

Návrh

Meranie fyzikálnych veličín pomocou meracieho prístroja

Dáša Keszeghová, Anna Rebeka Sojka, Matúš Gál, Jakub Švorc
10-31-2019

Obsah

1.	Špecifikácia vonkajších interfejsov.....	2
1.1.	Komunikácia s inými zariadeniami.....	2
1.1.1.	Pripojenie multifunkčného meracieho prístroja k počítaču	2
1.1.2.	Komunikácia multifunkčného meracieho prístroja s počítačom	2
1.1.3.	Komunikácia s $\pi\pi$ – grafom.....	3
1.1.4.	Použité technológie	3
2.	Formáty súborov	4
2.1.	Ukladanie meraní	4
2.2.	Exportovanie meraní.....	4
3.	Návrh používateľského rozhrania	5
4.	Návrh implementácie.....	7
4.1.	Component diagram	7
4.2.	Class diagram	8
4.3.	Sequence diagram.....	9

1. Špecifikácia vonkajších interfejsov

1.1. Komunikácia s inými zariadeniami

1.1.1. Pripojenie multifunkčného meracieho prístroja k počítaču

Aplikácia komunikuje s multifunkčným meracím prístrojom (MT-1820), ktorý má užívateľ pripojený k počítaču pomocou USB kábla a cez USB port. Aplikácia si bude automaticky detegovať port, v prípade keď sa nepodarí automaticky detegovať port, aplikácia požiada používateľa o zadanie portu ručne.

Prístroj sa pripája pomocou COM portov. V prípade, že COM port nie je nainštalovaný aplikácia ci ho doinštaluje potrebný COM port pomocou inštalera (CP210x_VCP_Windows\CP210xVCPInstaller_x64.exe) .

1.1.2. Komunikácia multifunkčného meracieho prístroja s počítačom

Na to aby prístroj mohol komunikovať s počítačom a taktiež aj s aplikáciou musí byť v móde REL/RS232 („Relative Value Measurement mode“), tento mód sa dosiahne podržaním tlačidla RS232/REL na prístroji. V prípade, že tento mód nie je zapnutý počítač nevie komunikovať s prístrojom.

Prístroj komunikuje posielaním 14 bajtov pričom každý bajt má svoj vlastný význam: (význam jednotlivých bajtov je možné nájsť na nasledujúcom odkaze - <https://github.com/drahoslavzan/Proskit-MT1820-Probe/blob/master/proskit.cc>)

Význam jednotlivých bajtov:

- 00 - znamienko (+/-)
- 01 – 04 hodnota, ktorú prístroj nameral
- 05 – medzera
- 06 – desatinná čiarka (za ktorú cifru z nameranej hodnoty sa má napísať desatinná čiarka)
- 07 – 08 – príznaky
- 09 – 10 – jednotka, v ktorých prístroj meria (fyzikálna veličina)
- 11 – stupnica na dolnej časti displeja prístroja
- 12 – 13 – nový riadok

Jednotky, v ktorých prístroj meria (09-10 bajt):

- UNIT_hFE = 0x0010
- UNIT_mV = 0x4080 (napätie vo voltoch)
- UNIT_V = 0x0080 (napätie v milivoltoch)
- UNIT_Ohm = 0x0020 (odpor v Ohmoch)
- UNIT_kOhm = 0x2020 (odpor v kilo-Ohmoch)
- UNIT_MOhm = 0x1020 (odpor v mega-Ohmoch)
- UNIT_DIODE_V = 0x0480
- UNIT_F = 0x0004 (teplota v stupňoch Fahrenheit)
- UNIT_Hz = 0x0008
- UNIT_DUTY_Hz = 0x0200
- UNIT_C = 0x0002 (teplota v stupňoch Celzia)
- UNIT_uA = 0x8040 (elektrický prúd v mikroampéroch)
- UNIT_mA = 0x4040 (elektrický prúd v miliampéroch)
- UNIT_A = 0x0040 (elektrický prúd v ampéroch)

