

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

## GENEROVÁNÍ GRAFŮ V DIADEMU Z LABVIEW

LABVIEW GRAPHS GENERATION VIA DIADEM

### BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

Karel Kučera

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jan Kunz

BRNO 2020



# Bakalářská práce

bakalářský studijní program **Automatizační a měřicí technika**

Ústav automatizace a měřicí techniky

**Student:** Karel Kučera

**ID:** 203577

**Ročník:** 3

**Akademický rok:** 2019/20

**NÁZEV TÉMATU:**

## Generování grafů v DIAdemu z LabVIEW

### POKYNY PRO VYPRACOVÁNÍ:

Cílem bakalářské práce je vytvořit program, který bude generovat vektorové i bitmapové grafy z prostředí LabVIEW pomocí programu DIAdem. Zadání lze shrnout do následujících bodů:

- 1) Seznamte se s programem DIAdem a zpracujte přehled možností komunikace mezi programy LabVIEW a DIAdem.
- 2) Vytvořte program, který bude automaticky generovat 2D, včetně XY a intenzitního, a 3D grafy z prostředí LabVIEW za pomoci programu DIAdem. Tento program bude plně nastavitelný z prostředí LabVIEW.
- 3) Existující program doplňte o možnost uživatelských změn parametrů, například popisky a číslování os, fonty, barvy, legenda atd.
- 4) Výsledný program upravte do formy knihovny do LabVIEW (.lvlib) a řádně jej, včetně obsluhy a vlastností, zdokumentujte.

### DOPORUČENÁ LITERATURA:

VLACH, Jaroslav, Josef HAVLÍČEK a Martin VLACH. Začínáme s LabVIEW. Ilustroval Viktorie VLACHOVÁ. Praha: BEN - technická literatura, 2008. ISBN 978-80-7300-245-9

National Instruments. NI DIAdem: Getting Started with DIAdem. 2017.

<http://www.ni.com/pdf/manuals/373422p.pdf>

**Termín zadání:** 3.2.2020

**Termín odevzdání:** 8.6.2020

**Vedoucí práce:** Ing. Jan Kunz

**doc. Ing. Václav Jirsík, CSc.**  
předseda rady studijního programu

### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Cílem této práce je vytvoření programu který bude vytvářet a ukládat grafy pomocí programu DIAdem. Program je rozdělen do dvou částí kdy jedna se nachází v prostředí LabVIEW a druhá jako script v jazyce VBS. Výsledné řešení dokáže vytvořit a parametrizovat graf čistě z prostředí LabVIEW bez nutnosti přímého spouštění scriptu.

## **KLÍČOVÁ SLOVA**

LabVIEW, DIAdem, Graf, VBS, VI, Export, Komunikace

## **ABSTRACT**

The aim of this work is to create a program, which will create charts and save using DIAdem. The program is divided into two parts, one in LabVIEW and the second as a script in VBS. The resulting solution can create and parameterize the graph purely from the LabVIEW environment without the need to run the script directly.

## **KEYWORDS**

LabVIEW, DIAdem, chart, VBS, VI, Export, Communication

KUČERA, Karel. *Generování grafů v DIAdemu z LabVIEW*. Brno, 2020, 39 s. Bachelářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce: Ing. Jan Kunz



## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Generování grafů v DIAdemu z LabVIEW“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno 8.6.2020 .....

.....

podpis autora

## PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Janu Kunzovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

# Obsah

|                                           |           |
|-------------------------------------------|-----------|
| Úvod                                      | 9         |
| <b>1 Grafy</b>                            | <b>10</b> |
| 1.1 Typografické zásady grafů             | 10        |
| 1.1.1 Nadpisy, popisky a vysvětlivky      | 10        |
| 1.1.2 Osy a měřítka                       | 11        |
| 1.1.3 Estetika                            | 11        |
| <b>2 LabVIEW a DIAdem</b>                 | <b>12</b> |
| 2.1 Virtuální instrumentace               | 12        |
| 2.2 LabVIEW                               | 12        |
| 2.2.1 Front panel                         | 13        |
| 2.2.2 Block Diagram                       | 13        |
| 2.2.3 Grafy                               | 13        |
| 2.2.4 3D grafy                            | 14        |
| 2.2.5 Přímé ukládání Grafů                | 15        |
| 2.2.6 Práce se soubory                    | 15        |
| 2.2.7 Formáty souborů                     | 15        |
| 2.3 DIAdem                                | 16        |
| 2.3.1 NAVIGATOR                           | 16        |
| 2.3.2 REPORT                              | 16        |
| 2.3.3 SCRIPT                              | 17        |
| <b>3 volání programů LabVIEW a DIAdem</b> | <b>18</b> |
| 3.1 Přímé volání                          | 18        |
| 3.2 Nepřímé volání                        | 18        |
| <b>4 Návrh</b>                            | <b>19</b> |
| 4.1 Návrh Parametry                       | 19        |
| 4.2 Požadavky                             | 19        |
| <b>5 Implementace</b>                     | <b>20</b> |
| 5.1 Popis chování LabVIEW                 | 20        |
| 5.1.1 Předání hodnot                      | 21        |
| 5.1.2 Rozpoznání grafu                    | 21        |
| 5.1.3 Uložení Dat                         | 21        |
| 5.1.4 Parametry                           | 22        |
| 5.1.5 TDMS soubor                         | 22        |

|          |                                         |           |
|----------|-----------------------------------------|-----------|
| 5.1.6    | spuštění DIADEMU . . . . .              | 22        |
| 5.2      | Popis chování DIAdem . . . . .          | 23        |
| 5.2.1    | Smazání starých dat a objektů . . . . . | 24        |
| 5.2.2    | Načtení nových dat . . . . .            | 24        |
| 5.2.3    | Souřadnicový systém . . . . .           | 24        |
| 5.2.4    | Křivka . . . . .                        | 24        |
| 5.2.5    | Uložení grafu . . . . .                 | 24        |
| <b>6</b> | <b>Výsledky studentské práce</b>        | <b>25</b> |
| 6.1      | 2D graf - Lineární osy . . . . .        | 25        |
| 6.1.1    | LabVIEW . . . . .                       | 26        |
| 6.1.2    | Aplikace . . . . .                      | 27        |
| 6.2      | 2D graf - Logaritmické osy . . . . .    | 29        |
| 6.2.1    | LabVIEW . . . . .                       | 29        |
| 6.2.2    | Aplikace . . . . .                      | 30        |
| 6.3      | 3D graf . . . . .                       | 31        |
| 6.3.1    | LabVIEW . . . . .                       | 31        |
| 6.3.2    | Aplikace . . . . .                      | 32        |
| 6.4      | Vzhled grafů . . . . .                  | 36        |
| <b>7</b> | <b>Problematika</b>                     | <b>37</b> |
| 7.1      | Kompatibilita . . . . .                 | 37        |
| 7.2      | spuštění . . . . .                      | 37        |
| 7.3      | Problematické grafy . . . . .           | 37        |
|          | <b>Závěr</b>                            | <b>38</b> |
|          | <b>Literatura</b>                       | <b>39</b> |

# Seznam obrázků

|      |                                                                      |    |
|------|----------------------------------------------------------------------|----|
| 1.1  | Porovnání popisků grafu . . . . .                                    | 11 |
| 2.1  | Porovnání nastavení zobrazení . . . . .                              | 14 |
| 5.1  | Vývojový diagram pro LabVIEW část . . . . .                          | 20 |
| 5.2  | Vývojový diagram pro DIAdemovou část . . . . .                       | 23 |
| 6.1  | Vzhled grafu přímo ve vývojovém prostředí . . . . .                  | 26 |
| 6.2  | Export LabVIEW grafu s úpravou vzhledu . . . . .                     | 26 |
| 6.3  | Export grafu bez úprav pomocí vytvořeného programu . . . . .         | 27 |
| 6.4  | Export grafu s úpravou vzhledu pomocí vytvořeného programu . . . . . | 28 |
| 6.5  | Vzhled grafu přímo ve vývojovém prostředí . . . . .                  | 29 |
| 6.6  | Export LabVIEW grafu s úpravou vzhledu . . . . .                     | 29 |
| 6.7  | Export grafu s úpravou vzhledu pomocí vytvořeného programu . . . . . | 30 |
| 6.8  | Vzhled grafu přímo ve vývojovém prostředí . . . . .                  | 31 |
| 6.9  | Export LabVIEW grafu s úpravou vzhledu . . . . .                     | 32 |
| 6.10 | Export grafu bez úprav pomocí vytvořeného programu . . . . .         | 33 |
| 6.11 | Export grafu s úpravou vzhledu pomocí vytvořeného programu . . . . . | 33 |
| 6.12 | Pohled XZ . . . . .                                                  | 34 |
| 6.13 | Pohled XY . . . . .                                                  | 34 |
| 6.14 | Pohled YZ . . . . .                                                  | 35 |
| 6.15 | Export grafu bez úprav pomocí vytvořeného programu . . . . .         | 36 |

# Úvod

Vývojové prostředí LabVIEW, které vytváří společnost National Instruments, (NI) je vhodné pro měření a analýzu dat, nebo pro řízení a vizualizaci technologických procesů. V oblasti analýzy dat je důležité grafické zpracování naměřených dat, protože právě z tvaru průběhu jsme schopni zhodnotit výsledek. LabView ale nenabízí nejlepší možnost generování grafů a proto se v této práci budu zabývat možností využití programu DIAdem, který je vytvořen od stejné společnosti, pro generování a nastavení ideálních grafů přímo z prostředí LabVIEW.

