedeny všechny naměřené hodnoty odezev pro nastavení napětí při komunikaci přes různá přístrojová rozhraní.

**Tab. 11:** Čas nastavení napětí na jednom zdroji pro různá rozhraní.

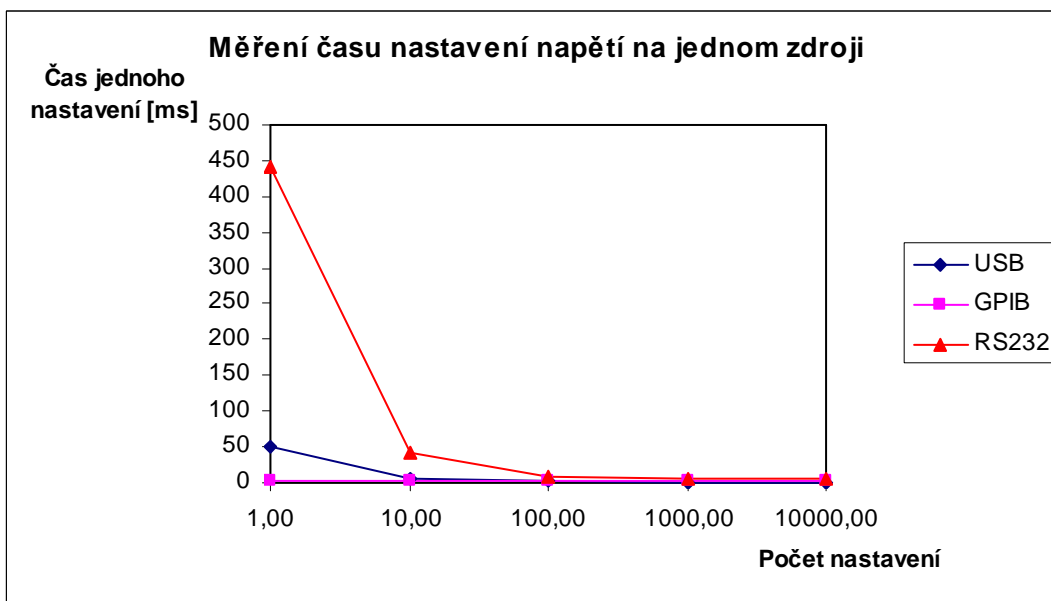
Počet nastavení	Čas pro USB [ms]	Čas pro GPIB [ms]	Čas pro RS232 [ms]
1	50	3	441
10	6	3	42
100	1	3	8
1000	1	3	6
10000	1	3	5
<b>Aritmetický průměr</b>	-	3	-
<b>Přesnost měření</b>	-	1	-
<b>Střední kvad. chyba</b>	-	1	-

**Tab. 12:** Čas nastavení napětí na obou zdrojích pro různá rozhraní.

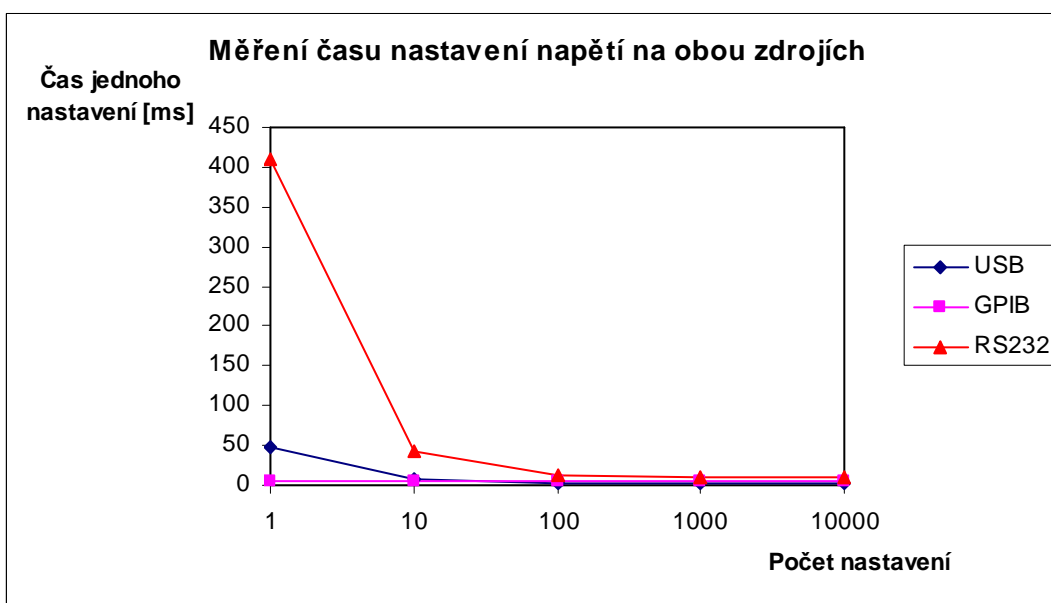
Počet nastavení	Čas pro USB [ms]	Čas pro GPIB [ms]	Čas pro RS232 [ms]
1	49	5	409
10	7	5	42
100	2	5	13
1000	2	5	11
10000	2	5	11
<b>Aritmetický průměr</b>	-	5	-
<b>Přesnost měření</b>	-	1	-
<b>Střední kvad. chyba</b>	-	1	-

Bohužel měření času nastavení napětí se nepovedlo provést pro všechna rozhraní. Chybí rozhraní LXI, pro které byly naměřeny hodnoty velmi blízké nebo rovné 0 a zdroj hlásil chybu. Rovněž při měření přes USB a GPIB rozhraní zdroj vykazoval chybu. Tato chyba se projevila tak, že na displeji blikalo upozornění *Fail 4*, které není nikde v návodu [7] specifikováno. V případě LXI a USB jsou data posílána v paketech o různé bytové velikosti. Z tohoto důvodu může paket obsahovat až několik příkazů pro nastavení napětí a při nulovém časování je pak přístroj přehlcnen příkazy, které nedokáže zpracovat.

Ve níže uvedených Grafech 3 a 4 je možné opět pozorovat rozdíl u první hodnoty pro rozhraní USB a RS232, který dokazuje nutnost inicializace sériového portu před každým měřením.



Graf 3



Graf 4

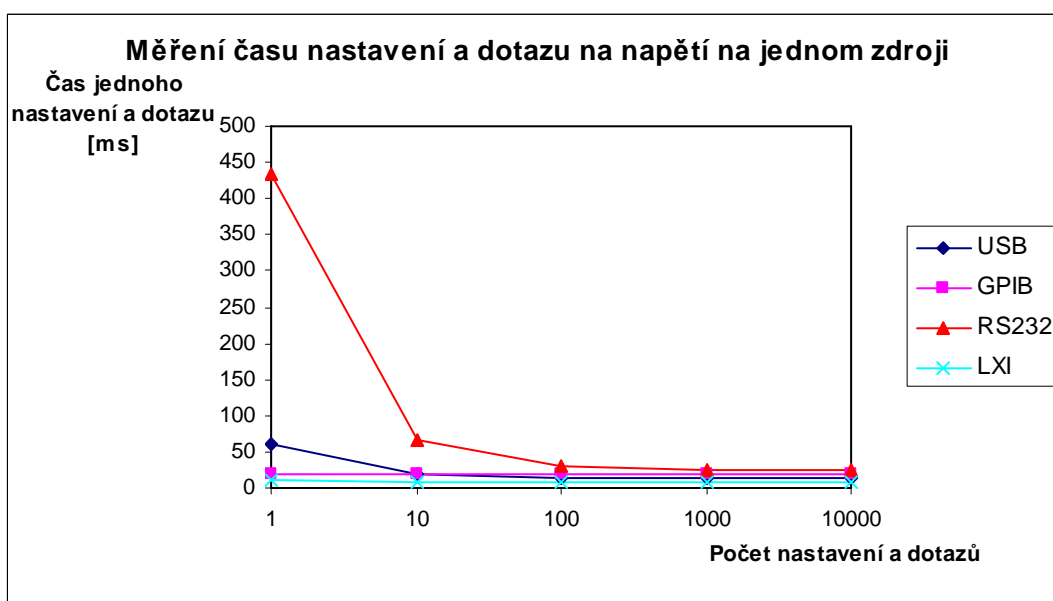
#### 4.1.3 Měření času odezvy při nastavení i dotazu na napětí

Tab. 13: Čas odezvy nastavení a dotazu na napětí na jednom zdroji pro různá rozhraní.

