

**ZÁPADOČESKÁ UNIVERZITA V PLZNI  
FAKULTA ELEKTROTECHNICKÁ**

**KATEDRA ELEKTROMECHANIKY A VÝKONOVÉ ELEKTRONIKY**

## **BAKALÁŘSKÁ PRÁCE**

**Software pro měření cyklické voltametrie  
elektrochemických senzorů**



## **Anotace**

Předkládaná bakalářská práce se zabývá vývojem softwaru v prostředí LabVIEW. Program provádí měření cyklické voltametrie na přístroji Keithley 2636A.

## **Klíčová slova**

Keithley 2636A, LabVIEW, elektrochemické senzory, cyklická voltametrie

## **Abstract**

The bachelor theses presents the development of the software in LabVIEW environment. This program do measure of cyclic voltammetry.

## **Key words**

Keithley 2636A, LabVIEW, electrochemical sensor, cyclic voltammetry

## **Prohlášení**

Předkládám tímto k posouzení a obhajobě bakalářskou práci, zpracovanou na závěr studia na Fakultě elektrotechnické Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této bakalářské práce.

Dále prohlašuji, že veškerý software, použitý při řešení této bakalářské práce, je legální.

V Plzni dne 8.6.2012

Michal Bulva

.....

## **Poděkování**

Tímto bych rád poděkoval vedoucímu bakalářské práce Ing. Petru Kuberskému za vřelý přístup v průběhu práce.

# Obsah

<b>OBSAH.....</b>	<b>7</b>
<b>ÚVOD.....</b>	<b>9</b>
<b><u>1 ELEKTROCHEMICKÉ SENZORY.....</u></b>	<b><u>11</u></b>
1.1 POTENCIOMETRIE.....	11
1.2 AMPÉROMETRIE.....	12
<b><u>2 CYKICKÁ VOLTAMETRIE.....</u></b>	<b><u>13</u></b>
<b><u>3 VÝVOJ MĚŘICÍHO SOFTWARE.....</u></b>	<b><u>14</u></b>
3.1 ÚVOD DO LABVIEW.....	14
3.1.1 Čelní panel (Front Panel).....	15
3.1.2 Blokový diagram (Block Diagram).....	15
3.1.3 Datový tok – vzájemné předávání parametrů.....	15
3.1.4 Palety.....	15
3.1.5 Virtual Instrument.....	15
3.2 KEITHLEY 2636A.....	16
3.3 KOMUNIKACE LABVIEW S PŘIPOJENÝMI ZAŘÍZENÍMI.....	16
3.4 VYTVOŘENÍ PROJEKTU.....	17
3.5 VYTVOŘENÍ VI.....	17
3.6 KNIHOVNA KEITHLEY 2600.....	18
3.7 ČÁSTI PROGRAMU.....	18
3.7.1 Initialize.....	18
3.7.2 Node Format.....	19
3.7.3 Reset SMU.....	19
3.7.4 Source Function.....	19
3.7.5 Source Autorange.....	20
3.7.6 Measure Autorange.....	20
3.7.7 Measure Range.....	20
3.7.8 NPLC.....	21
3.7.9 Measure Sense Mode.....	21
3.7.10 Source Level.....	21
3.7.11 Source Output Enable.....	22
3.7.12 Cyklus opakování.....	22
3.7.12.1 Aktuální hodnota pilového průběhu.....	22
3.7.12.2 Source Level.....	22
3.7.12.3 Measure.....	23
3.7.12.4 Zápis hodnot do souboru.....	23
3.7.12.5 Vykreslování grafu.....	24
3.7.12.6 Přesné čekání.....	24
3.7.13 Ukončení cyklu a celého programu.....	24
3.8 GENERÁTOR PILY.....	24
3.8.1 Parametry pilového průběhu.....	25
3.8.2 Výpočet vzorkovací frekvence.....	25
3.8.3 Výpočet amplitudy.....	25
3.8.4 Přepínání mezi růstem a poklesem amplitudy.....	25
3.8.5 Výpočet počtu period.....	26
3.8.6 Doplnující ošetření.....	26
3.8.7 Počáteční nastavení hodnot signálem reset.....	26
<b><u>4 ZÁVĚR.....</u></b>	<b><u>28</u></b>
<b><u>5 POUŽITÁ LITERATURA.....</u></b>	<b><u>29</u></b>

**6 PŘÍLOHY.....1**



## Úvod

V dnešní době je bezpečnost práce řazena na přední příčky. Jedním ze sledovaných parametrů je množství chemických látek v ovzduší pracovního prostředí. Různá pravidla nebo i přímo zákony stanovují bezpečné limity, jejichž dodržování by mělo minimalizovat škodlivé účinky nebezpečných látek na člověka. Základem pro splnění limitů těchto látek je měření jejich koncentrací doplněné o konkrétní akční zásah.

Komplexním řešením pro měření koncentrací je analyzátor plynů velmi často pracující na principu infračerveného spektrometru. Ovšem nutno podotknout, že takovýto přístroj má vysoké pořizovací náklady, kde dalšími cenovými položkami je servis a kalibrace.

Méně robustním řešením než analyzátory jsou senzory zaměřené na měření jedné konkrétní látky popř. skupiny látek. Nejčastější principy senzorů jsou teplotně vodivostní, katalytické, polovodičové, elektrochemické, optické, magnetické a ionizační. Obecně nelze říci, který princip je nejvhodnější, vždy záleží na aktuální potřebě měření o jaký prvek jde, jaké se vyskytují jeho koncentrace, jestli se měří průběžně nebo v časových intervalech, případně jak dlouho z hlediska trvanlivosti má senzor fungovat atp.. V další části se budu věnovat elektrochemickým senzorům.

Výhodou elektrochemických senzorů je snadná dostupnost, malá velikost a to až v řádu milimetrů, relativně snadný vývoj celého měřicího zařízení a v neposlední řadě cena samotného senzoru. Na druhou stranu jejich hlavní nevýhodou je postupné snižování jejich citlivosti v závislosti na koncentraci a době, po kterou je senzor koncentraci vystaven.