# 1 Grafy

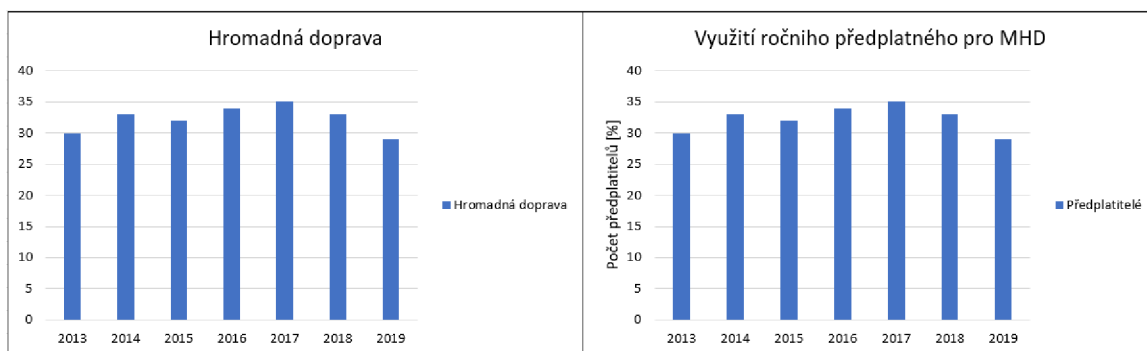
Hlavním významem grafu je správná komunikace dat, která nám umožňuje efektivně a přehledně znázornit vztahy a trendy čísel. Oproti uspořádání v tabulce je tedy největší výhodnou názornost a snadná prezentace dalším osobám. Nevýhodou ovšem bývá, že graf neudává přesně číselné hodnoty a proto je nutné si uvědomit kdy je vhodné použít pro prezentaci graf a kdy tabulku. Tabulka je tedy většinou použita všude kde je žádoucí sdělit přesné výsledky a nebo je dat málo. Graf se naopak hodí pro velké množství dat a jejich porovnání. Při použití grafu je ještě nutné si uvědomit jaký typ grafu by se měl použít na základě sdělení, které má být předáno. Například pro porovnání diskrétních hodnot je vhodný sloupcový graf, ale pro znázornění časové změny hodnot je výhodnější použít graf spojnicový. Dalším důležitým rozhodnutím při tvorbě grafu je volba množství zobrazených dat. Každý graf je omezený svými rozměry a schopností jednotlivá data od sebe rozlišit. Pokud se tato omezení překročí, dojde k znehlednění výsledného čtení. Data s průběhy dat se nejčastěji rozlišují pomocí barvy, zde je ale nutné myslet na vhodnou volbu barvy. Pokud se například zvolí méně výrazná barva, může se stát, že na některých zařízeních nebude barva vůbec vidět. Dále se může nastat situace kdy výsledná práce bude tisknuta v černobíle verzi, která může ohrozit čitelnost grafů. [6] [7]

## 1.1 Typografické zásady grafů

Cílem typografie je čtenáři zajistit co nejlehčí a nejefektivnější čtení výsledné práce. Taktéž i zamezit možným chybám. Práce které dodržují typografická doporučení působí profesionálněji.

### 1.1.1 Nadpisy, popisky a vysvětlivky

Nadpis grafu je určený pro připravení čtenáře na sdělení. Je proto vhodné si uvědomit, které informace je vhodné napsat přímo do názvu a které do popisku grafu. Název grafu by tedy měl být snadno čitelný a přesně popisující co graf sděluje. Toto sdělení by mělo být zřejmé čistě z grafu bez nutnosti dalšího textového popisu, proto je zapotřebí aby všechny osy byly správně popsány včetně jednotek, které se na nich vyskytují. Stejně tak se přidává legenda a jeli to žádané může být přidána další vysvětlivka. Pro dobrý vzhled je vhodné ohlídat aby se v jednom grafu neobjevovalo více fontů. Stejně jako s daty je u popisek nutné si uvědomit, zda je vhodné jej do grafu umístit. Pokud se v oblasti vyskytuje příliš mnoho popisků, může to zapříčinit znehlednění grafu. [6] [7]



Obr. 1.1: Porovnání popisků grafu

Na obrázku č. 1.1 jsou znázorněny dva grafy, které se liší pouze v popiskách. Graf který je umístěn na levé straně je bez popsaní v textu nic neříkající. Při použití správných popisků (graf na pravé straně), je graf již možné použít a vysvětlit samostatně. Například z nic neříkajícího grafu, se stal přehled kolik procent populace daného města využívá ročního předplatného.

### 1.1.2 Osy a měřítka

Volbou osy lze změnit celkové vyznění grafu. Pokud je cílem grafu ukázat určitou část dat, není nutné zobrazit celou oblast sběru dat, ale pouze úsek který je důležitý. Stejně tak je ale nutné myslet na nebezpečí manipulací grafy. Pokud je například cílem grafu porovnání hodnot, pak je možné volbou osy výrazně ovlivnit způsob vnímání čtenáře. Stejně tak volbou rozměrů grafu je možné vzhled strmosti růstu nebo poklesu dat. [6] [7]

### 1.1.3 Estetika

Při tvorbě grafu záleží také na použitém programu při jeho tvorbě. Mnoho programů jsou zaměřena pro práci v oblasti managementu, nikoliv v té akademické a proto tyto programy přidávají ozdoby aby výsledný graf vypadal zajímavěji. Ty ozdoby ovšem můžou v některých případech zkreslit význam jednotlivých úseků grafu. Odborný název pro akademicky nežádoucí prvky grafu je chartjunk. Pod toto označené spadá vše co data vizuálně zanáší a přebíjí (například příliš silná mřížka). Mezi efekty, které nejvíce zkreslují výsledná data, patří 3D efekty pro jinak 2D data. [6]



## 2 LabVIEW a DIAdem

### 2.1 Virtuální instrumentace

Virtuální instrumentace dále jen VI umožňuje hned několik aplikací. Běžně se používá pro získání dat do elektronické podoby a to tak, že program z VI se připojí ke skutečným přístrojům, které jsou zapojené na pracovišti a provádí stejná měření jako člověk, který by přístroje obsluhoval. Dále umožňuje okamžité zpracování naměřeného signálu včetně úprav signálu. Výhodou VI je, že některé nákladné přístroje, které by byly potřeba při reálném zapojení, lze nahradit elektronickým kódem a tím snížit náklady na měření, například generátory šumu, průběhů, filtry a pod. National Instruments navíc dává možnost připojení svých vstupních a výstupních karet a tím mít jistotu, že získaná data budou řádně přečtena. Další výhodou je, že pokud bude program VI dobře vytvořený, pak se sníží nejistota měření vzniklá chybou obsluhy měření. Dále lze měření převést z laboratoře kamkoliv jinam, takže obsluha může měřit centrálně z jednoho místa bez nutnosti reálného kontaktu s měřícím přístrojem. Bohužel pořízení programu pro VI může být dost nákladné, takže se jeho použití nehodí na všechny aplikace. Existuje několik programů pro VI, například labVIEW, EFLAB [1]

### 2.2 LabVIEW

LabVIEW je vývojové prostředí od firmy National Instruments, které započalo vývoj s představou, že člověk, který je schopen vytvořit blokové schéma své problematiky, pak již intuitivně vytvoří program. Výsledné prostředí je tedy grafický programovací jazyk, který umožňuje rychlejší tvorbu programu než klasický textový. Jazyk je tvořen z funkcí ve tvaru bloků, které jsou spojovány virtuálními vodiči. Grafický způsob zápisu umožňuje také snadnější tvorbu paralelních aplikací pro více jádrové počítače. Název virtuální instrumentace se zkratkou VI si LabVIEW převzalo pro označení přípon svých souborů. Prostředí se skládá ze dvou oken, front panelu a blokového diagramu. Při své práci budu nejvíce řešit možnost komunikace s dalšími programy a předávání si parametrů pomocí souborů. [1]

### 2.2.1 Front panel

Jedná se o úvodní okno, které se spustí při spuštění prostředí. Toto okno je určeno přímo pro uživatele a určuje výsledný vzhled a chování aplikace. Vzhled je dán rozložením objektů jako jsou grafy a textové pole. Pomocí těchto polí poté uživatel může přímo měnit nastavení programu (například zadat rozlišení grafu). Stejně tak může program uživatele informovat o průběhu programu, například textovým výstupem nebo rozsvícením kontrolky. [1]

### 2.2.2 Block Diagram

Okno diagramu je přidružené k Front panelu a je možné jej kdykoliv otevřít. Zde se programuje samotný algoritmus a aplikace pomocí spojování jednotlivých funkcí. Je dobré si uvědomit, že pokud je to možné, pak se LabVIEW snaží, aby výsledný program běžel paralelně. [1]

### 2.2.3 Grafy

LabVIEW nabízí velkou variabilitu v reprezentaci grafických výsledků. Tyto reprezentace mohou být rozděleny na dvě velké skupiny a to na 2D a 3D grafy. Dvourozměrné grafy navíc lze dále dělit podle principu zobrazení na takzvaný "Chart" nebo "Graph".

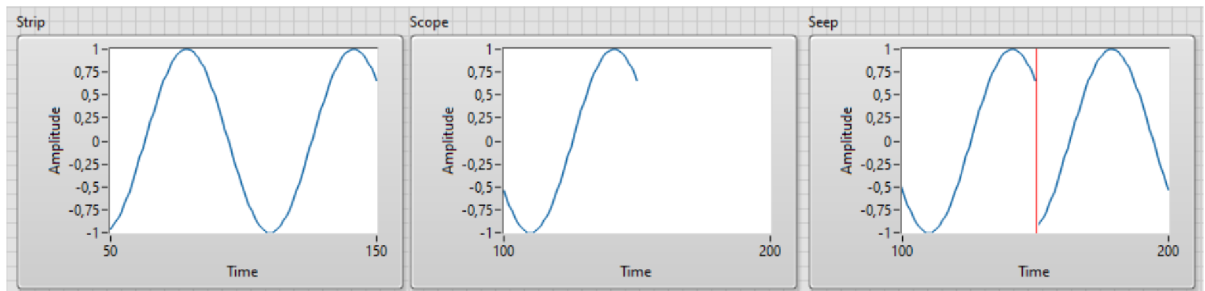
#### Chart

Jedná se o průběžný zapisovač, který se nejčastěji využívá pro zobrazení změny veličiny v čase. Vstupní data jsou předána buď postupně nebo jako celistvý blok. Chart k vstupnímu číslu, které je reprezentováno na ose Y časovou hodnotu na ose X. Data jsou vykreslována okamžitě jak do grafu přijdou a dokud nenastane funkce pro vymazání dat pak všechny body zůstávají v historii. Nejčastější zobrazovač tohoto typu se nazývá Waveform chart. Jak již bylo zmíněno, je využíván pro časové zobrazení kdy jsou hodnoty zobrazovány v přesných časových intervalech. Způsob zobrazování lze volit následovně:

**Strip** Jedná se o výchozí nastavení, kdy jsou data vykreslována tak jak přišly a jakmile dojdou na konec grafu dojde k rozšíření časového rozsahu dokud data nepřestanou chodit

**Scope** Jak již název prozrazuje, jeho funkce je obdobná jako zobrazovač klasického osciloskopu. Pokud tedy data dosáhnou prvního, okraje dojde k novému vykreslování opět zleva

**Sweep** Činnost je obdobná jako u Scope způsobu, ale nová data jsou zobrazována od pohybující se svislé značky a nedochází tak k okamžitému přemazání předešlých průběhů



Obr. 2.1: Porovnání nastavení zobrazení

Na obrázku č. 2.1 lze vidět porovnání jednotlivých nastavení chart grafů. Graf vlevo je nastavený typu strip. Graf uprostřed je typu scope a vpravo je nastavení typu sweep. Na těchto grafech lze, vidět že stejná data můžou v závislosti na prezentaci zobrazení působit dosti odlišně.