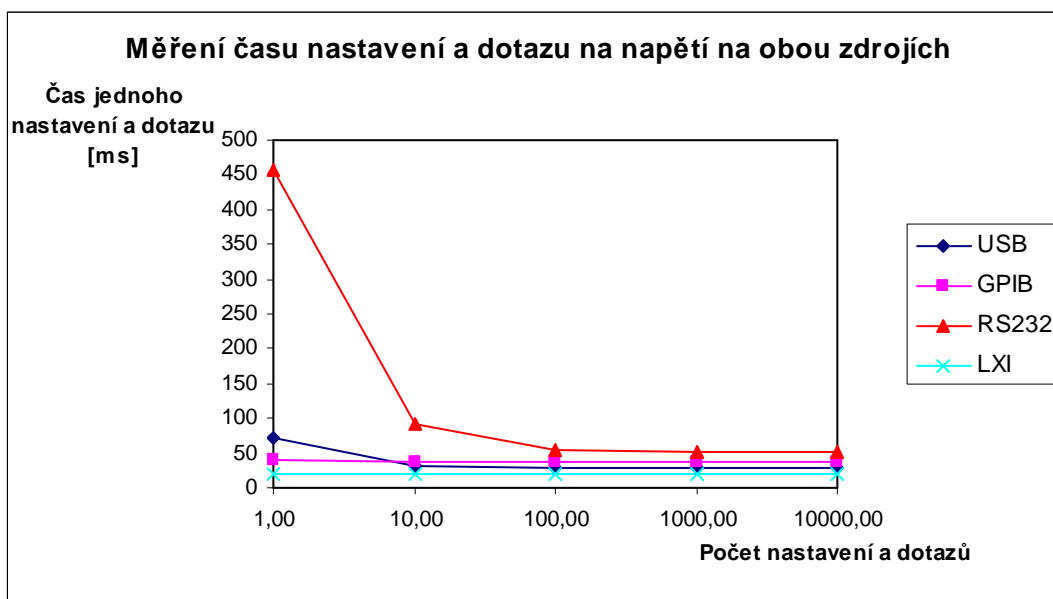
Počet nastavení a dotazů	Odezva pro USB [ms]	Odezva pro GPIB [ms]	Odezva pro RS232 [ms]	Odezva pro LXI [ms]
1	61	19	434	10
10	19	19	66	9
100	14	19	30	9
1000	14	19	26	9
10000	14	19	26	9
<b>Aritmetický průměr</b>	-	19	-	9
<b>Přesnost měření</b>	-	1	-	1
<b>Střední kvad. chyba</b>	-	1	-	1

**Tab. 14:** Čas odezvy nastavení a dotazu na napětí na obou zdrojích pro různá rozhraní.

Počet nastavení a dotazů	Odezva pro USB [ms]	Odezva pro GPIB [ms]	Odezva pro RS232 [ms]	Odezva pro LXI [ms]
1	73	39	457	20
10	32	38	92	19
100	28	38	56	19
1000	28	38	52	19
10000	28	38	52	19
<b>Aritmetický průměr</b>	-	38	-	19
<b>Přesnost měření</b>	-	1	-	1
<b>Střední kvad. chyba</b>	-	1	-	1



**Graf 5**



**Graf 6**

Toto měření bylo uskutečněno pro srovnání se součtem naměřených odezev z předchozích dvou měření (dotaz a nastavení). Za tímto účelem byly vytvořeny Tab. 15 a Tab. 16, ve kterých jsou sečteny odezvy z těchto předchozích měření. Jelikož by v případě RS232 a USB byla započítána jedna doba inicializace navíc, byla tato hodnota jednou odečtena (více o době inicializace sériového portu v následující podkapitole 4.2). V případě USB byla odečtena hodnota inicializace sériového portu 48 ms a RS232 406 ms.

**Tab. 15:** *Součet naměřených hodnot z předchozích měření pro jeden zdroj.*

Počet nastavení a dotazů	Odezva pro USB [ms]	Odezva pro GPIB [ms]	Odezva pro RS232 [ms]	Odezva pro LXI [ms]
1	60	17	459	7+x
10	15	17	61	7+x
100	11	17	34	7+x
1000	10	17	32	7+x
10000	10	17	31	7+x

**Tab. 16:** *Součet naměřených hodnot z předchozích měření pro oba zdroje.*

Počet nastavení a dotazů	Odezva pro USB [ms]	Odezva pro GPIB [ms]	Odezva pro RS232 [ms]	Odezva pro LXI [ms]
1	70	33	457	12+x
10	25	33	82	12+x
100	22	33	54	12+x
1000	21	33	52	12+x
10000	21	33	52	12+x

Bohužel pro hodnoty naměřené pomocí LXI nebylo možné spočítat jejich součet, proto byly ponechány ve tvaru hodnota pro dotaz napětí + x, kde x je nenaměřená hodnota doby nastavení napětí.

#### 4.1.4 Porovnání naměřených a spočítaných hodnot pomocí teoretické rychlosti

V Tab. 17 (viz níže) je shrnutí všech naměřených hodnot z podkapitol 4.1.1-4.1.3. V Tab. 18 (viz níže) jsou uvedeny vypočítané hodnoty z teoretických přenosových rychlostí. Při počítání odezvy pro RS232 bylo počítáno s rychlostí 9600 b/s, tedy 9,6 b/ms. Při počítání odezvy pro USB byl hlavním faktorem jeden rámeček, který trvá přesně 1 ms. Naopak pro GPIB a LXI se již vycházelo z přenosových rychlostí 500 kB/s a 100 MB/s. U LXI je nutné připomenout, že data se posílají v paketech o neznámých bytových velikostech, jelikož standard nedefinuje velikost datového pole. Pro spočítání odezvy byla tedy délka datového pole zvolena 1500 B, která odpovídá standardu pro Ethernet, který s LXI úzce souvisí.

Je nutné rovněž připomenout, že hodnoty, s kterými byly počítány jednotlivé odezvy, se uvádí jako maximální možné, ale reálné hodnoty přenosových rychlostí jsou ve většině případů nižší. Rozdíl, který vyjde po odečtení naměřených a teoretických hodnot je považován za reakční dobu přístroje (rychlost elektroniky) zdroje Sorensen XEL 30-3DP.

**Tab. 17:** *Naměřené hodnoty odezev rozhraní.*

Druh příkazu	Délka v bytech	RS232 [ms]	USB [ms]	GPIO [ms]	LXI [ms]
Dotaz na napětí	11	20	9	14	7
Nastavení napětí	4	5	1	3	-
Nastavení i dotaz na napětí	15	26	14	19	9

**Tab. 18:** *Hodnoty odezev spočítané pomocí teoretických přenosových rychlostí rozhraní.*

Druh příkazu	Délka v bytech	RS232 [ms]	USB [ms]	GPIO [ms]	LXI [ms]
Dotaz na napětí	11	11,458	2,000	0,026	0,030
Nastavení napětí	4	4,167	1,000	0,008	0,015
Nastavení i dotaz na napětí	15	15,625	2,000	0,034	0,030

#### 4.1.5 Měření pro charakterizaci LXI

Tato měření byla prováděna ve dvou různých sítích. První z nich je interní síť Univerzity Palackého v Olomouci a provozovatel druhé sítě je občanské sdružení Haná Free Net, jejíž server je umístěn ve společnosti FOFRnet v budově RCO v Olomouci. Pro každou z těchto sítí se měnil i druh připojení. Připojení počítače do sítě bylo provedeno jak přes UTP kabel tak i pomocí wi-fi. Všechny naměřené i spočítané hodnoty v Tab. 19-22 jsou uváděny v milisekundách.

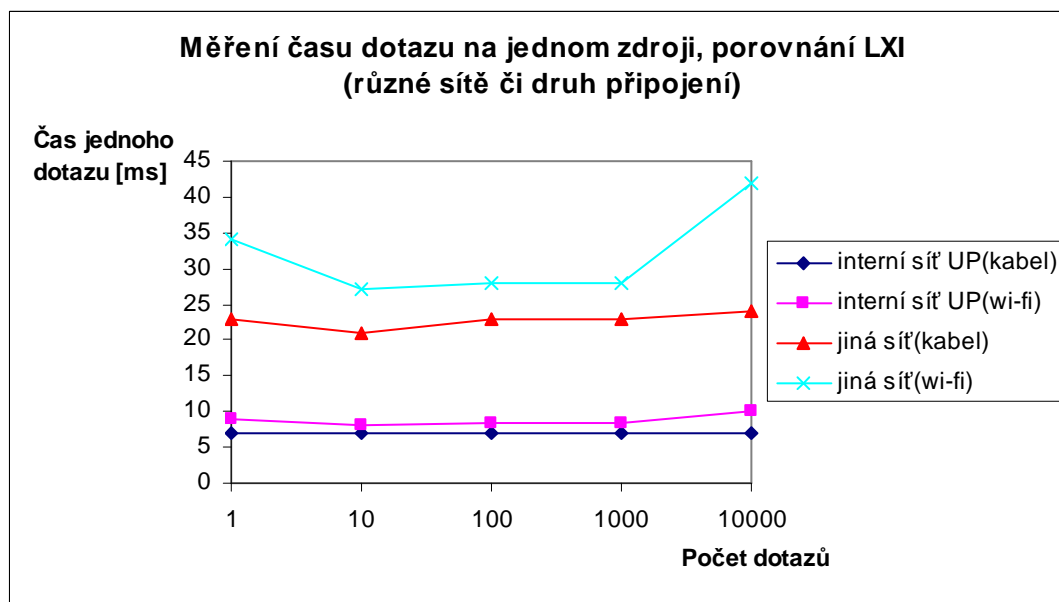
**Tab. 19:** *Čas odezvy dotazu na napětí na jednom zdroji pro různé sítě.*