1.1.3. Komunikácia s $\pi\pi$ – grafom

- $\pi\pi$ – graf je zariadenie, ktoré mechanickým pohybom ovládača číta grafy pomocou zvukového signálu
- graf posiela 4 hodnoty v rozsahu od 0 do 1023, tieto hodnoty indikujú, na ktorom bode v grafe sa práve nachádzame
- aplikácia kreslí čiarové grafy a preto stačí pracovať s prvou hodnotou, ktorú $\pi\pi$ -graf posiela, na základe načítanej hodnoty aplikácia vydá zvuk príslušnej výšky podľa hodnoty na zadanej pozícii v grafe

1.1.4. Použité technológie

- Python – Jadro aplikácie bude napísané v tomto jazyku, kvôli veľkej podpore knižníc a modulov na riešenie projektu.
- wxPython – Modul na tvorbu interface-u a interaktívnej časti aplikácie. Nakoľko zabudovaný pythonovský modul tkInter nepodporuje čítanie obrazovky, je tento nástroj najvhodnejší, aj vzhľadom na fakt, že čítač obrazovky NVDA je napísaný aj pythone.
- NVDA – program na čítanie obsahu obrazovky. Interaktívna aplikácia ktorá tvorí zvukový výstup na opísanie označeného prvku obrazovky alebo na opísanie prvku, na ktorý ukazuje myš počítača. Program používajú žiaci, pre ktorých je výsledok projektu určený. Jeho používanie a prístup k nemu je zadarmo.
- JAWS – ďalší čítač obrazovky, ktorý je tiež používaný žiakmi a však jeho licencia a prístup k nemu je platený.

2. Formáty súborov

Aplikácia bude pracovať s dvomi formátmi súborov:

2.1. Ukladanie meraní

Merania sa budú dať uložiť vo formáte .pickle, tieto súbory sa budú dať späťne otvoriť pomocou aplikácie.

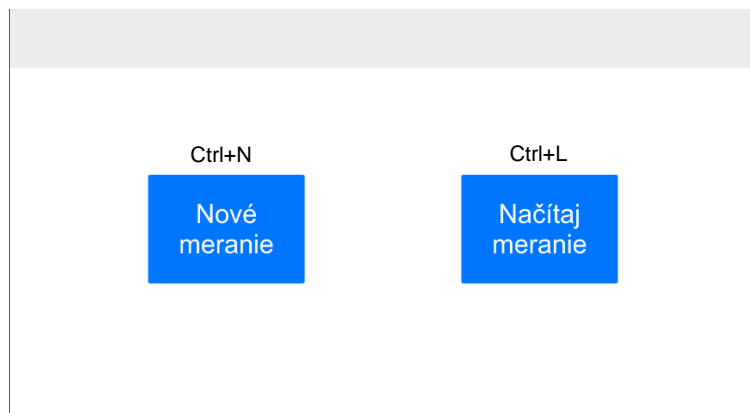
Pickle je modul používaný na serializáciu a deserializáciu pythonovských štruktúr. Pomocou tohto modulu jednoducho skonvertujeme celé meranie na prúd bajtov, ktorý sa dá jednoducho uložiť. Neskôr sa opäť vieme vrátiť do pôvodnej objektovej hierarchie. Na používanie tohto modulu netreba nič inštalovať, stačí použiť import z knižnice – „import pickle“. Na serializáciu budeme používať pickle.dumps() a deserializáciu pickle.loads(). Serializovať sa budú inštancie triedy Measurement.

2.2. Exportovanie meraní

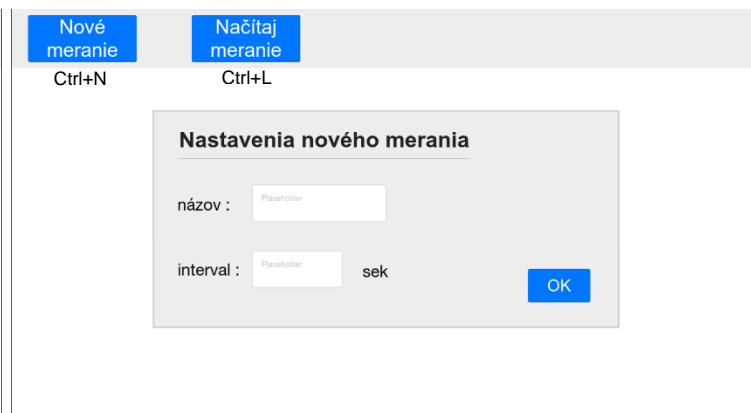
Žiaci si budú môcť exportovať namerané údaje vo forme tabuľky a grafu do Excelu vo formáte .xlsx.