## Graph

Jedná se o statický grafický zobrazovač. Pro zobrazení průběhů je nutné data předem připravit tak, aby popisovali celý průběh a následně je předat jako celek. Mezi dva nejčastější příklady těchto zobrazovačů patří Waveform a XY graph. Funkce obou příkladů se nijak zvlášť neliší.

### 2.2.4 3D grafy

Druhým velkým rozdělením grafů je dělení do 2D a 3D grafů. Do 2D grafů patří výše zmíněné. Tří rozměrné grafy se liší už v samotném LabVIEW a to tím, že jsou jiného typu, respektive spadají do tříd kdy každý typ grafu je samostatnou třídou. Základem každé této třídy je obecná třída popisující graf. Jelikož je základ každé třídy stejný, lze využít pro manipulaci s daty stejné metody pro každou ze tříd. Vstupní data bývají upravovány podle zvolené třídy a mohou být buď jako 1D nebo 2D pole hodnot.

## 2.2.5 Přímé ukládání Grafů

Každý z grafických zobrazovačů lze uložit přímo jako obrázek. Tyto obrázky ale mají pouze omezené možnosti formátování a jejich využití v dokumenty tedy nemusí být vždy vhodná volba. Kromě exportu do obrázku je ještě možné exportovat do:

- Excelu
- Clipboardu
- DIAdemu

v těchto případech se dle zvolených možností přenesou pouze data tvořící průběhy za účelem následného zpracování, uložení nebo zobrazení.

## 2.2.6 Práce se soubory

Pro práci se soubory nabízí LabVIEW paletu File I/O, která obsahuje funkce, které dokáží:

- otevřít, vytvořit a uzavřít soubory,
- práci s tabulkami,
- změnit charakteristiku souboru,
- přejmenovat soubory a adresáře,
- přesunout soubory a adresáře,
- číst a zapisovat data.

Při každé práci se souborem je zapotřebí nadefinovat cestu k souboru (File path). Poté co je soubor otevřen, je reprezentován referenčním číslem (refnum) [1]

## 2.2.7 Formáty souborů

LabView pracuje s třemi základními formáty souborů, které umožňují spolupráci s textovými editory, tabulkovými kalkulátory a nebo mezi jednotlivými VI (Virtuálními Instrumenty)

**Textový datový tok** Nejjednodušší typ formátu, ve kterém se data ukládají a čtou jako ASCII znaky

**Binární datový tok** V tomto formátu každé číslo zabírá pevně danou délku bitů. Zápis i čtení je velice efektivní a výsledné, soubory zabírají málo místa

**Datalog** Speciální formát, který se využívá jen v rámci prostředí LabView, je schopen efektivně pracovat s datovými strukturami. Jako jsou řetězce, klastry a čísla. Struktura výsledného souboru je podobná tabulkovému zápisu. [1] [2] [3]

## 2.3 DIAdem

DIAdem je program určený ke zpracování předem naměřených dat. Jednou z jeho výhod je, že je optimalizovaný pro velké množství dat a umožňuje efektivní hledání konkrétních dat. Pro zpracování umožňuje transformování dat pomocí specifických analytických funkcí (např. Fourierova transformace), vyhodnocení dat a následně i úpravu a tisk grafu. Dále pro možnost automatizace zpracování dat, které by bylo pokaždé obdobné, umožňuje napsání scriptu v jazyce VBS (Visual Basic Script), který je pak spustitelný a tím se ušetří výsledný čas a práce. Ve své práci se zaměřím především na tvorbu grafů v kolonce REPORT a právě na psaní skriptu. [2] [4] [5]

### 2.3.1 NAVIGATOR

Jedná se o úvodní a základní okno, které se v DIAdemu vyskytuje. Zde je možné prohledávat soubory počítače, aby se do diadému mohl nahrát soubor s daty, které má zpracovat. Při instalaci programu DIAdem se do počítače stáhnou i dema (předem vytvořené data, grafy a skripty), která ukazují a popisují možnosti práce s celým programem. Právě za pomoci navigátoru je možné tyto ukázková data nahrát, popřípadě spustit ukázkový script. [2] [5]

### 2.3.2 REPORT

Záložka Report je jednou z oken, které DIAdem vytváří a slouží k vytvoření výstupu zpracovaných dat dle potřeby přání uživatele. Celý výstup se dá volně upravovat dle přání a umožňuje přidávání různých objektů (grafy, tabulky, loga, text a další). Výsledný návrh se poté může uložit různými způsoby, a to od uložení do obrázku v různých formátech, až po uložení do PDF souboru, popřípadě poslat přímo na tisk do tiskárny. Pokud uživatel očekává, že grafický návrh, který použil, se bude často opakovat, může si rozložení uložit a při příští práci znovu nahrát a tím urychlit svou práci. Záložka report je funkce, která bude využita pro vytváření grafů. [2] [5]

### 2.3.3 SCRIPT

Záložka určená pro vytváření scriptu jazyka VBS. Zdrojový kód se dá vytvořit různými způsoby a krom klasického psaní má funkce, které i nezkušeným uživatelům umožní lehčí cestou dosáhnout tíženého výsledku. První z možností je mód nahrávání, který poté co je spuštěný, nahrává veškerou činnost, kterou uživatel v programu dělá a pokud to je možné, pak akce převede na sadu příkazů, které zajistí stejnou činnost jako akce uživatele. Druhou možností je zkopírování existujícího objektu (například v záložce report) na zdrojový kód skriptu, který se pak jen vloží do úseku, kde jej uživatel chce mít. Při psaní je dobré si uvědomit, že téměř ke všemu v DIAdemu se dá přistupovat jako k objektu. [2] [4] [5]

## 3 volání programů LabVIEW a DIAdem

### 3.1 Přímé volání

#### Volání DIAdemu z LabVIEW

Pro LabView byly vytvořeny speciální VIs programy, které jsou určené pro práci s programem Diadem. Tyto VIs se nenacházejí přímo v knihovně funkcí, a proto se musí přetahovat z místa uložení, které nejčastěji bývá „/National Instruments/LabVIEW VERSIOR/vi.lib/ExportToExcel“. Po vložení do programu se s nimi pracuje jako s běžnými oficiálními funkcemi. Práce se scriptem se skládá ze 3 VIs, a to sice z otevření VBS skriptu, spuštění a následně uzavření. Pokud chceme, aby script pracoval s daty vytvořenými pomocí LabVIEW, je zapotřebí vytvořit sdílenou složku, do které budou přistupovat oba programy a navzájem si předávat informace. [5]

#### Volání LabVIEW z DIAdemu

Stejně jako jde spustit DIAdemový script z LabVIEW, to jde i opačně. V LabVIEW je nutné vytvořit subVI, který bude mít definované vstupy a výstupy. Poté již stačí jen vytvořit skript, který zapomocí funkcí, které jsou vytvořené přímo pro DIAdem, otevře a spustí námi vytvořený program. [5]

### 3.2 Nepřímé volání

LabView má možnost pro práci s příkazovým řádkem. Je tedy možné vytvořit skript pro příkazový řádek, který bude spouštěn přímo z LabView a uživatel tak nebude nucen sám přistupovat do samotného příkazového řádku. Práce s příkazovým řádkem může probíhat dvojím způsobem.

1. Labview spustí příkazový řádek na pozadí a začne vykonávat skript. Výsledek pak navrátí zpět do prostředí jako řetězec. Důležité je zmínit, že pokud je řádek spouštěn na pozadí, pak se zbytek programu nevykonává, dokud neskončí veškeré úkony, které byly příkazovému řádku dány. V tomto případě uživatel nevidí vykonávání skriptu, protože se neotevře žádné viditelné okno.
2. Druhou možností je, že se otevře nové okno s příkazovým řádkem, který začne vykonávat skript. Rozdílem oproti prvnímu řešení je především to, že program nečeká na vykonání příkazového řádku, a pokud může, pak pokračuje s vykonáváním zbytku programu. Dalším rozdílem je, že uživatel přímo vidí otevřený příkazový řádek včetně průběhu a výsledku skriptu. [5]

## 4 Návrh

Při navrhování funkce programu jsem se snažil o vytváření co nejmenšího množství přebytečných souborů. Dalším důležitým faktorem návrhu bylo přehledné nastavování parametrů grafů. Vzhled aplikace ve vývojovém prostředí se snaží co nejvíce napodobit vzhled běžných LabVIEW funkcí. Původní návrh tedy byl pouze 2 soubory programu a jeden soubor pro výměnu dat a parametrů. Množství souborů ale bylo rozšířeno z důvodu třírozměrných grafů. Výsledný návrh je tedy vytvořen na 4 programové soubory. Struktura programu je navržena tak, aby soubory mohly být uloženy kdekoliv v počítači, ale musí zůstat ve složkách v jakých byly vytvořeny.

### 4.1 Návrh Parametry

Ovládání je navrženo, tak aby měl uživatel možnost kompletně upravit veškeré vlastnosti programu. Parametry které jsou navrženy k pravě jsou:

- Velikost grafu
- Barva pozadí grafu
- Barva os
- Font písma programu
- Velikost popisků
- Zvolení textu popisků os a grafu
- Tloušťka os
- Nastavení Min a Max hodnoty na osách
- Nastavené hlavních a vedlejších jednotek na ose
- Možnost zapnutí/vypnutí hladké křivky
- Zvolení typu os (Lineární, Logaritmické)
- Zapnutí vypnutí mřížky
- Nastavení vzhledu mřížky
- Zapnutí/Vypnutí legendy
- Úprava legendy
- Možnost zvolení složky pro uložení grafu
- Možnost zvolit formát uložení grafu

