

Jacek Szlachciak

## **Projekt oprogramowania dla systemu wieloparametrycznego tworzonego na Wydziale Fizyki Politechniki Warszawskiej**

### **Spis treści**

1 Wstęp .....	3
2 Określenie wymagań na system i specyfikacja istotnych parametrów systemu .....	3
3 Określenie wymagań na oprogramowanie i architektura systemu informatycznego .....	3
3.1 Wymagania dotyczące oprogramowania .....	3
3.1.1 Język programowania.....	3
3.1.2 Biblioteki graficzne .....	3
3.1.3 Analiza danych .....	4
3.2 Architektura systemu wieloparametrycznego .....	4
3.2.1 Wstęp.....	4
3.2.2 Interakcja pomiędzy elementami systemu.....	5
3.2.3 Biblioteka urządzeń wirtualnych .....	6
3.2.3.1 Wstęp .....	6
3.2.3.2 Spektrometryczny przetwornik analogowo-cyfrowy (spectroscopy ADC) .....	7
3.2.3.3 Zegar (Timer).....	7
3.2.3.4 Licznik (Counter) .....	8
3.2.3.5 Blok paternu i koincydencji .....	8
3.2.4 System zbierania danych .....	11
3.2.5 System prezentacji danych .....	12
3.2.5.1 Wstęp.....	12
3.2.5.2 Prezentacja danych .....	12
3.2.5.2.1 Tryb graficzny .....	12

3.2.5.2.2 Tryb tekstowy.....	12
3.2.5.3 Rodzaje prezentowanych wykresów .....	13
3.2.5.3.1 Wykresy jednoparametryczne .....	13
3.2.5.3.1.1 Wstęp .....	13
3.2.5.3.1.2 Rodzaje wykresów .....	13
3.2.5.3.1.3 Atrybuty wykresów (opis najważniejszych atrybutów).....	15
3.2.5.3.1.4 Operacje możliwe do przeprowadzania na wykresach .....	16
3.2.5.3.2 Wykresy wieloparametryczne 2D .....	19
3.2.5.3.2.1 Wstęp .....	19
3.2.5.3.2.2 Rodzaje wykresów .....	19
3.2.5.3.2.3 Atrybuty wykresów (opis najważniejszych atrybutów).....	19
3.2.5.3.2.4 Operacje możliwe do przeprowadzania na wykresach 2D.....	21
3.2.5.3.3 Wykresy wieloparametryczne 3D .....	24
3.2.5.3.3.1 Wstęp .....	24
3.2.5.3.3.2 Rodzaje wykresów 3D .....	24
3.2.5.3.3.3 Atrybuty wykresów (opis najważniejszych atrybutów).....	24
3.2.5.3.3.4 Operacje możliwe do przeprowadzania na wykresach 3D.....	25
3.2.6 System analizy danych. ....	26
3.2.6.1 Wstęp .....	26
3.2.6.2 Lista dostępnych funkcji z zaznaczeniem wykorzystania biblioteki ROOT .....	26
3.2.6.2.1 Widma jednowymiarowe .....	26
3.2.6.2.2 Widma dwuwymiarowe 2D .....	27
3.2.7 System sterowania.....	27
3.2.7.1 Wstęp .....	27
3.2.7.2 Sterowanie pomiarem .....	27
3.2.7.3 Kontrola przebiegu pomiaru .....	28
3.2.7.4 Formularz opisu eksperymentu .....	29

# 1 Wstęp

Niniejsze opracowanie jest projektem oprogramowania dla systemu wieloparametrycznego tworzonego na Wydziale Fizyki Politechniki Warszawskiej.

Autor tej pracy jest współtwórcą (wraz z śp. dr Radomirem Kupczakiem) systemu wieloparametrycznego RAJ. System ten został skonstruowany w Instytucie Problemów Jądrowych w 1991 roku.

## 2 Określenie wymagań na system i specyfikacja istotnych parametrów systemu

Na obecnym etapie projektowania systemu nie jest jeszcze znana końcowa specyfikacja części sprzętowej systemu wieloparametrycznego.