Počet dotazů	LXI - interní síť (kabel)	LXI - interní síť (wi-fi)	LXI - jiná síť (kabel)	LXI - jiná síť (wi-fi)
1	7	9	23	34
10	7	8	21	27
100	7	8	23	28
1000	7	8	23	28
10000	7	10	24	42
<b>Aritmetický průměr</b>	7	9	23	32
<b>Přesnost měření</b>	1	1	1	5
<b>Střední kvad. chyba</b>	1	1	1	2

**Tab. 20:** Čas odezvy dotazu na napětí na obou zdrojích pro různé sítě.

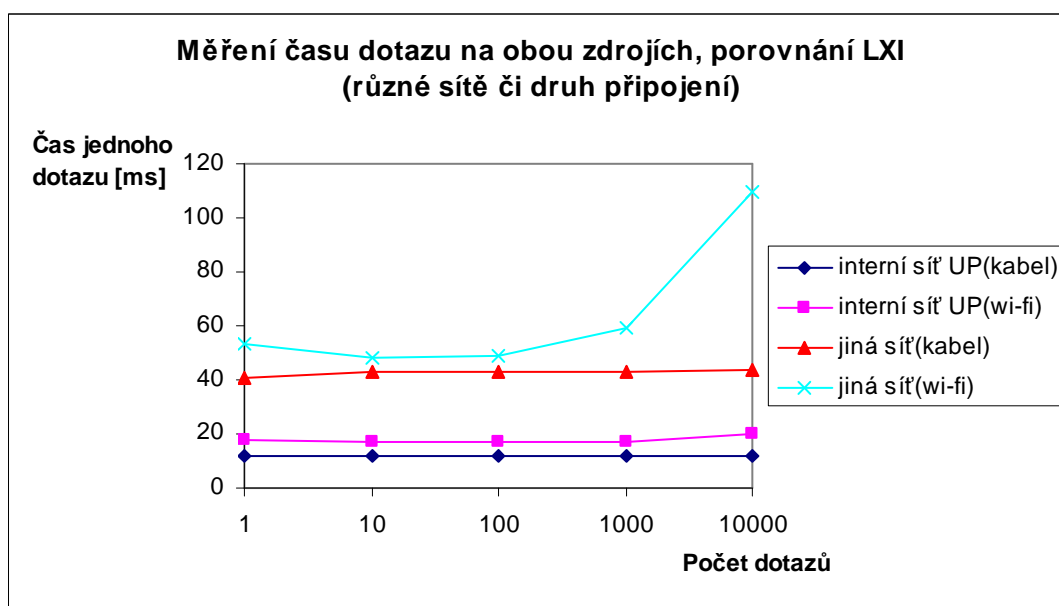
Počet dotazů	LXI - interní síť (kabel)	LXI - interní síť (wi-fi)	LXI - jiná síť (kabel)	LXI - jiná síť (wi-fi)
1	12	18	41	53
10	12	17	43	48
100	12	17	43	49
1000	12	17	43	59
10000	12	20	43	110
<b>Aritmetický průměr</b>	12	18	43	60
<b>Přesnost měření</b>	1	1	1	20
<b>Střední kvad. chyba</b>	1	1	1	10

Naměřené hodnoty v Tab. 19 a Tab. 20 jsou relativně konstantní. Výjimku tvoří časy naměřené při komunikaci přes wi-fi, především tedy z jiné sítě. Tento výsledek byl očekáván, jelikož připojení přes wi-fi je mnohem méně stabilní. Zajímavý je také fakt, že naměřené hodnoty odezev v případě 10 dotazů při připojení přes wi-fi v jiné, než interní síti UP v Olomouci byly naměřeny nejmenší hodnoty. Bohužel nelze přesně určit příčinu tohoto jevu z důvodu velkého rozsahu naměřených odezev (nestability wi-fi). Jedno vysvětlení tohoto jevu by ale mohlo být to, že přenášená data pro 10 dotazů na okamžitou hodnotu napětí z hlediska bytové velikosti tvoří např. jeden paket. Jelikož velikost datového pole paketu u LXI není standardem definována, je toto tvrzení pouze spekulace.



**Graf 7**





**Graf 8**

Ze stejného důvodu, který je uveden v části 4.1.2, nebyla provedena měření času nastavení napětí.

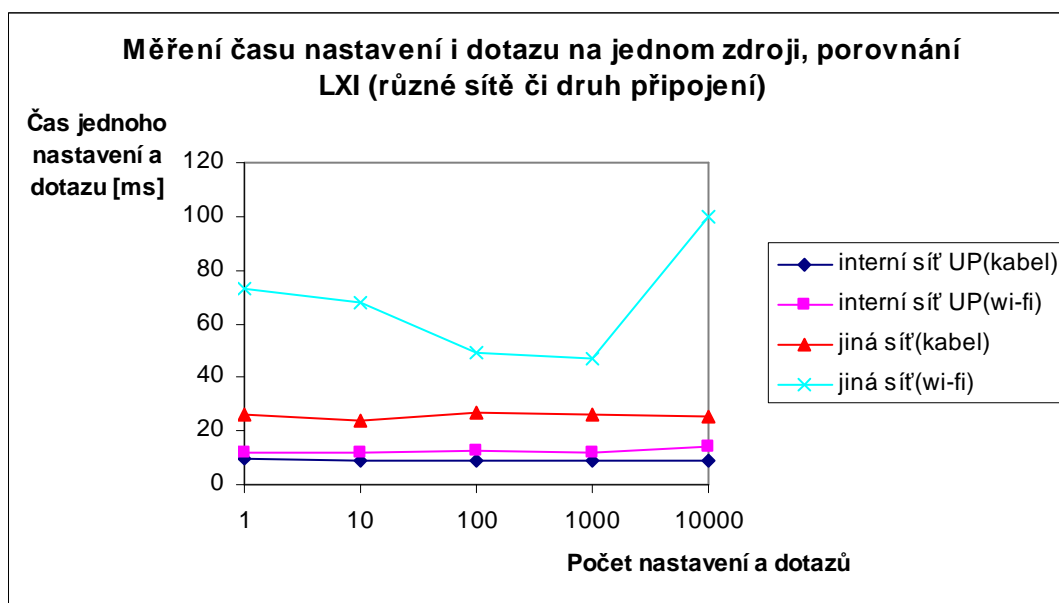
**Tab. 21:** Čas odezvy nastavení a dotazu na napětí na jednom zdroji pro různé sítě.

Počet nastavení a dotazů	LXI - interní síť (kabel)	LXI - interní síť (wi-fi)	LXI - jiná síť (kabel)	LXI - jiná síť (wi-fi)
1	10	12	26	73
10	9	12	24	68
100	9	13	27	49
1000	9	12	26	47
10000	9	14	25	100
<b>Aritmetický průměr</b>	9	13	26	70
<b>Přesnost měření</b>	1	1	1	20
<b>Střední kvad. chyba</b>	1	1	1	10

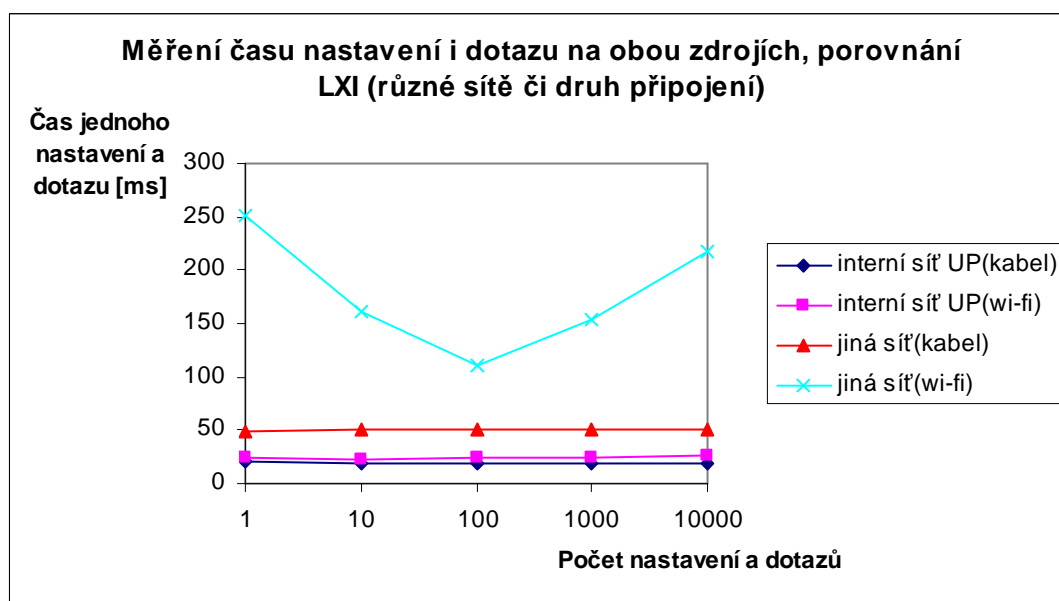
**Tab. 22:** Čas odezvy nastavení a dotazu na napětí na obou zdrojích pro různé sítě.