V současné době probíhá výzkum a vývoj nové verze elektrochemických senzorů, které mají mít řádově vyšší životnost. S tímto je spojeno i opakované proměřování a hledání optimálních pracovních podmínek senzorů. Určité možnosti nabízí cyklická voltametrie.

## Seznam symbolů

NI	National Instruments
VI	Virtuální přístroj označující hotové řešení v LabVIEW
subVI	Část virtuálního přístroje
PC	Osobní počítač
GPIB, VXI, PXI, Seriál, Ethernet, USB	druhy datových sběrnic
TCP/IP	komunikační protokol v síti
VISA	Komunikační standard LabVIEW
CR	Znak pro začátek řádku při práci s textem (carriage return)
LF	Znak pro další řádek při práci s textem (line feed)
csv	Formát souboru, v němž jsou hodnoty oddělené čárkou

# 1 Elektrochemické senzory

Obecně by se dalo říci, že vývoj senzorů je spjat s vývojem technologií, výrobních postupů a je tak silně vázán na průmysl.

Nejstarší elektrochemické senzory se datují k roku 1950 a jejich využití bylo pro sledování koncentrace kyslíku. Jejich další vývoj souvisel se zvyšováním požadavků na bezpečnost práce, kde bylo potřeba měřit toxické a hořlavé plyny. Kolem roku 1980 započala jejich miniaturizace, která zapříčinila jejich dostupnost pro detekci mnoha rozdílných toxických plynů [1].

Fyzická velikost, geometrie, výběr proměnlivých komponentů a konstrukce elektrochemických senzorů obvykle záleží na zamýšleném použití. Docela často konečný návrh vyústí v kompromis parametrů senzoru. Dle očekávaného použití se pak funkce samotného elektrochemického senzoru různí. V důsledku toho lze očekávat rozdílná omezení (možnosti) z hlediska citlivosti, selektivity, reakčního času a provozní životnosti [1].

Základní princip je dán elektrochemickými reakcemi, které nastávají na určité elektrodě v daném elektrolytu. K těmto reakcím dochází na třech fázových rozhraních mezi analyzovanou kapalinou (resp. plynem), kovovou elektrodou a elektrolytem v tekuté nebo tuhé fázi. U elektrody dochází buď k anodické oxidaci resp. katodické redukci (tj k přijímání resp. odevzdávání elektronů). Reakci lze ovlivnit volbou elektrody, elektrolytu a případně elektrodovým potenciálem. Je-li elektroda ponořena do roztoku elektrolytu, který obsahuje ionty kovu, z něhož je zhotovena, vytváří se mezi ní a roztokem elektrický potenciál, který závisí na aktivitě a tedy na koncentraci iontů kovu v roztoku. Je-li elektroda z neušlechtilého kovu, kationty z elektrody pronikají do elektrolytu a na elektrodě vzniká kladný potenciál vůči roztoku. Naopak je-li elektroda z kovu ušlechtilého, část kationtů z elektrolytu se vyloučí na povrch elektrody a potenciál bude záporný. Dle způsobu vyhodnocování potenciálu elektrody existují metody potenciometrie a ampérometrie[2].

## 1.1 Potenciometrie

Je založena na bezproudovém měření rozdílového potenciálu mezi měřicí a srovnávací (referenční) elektrodou, jejíž potenciál na styku s roztokem elektrolytu je nezávislý na koncentraci iontů. Potenciál měřicí elektrody je dán Nernstovou rovnicí

$$\varphi = \varphi_0 \pm \frac{RT}{nF} \ln \alpha_i = \varphi_0 \pm \frac{2,303 RT}{nF} \log \alpha_i \quad (1.1)$$

$\varphi$	potenciál měřicí elektrody
$\varphi_0$	standardní potenciál (při aktivitě $\alpha=1$ )
$R$	plynová konstanta $R=8,3144 \text{ J} \cdot \text{K}^{-1} \cdot \text{mol}^{-1}$
$T$	termodynamická teplota [K]
$n$	valence (oxidační číslo) iontů
$F$	Faradayova konstanta $F=9,64870 \cdot 10^4 \text{ C} \cdot \text{mol}^{-1}$
$\alpha_i$	aktivita iontů, kladné znaménko platí pro kationty, záporné pro anionty

Měřicí a srovnávací elektroda vytvářejí při ponoření do elektrolytu elektrický člunek, jehož napětí je dané rozdílem potenciálů na obou elektrodách.

Potenciometrické metody se dále dělí dle toho, zdali se měří hodnota pH, redoxní potenciál nebo koncentrace iontů [2].

## 1.2 Ampérometrie

Je založena na měření proudu procházejícího mezi dvěma elektrodami ponořenými do roztoku elektrolytu, přičemž je obvykle do měřicího obvodu zapojen zdroj stejnosměrného napětí. Jedna z elektrod (označována jako elektroda pracovní, měřicí nebo jako katoda) je polarizovatelná (např. z Pt, Au), druhá elektroda (označovaná jako vztažná nebo jako anoda) je nepolarizovatelná (Pb, Ag/AgCl, Zn aj.). Jsou-li elektrody ponořené do roztoku, který obsahuje danou analyzovanou látku, vytvoří se chemickou reakcí na rozhraní roztoku a měřicí elektrody depolarizátor [2].

## 2 Cyklická voltametrie

Cyklická voltametrie (cyclic voltammetry) je jednou z mnoha metod odvozených od polarografie, při které prochází zkoumanou soustavou elektrický proud. Při CV je zkoumaný roztok podroben potenciálu vloženému na elektrody tak, že potenciál je lineárně zvyšován od počátečního (initial) k „zlomovému“ (vertex) potenciálu, což je tzv. dopředný (forward) scan, a poté je snižován ke konečnému (final) potenciálu (zpětný – reverse scan). Počáteční potenciál je zpravidla shodný s konečným potenciálem, dopředný a zpětný scan pak tvoří jeden cyklus. Podle potřeby se provádí jeden nebo více cyklů, pokud je technika omezena jen na polovinu cyklu, hovoříme o LSV technice (linear sweep voltammetry). Rychlost, s jakou je potenciál měněn (scan rate), určuje časové okno experimentu.