Namerané údaje budú v Exceli zapísane v dvoch riadkoch, pričom prvý bude obsahovať čas a druhý nameranú hodnotu. Graf bude čiarový, na x-ovej osi bude čas a na y-ovej bude nameraná hodnota. Namerané dvojice hodnôt budú vykreslené na grafe a spojené spojovacou čiarou kontrastnej farby.

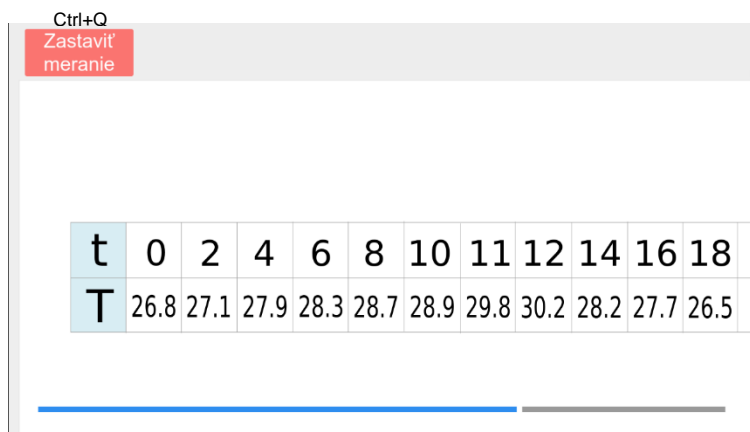
3. Návrh používateľského rozhrania



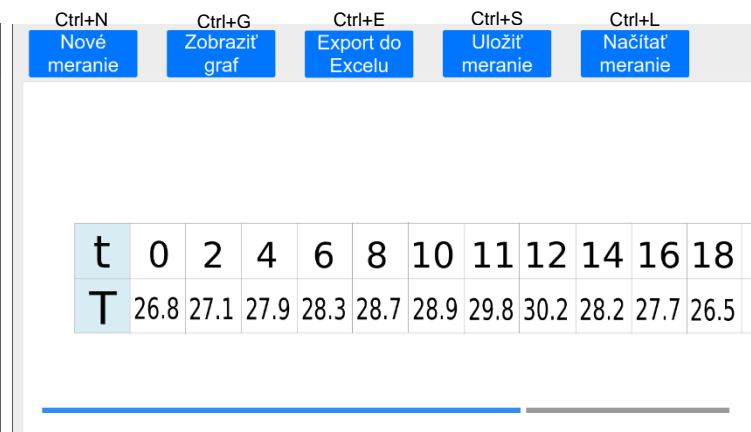
obr. 1 - štart programu



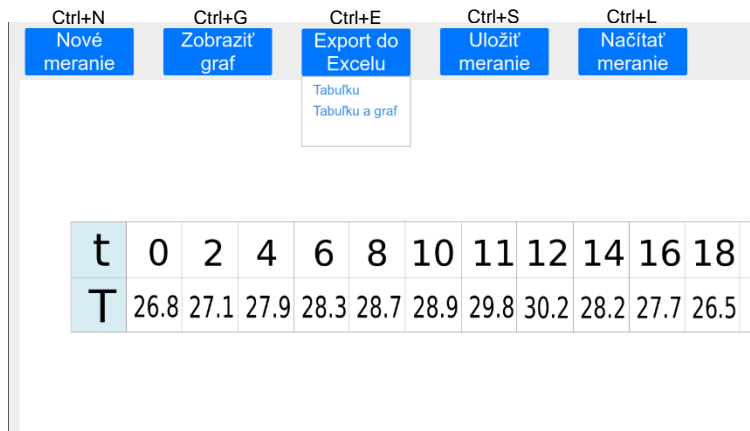
obr. 2 – nové meranie



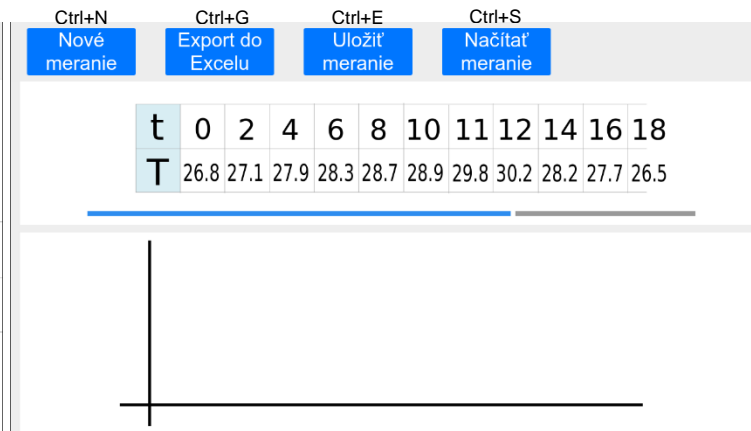