Počet nastavení a dotazů	LXI - interní síť (kabel)	LXI - interní síť (wi-fi)	LXI - jiná síť (kabel)	LXI - jiná síť (wi-fi)
1	20	24	48	252
10	19	23	51	162
100	19	25	51	110
1000	19	24	51	154
10000	19	26	50	218
<b>Aritmetický průměr</b>	19	24	50	180
<b>Přesnost měření</b>	1	1	1	40
<b>Střední kvad. chyba</b>	1	1	1	20

Všechny naměřené hodnoty jednoznačně dokazují, že komunikace v místní síti (LAN) je vždy rychlejší, a to někdy až více jak desetinásobně v případě připojení pomocí wi-fi (viz Tab. 22).



**Graf 9**



**Graf 10**

Z Grafů 7-10 lze vyčíst, že komunikace s přístrojem v LAN je nejen rychlejší, ale také naměřené hodnoty jsou mnohem více stabilní (než hodnoty pro wi-fi v jiné síti).

## 4.2 Měření doby inicializace sériového portu

Pomocí měření v předchozí kapitole bylo zjištěno, že inicializace výrazně ovlivňuje časy odezvy při menším počtu příkazů. Dobu této inicializace bylo nutné zjistit. První cestou ke zjištění doby inicializace bylo přímé změření jejího průběhu pomocí vytvořeného programu dle podkapitoly 3.3. Druhou myšlenkou byl jednoduchý výpočet za pomoci první a poslední hodnoty naměřené z předchozích měření. První naměřená hodnota obsahuje čas odezvy a dobu inicializace. Poslední hodnota obsahuje rovněž čas odezvy, ale jen nepatrný zlomek doby inicializace (přesně  $\frac{1}{10000}$  doby inicializace), kterou lze v tomto případě zanedbat. Dobu inicializace lze tedy získat rozdílem první a poslední naměřené hodnoty (přesněji hodnoty pro 1 dotaz a hodnoty pro 10 000 dotazů).

Všechny naměřené i spočítané hodnoty pro dobu inicializace sériového portu byli umístěny do Tab. 23.

**Tab. 23:** Doba inicializace sériového portu.

Číslo měření	Inicializace RS232 [ms]	Inicializace RS232* [ms]	Inicializace USB [ms]	Inicializace USB* [ms]
1.	310	404	46	49
2.	314	413	45	50
3.	316	398	44	47
4.	317	408	44	47
5.	316	405	44	45
<b>Aritmetický průměr</b>	315	406	45	48
<b>Přesnost měření</b>	2	4	1	2
<b>Střední kvad. chyba</b>	1	1	1	1

Hodnoty v Tab. 23 označené jako \* jsou ty, které byli spočítány. Ostatní jsou naměřené pomocí programu. V případě měření doby inicializace při komunikaci přes USB se hodnoty doby inicializace od sebe výrazně neliší, bohužel nelze konstatovat to samé o hodnotách pro RS232. Zde je rozdíl mezi naměřenou a spočítanou hodnotou až 100 ms.

## 4.3 Porovnání hodnot ze zdroje a hodnot naměřených multimetrem

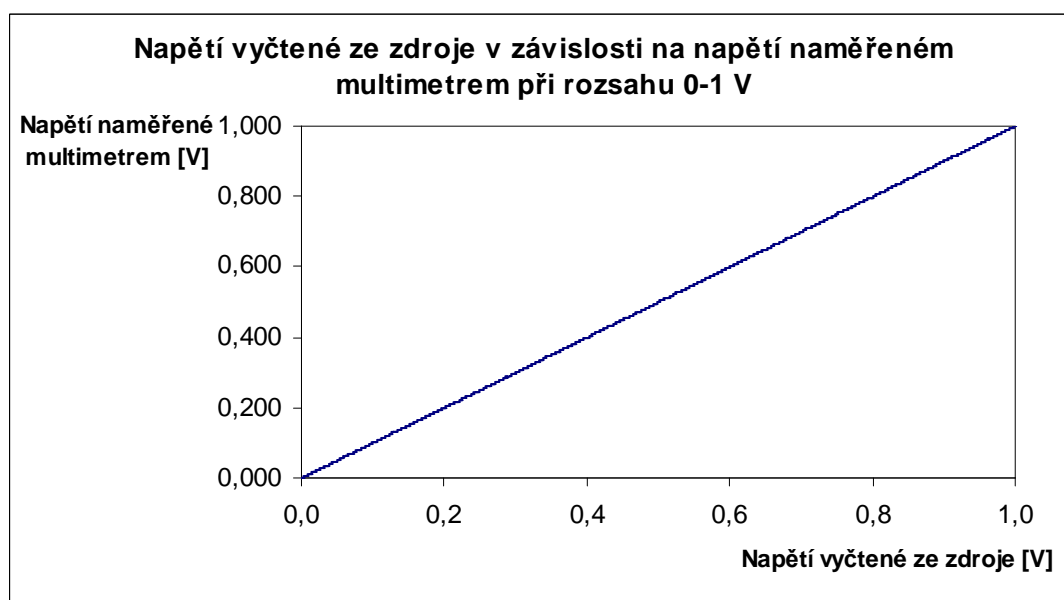
Smyslem tohoto měření bylo ověření přesnosti měření udávané v manuálu programovatelného napájecího zdroje Sorensen XEL 30-3DP [7] za pomoci digitálního multimetru Agilent 34401A. Samotný průběh měření začínal na zvolené počáteční hodnotě napětí. Po době 4 s (z toho 2 s měl digitální multimetr na ustálení) se napětí na zdroji změnilo o krok, který dosahoval pro každé měření jiné hodnoty. Zvolené časování by mělo

být dostatečné, jelikož součet doby zpracování příkazu a rychlosti změny napětí bez zátěže při zvyšování napětí je roven 65 ms. Zbylý čas tedy může být využit pro chod programu. Doba náběhu napětí na výstup je sice menší než 15 ms, v tomto případě ji není nutné započítávat, jelikož zapnutí napěťových výstupů proběhlo již před počátkem cyklu měření. Měření bylo ukončeno tím, když postupně měnící se hodnota napětí dosáhla konečné hodnoty napětí. Rozsah měření byl tedy definován dvěmi hodnotami (počáteční a konečné napětí). Hodnoty vyčtené ze zdroje a hodnoty naměřené digitálním multimetrem byli rovněž odečteny a tak byla spočítána jejich odchylka. Blokovaný diagram použitého programu je popsán v podkapitole 3.4.

Ačkoliv byla nastavena na digitálním multimetru přesnost 0,001 V, hodnoty byli vyčítány s přesností 0,000001 V (celý obsah bufferu). Proto byli v tabulkách uvedených v Příloze 3 zaokrouhleny na 3 desetinná místa.