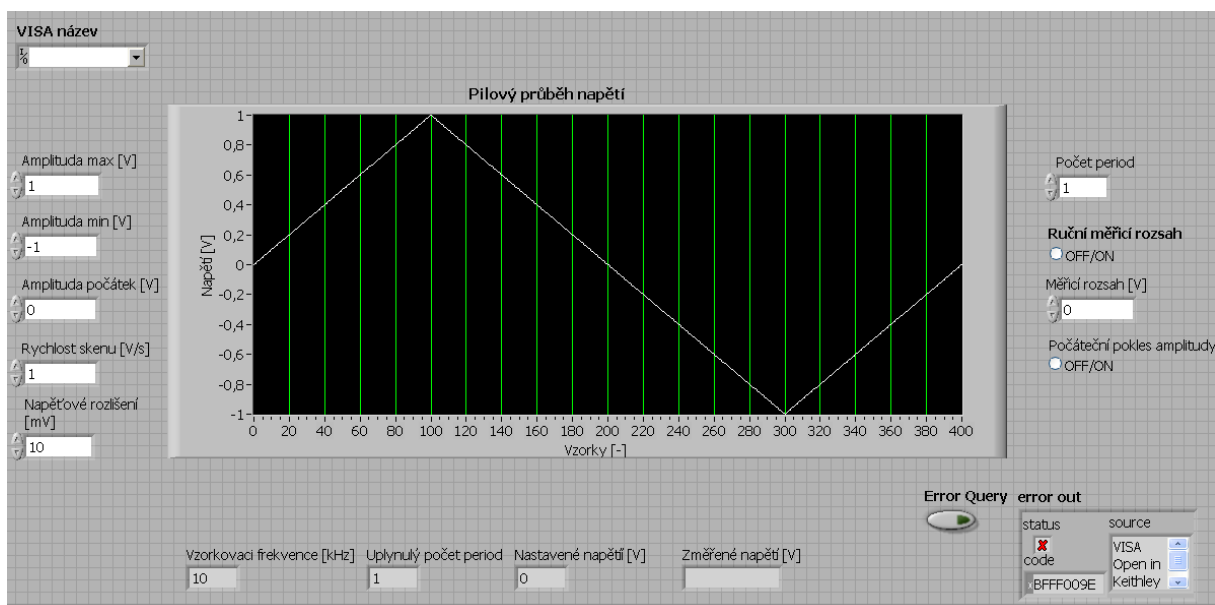
Klíčovým rysem cyklické voltametrie je možnost při zpětném scanu reoxidovat či znovu zredukovat produkt vzniklý během dopředného scanu. Ze separace píků ve voltagramu, z jejich tvarů, poměrů jejich výšek a ze změn těchto parametrů s rychlostí scanu lze získat rozmanité informace o kinetice přenosu náboje, usoudit na existenci reakcí, které přenosu náboje předcházejí nebo jej následují apod.. CV voltamogram může být komplikovaný, pokud se některá látka zapojená do sekvence elektrodových reakcí adsorbuje na povrch elektrody. Redoxní přeměny adsorbovaných látek totiž obecně probíhají při potenciálech odlišných od potenciálů, při kterých se přeměňují látky volně přítomné v roztoku. Potenciál píku odpovídající oxidaci adsorbované látky je stejný jako potenciál píku odpovídající její redukci. Proto je voltagram adsorbované látky symetrický podle proudové osy a lze jej snadno rozeznat [3].

Z uvedeného vyplývá, že pro účely cyklické voltametrie je vhodné používat pilový průběh napětí.

### 3 Vývoj měřicího softwaru

V následujícím textu popisují postupný vývoj softwaru v prostředí LabVIEW. Zpočátku uvádím nezbytné části k pochopení způsobu práce s LabVIEW. Poté obecně seznamuji s přístrojem Keithley. A následně v krocích popisují konkrétní postup tvorby výsledného softwaru určeného pro cyklickou voltametrii.

Jelikož správné průběžné vytváření představy je důležité pro snadné pochopení výsledné práce, tak nejdříve zobrazují ovládací rozhraní a až teprve v pozdějších částech její kontinuálně dovysvětlím. Toto rozhraní je uvedeno na *Obr. 3.1*.



*Obr. 3.1 Ovládací rozhraní*

#### 3.1 Úvod do LabVIEW

LabVIEW může být chápán jako nástroj virtuální instrumentace, přičemž přesný význam této zkratky je *Laboratory Virtual Instruments Engineering Workbench* čili *laboratorní pracoviště virtuálních přístrojů*.

Hlavním cílem virtuální instrumentace je nahradit využití technických prostředků řešením virtuálním za přispění programového vybavení, zejména pak grafickými a vizuálními nástroji, a uživateli tak zprostředkovat maximální názornost. Jde o produkt americké firmy National Instruments (NI) [4].

### 3.1.1 Čelní panel (Front Panel)

Čelní panel tvoří uživatelské rozhraní zvolené aplikace a určuje její vzhled a chování. Přes jeho objekty (ovládací a indikační prvky) lze řídit běh aplikace, zadávat parametry a získávat informace o zpracovaných výsledcích. V okně čelního panelu tedy uživatel vytváří jak vnější vzhled aplikace, tak i prvotní představu o chování programu na základě vstupů a výstupů [4].

### 3.1.2 Blokový diagram (Block Diagram)

Obrazovka blokového diagramu je druhým sdruženým oknem každé aplikace. Na této obrazovce uživatel definuje vlastní algoritmus programu, tedy propojení prvků z čelního panelu a jejich parametry. Každý zde obsažený prvek má podle své funkce nedefinované vstupní či výstupní připojovací body (piny). Jednotlivé prvky vzájemně propojené tvoří zdrojový kód programu, který je takto realizován grafickými objekty [4].

### 3.1.3 Datový tok – vzájemné předávání parametrů

Vykonávání programu v LabVIEW je řízeno datovým tokem a nikoli lineárním vykonáváním řádků kódu, jak je obvyklé u textových programovacích jazyků, kde jsou jednotlivé části programu řazeny sekvenčně. Datový tok tak jednoznačně určuje směr provádění programu [4].