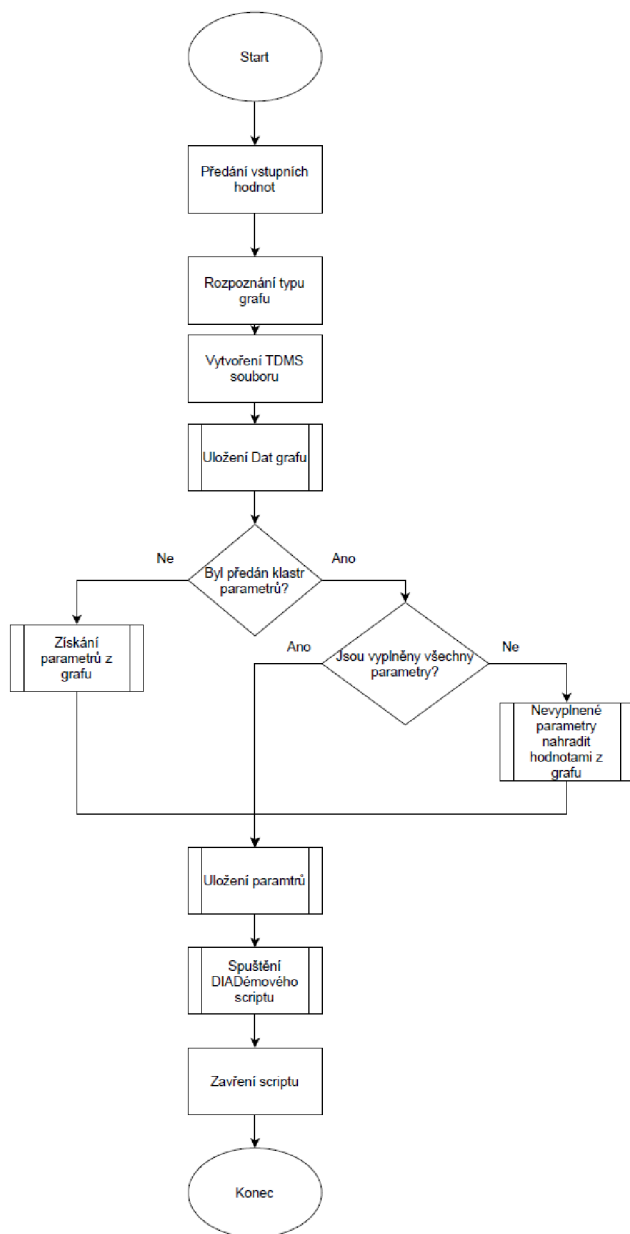
### 4.2 Požadavky

Program má pouze jeden vstup, který je vyžadován a tím je předaný graf. Všechny ostatní vstupy a nastavení nejsou vyžadovány. Pokud nejsou předány nebo nejsou vyplněny, program použije vnitřní nastavení pro úpravu grafu do výsledné podoby



# 5 Implementace

## 5.1 Popis chování LabVIEW



Obr. 5.1: Vývojový diagram pro LabVIEW část

### 5.1.1 Předání hodnot

Při vložení programu do VI je pouze jeden drát, který je nutný aby byl zapojený pro funkční program a tím je reference na graf, který má být uložen. Jako další parametry které, ale již nejsou povinné pro chod programu jsou:

**Error klastr:** Pro chod programu není povinný, ale přesto je doporučený jej zapojit aby se co nejvíce zamezilo nesprávnému průběhu programu.

**Klastr parametrů:** Slouží pro úpravu vzhledu grafu. Klastr je vytvořen jako typová definice, lze jej tedy vyvolat pro ovládání popřípadě jako konstanta. Nevyplněné číselné hodnoty jsou označeny jako NaN (Not a number). Pro předání klastru nemusí být vyplněny všechny položky, ale stačí jen ty které je žádáno upravit. Pokud klastr nebude předán nebo v něm bude nevyplněná položka pak je výsledný graf nastaven podle parametrů získaných z předaného grafu. Výsledný vzhled tedy lze do jisté míry nastavit úpravou grafu v LabVIEW.

**Path:** Pokud si uživatel přeje výsledný graf uložit na konkrétní místo v počítači, pak jednoduše předá cestu s žadáním místem. Pokud políčko zůstane nevyplněno, pak se graf uloží do předem nastavené složky.

### 5.1.2 Rozpoznání grafu

Příchozí reference na graf je přijata jako obecný grafický objekt. Pro bližší práci s daty je tedy nutné, aby byl graf přesně určen. Z obecného objektu lze krom jiných věcí získat jméno a ID typu grafu. Této vlastnosti je využito, neboť na základě těchto parametrů lze referenci více specializovat a následně pracovat s daty které se nacházejí uvnitř.

### 5.1.3 Uložení Dat

#### Získání dat

Před samotným uložením dat je nutné je nejprve získat. Možnosti získání hodnot se liší podle typu grafu.

**2D Graph** Hodnoty, které předává reference grafu jsou v datovém typu variant, který není možné přímo uložit. Je tedy nutné data nejprve převést do formátu, který je totožný se vstupem do grafu. Na základě známého grafu a známého formátu kterým je možné vykreslit křivku v grafu je možné variant správně převést.

**2D chart** Jak již bylo řečeno, chart se nejčastěji používá pro sledování časové změny. Data jsou do něj postupně zasílána, a proto je nelze jednoduše získat jako v předchozím případě. Data jsou tedy získána pomocí exportu do schránky.

**3D Graph** Na rozdíl od 2D grafů jsou typu třídy a proto se data z variantu musí převést na obecnou třídu trojrozměrného grafu, ze které se za pomoci vytvořených metod lze dostat až k datům která definují křivky

## **Uložení**

Po získání dat je nutné je upravit tak, aby bylo možné je uložit. Nejčastěji tedy převést na pole hodnot na osách nebo jako waveform. Při ukládání dojde k pojmenování skupiny jako data. Následně se pojmenují samotné kanály podle os ke kterým patří.

### **5.1.4 Parametry**

Parametry se předávají pomocí klastru. Pokud žádný klastr nepřišel pak použije obecný nevyplněný klastr. Pro každý z parametrů dojde k porovnání hodnoty z klastru. Pokud program přečte nevyplněnou hodnotu, pak ji nahradí hodnotou z grafu. Poté co dojde porovnání všech hodnot se již opravený klastr rozbalí a převede do pole hodnot pro následující uložení. Spolu s polem parametrů vznikne i stejně velké pole stringů pro pojmenování kanálů parametrů. Při ukládání pak stejně jako u dat dojde ještě k pojmenování nové skupiny aby došlo ke snadnému rozlišení. Po uložení se s TDMS souborem již nebude pracovat a LabVIEW jej může uzavřít, aby s ním byl připravený pracovat DIADEM.

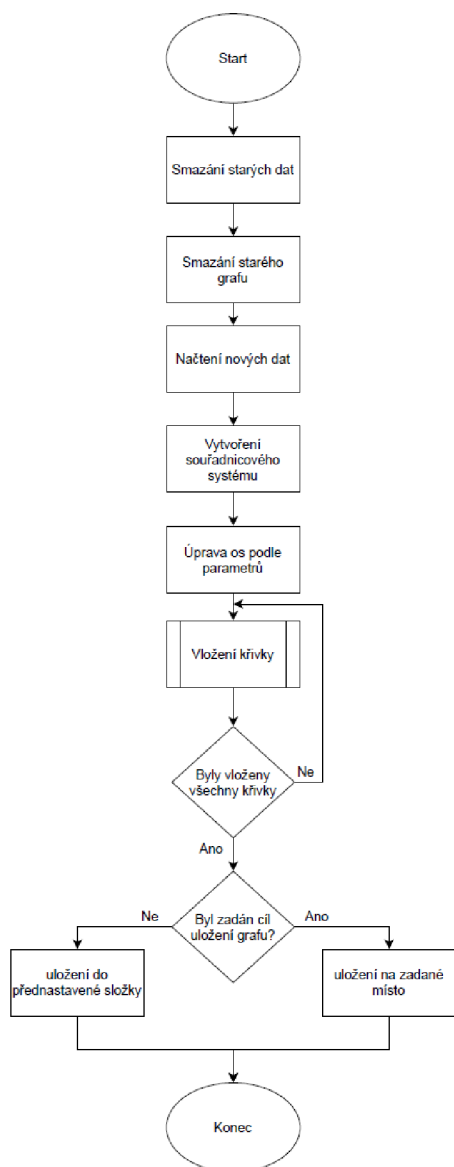
### **5.1.5 TDMS soubor**

Pro komunikaci mezi částmi v LabVIEW a DIADEM je využíván TDMS soubor. Ten se ukládá do předem vytvořené složky, která se nachází v projektu spolu s hlavním programem. Pokud dojde ke smazání složky nebo přesunutí mimo předpokládané umístění dojde k novému vytvoření složky. Jeden TDMS soubor má dvě skupiny, jednu pro data grafu a druhou pro parametry. Celý soubor je dočasný a po dokončení programu dojde k jeho následnému smazání.

### **5.1.6 spuštění DIADEMU**

Pro spuštění scriptu se využívají funkce pro to vytvořené přímo od společnosti NI. Tyto funkce jsou tři, tedy pro otevření scriptu, jeho spuštění a následné ukončení. Pro spuštění je nutné znát umístění souboru, ten je uložený v samostatné složce tak, aby se v souborech dalo snáze vyznat. Tato cesta je spojena s umístěním VI souboru. Po dokončení práce scriptu dojde k jeho ukončení a VI na svém výstupu vrátí přijatou referenci, zadanou cestu a error klastr pro případný error handler

## 5.2 Popis chování DIAdem



Obr. 5.2: Vývojový diagram pro DIAdemovou část

### **5.2.1 Smazání starých dat a objektů**

Při spuštění DIAdemu dojde k nahrání dat, které mají sloužit jako příklad funkce. Tyto reprezentativní data ale nejsou žádána a proto hned po spuštění dojde k jejich smazání. Stejně tak se ve složce report sami vytvoří objekty. Ze stejného důvodu script smaže i tyto objekty a tím dosáhne úplné volnosti při dalším vytváření grafu.

### **5.2.2 Načtení nových dat**

Poté co jsou všechna stará data vymazána, skript načte TDMS soubor s daty a s parametry. Pro přehlednější práci s parametry je každý z parametrů uložen do proměnné v scriptu. Každý parametr je v souboru uložený pod odpovídajícím jménem a tak není nutné řešit pořadí parametrů. Výjimkou jsou parametry jmen (názvy os, grafu, cesty). Ty jsou uloženy na pevně daném pořadí. Jelikož jejich název je zároveň parametrem, který je u každého grafu jiný. Jsou tedy načítány podle svého indexového čísla.

### **5.2.3 Souřadnicový systém**

Podle rozměrů grafu může script vytvořit buď dvourozměrný souřadnicový systém, nebo třírozměrný. Dále všechny další úpravy jako je mřížka a nebo v případě 3D grafu změnit úhel pohledu. Veškeré volby vzhledu jsou upravitelné.

### **5.2.4 Křivka**

Do vytvořeného grafu je zapotřebí ještě vložit křivku. Způsob vkládání křivek se liší v závislosti na typu grafu a typu křivky která má být vložena. Vkládání probíhá tak dlouho, dokud nedojde ke vložení všech křivek. Při vložení je automaticky volena barva, aby bylo možné jednotlivé průběhy rozlišit.

### **5.2.5 Uložení grafu**

Závěrem programu dojde k uložení grafu podle žádaného formátu do zadané složky. Soubor je pojmenován jako stejně jako graf. Po uložení je práce scriptu hotová a ukončí se.