#### 4.3.1 Měření při rozsahu 0-1 V

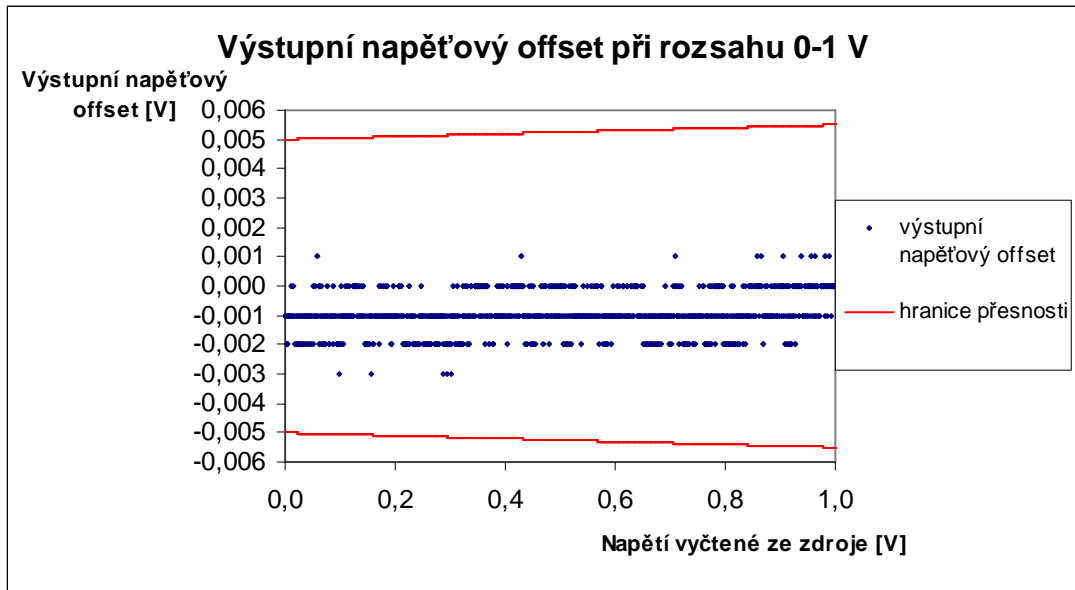
Počáteční hodnota napětí byla stanovena na 0 V, jeden krok změny napětí na 0,001 V (1 mV) a konečné napětí na 1 V. Počet měření byl tedy roven přesně 1000. Jelikož je počet měření poměrně vysoký, byla tabulka s naměřenými hodnotami umístěna do textového souboru Přílohy 3. Hodnoty byli vyneseny do Grafu 11.



Graf 11

Z Grafu 11 nejsou patrné odchylky naměřených hodnot, proto byl vytvořen Graf 12, který lépe zobrazuje vstupní napěťový offset napájecího zdroje. Nicméně žádná odchylka v podobě napěťového offsetu nepřesáhla hranici přesnosti definované pro napájecí zdroj

(přesnost napájecího zdroje je uváděna jako 0,05% z vyčtené hodnoty + 5 mV). Tato hranice je rovněž zobrazena v Grafu 12.

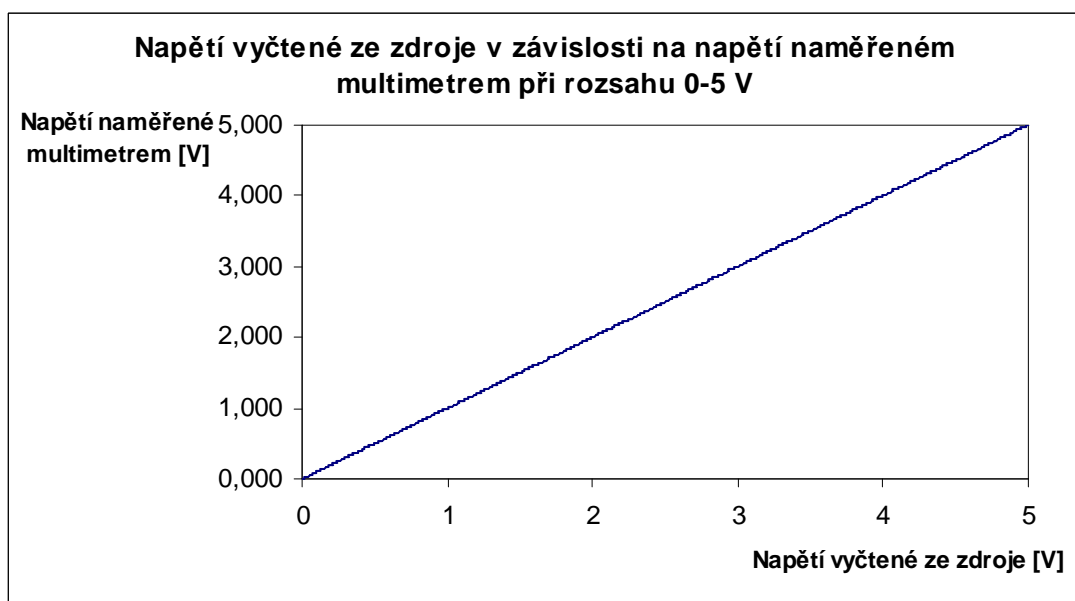


Graf 12

Při zvoleném rozsahu 0-1V je offset skoro vždy záporný. Přesnost měření byla spočítána jako  $\pm 0,001$  V.

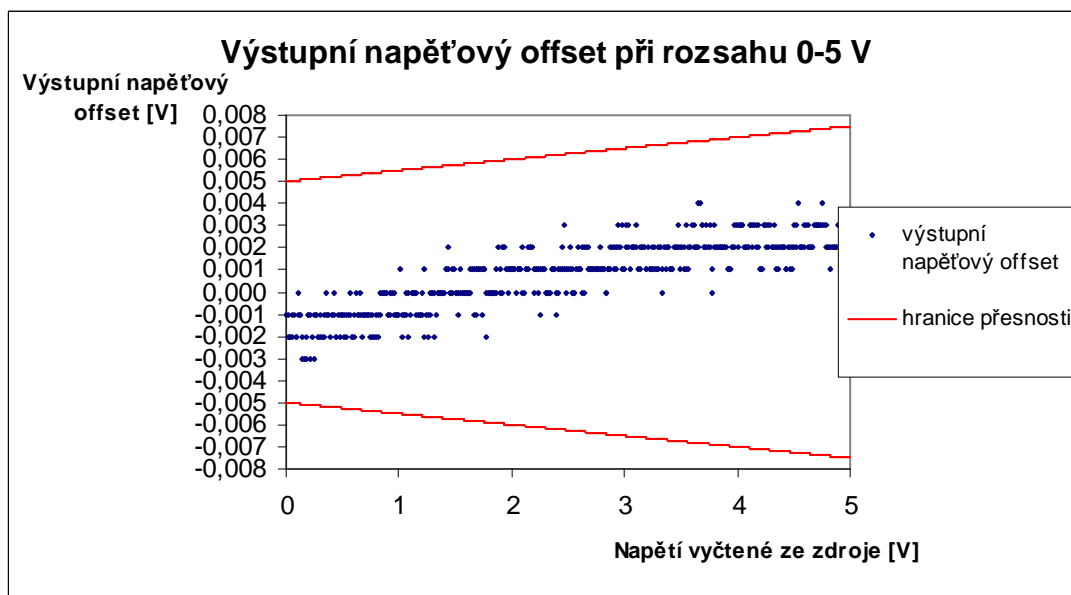
#### 4.3.2 Měření při rozsahu 0-5 V

Měření opět začínalo na hodnotě 0 V a skončilo když bylo dosaženo napětí 5 V. Krok změny napětí byl nyní ovšem navýšen na hodnotu 0,01 V (10 mV). Tabulka byla opět umístěna do Přílohy 3. Rovněž bylo provedeno vynesení hodnot do Grafu 13.



Graf 13

Pro lepší posouzení výstupního napěťového offsetu při rozsahu 0-5 V byl opět vytvořen následující Graf 14. Hodnoty odchylek nepřesáhli definovanou přesnost.