### 3.1.4 Palety

Jsou grafickými panely obsahující různé nástroje a objekty pro vývoj aplikací. Jejich využití je prakticky od definování proměnných přes vytváření logických struktur až po funkce vykonávající složitější algoritmy. Palety mohou být vkládány na *přední panel* nebo do *blokového diagramu*, ale nutno podotknout, že mají rozdílný obsah [4].

### 3.1.5 Virtual Instrument

*Virtual Instrument* označován zkratkou VI v překladu znamená *virtuální přístroj*. Tento druh virtualizace by měl být chápán jako finální řešení daného problému, jež je tvořeno blokovým diagramem a čelním panelem, ve kterých jsou obsaženy používané palety vzájemně propojené skrze datový tok.

Pokud VI není finálním řešením, pak je označováno jako subVI, které je ekvivalentem k funkcím textového programovacího jazyka.

### 3.2 Keithley 2636A

Přístroj Keithley řady 2600 je označován jako systém obsahující zdroj a měřicí přístroj, který nabízí elektronické části a polovodičová zařízení vyrobená za účelem škálovatelnosti, vysokého výkonu a nákladově efektivního řešení pro práci se signály stejnosměrnými (DC), pulzními a nízkofrekvenčními (AC) [5].

Tyto přístroje je možné ovládat dvojím způsobem. První možností je manuální nastavování přes tlačítka jeho čelního panelu, které se od běžných přístrojů příliš neliší. Druhou možností je nastavování skrze příkazy komunikačního protokolu, který je ve své podstatě souborem řetězců reprezentující příkazy [5].

Použití komunikačního protokolu pro účely cyklické voltametrie je téměř nutností. Tímto lze přístroji předkládat požadované hodnoty výstupního napětí a měřit skutečné hodnoty napětí a proudu. Dalším důležitým příkazem, i přestože protokol jich má samozřejmě mnohem více, je uvedení přístroje do režimu 4-wire, který při doporučeném zapojení udržuje na sledovaném výstupu požadované napětí (viz literatura [5]).

Jelikož v další části jsou uvedeny některé obrázky s kanálem A, tak pouze dovysvětlím, že tento přístroj má kanál A a kanál B a výsledná práce dovoluje jeden z nich zvolit.

### 3.3 Komunikace LabVIEW s připojenými zařízeními

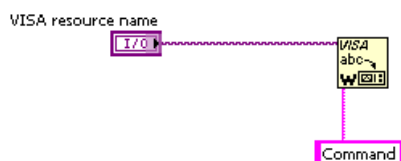
Jelikož program je spuštěn na PC, tak je potřeba řešit spojení PC a přístroj. Přístroj nabízí rozhraní GPIB nebo TCP/IP. Vzhledem k tomu, že byl k dispozici převodník GPIB – USB, tak byla dána přednost tomuto řešení.

Pro správnou funkčnost bylo zapotřebí nainstalovat na PC ovladače převodníku, které jsou dodávány výrobcem a následně propojit přístroj skrze převodník s USB portem počítače. Posléze bylo velmi snadné v LabVIEW navázat spojení se zařízením.

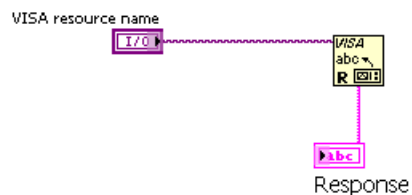
Komunikace je v LabVIEW řešena velmi univerzálně. Společnost NI používá pro komunikaci s vnějšími zařízeními standard VISA, který slouží pro konfiguraci, programování a ladění přístrojových systémů přes GPIB, VXI, PXI, Seriál, Ethernet nebo USB rozhraní [4.3].

Nejdříve je potřeba navázat spojení, o což se postará funkce VISA open z ovladačů VISA. Poté pro poslání příkazu je postačující použít funkci VISA write, které je předtím předán parametr typu řetězec, jež představuje příkaz, a který je takto následně odeslán (viz Obr. 3.2). Pro zpracování odpovědi se použije funkce VISA read, jejíž výstupem je řetězec příchozích dat (viz Obr. 3.3).





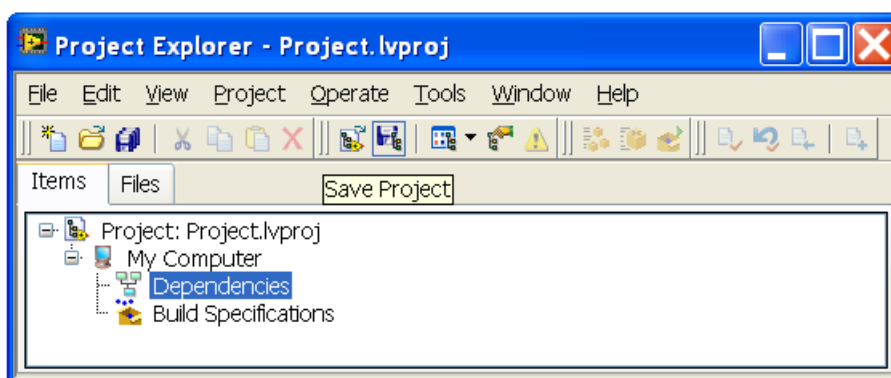
Obr. 3.2 VISA write



Obr. 3.3 VISA read

### 3.4 Vytvoření projektu

Po spuštění LabVIEW lze vytvořit nový projekt postupem **File**→**New Project**, poté se objeví následující okno (Obr. 3.4).