obr. 3 – počas merania



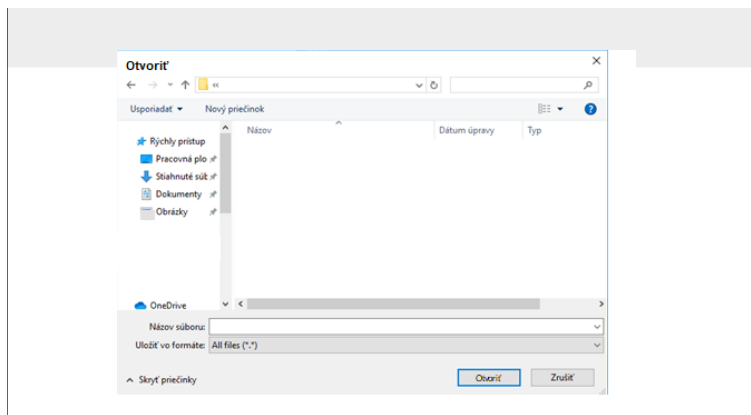
obr. 4 – po meraní / načítané meranie



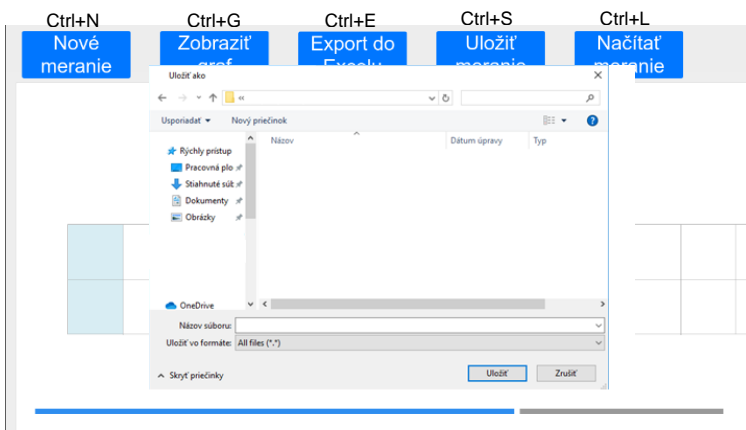
obr. 5 – možnosti exportu



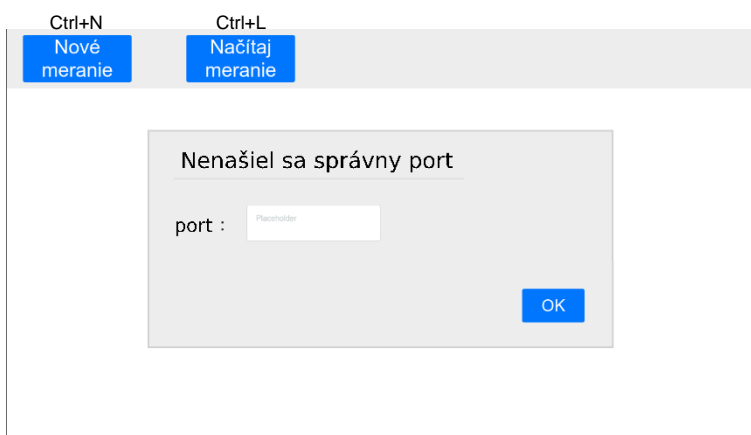
obr. 6 – zobrazenie grafu



