

Tworzenie plików konfiguracyjnych

Skrócony opis formatu JSON

Pliki konfiguracyjne mają format JSON. [Specyfikacja](#). W skrócie, w formacie json występują dwa elementy: obiekty (`{}`) i listy (`[]`). Obiekt to zbiór kluczy i odpowiadającym im wartości rozdzielonych dwukropkiem `:`, np:

```
{
  "enabled": true,
  "name": "Quadrupole",
  "age": 18
}
```

Klucze muszą zawierać się w cudzysłowach i być wartościami tekstowymi. Wartość może być liczbą, tekstem, wartością `true/false`, bądź innym obiektem lub listą. Wartości tekstowe muszą zawierać się w cudzysłowie. Lista to uporządkowany zbiór wartości, np:

```
[
  "First",
  "Second",
  3,
  True,
  {
    "name": "jan"
  }
]
```

Podobnie jak w przypadku obiektu, każda wartość w liście może być liczbą, tekstem, itp.

Po każdym elemencie poza ostatnim w liście lub obiekcie musi znaleźć się **przecinek!**.

Konstrukcja pliku konfiguracyjnego

Każdy plik konfiguracyjny zawiera jeden obiekt JSON. W obiekcie musi znaleźć się lista kontrolerów `controllers` i opcjonalnie obiekt `defaults`:

```
{
  "controllers": [
    ...
  ],
  "defaults": {
    ...
  }
}
```

Kontroler

Kontroler to inaczej pojedyncze urządzenie pomiarowe np: multimetr, system EuroMeasure, próżniomierz. Każdy kontroler musi zawierać pole typu `type` definiujące rodzaj urządzenia, np. `HP34401A` oraz listę parametrów jakie chcemy obsługiwać: `params`. Parametry i dostępne kontrolery opisane są dalej. Poza nimi może zawierać opcjonalne elementy konfigurujące urządzenie i połączenie z nim. Wszystkie dostępne kontrolery opisane są w rozdziale [dostępne kontrolery](#)

```
{
  "type": "HP34401A",
  "ip": "169.254.100.30",
  "speed": "Slow",
  "params": [
    {
      "type": "VDC",
      "name": "Voltage",
    }
  ]
}
```

Parametr

Parametr to pojedyncza wartość ustawiana bądź czytywana z instrumentu (np. napięcie w woltomierzu, częstotliwość w generatorze). Każdy parametr musi posiadać pola typu `type` i nazwy `name`. Typ definiuje rodzaj danego parametru. Każdy z kontrolerów wspiera określone rodzaje parametrów (np. dla `HP34401A` może to być `VDC`, `IDC` itp.). Typ musi być dokładnie taki sam jak definiuje kontroler. Nazwa parametru to nazwa widziana przez użytkownika programu, **musi być unikalna**. Dodatkowo, każdy parametr może posiadać pola:

- `unit` - jednostka wyświetlana w programie (nadpisanie domyślnej)
- `default` - domyślna wartość ustawialnego parametru ustawiana przy uruchomieniu programu
- `eval_get` - wzór na podstawie którego przekształcana jest wartość odczytana z instrumentu. Więcej w [wzory](#).
- `eval_set` - wzór na podstawie którego przekształcana jest wartość ustawiana na instrumencie. Więcej w [wzory](#).
- `priority` - wartość numeryczna definiująca kolejność odczytywania/ustawiania parametrów w obrębie jednego cyklu pomiarowego. Im większa wartość tym później

Niektóre typy parametrów mogą przyjmować również dodatkowe argumenty, więcej informacji opisane w [dostępnych kontrolerach](#)

Wzory

Równania opisujące wartości ustawiane/odczytywane są w formacie języka Python. Dostępne są proste operacje matematyczne: `+`, `-`, `*`, `/`, `**` - potęgowanie, `%` - dzielenie modulo. W równaniach można

odwoływać się do włączonych parametrów wirtualnych **a**, **b**, **c** i **d**. W przypadku **eval_get** surowa wartość odczytana z instrumentu znajduje się w zmiennej **x**. Na przykład, aby ustawić dany parametr na $a + b \cdot 10$:

```
"eval_set": "a+b*10"
```

Aby skalować odczytane napięcie razy 0.001:

```
"eval_get": "x * 0.001"
```

Aby użyć bardziej złożonych operacji matematycznych należy zaimportować moduł **math**. Np:

```
"eval_get": "__import__('math').sin(x)"
```

W tym przykładzie obliczony zostanie sinus wartości zmierzonej.