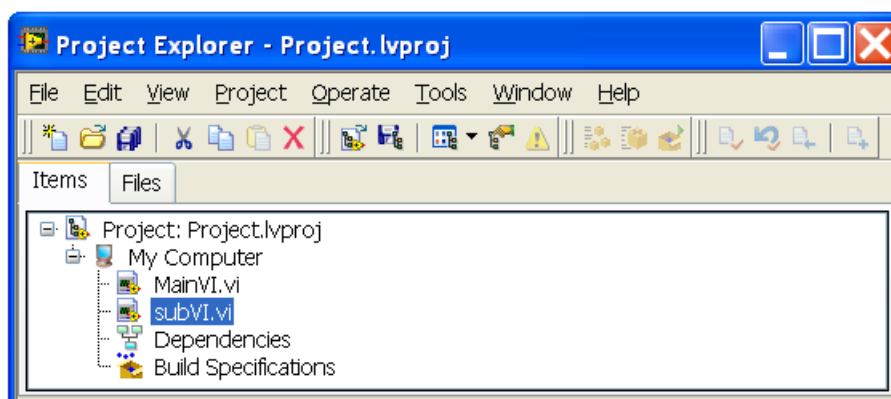


Obr. 3.4 Nový projekt

Tento projekt v současnosti nebude nic neobsahovat. Později může obsahovat vytvořené podprogramy, použité knihovny a případně vytvořenou spustitelnou *exe* aplikaci.

### 3.5 Vytvoření VI

Do tohoto projektu je vhodné vložit hlavní virtuální přístroj (VI), který slouží jako prvotní přední panel a současně blokový diagram, a který by měl být hlavním abstraktem celé práce. Jednotlivé nižší abstrakce by měly být realizovány pomocí tzv. subVI, které by měly být uloženy rovněž zde (Obr. 3.5).



Obr. 3.5 Virtuální přístroj v projektu

### 3.6 Knihovna Keithley 2600

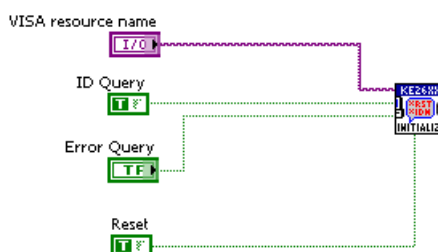
Pro komunikaci s přístrojem slouží komunikační protokol přístroje, kde příkazy a odpovědi jsou zprostředkovány ovladači VISA. Pro posláni příkazu přístroji je postačující naplnit řetězec některým příkazem komunikačního protokolu a poté provést zápis přes funkci VISA write. Obdobně to platí pro příjem odpovědi, kde nás může zajímat příchozí řetězec, který bude vyčten přes funkci VISA read.

Pokud bychom používali tento zápis komunikace, pak by vznikl jeden nepřehledný kus kódu, třebaže i grafického. Ovšem výrobce Keithley nabízí lepší řešení, a tím je využití knihoven přístrojů Keithley řady 2600 určených pro prostředí LabVIEW. Následující část tyto funkce blíže popisuje.

### 3.7 Části programu

#### 3.7.1 Initialize

Inicializuje GPIB rozhraní spojení PC – přístroj. Toto subVI může rovněž inicializovat TCP/IP, USB nebo sériové rozhraní. Pro představu je uveden obrázek (Obr. 3.6).



Obr. 3.6 Initialize

Úkolem je zavolat funkci *VISA open*, následně *VISA clear*, což povede k navázání spojení. Poté vzhledem k tomu, že je aktivováno *ID Query*, je poslán příkaz *\*IDN?* pomocí *VISA write*, na který je vráceno přes *VISA read* tovární číslo modelu, sériové číslo a firmware jednotky (dále v následujícím textu kvůli přehlednosti už nebudu uvádět odeslání příkazu přes *VISA write* a obdobně ani příjem odpovědi přes *VISA read*). Pokud by po těchto krocích nastala chyba, pak následně dojde k uzavření spojení přes funkci *VISA close*.

### 3.7.2 Node Format

Účelem této funkce je nastavit na výstupním řetězci hodnotu *localnode.*, který slouží pro komunikaci s místní jednotkou přístroje (Obr. 3.7).



Obr. 3.7 Node Format

Toho je dosaženo tak, že na vstup celočíselného typu je přivedena hodnota 0. Čehož je následně využito při volání všech příkazů směřujících na přístroj.

### 3.7.3 Reset SMU

Provede obnovení kanálu A i kanálu B do defaultního stavu (Obr. 3.8).

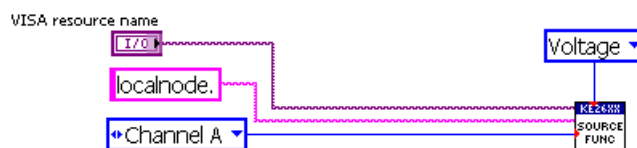


Obr. 3.8 Reset SMU

Volba resetování kanálu je tu skrze booleanské proměnné, které jsou v aktivním stavu. Konkrétně je poslán příkaz *localnode.smua.reset()* a *localnode.smub.reset()*.

### 3.7.4 Source Function

Nastavuje zdroj přístroje na napěťový zdroj (na Obr. 3.9 je uvedeno nastavení pro kanál A).

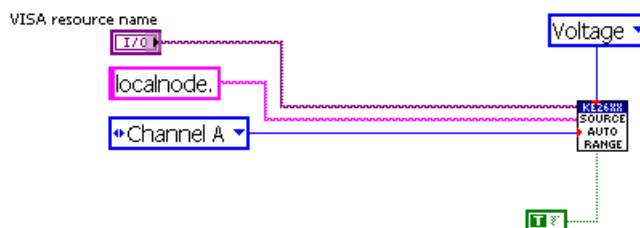


Obr. 3.9 Source Function

Zde byl použit příkaz `localnode.smua.source.func = smua.OUTPUT_DCVOLTS`.

### 3.7.5 Source Autorange

Slouží pro aktivaci automatického rozsahu výstupního zdroje nastaveného předchozí funkcí (Obr. 3.10).

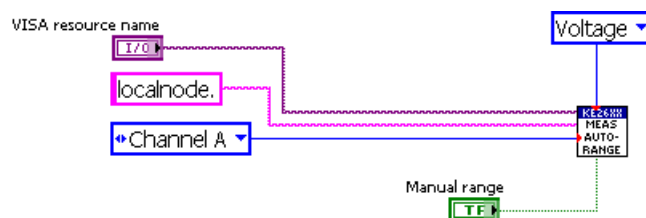


Obr. 3.10 Source Autorange

Volán příkaz `localnode.smuX.source.aurangev = "1"`. Přičemž "1" je typu string neboli řetězec a způsobuje aktivaci. Hodnota "0" by provedla deaktivaci.