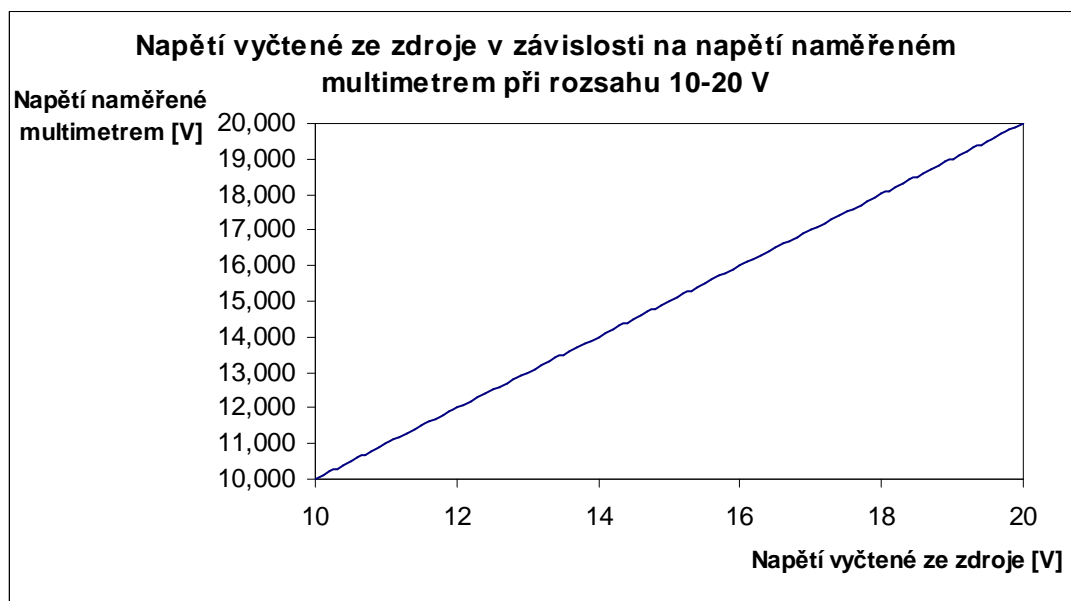


**Graf 14**

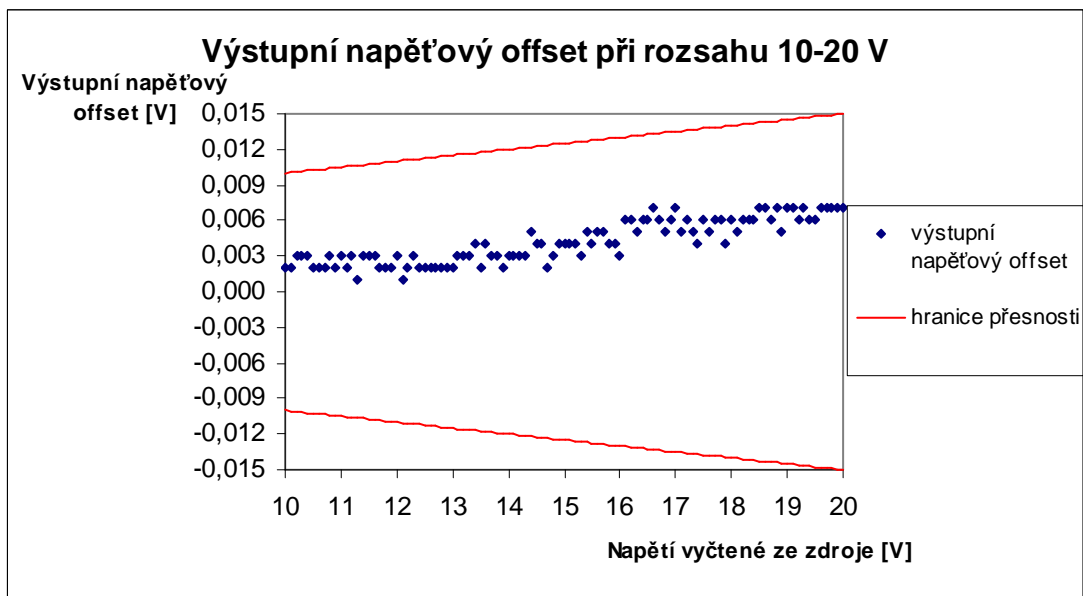
Při rozsahu 0-5 V lze pozorovat, že offset je do napětí cca 1,5 V záporný a od napětí vyšších než 1,5 V nabývá kladných hodnot. Přesnost měření je spočítána  $\pm 0,001$  V.

### 4.3.3 Měření při rozsahu 10-20 V

Při tomto měření byl rozsah měření stanoven od 10 V do 20 V a jeden krok změny napětí na hodnotu 0,1 V (100 mV). Tabulka byla umístěna do Přílohy 3.



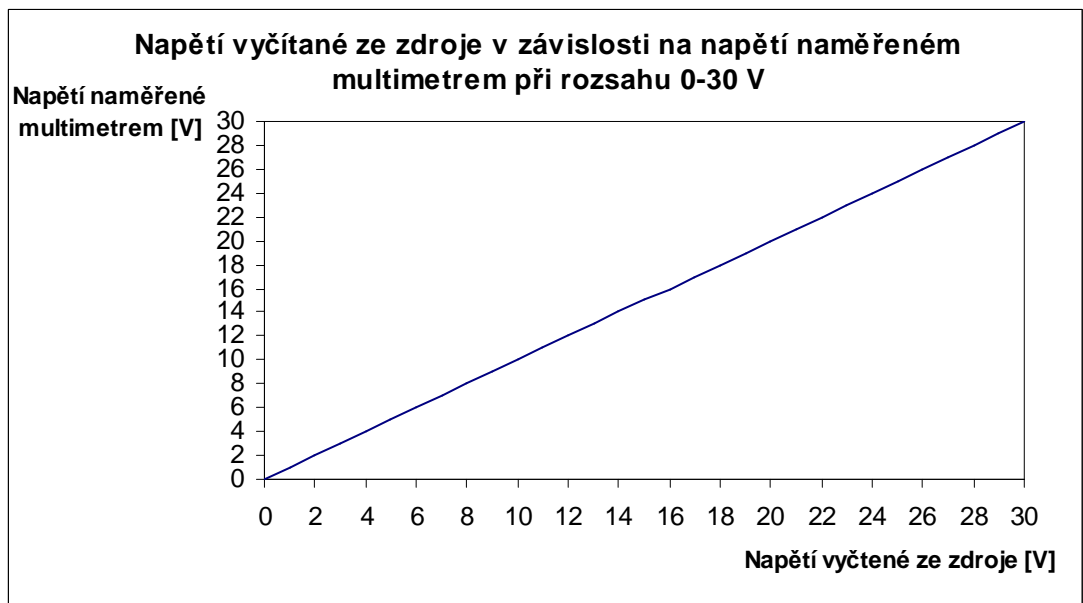
**Graf 15**

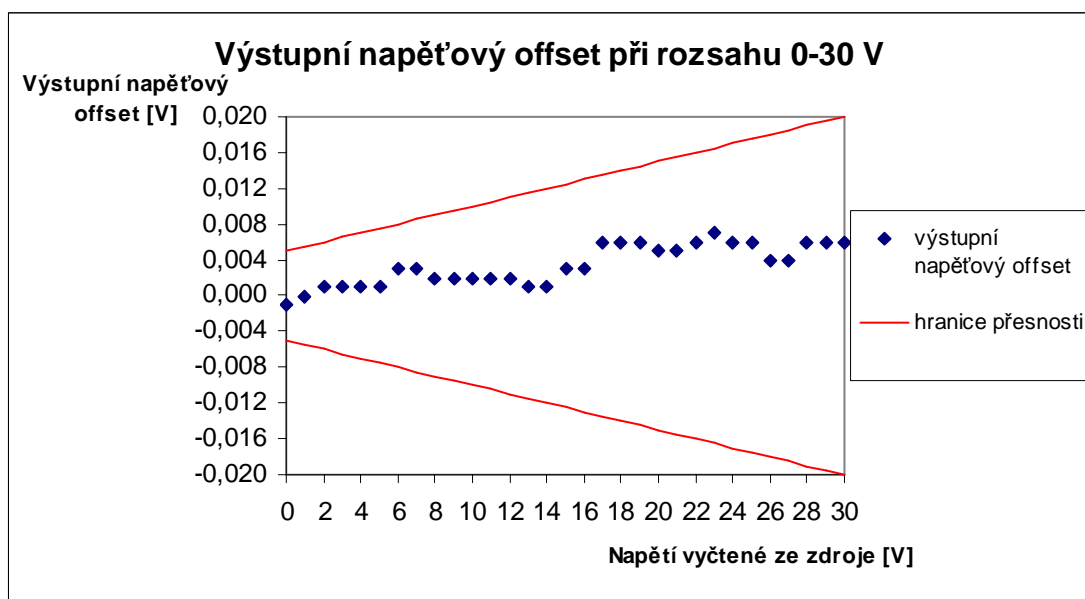


V porovnání s hodnotou přesnosti všechny splňovali limit uvedený v manuále napájecího zdroje. Přesnost měření byla spočítána na  $\pm 0,004$  mV.

#### 4.3.4 Měření při rozsahu 0-30V

Počáteční hodnota napětí byla nastavena na 0 V, konečná hodnota napětí na 30 V a krok změny napětí byl zvolen 1 V. Počet měření je tedy 30 a tabulka s výslednými hodnotami je umístěna v Příloze 3. Graf 17 popisuje průběh celého měření a Graf 18 výstupní napěťový offset napětí.





**Graf 18**

Z Grafu 18 je patrné, že ani jedna z odchylek opět nepřekračuje hranice přesnosti, které jsou uvedeny v manuálu napájecího zdroje. Přesnost měření je určena  $\pm 0,004$  mV.