Przewiduje się wykorzystanie bloku firmy CAEN V1495 (General Purpose VME Board), który mógłby pełnić rolę bloku paternu i koincydencji oraz funkcję zegara systemowego i liczników.

Zakłada się, że system będzie mógł rejestrować nie więcej niż 15 parametrów.

## 3 Określenie wymagań na oprogramowanie i architektura systemu informatycznego

### 3.1 Wymagania dotyczące oprogramowania

#### 3.1.1 Język programowania

Wymagane jest, aby został zastosowany język oprogramowania zorientowany obiektowo. Dodatkowo powinien być to język umożliwiający korzystanie z dostępnych bibliotek graficznych i numerycznych stanowiących uznany standard.

Powyższe wymagania spełnione są przez dwa języki: C++ i Java

#### 3.1.2 Biblioteki graficzne

Pisanie bibliotek graficznych jest zadaniem stanowiącym duży projekt sam w sobie. Nie ma wyraźnego powodu, aby ten komponent był tworzony od samego początku. Uwaga programisty (programistów) powinna być raczej skupiona na tworzeniu głównego systemu, a nie tworzeniu bibliotek graficznych. Biblioteki te w postaci podstawowej są często darmowe i mogą stanowić dobry początek dla pierwszej fazy tworzonego systemu, a w postaci zaawansowanej, przy koszcie nieprzekraczającym 500 EUR, ich możliwości w pełni zaspokoją potrzeby systemu wieloparametrycznego.

Jako sprawdzony produkt proponuję bibliotekę TeeChart firmy Steema Software (<http://www.steema.com/products/teechart/overview.html>). Jest to bardzo rozbudowana biblioteka, która może pracować z różnymi zintegrowanymi środowiskami do tworzenia oprogramowania (np. CodeGear Delphi, Microsoft Visual Studio, JBuilder, IBM Eclipse, Sun

Netbeans, IntelliJ IDEA, Oracle JDeveloper) i może być używana w popularnych językach obiektowych (np. C++, Java).

### 3.1.3 Analiza danych

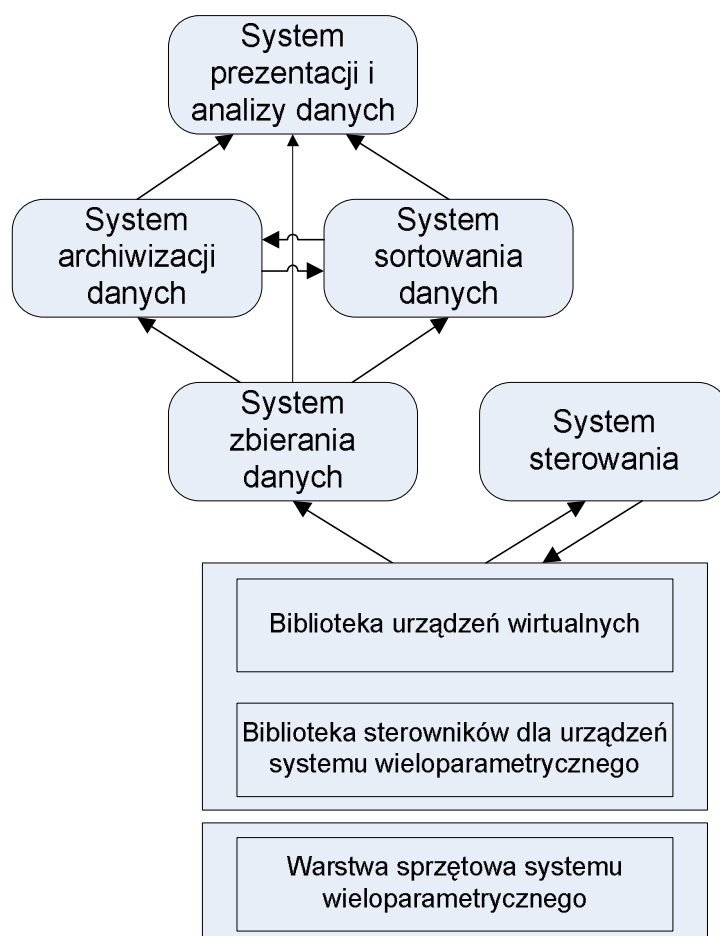
Podobnie jak w przypadku bibliotek graficznych nie należy tworzyć zaawansowanych bibliotek do analizy danych. Stworzenie takich bibliotek wymaga dobrej znajomości zasad pisania algorytmów numerycznych, długiego okresu testowania, weryfikacji poprawności i dokładności obliczeń. Zamiast tego proponuję wykorzystanie sprawdzonej (uznanej jako standard) i kompletnej (spełniającej wymagania systemu wieloparametrycznego) biblioteki numerycznej wchodzącej w skład systemu ROOT, który został stworzony w ośrodku naukowo-badawczym CERN (<http://root.cern.ch>).