### 3.7.6 Measure Autorange

Pokud uživatel nezvolil manuální přepínání rozsahů, pak je tato funkce prováděna, jinak nikoliv. Jejím účelem je aktivovat automatické přepínání měřených rozsahů (Obr. 3.11).

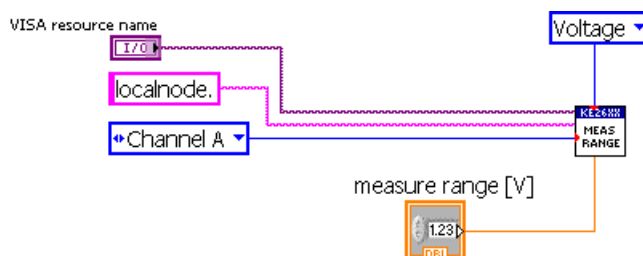


Obr. 3.11 Measure Autorange

Volán příkaz `localnode.smua.measure.aurangev = "1"`. Přičemž "1" způsobuje aktivaci a "0" deaktivaci.

### 3.7.7 Measure Range

Pokud uživatel zvolil manuální přepínání rozsahů, pak se tato funkce provede namísto funkce *Measure Autorange* (Obr. 3.12).

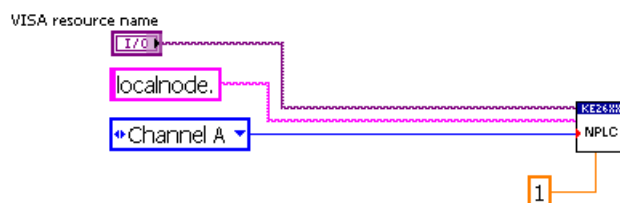


Obr. 3.12 Measure Range

Uživatel takto může přes vstupní hodnotu *measure range* nastavit požadovaný rozsah měřeného napětí. Volány příkazy `localnode.smua.measure.rangev = hodnota`, kde *hodnota* je rozsah napětí a je to číslo s plovoucí řadovou čárkou (v LabVIEW označováno jako DBL).

### 3.7.8 NPLC

Určuje rychlost měření signálu. Tento údaj je závislý na napájecí frekvenci elektrického síťového rozvodu (Obr. 3.13).

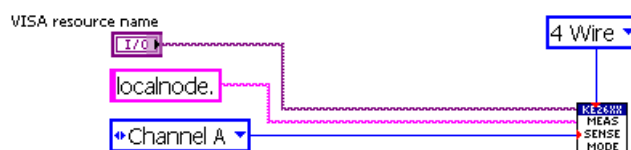


Obr. 3.13 NPLC

Pro bližší informace je vhodné použít datasheet přístroje (viz literatura [5]). Z příkazů je použit příkaz `localnode.smua.measure.nplc = hodnota`.

### 3.7.9 Measure Sense Mode

Tato funkce je volána z důvodu přepnutí přístroje do režimu 4–drát (Obr. 3.14). Pro více informací je vhodné použít literaturu (viz literatura [5]).

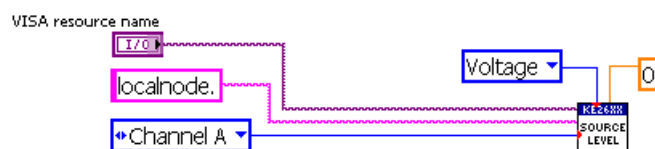


Obr. 3.14 Measure Sense Mode

Volán příkaz `localnode.smua.sense = smua.SENSE_REMOTE`.

### 3.7.10 Source Level

Tato funkce má nastavit hodnotu zdroje napětí. V této části jí je využita pro nastavení nulového napětí (Obr. 3.15).

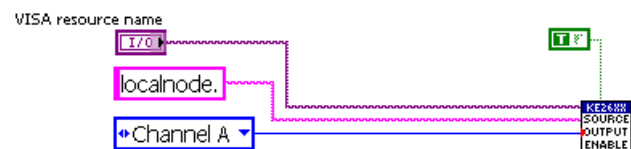


Obr. 3.15 Source Level

Volán příkaz `localnode.smua.source.levelv = hodnota`.

### 3.7.11 Source Output Enable

Touto funkcí je aktivován výstup zdroje napětí (Obr. 3.16).



Obr. 3.16 Source Output Enable

Vykonán příkaz `localnode.smua.source.output = "1"`.

### 3.7.12 Cyklus opakování

Zde jsou uvedeny části, které je potřeba cyklicky provádět. Účelem smyčky je opakovaně počítat hodnotu napětí, kterou má přístroj nastavit, průběžně zaznamenávat naměřené hodnoty napětí a proudu s časovou značkou do souboru a čekat přesně definovaný čas do dalšího opakování smyčky. Současně je zobrazován nastavovaný průběh napětí, který by měl odpovídat pilovému průběhu. Smyčka je automaticky ukončena po splnění požadovaného počtu period.

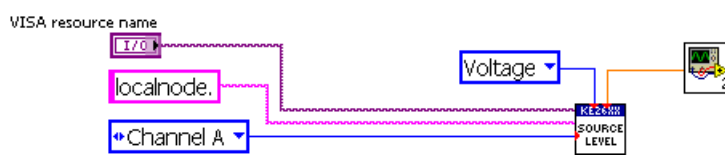
### 3.7.12.1 Aktuální hodnota pilového průběhu

Z cyklické voltametrie vyplývá, že je žádoucí vytvářet pilový průběh napětí. Výsledný pilový průběh je založen na opakovaném volání subVI generátoruPily, o což se stará cyklus smyčky. Toto subVI po každém zavolání vrací dvě hodnoty. První hodnota odpovídá aktuální hodnotě amplitudy pilového průběhu. Druhá hodnota odpovídá přesnému času v milisekundách, po který se čeká, než je ukončen jeden cyklus smyčky. Tento čas je během programu konstantní. Ve výsledku to znamená, že opakovaným prováděním postupně vzniká vzorkovaný pilový průběh. Opakování se provádí do doby, než uplyne zadaný počet period. Podotýkám, že v principu je využíváno toho, že subVI si pamatuje naposledy vypočtené hodnoty proměnných používaných uvnitř tohoto subVI.