## 6 Výsledky studentské práce

Cílem práce bylo dosažení možnosti uložení lepších grafů než nabízí samotné LabVIEW, proto jsem se rozhodl že nejlepší možnost zhodnocení výsledků práce je porovnáním grafů exportovaných programem LabVIEW a grafů které jsou výsledkem exportu z mé práce

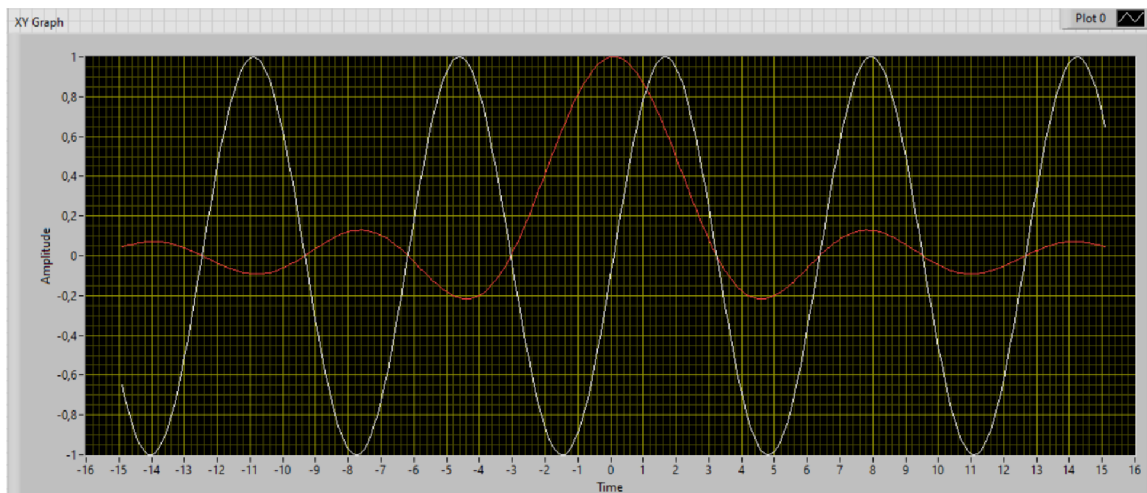
### 6.1 2D graf - Lineární osy

Pro porovnání výsledků jsem si zvolil průběh popsany jako:

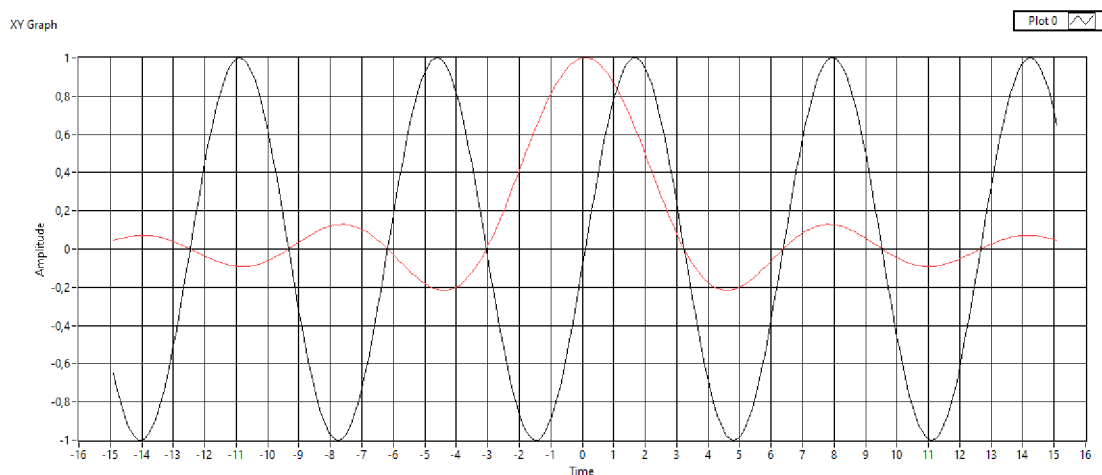
$$f(x) = \frac{\sin(x)}{x}$$

Kdy  $x$  narůstá hodnot od -15 do 15. spolu s průběhem klasické sinusovky pro znázornění možnosti zobrazení několika křivek v jednom grafu.

## 6.1.1 LabVIEW



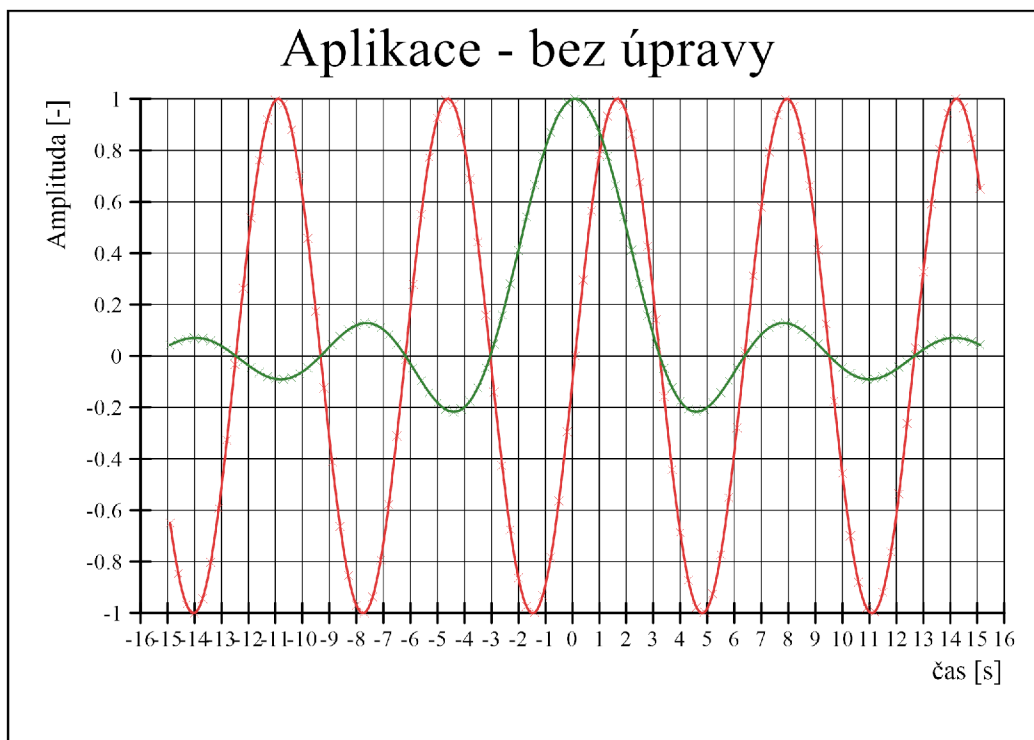
Obr. 6.1: Vzhled grafu přímo ve vývojovém prostředí



Obr. 6.2: Export LabVIEW grafu s úpravou vzhledu

Z grafů č. 6.1 a 6.2 lze vidět, že export je sice možný, ale z klasických formátů jen do BMP. Výsledný graf sice není úplně nepoužitelný, ale lze vidět že se v něm vyskytují nedostatky, které nelze odstranit. Mezi tyto nedostatky patří tloušťka os. Osy nemají žádné rozlišení od mřížky v podobě zvýraznění a proto není hned zřejmé, kde osy leží. Dalším nedostatkem je chybějící sekundární mřížka a možnost nastavit výraznost čar mřížky. Další problematikou je, že velikost exportovaného grafu je přímo odpovídá velikosti grafu ve front panelu.

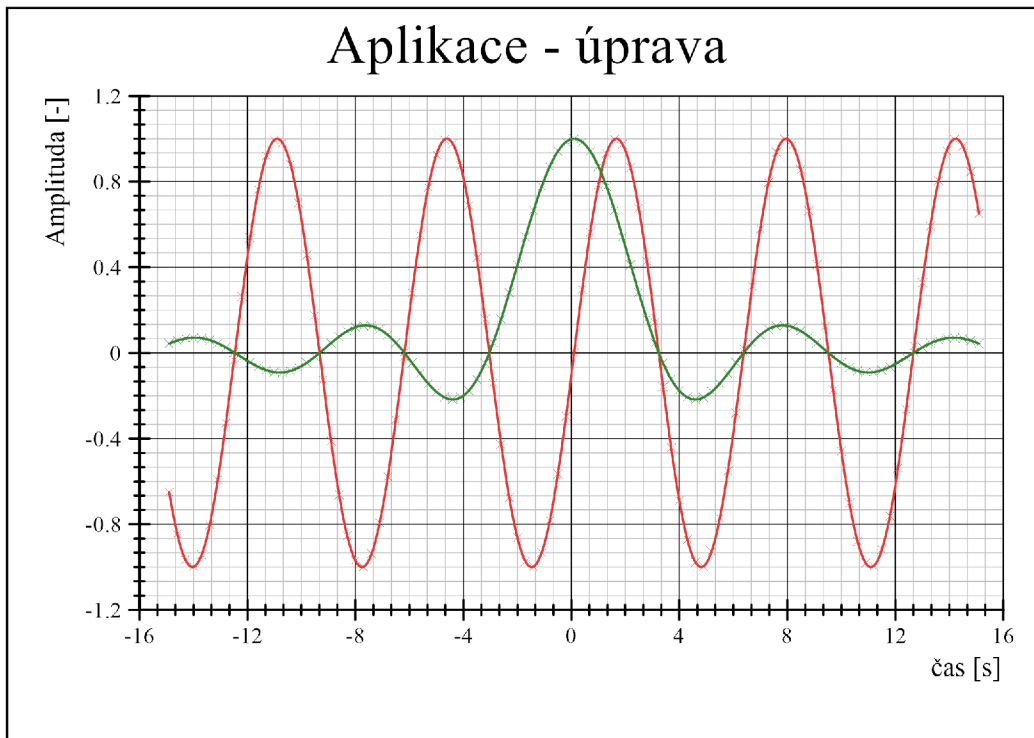
## 6.1.2 Aplikace



Obr. 6.3: Export grafu bez úprav pomocí vytvořeného programu

Porovnáním exportovaného grafu pomocí prostředí LabVIEW (obrázek č. 6.2) a vytvořené aplikace bez předání parametrů na obrázku č. 6.3 si lze všimnout, že se aplikace snaží co nejvíce napodobit vzhled grafu v LabVIEW. Rozdíl však je v tloušťce čar a popisků os.