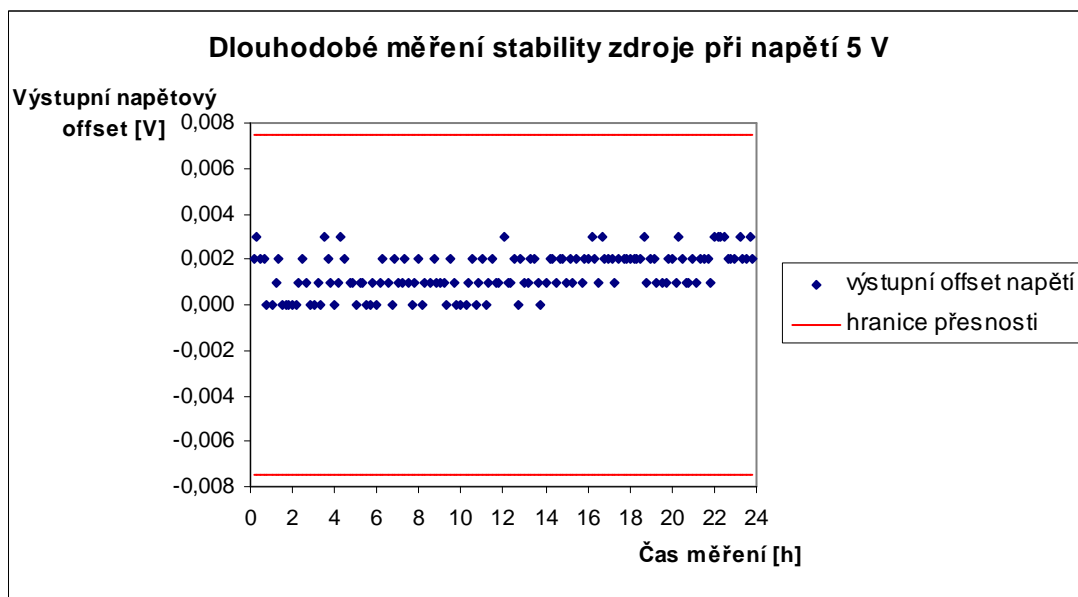
#### **4.4 Měření stability programovatelného napájecího zdroje**

Podstata tohoto měření byla velmi podobná měření v podkapitole 4.3. Rozdíl byl v tom, že napětí bylo nastaveno na počátku měření na jednu hodnotu, která se již v průběhu měření neměnila a v intervalu 10 minut bylo pokaždé změřeno napětí na výstupech napájecího zdroje pomocí digitálního multimetru. Smyslem bylo prověřit jestli se napětí na výstupech ustálí na nějaké hodnotě napětí, nebo jestli se během měření nezmenší alespoň výstupní offset. Každé měření bylo prováděno po dobu jednoho dne (1440 minut). Měření probíhala pro dvě různé hodnoty napětí (5 V a 30 V).

##### **4.4.1 Stabilita programovatelného napájecího zdroje při napětí 5 V**

Z důvodu velkého počtu naměřených dat byla tabulka naměřených hodnot umístěna do Přílohy 3. Všechny hodnoty z ní byly vloženy do Grafu 19 (viz níže), ve kterém je rovněž zobrazena hranice definovaná manuálem jako přesnost zdroje. Počáteční hodnota napětí byla zvolena 5 V.



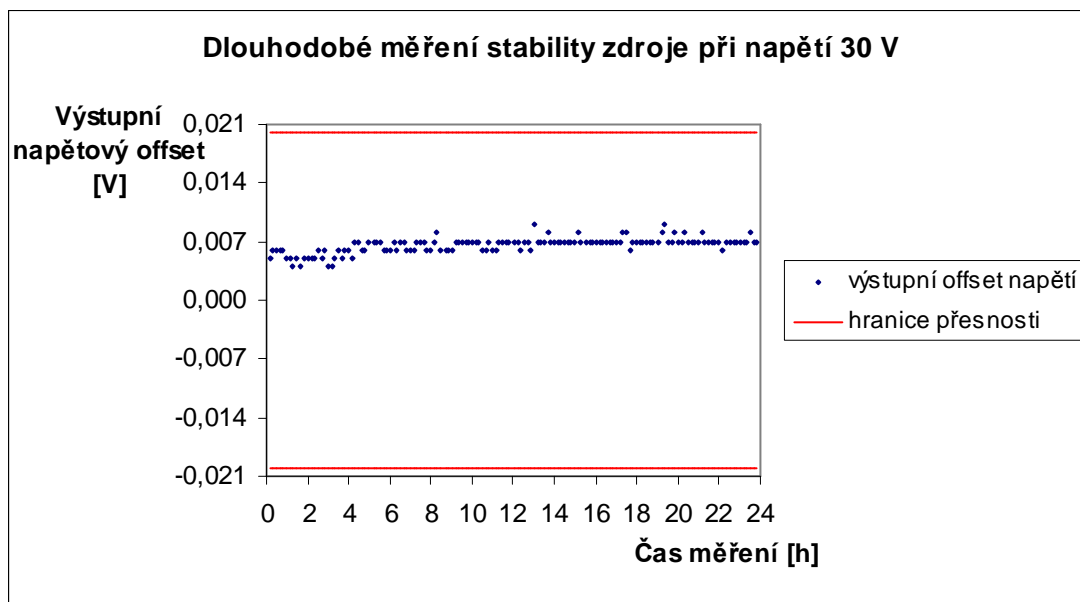


**Graf 19**

Z Grafu 19 je možné usoudit, že výstupní napěťový offset měl tendenci s rostoucím časem jakkoliv růst. Přesnost měření je určena  $\pm 0,001$  V.

#### 4.4.2 Stabilita programovatelného napájecího zdroje při napětí 30 V

Tabulka s hodnotami byla opět umístěna do Přílohy 3. Naměřené hodnoty byly vyneseny do Grafu 20. Počáteční hodnota napětí byla změněna na 30 V.



**Graf 20**

Z počátku měření se výstupní napěťový offset poměrně často měnil. V čase přibližně 14 hodin od počátku měření se již začal ustalovat na hodnotě 7 mV s občasnými výkyvy. Přesnost měření byla spočítána jako  $\pm 0,007$  V.

## Závěr

Hlavním cílem této bakalářské práce bylo seznámení se s obsluhou programovatelného napájecího zdroje Sorensen XEL 30-3DP. V první kapitole byly podrobně popsány jednotlivá přístrojová rozhraní, kterými je vybaven výše zmíněný napájecí zdroj. V této kapitole byli rovněž vysvětleny principy komunikace pro tyto rozhraní. Druhá kapitola obsahuje důležité technické parametry zařízení, které byly využity pro realizaci všech měření. Další část práce je věnována tvorbě jednotlivých programů, které byli implementovány ve vývojovém programovacím prostředí LabVIEW. Pro snadnější orientaci v částech kódů byli do textu (nebo do příloh) vloženy obrázky blokových diagramů.

K prezentaci veškerých naměřených hodnot byla použita nejčastěji forma přehledných tabulek a grafů. Bohužel při porovnání naměřených odezev spolu se spočítanými hodnotami za pomoci teoretických přenosových rychlostí došlo k nemalým rozdílům. Tyto rozdíly mohou mít u každého rozhraní několik příčin. V případě sériového rozhraní RS232 se na rozdíl nejvíce podílí zvýšení reakční doby napájecího zdroje programovým řízením XON/XOFF. Při komunikaci přes USB byly teoretické hodnoty odezev spočítány dle počtu předpokládaného počtu rámců, který se může lišit od skutečného počtu. Sběrnice GPIB má poměrně složitý cyklus přenosu dat, který může také celý proces přenosu značně zpomalit a v případě LXI, kde je rozdíl nejvyšší, je nutné určitě brát v úvahu zatíženost sítě a také neurčitost délky datového pole v paketu. Na druhou stranu všechny naměřené hodnoty splňují limit stanovený manuálem s výjimkou těch, které byli naměřeny přes wi-fi.