Z důvodu přehlednosti předkládaných částí smyčky je podrobnější popis uveden po jeho skončení.

### 3.7.12.2 Source Level

Tato funkce byla zmíněna už výše. Zde ve smyčce se už nastavuje konkrétní hodnota napětí, která je generována vytvořeným subVI generátorPily (Obr. 3.17).

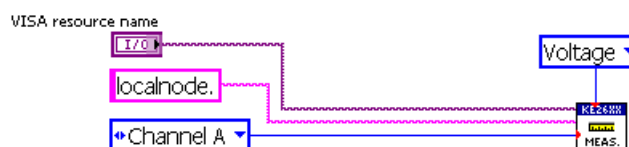


Obr. 3.17 Source Level

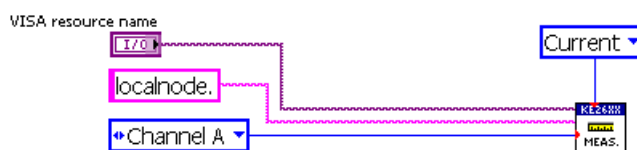
Hodnota je odeslána přístroji, který ji nastaví na svém napěťovém výstupu.

### 3.7.12.3 Measure

Z přístroje vyčte naměřenou hodnotu a vyvede ji na výstup jako řetězec. Nejdříve je tato funkce použita pro měření napětí (Obr. 3.18) a poté pro měření proudu (Obr. 3.19).



Obr. 3.18 Measure Voltage



Obr. 3.19 Measure Current

Volaný příkaz je `printnumber(localnode.smua.measure.v())` v prvním případě a ve druhém případě je `printnumber(localnode.smua.measure.i())`.

#### 3.7.12.4 Zápis hodnot do souboru

Před vstupem do smyčky je otevřen soubor. Do souboru je průběžně ukládána časová známka v ms, nastavené a změřené napětí ve voltech, změřený proud v ampérech a nakonec je přidán znak CR a LF, aby příští zápis do souboru byl proveden na začátek nového řádku (jde o řešení pro operační systém MS Windows).

Ukládané hodnoty do souboru jsou mezi sebou odděleny středníkem, což téměř odpovídá formátu csv (comma separated values), kde hodnoty jsou odděleny čárkou. Takto vytvořený soubor je možné načíst tabulkovým procesorem bez žádných dalších nutných úprav, který má větší možnosti zpracování dat.

#### 3.7.12.5 Vykreslování grafu

V každém cyklu je generovaná hodnota, která je posílána do zdroje napětí, zobrazována do grafu, jež je nastaven na automatické přizpůsobení velikosti zobrazovaných hodnot. Před prvnotním vstupem do smyčky je graf vymazán od možných předchozích hodnot.

#### 3.7.12.6 Přesné čekání

Přesným čekáním vzniká vzorkovaný průběh. Proto je nezbytné odpočítávat čas od předchozího posledního čekání, jinak by výsledný průběh neměl jasně definovaný tvar a byl by závislý na čase stráveném vnitřním výpočtem.

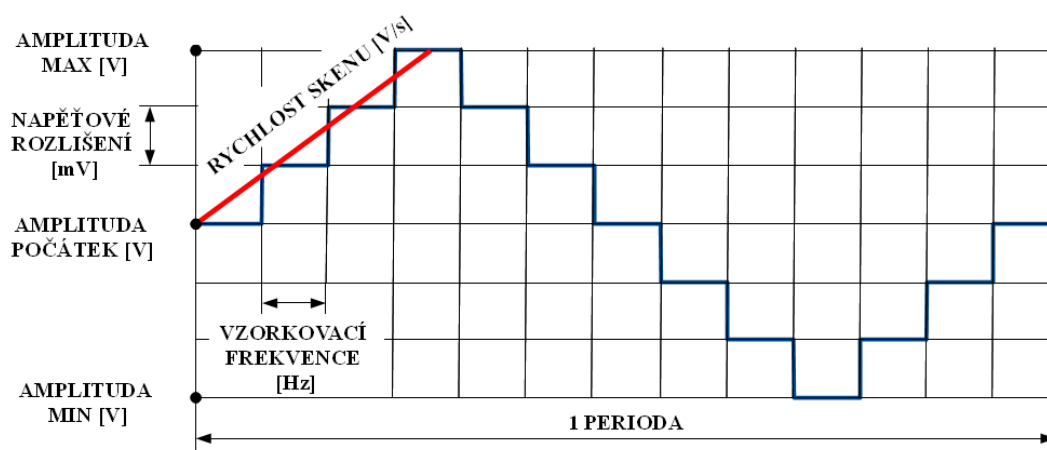
#### 3.7.13 Ukončení cyklu a celého programu

Před ukončením programu je provedeno nejdříve uzavření souboru, poté vypnutí výstupního napětí zdroje a následně je ukončeno spojení s přístrojem.



### 3.8 Generátor pily

Generátor pily slouží pro vytvoření pilového průběhu, který je využíván pro nastavování napětí přístroje. Podotýkám, že jeho nedílnou součástí je opakovací smyčka. Pro základní představu řešení je na *Obr. 3.20* naznačen jeden z možných průběhů, na kterém jsou zobrazeny vstupní požadované hodnoty vzhledem k vytvořenému průběhu (obrázek je pouze ilustrativní).



*Obr. 3.20 Pilový průběh*

#### 3.8.1 Parametry pilového průběhu

Parametry je možné měnit následujícími proměnnými, které jsou přehledně zobrazeny v tabulce *Tab. 3.1*.