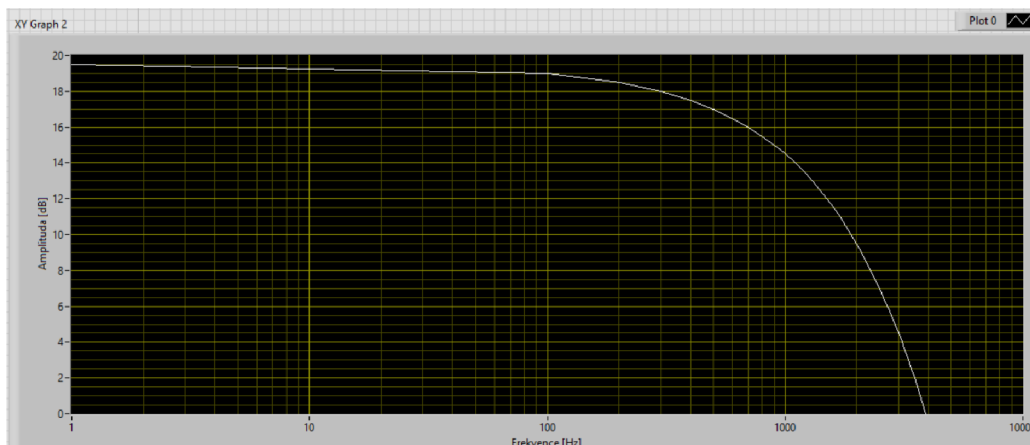
Obr. 6.4: Export grafu s úpravou vzhledu pomocí vytvořeného programu

Obrázek č. 6.4 ukazuje, že změnou předaných parametrů pak lze výsledný vzhled upravit. Pokud jsou předávaná data volena podle typografických zvyklostí, lze dosáhnout přehledného grafu

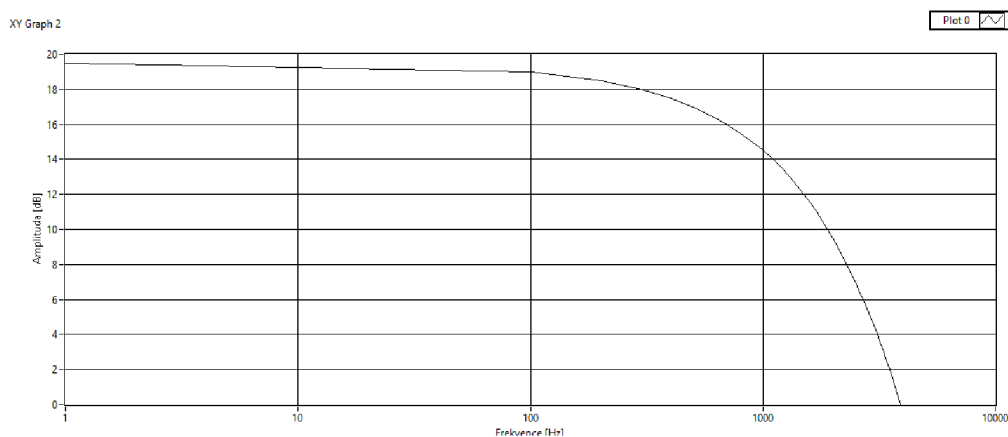
## 6.2 2D graf - Logaritmické osy

Porovnání grafů s použitím logarotmických měřítek jsem se rozhodl na průběhu, který trvá od 1 do 10000 s postupnou změnou amplitudy. Signál je inspirovaný průběhem typu dolní propust

### 6.2.1 LabVIEW

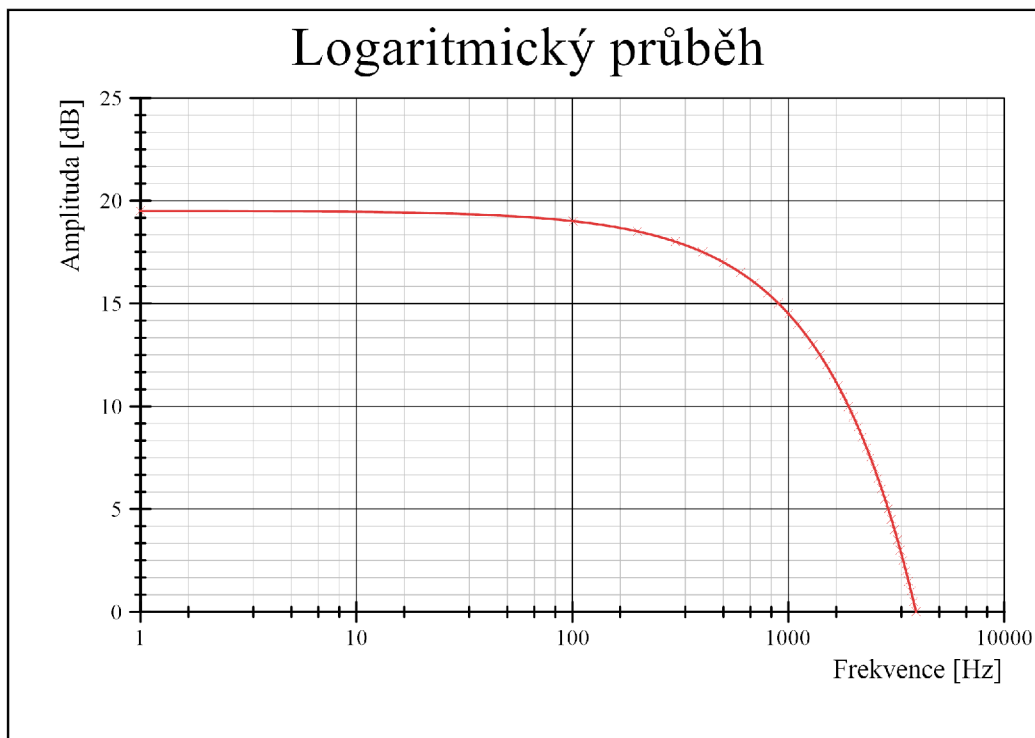


Obr. 6.5: Vzhled grafu přímo ve vývojovém prostředí



Obr. 6.6: Export LabVIEW grafu s úpravou vzhledu

Obrázky č. 6.5 a 6.6 znázorňují, že export funguje stejně jako u lineárních os. V tomto konkrétním případě jde ale mnohem více vidět malá schopnost rozlišení průběhu od mřížky grafu. Graf tedy vyžaduje více soustředění, aby byl srozumitelný.



Obr. 6.7: Export grafu s úpravou vzhledu pomocí vytvořeného programu

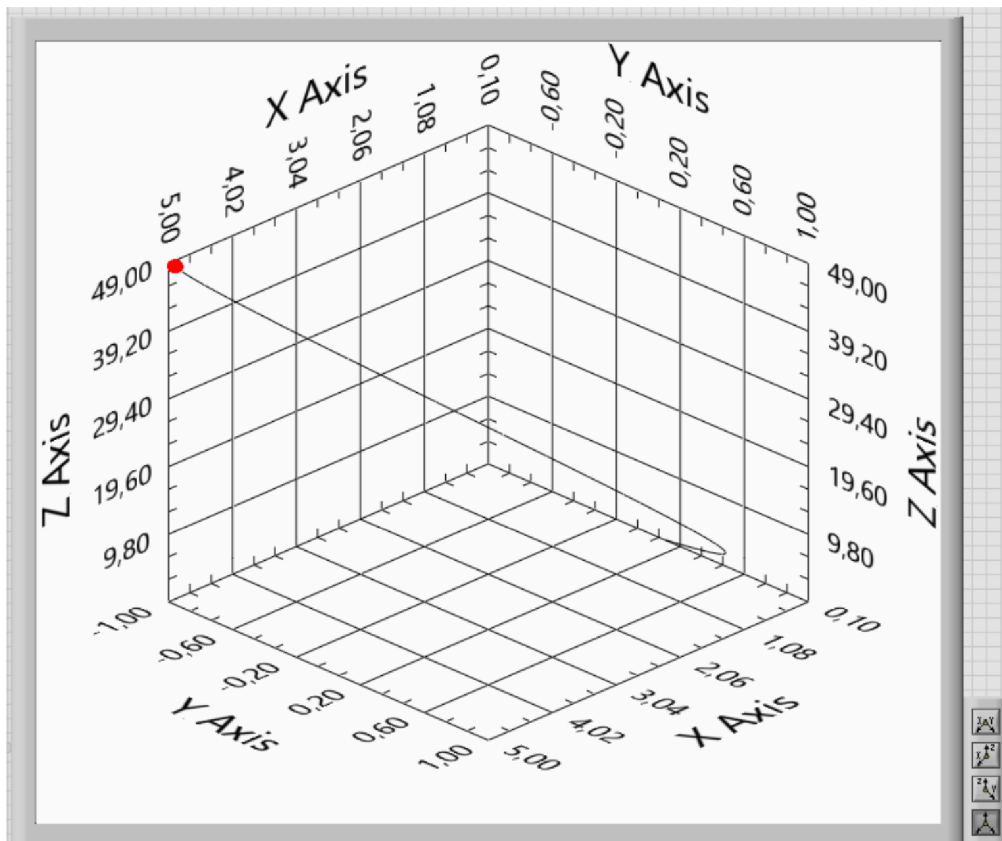
## 6.2.2 Aplikace

Aplikace bez předaných parametrů se opět snaží, aby graf co nejvíce odpovídal předloze v LabVIEW. Stejně jako v předchozím případě lze vzhled úplně změnit pomocí parametrů. Výsledný vzhled grafu může tedy vypadat například jako na obrázku č. 6.7

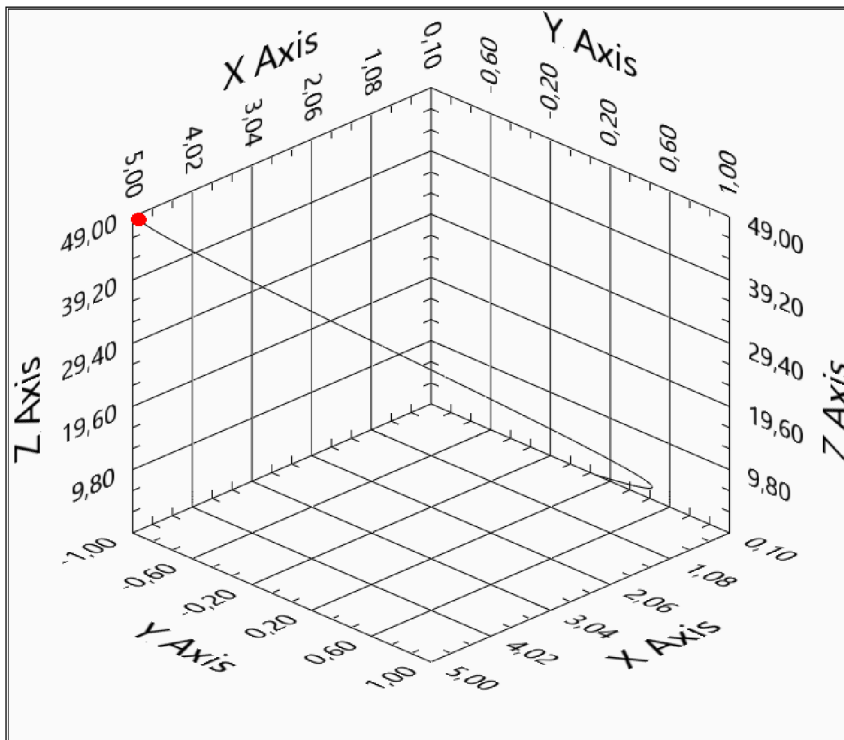
## 6.3 3D graf

Pro zobrazení funkčnosti 3D grafů jsem si vybral graf typu comet. Průběh v grafu je zaznamenána sinusovka, kdy na ose Z je index pořadí získaného bodu.