Naturalnym językiem programowania dla biblioteki ROOT jest interpreter języka C++ – CINT. Stworzono również biblioteki dla języka Python – PyROOT. Istnieje także ograniczone wsparcie dla języka Java – javaROOT.

## 3.2 Architektura systemu wieloparametrycznego

### 3.2.1 Wstęp

Poniższy rysunek przedstawia schemat systemu wieloparametrycznego. Poszczególne elementy systemu przedstawione na schemacie zostały opisane w szczegółowy sposób w następujących podrozdziałach.



Rysunek 1: Architektura systemu wieloparametrycznego

### 3.2.2 Interakcja pomiędzy elementami systemu

*Warstwa sprzętowa* systemu wieloparametrycznego obejmuje rzeczywisty sprzęt użyty do budowy systemu taki jak przetworniki analogowo-cyfrowe, liczniki, układ paternu i koincydencji itp. Dla każdego z urządzeń należy napisać dedykowany sterownik (element *biblioteki sterowników*), który będzie miał ściśle ustaloną funkcjonalność. Interfejs każdego urządzenia danego typu opisany będzie zgodnie ze specyfikacją *biblioteki urządzeń wirtualnych*. Główną funkcją tej biblioteki jest ukrycie różnorodności urządzeń rzeczywistych (np. różne typy przetworników ADC pochodzące od różnych producentów) i przedstawienie tych urządzeń jako ujednoczonych uniwersalnych urządzeń dla systemu zbierania danych. Takie podejście powoduje, że przy zmianie urządzenia fizycznego niezbędne będzie napisanie tylko nowego sterownika, a pozostała część oprogramowania dla systemu wymagać będzie co najwyżej rekompilacji.

*System zbierania danych* odczytuje dane z urządzeń rzeczywistych poprzez *bibliotekę urządzeń wirtualnych*. System ma za zadanie serializację odczytywanych danych i prezentowanie tych danych wyższym warstwom (głównie *systemom archiwizacji i sortowania danych*) jako danych w postaci „list”.

*System sterowania* poprzez *bibliotekę urządzeń wirtualnych* zapewnia interakcję ze sprzętem (zapis parametrów, odczyt stanu, inicjalizację urządzeń, ich kasowanie, itp.).

Praca *systemów sterowania i zbierania danych* jest niezależna od *warstwy sprzętowej* i ściśle zdefiniowana przez funkcjonalność dostarczoną przez *bibliotekę urządzeń wirtualnych*.

*System archiwizacji danych* służy do przechowywania danych w pamięci masowej (dysk twardy, macierz dyskowa, taśma magnetyczna, itp.). System ten zapisuje dane pochodzące z *systemu zbierania danych* (dane w postaci „list”) i *systemu sortowania danych* (widma jedno- i wieloparametryczne, wykresy czasowe). Dane udostępniane przez ten system mogą być wykorzystywane przez *system prezentacji i analizy danych* (prezentacja i obróbka danych off-line) oraz przez *system sortowania danych* (sortowanie danych w trybie off-line).