Aby parametry wirtualne były dostępne należy dodać np. `"uses_b": true`, w głównym obiekcie w pliku konfiguracyjnym.

Dostępne kontrolery

EuroMeasure

Kontroler typu **EuroMeasure** obsługuje system EuroMeasure. Do obsługi dowolnej liczby kart systemu potrzebny jest w pliku konfiguracyjnym tylko jeden kontroler.

Opcje wymagane

- **port** - Nazwa portu szeregowego do którego podłączone jest urządzenie, np. **"COM11"**. W systemie windows można ją znaleźć w Menedżerze urządzeń (win+x n). **Częstym powodem niedziałania systemu jest samoistna zmiana nazwy portu**, co dzieje się przy ponownym uruchomieniu. Jeśli program nie uruchamia się poprawnie należy sprawdzić tą wartość.

Dostępne typy parametrów według kart

- generator RF
 - **generator_amplitude** - Amplituda sygnału w danym kanale
 - **generator_frequency** - Częstotliwość sygnału w danym kanale
 - **pid_p** - Parametr P kontrolera PID
 - **pid_i** - Parametr I kontrolera PID
 - **pid_d** - Parametr D kontrolera PID
 - **pid_state** - Stan kontrolera PID (włączony - 1, wyłączony - 0)
 - **pid_setpoint** - Nastawa kontrolera PID
- HVPSU (4-kanalowy zasilacz precyzyjny)
 - **hvpsu_voltage** - Nastawa napięcia wyjściowe danego kanału

- SourcePSU/6kV-PSU (1-kanałowe zasilacze wysokiego napięcia)
 - `source_psu_set_voltage` - Nastawa napięcia wyjściowego
 - `source_psu_set_current` - Nastawa ograniczenia prądowego
 - `source_psu_measured_voltage` - Zmierzone napięcie wyjściowe
 - `source_psu_measured_current` - Zmierzony prąd wyjściowy
- Voltmeter (4-kanałowy woltomierz)
 - `voltmeter_voltage` - Zmierzone napięcie dla danego kanału

Pola parametrów

Parametry udczytywane z kart posiadających więcej niż jeden kanał muszą mieć ustawione pole `channel` na wartość odpowiadającą numerowi używanego kanału

Jeśli w systemie zainstalowana jest więcej niż jedna karta danego typu, rozróżniane są one adresem. Adres jest własnością danej karty, ustawianą na płycie drukowanej. Aby użyć parametru z kart których zainstalowane jest więcej niż 1, należy ustawić pole `address` na odpowiednią wartość.

Przykład

W systemie zainstalowane są dwie karty Source_PSU, i jedna karta woltomierza.

Na karcie Source_PSU z adresem 8 zadawane jest napięcie wyjściowe (wyświetlana nazwa: `V_1`) i monitorowana jest jego faktyczna wartość (`V_1_monitor`).

Na karcie Source_PSU z adresem 9 napięcie wyjściowe ustawiane jest jako $a+10*b$ (`V_2`) i minitorowany jest prąd (`V_2_current`).

Na karcie woltomierza odczytywany jest kanał 1 i 2.

Pierwszy kanał woltomierza nazwano (`Voltage_1`)

Kanał drugi woltomierza wykorzystano do monitorowania prądu przy użyciu rezystora 1k, napięcie jest automatycznie przeliczane na prąd i wyświetlane z poprawną jednostką ($I=U/1000$). Ten kanał nazwano (`Resistor_current`)

```
{
  "type": "EuroMeasure",
  "port": "COM4",
  "params": [
    {
      "type": "source_psu_set_voltage",
      "address": 8,
      "name": "V_1"
    },
    {
      "type": "source_psu_measured_voltage",
      "address": 8,
      "name": "V_1_monitor"
    },
    {
      "type": "source_psu_set_voltage",
```

```
        "address": 9,  
        "eval_set": "a+10*b",  
        "name": "focus_V"  
    },  
    {  
        "type": "source_psu_measured_current",  
        "address": 9,  
        "name": "V_2_current"  
    },  
    {  
        "type": "voltmeter_voltage",  
        "channel": 1,  
        "name": "Voltage_1"  
    },  
    {  
        "type": "voltmeter_voltage",  
        "channel": 2,  
        "eval_get": "x/1000",  
        "unit": "A",  
        "name": "Resistor_current"  
    }  
]  
}
```