### 6.3.1 LabVIEW



Obr. 6.8: Vzhled grafu přímo ve vývojovém prostředí



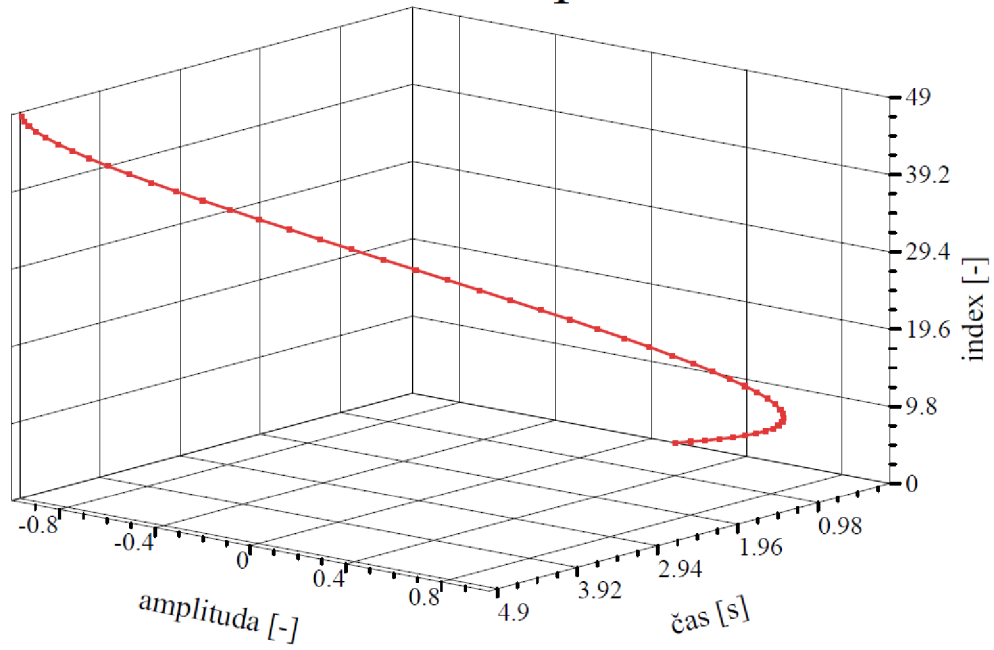
Obr. 6.9: Export LabVIEW grafu s úpravou vzhledu

V případě 3D lze z grafu č. 6.8 a 6.9 vidět, že LabVIEW export přímo odpovídá vzhledu v prostředí. Stejně jako u 2D grafů ale platí, že velikost exportovaného grafu je totožná s reálnou velikostí použitou ve vývojovém prostředí.

### 6.3.2 Aplikace

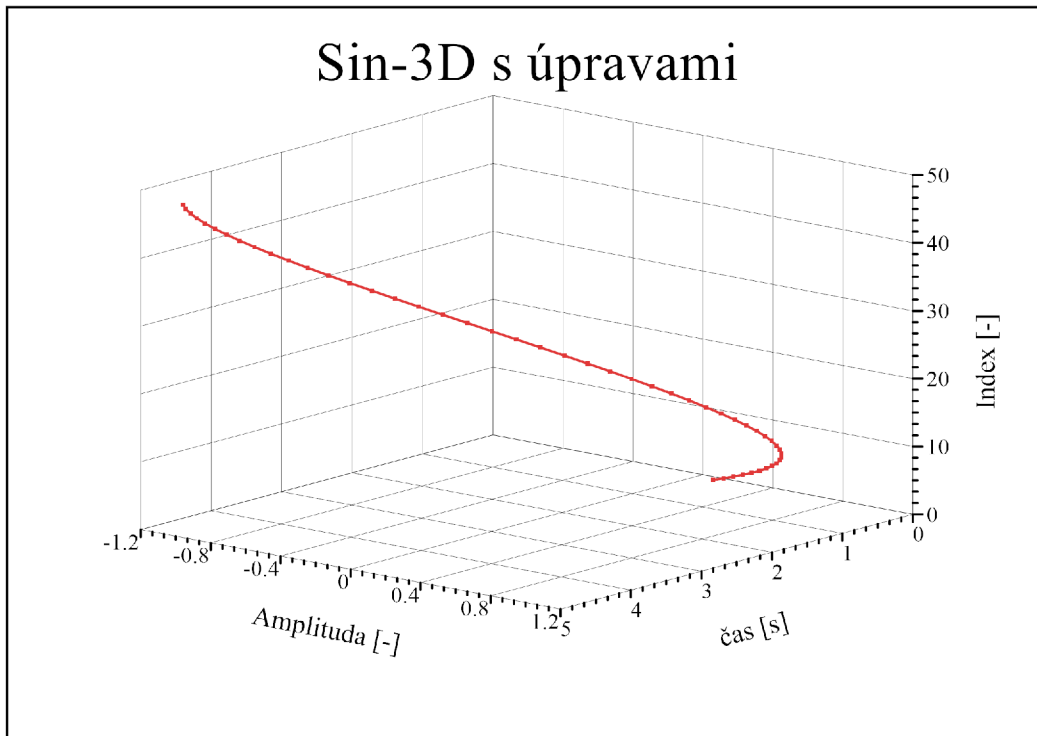
Aplikace se chová stejně jako v předchozích případech a to tak, že bez předaných parametrů se snaží co nejvíce napodobit LabVIEW verzi. V případě 3D grafu je variabilita možností na úpravu vzhledu více omezená než v případě 2D grafů. Přesto je však možné graf upravit do vhodného vzhledu jako je tomu u grafů č. 6.10 a 6.11