obr. 7 – načítanie merania



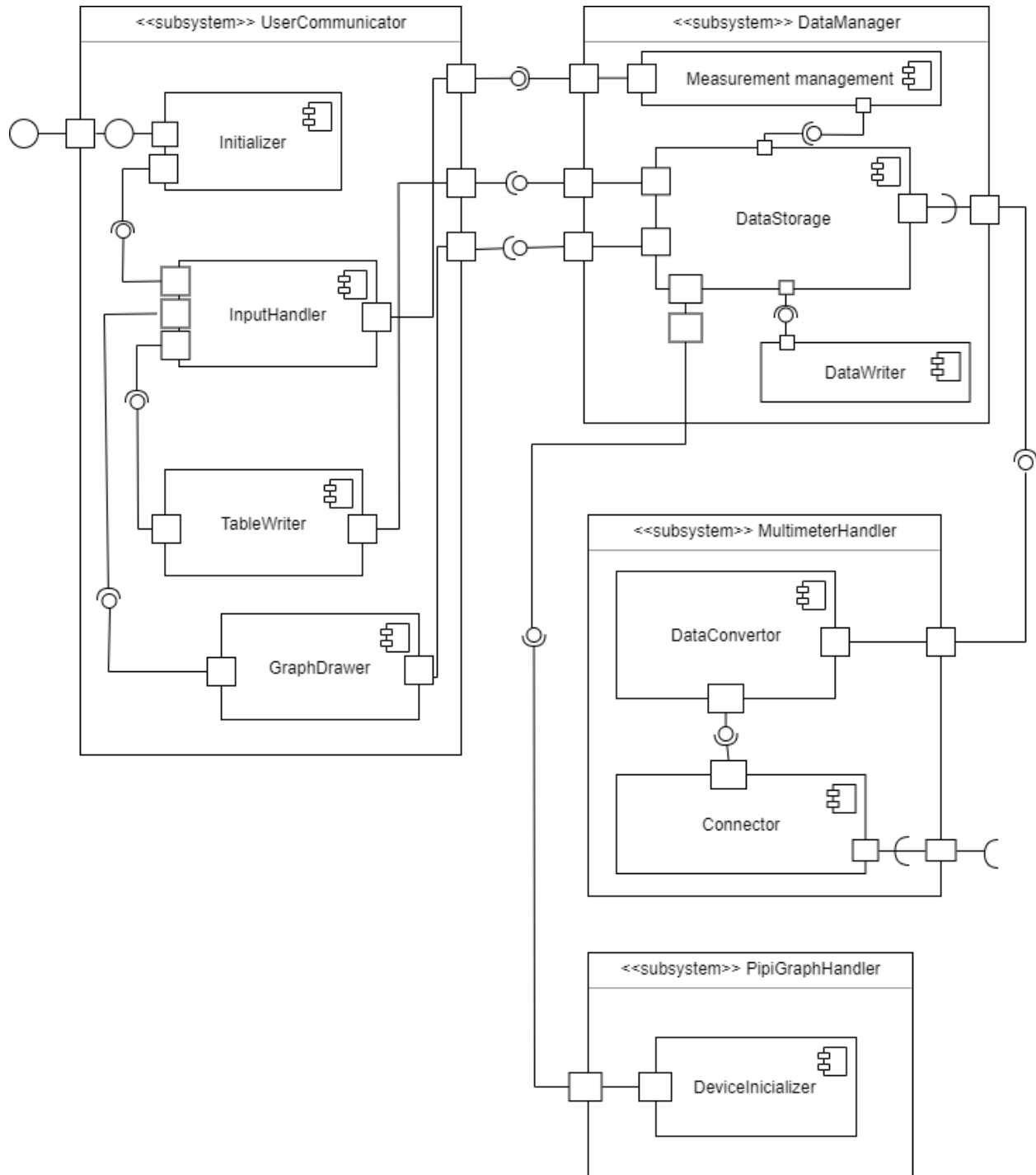
obr. 8 – ukladanie merania



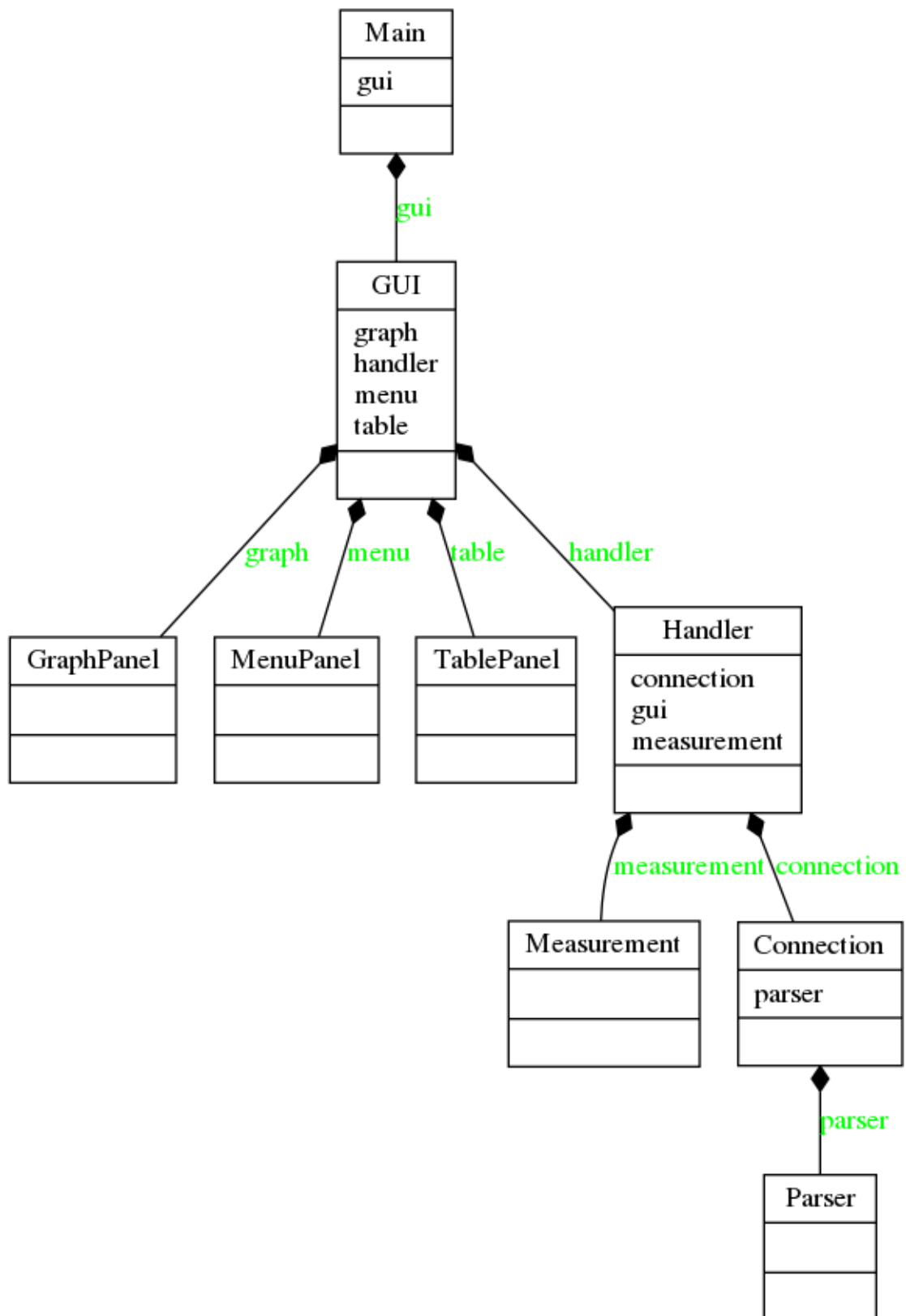
obr. 9 – chybové hlásenie o porte

4. Návrh implementácie

4.1. Component diagram



4.2. Class diagram



4.3. Sequence diagram