*System sortowania danych* służy do sortowania danych w postaci „list” w trybie pracy on-line i off-line. W trybie pracy on-line dane pobierane są z *systemu zbierania danych*, natomiast w trybie pracy off-line z *systemu archiwizacji danych*. System ten ma za zadanie sortowanie danych w postaci „list” na dane w postaci tablic jedno- i dwuwymiarowych przechowujących informacje o widmach jedno- i wieloparametrycznych oraz wykresach czasowych. Dane te następnie zapisywane są w *systemie archiwizacji danych* lub pobierane przez *system prezentacji i analizy danych*.

*System prezentacji i analizy danych* jest systemem, który wystawia bogaty interfejs graficzny dla końcowego użytkownika systemu wieloparametrycznego. Jego dwie główne funkcje to prezentacja danych bieżących pochodzących z przeprowadzanego eksperymentu (on-line) lub danych zarchiwizowanych (off-line) w postaci wykresów dwu i trójwymiarowych, manipulacja tymi danymi oraz ich analiza w oparciu o aparat matematyczny wykorzystujący własne procedury (funkcjonalność podstawowa) lub procedury pochodzące z zewnętrznych bibliotek np. ROOT (funkcjonalność rozszerzona). System ten zasilany jest głównie danymi pochodzącymi z *systemu sortowania danych* (tablice jedno- i dwuwymiarowe przechowujące informacje o widmach jedno- i wieloparametrycznych oraz wykresach czasowych). Oprócz tego, głównie dla celów diagnostycznych, system ten może pobierać dane w postaci „list” z *systemu zbierania danych* (tryb on-line) oraz *systemu archiwizacji danych* (tryb off-line).

### 3.2.3 Biblioteka urządzeń wirtualnych

#### 3.2.3.1 Wstęp

Główną funkcją biblioteki urządzeń wirtualnych jest ukrycie różnorodności urządzeń rzeczywistych (np. różne typy przetworników ADC pochodzące od różnych producentów) i przedstawienie tych urządzeń jako ujednoczonych uniwersalnych urządzeń dla systemu zbierania danych. Takie podejście powoduje, że przy zmianie urządzenia fizycznego niezbędne będzie napisanie tylko nowego sterownika, a pozostała część oprogramowania dla systemu wymagać będzie co najwyżej rekompilacji.

Każda grupa wirtualnych urządzeń posiada interfejs umożliwiający systemom sterowania i zbierania danych na:

- odczyt i zapis (inicjalizację) parametrów stałych,
- odczyt i zapis parametrów programowalnych,
- uruchomienie funkcji sterujących.

Wartość wszystkich parametrów stałych ustalana jest w momencie inicjalizowania każdego urządzenia wirtualnego. Część z tych parametrów może być odczytywana bezpośrednio z fizycznego urządzenia (o ile dana funkcjonalność jest dostarczana przez producenta), a część na stałe zaszyta w sterowniku, bądź ustawiana przez interfejs użytkownika w fazie przed inicjalizacją (np. adres urządzenia).

Parametry programowalne, decydujące o bieżących parametrach urządzenia (np. wzmocnienia, progi, wartości liczników), mogą być zmieniane w trakcie pracy systemu wieloparametrycznego, nie powinny być jednak zmieniane (do późniejszej decyzji) po starcie zbierania danych.

Funkcje sterujące dostarczają głównej funkcjonalności polegającej na odczycie przetworzonych danych odpowiadających rejestrowanemu zdarzeniu, zerowaniu i kasowaniu urządzeń, starcie i zatrzymaniu przetwarzania itp.

We wszystkich urządzeniach wirtualnych każdy parametr lub funkcja posiada atrybut „typ parametru/funkcji” pozwalający na określenie poziomu interakcji z tym parametrem (tabela poniżej). Wykorzystując ten atrybut interfejs systemu sterowania dostarcza użytkownikowi jedynie taką funkcjonalność, jaką posiada rzeczywiste urządzenie.