## sin-3D bez úprav

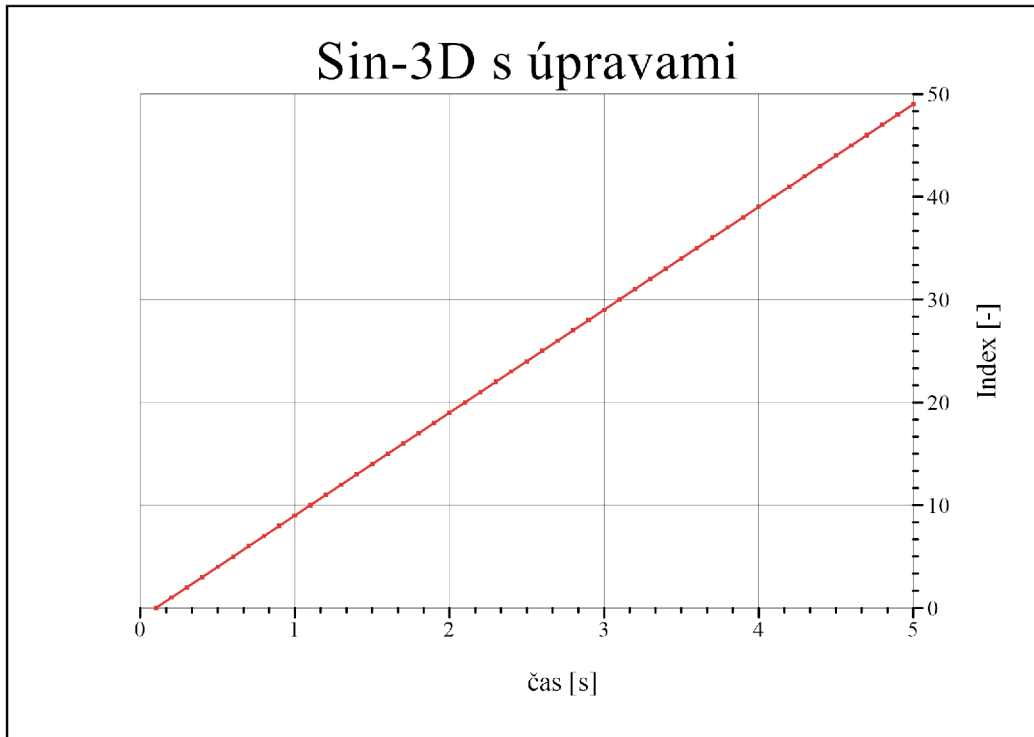


Obr. 6.10: Export grafu bez úprav pomocí vytvořeného programu

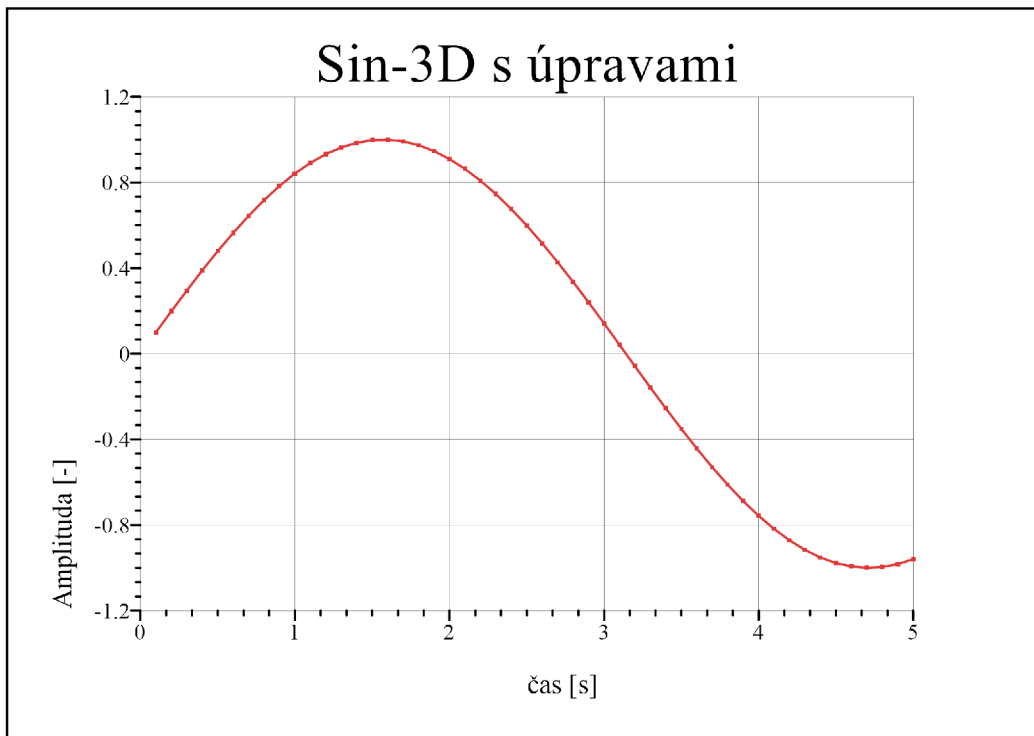
## Sin-3D s úpravami



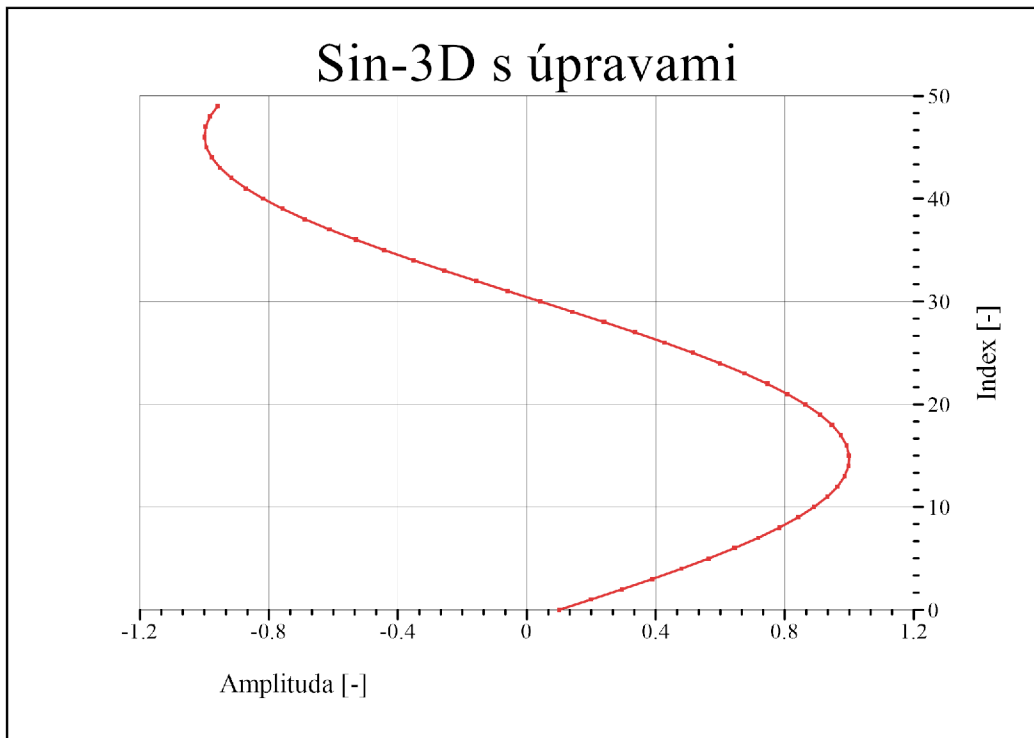
Obr. 6.11: Export grafu s úpravou vzhledu pomocí vytvořeného programu



Obr. 6.12: Pohled XZ



Obr. 6.13: Pohled XY



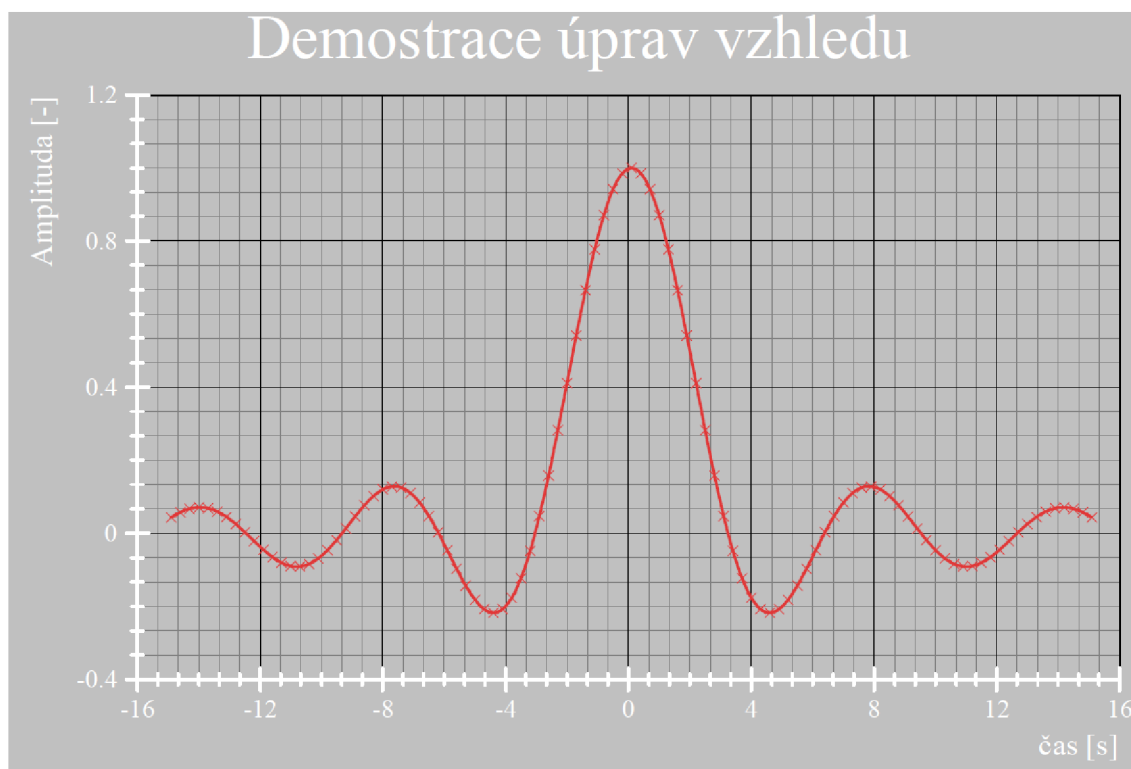
Obr. 6.14: Pohled YZ

Třírozměrné grafy je často zapotřebí posuzovat z různých pohledů. Grafy č. **6.12** **6.13** **6.14** tedy znázorňují uložení ve 3 zobrazeních a to XY, XZ a YZ. Tyto pohledy jsou zvoleny podle možností zobrazení v programu DIAdem.



## 6.4 Vzhled grafů

Vzhled grafů může být měněn s velkou možností voleb. Pro bližší znázornění jsem se rozhodl upravit výchozí 2D graf tak, aby možnosti změn byly jasně viditelné na obrázku č.6.15



Obr. 6.15: Export grafu bez úprav pomocí vytvořeného programu

## 7 Problematika

Při vývoji programu jsem narazil na několik problémů které je vhodné zmínit pro správné použití.

### 7.1 Kompatibilita

Prvním problémem je zpětná kompatibilita. Vývoj byl uskutečněn na LabVIEW verzi 2019 SP1 a DIAdem 2019 SP1. Výsledný program je uložený pro zpětnou kompatibilitu na verze 2018, ale přesto se může stát že se vyskytnou neočekávané chyby.

### 7.2 spuštění

Program je navržený tak aby sám spustil a zprovoznil DIAdem ale pro lepší funkci je doporučeno mít DIAdem předem spuštěný na pozadí aby se předešlo dlouhému spouštění programu které může mít za následek zpomalení práce na počítači.

### 7.3 Problematické grafy

Ve výsledném programu je několik problematických grafů které se mi nepovedlo z LabVIEW verze do DIAdemové tak aby si zachovaly stejný vzhled. Konkrétně se jedná o Intenzitní grafy a 3D grafy na principu zobrazování ploch. Jediný z 3D grafů nelze vygenerovat a tím je koláčový graf. DIAdem mi nedovolil koláčový graf vytvořit pomocí skriptu, ale příčinu se mi zjistit nepodařilo.

## Závěr

Během své bakalářské práce jsem se seznámil s programy DIAdem a LabVIEW. své poznatky o těchto prostředí jsem v bakalářské práci popsal a to včetně možností vzájemné komunikace těchto programů. Vzhled vytvářených grafů je brán na základě typografických doporučení. Výsledkem práce je LabVIEW knihovna funkcí, která umožňuje automatické generování grafu. Formát uložení grafu lze zvolit z několika možností (PNG, JPG, EMF, GIF, TIF, PDF, XPS, HTML, PPTX). Generování je prováděno pomocí DIAdemu, ale všechny funkce jsou řízeny pouze z LabVIEW. Pro úspěšné generování tedy není zapotřebí žádná znalost DIAdemu, stačí jen mít DIAdem i LabVIEW nainstalovaný a licencovaný v počítači. Generované grafy mohou být dvourozměrné, ale i třírozměrné. Funkce má pouze jediný povinný parametr a tím je reference grafu, který má být vygenerován. Všechny nezadané parametry jsou získány přímo z LabVIEW.

## Literatura

- [1] VLACH, Jaroslav, Josef HAVLÍČEK a Martin VLACH: *Začínáme s LabVIEW*. Ilustrovala Viktorie VLACHOVÁ. Praha: BEN - technická literatura, 2008. ISBN 978-80-7300-245-9
- [2] NI DIAdem: Getting Started with DIAdem [online]. 2017. [cit. 2019-11-25]. ISBN 373422P-01. Dostupné z: <http://www.ni.com/pdf/manuals/373422p.pdf>
- [3] Command: DataFileSave. National Instruments [online]. 2015 [cit. 2020-01-05]. Dostupné z: <http://zone.ni.com/reference/en-XX/help/370858M-01/comoff/datafilesave/>
- [4] DIAdem SCRIPT Tutorial Videos. National instruments [online]. 2017 [cit. 2020-01-05]. Dostupné z: <http://www.ni.com/tutorial/52619/en/>
- [5] National instruments: Automated test and automated measurements system [online]. [cit. 2020-01-05]. Dostupné z: <http://www.ni.com/cs-cz.html>
- [6] Tvorba efektivních grafů [online]. , 25 [cit. 2020-06-02]. Dostupné z: <https://is.muni.cz/th/vn8jk/tvorba-efektivnich-grafu.pdf>
- [7] GRIM, Tomáš. Tabulka a graf [online]. 2015 [cit. 2020-06-02]. Dostupné z: [http://tomasgrim.com/Grim/Grim USB Tabulka a graf 2015 web.pdf](http://tomasgrim.com/Grim/Grim%20USB%20Tabulka%20a%20graf%202015%20web.pdf)