*Tab. 3.1 Vstupní proměnné pilového průběhu*

Název	Účel
Amplituda max [v]	Maximální hodnota amplitudy
Amplituda min [v]	Minimální hodnota amplitudy
Amplituda počátek [v]	Počáteční hodnota amplitudy
Napěťové rozlišení [mv]	Velikost kroku amplitudy
Rychlost skenu [v/s]	Rychlost růstu resp. poklesu amplitudy
Vzorkovací frekvence [hz]	Frekvence generování hodnot

### 3.8.2 Výpočet vzorkovací frekvence

*Vzorkovací frekvence* je konstantní v rámci zadaných vstupních hodnot pilového průběhu. Její výpočet je podle následujícího vzorce.

$$\text{vzorkovací frekvence [kHz]} = \frac{\text{rychlost skenu} \left[ \frac{V}{s} \right]}{\text{napětové rozlišení [mV]}} \quad (3.1)$$

### 3.8.3 Výpočet amplitudy

Hodnota aktuální amplitudy při každém zavolání je inkrementována resp. dekrementována o hodnotu *napětového rozlišení*. Výsledkem je pak růst nebo pokles amplitudy.

$$\begin{aligned} \text{amplituda} &= \text{amplituda} + \text{napětové rozlišení} \\ &\text{resp.} \\ \text{amplituda} &= \text{amplituda} - \text{napětové rozlišení} \end{aligned} \quad (3.2)$$

### 3.8.4 Přepínání mezi růstem a poklesem amplitudy

K tomu aby docházelo k opakovanému střídání růstu a poklesu v zadaných mezích, což je potřeba pro pilový průběh, slouží přepínač. K přepnutí dojde, pokud následující hodnota amplitudy je při součtu resp. rozdílu s *napětovým rozlišením* větší než *amplituda max* resp. menší než *amplituda min*.

$$\begin{aligned} \text{if } ((\text{amplituda} - \text{napětové rozlišení}) < \text{amplitudaMin}) \quad \text{přepínač} &= 1 \\ &\text{else} \\ \text{if } ((\text{amplituda} + \text{napětové rozlišení}) > \text{amplitudaMax}) \quad \text{přepínač} &= 0 \end{aligned} \quad (3.3)$$

### 3.8.5 Výpočet počtu period

Během generování amplitudy jsou počítány počty přepnutí, čehož je využito při vyhodnocování počtu uplynulých period. Vychází se z principu, že po dvou přepnutích byla vygenerována jedna perioda. Tento výpočet je navíc upraven tak, aby v něm byla zahrnuta i hodnota *amplitudy počátek*.

$$\begin{aligned}
 & \text{if } (\text{přepínač} == 1) \\
 & \quad \text{if } ((\text{počet přepnutí} \geq 2) \wedge (\text{amplituda} \geq \text{amplituda počátek})) \\
 & \quad \quad \text{perioda} = \text{perioda} + 1 \\
 & \quad \quad \quad \text{resp.} \\
 & \text{if } (\text{přepínač} == 0) \\
 & \quad \text{if } ((\text{počet přepnutí} \geq 2) \wedge (\text{amplituda} \leq \text{amplituda počátek})) \\
 & \quad \quad \text{perioda} = \text{perioda} + 1
 \end{aligned} \tag{3.4}$$

K takto zvolenému výpočtu je doplněno ošetření správnosti výsledku při zadání *amplitudy počátek* rovné *amplitudě max* resp. *amplitudě min*.

### 3.8.6 Doplnující ošetření

Další ošetření, které je nutné provádět, je převádění zadávaných vstupních dat z čísla s plovoucí řadovou čárkou na číslo celočíselného typu. To se ukázalo jako nezbytné pro správnou funkčnost. Při tomto zvoleném algoritmu výpočtu pilového průběhu mohl vzniknout problém vyhodnocení přesného zlomu přepnutí z růstu amplitudy na pokles resp. naopak. Řešením bylo vynásobit vstupní proměnné s plovoucí řadovou čárkou, následně je převést na celočíselný typ, poté provádět vnitřní výpočty a před vrácením požadované hodnoty tuto proměnnou vhodně vydělit.

### 3.8.7 Počáteční nastavení hodnot signálem reset

Jelikož z použitého principu je nutné nastavit počáteční hodnoty generátoru pily, tak byl vytvořen signál reset. Při aktivním resetu dojde k vynulování proměnných *počtu zlomů*, *počtu period* a nastavení amplitudy na hodnotu *amplituda počátek*. V této části je rovněž načtena hodnota přepínače, která určuje, jestli generovaná pila bude zpočátku růst nebo klesat.

## **4 Závěr**

Výsledné řešení je naprogramováno ryze grafickým způsobem, ale nutno podotknout, že obdobného výsledku by bylo možné dosáhnout klasickou psanou formou programování. Otázkou by bylo, jestli by to bylo přehledné pro skupinu lidí, kterým je tento produkt určen.

Software byl vyvíjen konkrétně pro přístroje Keithley řady 2600. A bylo by na zváženu, jestli by bylo žádoucí zvýšit počet podporovaných přístrojů.

Z hlediska bezpečnosti senzoru, který tímto softwarem bude měřen, mohu říci, že použitý algoritmus generátoru pilového průběhu nedovoluje, aby výstupní signál byl větší než jaké jsou požadované meze.

Vzhledem k tomu, že jde o nový produkt, tak bych očekával, že svým dlouhodobým používáním vzniknou nápady o jeho doplnění.

## 5 Použitá literatura

- [1] *Electrochemical Sensors*, [online], 8.6.2012, [www.intlsensor.com](http://www.intlsensor.com)
- [2] Ripka, P, Ďaďo, S., Kreidl, M., Novák, J.: *Senzory a převodníky*, skripta ČVUT, Praha 2005
- [3] *Cyklická voltametrie*, [online], 8.6.2012, <http://fch.upol.cz/skripta/pok1>
- [4] Vlach, J.: *Začínáme s LabVIEW*, BEN, technická literatura, Praha 2008
- [5] *Series 2600 System SourceMeter Instruments*, datasheet, [online], [www.keithley.com](http://www.keithley.com)

## 6 Přílohy

### Příloha A - Blokový diagram virtuálního přístroje