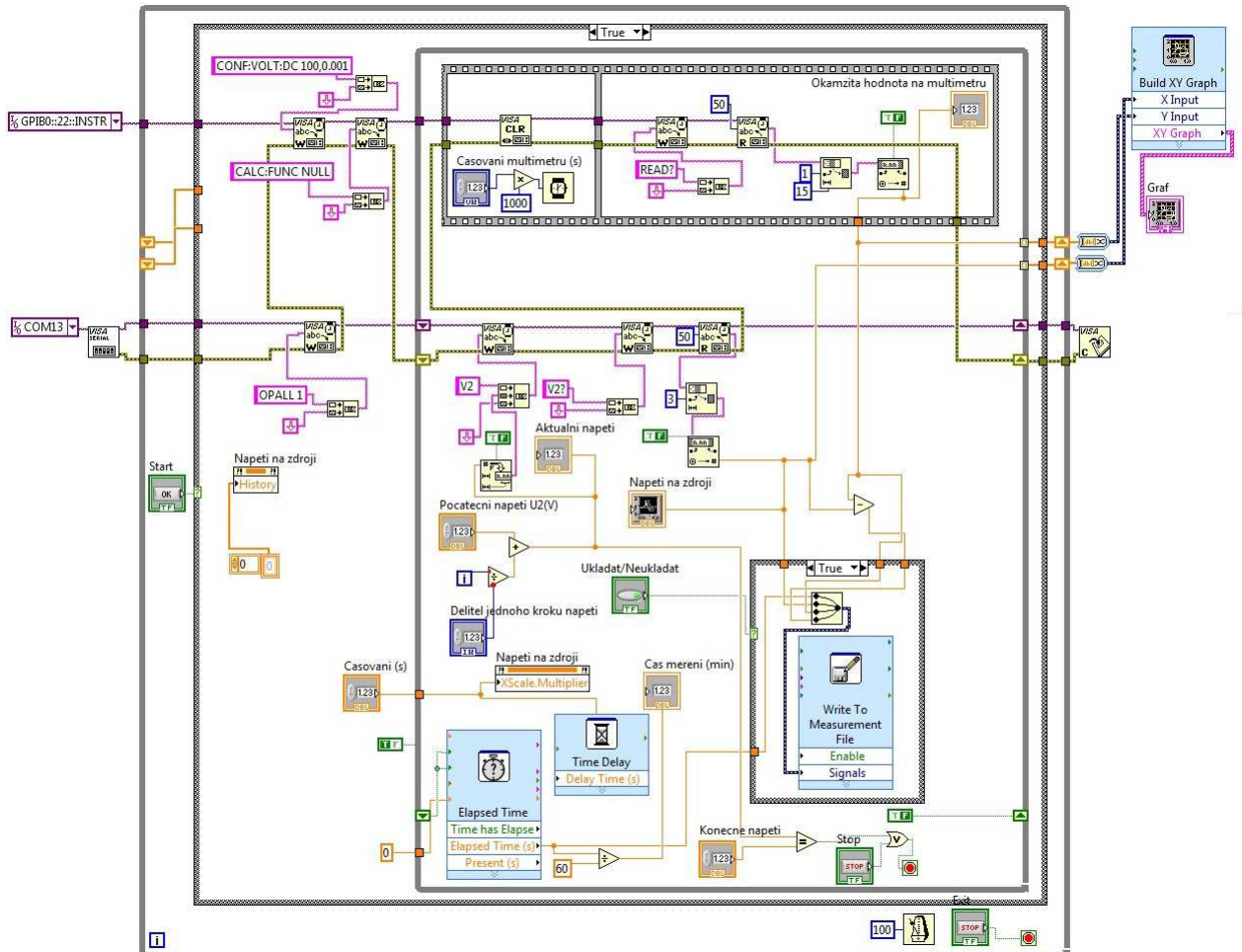
Při měření stability s konstantním i měnícím se napětím bylo prokázáno, že všechny odchylky jsou v intervalu přesnosti udávaném v manuálu.

## Seznam použitých zdrojů

- [1] Haasz V., Roztočil J., Novák J. : Číslicové měřicí systémy, Vydavatelství ČVUT, Praha 2000 (ISBN 80-01-02219-6)
- [2] Kubíček M. : Univerzální sériové rozhraní USB; Automa, FCC PUBLIC s. r. o, Praha 07/2000 (ISSN 1210-9592),  
dostupné z [http://www.odbornecasopisy.cz/index.php?id\\_document=27823](http://www.odbornecasopisy.cz/index.php?id_document=27823) (ze dne 23.2.2012)
- [3] Kainka B. : USB-Měření, řízení a regulace pomocí sběrnice USB, BEN-technická literatura, Praha 2002 (ISBN 80-7300-073-3)
- [4] Sběrnice GP-IB,  
dostupné z <http://webs.zcu.cz/fel/ket/+EMS/Prednes/GPIB.pdf> (ze dne 4.3.2012)
- [5] LXI Consortium,  
dostupné z [http://www.lxistandard.org/Documents/About/LXI%201\\_2\\_01%202007-11-26.pdf](http://www.lxistandard.org/Documents/About/LXI%201_2_01%202007-11-26.pdf) (ze dne 11. 3. 2012)
- [6] Peterka J. : Báječný svět počítačových sítí: Část XX. : Příběh Ethernetu; PC World, IDG Czech Republic, a. s., Praha 12/2006 (ISSN 1210-1079),  
dostupné z <http://www.earchiv.cz/b06/b1200001.php3> (ze dne 8.3.2012)
- [7] Sorensen XEL & XEL-P (GP) Series Operation Manual, AMETEK Programmable power, Inc., 2010
- [8] User manual Agilent 34401A, Agilent Technologies, Inc., 2007
- [9] M. Čejka: Použití počítačů v měřicí technice, VUT-FEKT v Brně, Brno 2002

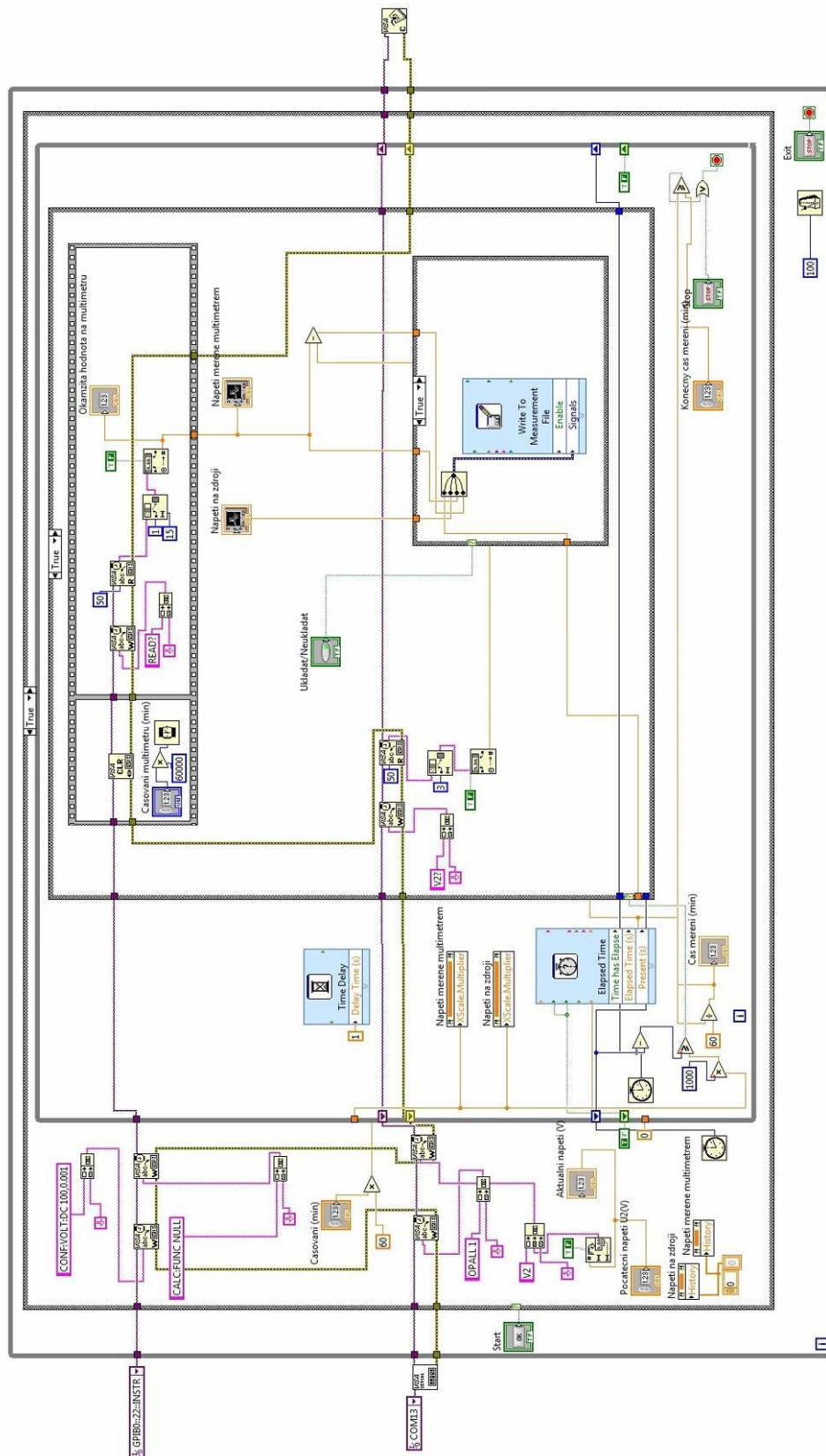
## Příloha 1:

**Blokový diagram VI pro porovnání hodnot vyčtených ze zdroje s hodnotami naměřenými digitálním multimetrem.**



## Příloha 2:

### Blokový diagram VI pro dlouhodobé měření stability zdroje.



### **Příloha 3:**

**Přiložené DVD s vytvořenými VI a tabulkami naměřených hodnot pro sledování stability zdroje.**