*Typ parametru/funkcji:*

<b>Wartość</b>	<b>Opis</b>
-1 lub 255 (do ustalenia)	parametr lub funkcja nie występuje w danym urządzeniu
0	brak możliwości zapisania i odczytania parametru lub wykonania i odczytania stanu funkcji
1	można odczytać parametr lub stan funkcji
2	można zapisać parametr lub wykonać funkcję
3	można odczytać i zapisać parametr lub odczytać stan i wykonać funkcję

### 3.2.3.2 Spektrometryczny przetwornik analogowo-cyfrowy (spectroscopy ADC)

#### Parametry programowalne:

- wzmacnienie sygnału wejściowego (Conversion Gain - Full scale resolution of the input signal) –  $2^n$
- zakres przetwarzania (Range – number of channels as the ADC's output limit) –  $2^n$
- przesunięcie cyfrowe (Digital Offset)
- próg dolny (LLD – Lower Level Discriminator for minimum input acceptance voltage)
- próg górny (ULD – Upper Level Discriminator for maximum input acceptance voltage)

#### Parametry stałe:

- model przetwornika (Model Number)
- numer seryjny (Factory Serial Number)
- adres urządzenia (Address)
- rozdzielczość (Channel Resolution)
- wzmacnienie sygnału wejściowego min/max
- zakres przetwarzania min/max
- przesunięcie cyfrowe min/max
- skok (offset) przesunięcia cyfrowego
- próg dolny min/max
- próg górny min/max
- zakres wejściowy (jest to największa wartość parametru jaką można zmierzyć) – parametr opisowy
- czas martwy (Dead Time) – parametr opisowy
- czas przetwarzania (Conversion Time) – parametr opisowy

#### Funkcje sterujące:

- zerowanie (reset)
- kasowanie (reject)
- dana gotowa do odczytu
- odczyt danej (read data)
- odczyt/kasowanie błędu (problem)
- odczyt/ustawienie trybu pracy On-line
- odczyt/ustawienie trybu pracy Off-line

### 3.2.3.3 Zegar (Timer)

#### Parametry programowalne:

- mierzony czas (Time Base)
- rozdzielczość czasu (Timer Resolution)
- kierunek upływu czasu (Count up/down)
- tryb pracy: przebieg pojedynczy lub cykliczny (One Time or Recycle Timer) – po odmierzeniu danego interwału zegar zatrzymuje się (pojedynczy) lub startuje ponownie (cykliczny)

**Parametry stałe:**

- model zegara (Model Number)
- numer seryjny (Factory Serial Number)
- adres urządzenia (Address)
- mierzony czas min/max
- rodzaje rozdzielczości czasu (Timer Resolution) mierzone w jednostkach czasu, zbiór wartości

**Funkcje sterujące:**

- zerowanie (reset)
- stop
- start
- odczyt danej (read data)
- odczyt/kasowanie błędu (problem)
- odczyt/ustawienie trybu pracy On-line
- odczyt/ustawienie trybu pracy Off-line start

**3.2.3.4 Licznik (Counter)****Parametry programowalne:**

- ilość impulsów (Count Number) do zliczenia, aby wygenerować sygnał końca

**Parametry stałe:**

- model licznika (Model Number)
- numer seryjny (Factory Serial Number)
- adres urządzenia (Address)
- zakres liczenia (Count Capacity)
- maksymalna częstotliwość impulsów (Maximum Counting Rate)

**Funkcje sterujące:**

- zerowanie (reset)
- stop
- start
- odczyt danej (read data)
- odczyt/kasowanie błędu (problem)
- odczyt/ustawienie trybu pracy On-line
- odczyt/ustawienie trybu pracy Off-line

**3.2.3.5 Blok paternu i koincydencji****Parametry programowalne:**

- maska aktywnych wejść (IN) koincydencyjnych (active inputs)  
zakładamy, że mamy 15 lub mniej wejść, bity w słowie 16 bitowym oznaczają, które wejścia są aktywne, a które zablokowane. Bit zerowy odpowiada pierwszemu wejściu
- maska paternu  
część stała słowa paternu dodawana (operacja bitowy OR) do paternu opisującego zarejestrowane zdarzenie, służy do dodawania do danych typu list danych z rejestrów



nie związanych bezpośrednio z rejestrowanym zdarzeniem, np. dana z zegara, licznika itp.

- tablica koincydencji i antykoincydencji
- czas koincydencji (Coincidence Time)

#### **Parametry stałe:**

- model bloku (Model Number)
- numer seryjny (Factory Serial Number)
- adres urządzenia (Address)
- czas koincydencji min/max
- skok czasu koincydencji

#### **Funkcje sterujące:**

- zerowanie (reset)  
kasowanie danych z bloku parametrów programowalnych i wpisanie wartości domyślnych
- tryb pracy (working mode)

##### **pattern input**

W tym trybie pracy system odczytuje te przetworniki, które zostały wskazane przez zarejestrowanie w czasie koincydencji odpowiednich sygnałów na wejściach IN bloku paternu i koincydencji i spełniły warunek koincydencji. Wybranie tego trybu powoduje, że mogą być odczytane ADC mimo tego, że nie dokonały konwersji sygnału wejściowego. Może to nastąpić w następujących przypadkach:

- uszkodzenie ADC
- przekroczenie górnego progu ADC
- niewłaściwe ustawienie przełączników na ADC
- brak przetwornika ADC

W trybie tym zostaną przeczytane wszystkie przetworniki wskazane przez blok paternu, a w wymienionych wyżej przypadkach dana będzie miała wartość 0.

##### **pattern LAM**

W tym trybie pracy, po stwierdzeniu koincydencji, system odczytuje tylko te przetworniki ADC, które wykazały gotowość do odczytu. Powoduje to, że przypadkach wymienionych w opisie trybu pattern input, wystąpi niezgodność między ilością wykrytych impulsów w czasie koincydencji, a ilością zarejestrowanych danych z ADC.

W tym trybie, w danych typu LIST, nie mogą wystąpić zerowe dane.

##### **pattern input & LAM**

W tym trybie pracy, po stwierdzeniu koincydencji, system porównuje czy sygnały gotowości odczytu ADC są zgodne z zarejestrowanymi impulsami w czasie koincydencji przez blok paternu. Przy niezgodności zdarzenie jest odrzucane. W tym trybie, w danych typu LIST, nie mogą wystąpić zerowe dane.

- tryb koincydencji (coincidence mode)  
**tryb bez koincydencji (no coincidence)**

Wybranie tego trybu powoduje, że system pracuje bez koincydencji, czyli przyjmuje wszystkie sygnały docierające do systemu zgodnie z ustawieniem maski aktywnych wejść koincydencyjnych. Zdarzenie zamykane jest po ustawionym czasie koincydencji.

## tryb z koincydencją (set coincidence)

Wybranie tego trybu powoduje, że system pracuje z koincydencją

### Typy koincydencji:

#### 1. Typ prosty wieloparametryczny

Przykłady:

Koincydencja  $1*2*5$  oznacza, że system wieloparametryczny zarejestruje zdarzenie, gdy w czasie koincydencji na wejściach 1, 2, 5 bloku *paternu* i *koincydencji* pojawią się sygnały. Pojawienie się w tym czasie sygnałów na innych wejściach spowoduje skasowanie zdarzenia. Koincydencja  $N2*4*N7$  oznacza, że system wieloparametryczny zarejestruje zdarzenie, gdy w czasie koincydencji na wejściach 2, 7 bloku *paternu* i *koincydencji* nie pojawią się sygnały, natomiast na wejściu 4 będzie zarejestrowany sygnał. Pojawienie się w tym czasie sygnałów na innych wejściach spowoduje skasowanie zdarzenia.

#### 2. Typ złożony wieloparametryczny

Przykład:

Typ złożony wieloparametryczny różni się od typu prostego tym, że pojawienie się w czasie koincydencji sygnałów niewyszczególnionych w wyrażeniu koincydencyjnym nie powoduje skasowania zdarzenia, a odczytane parametry związane z tymi sygnałami zostaną również przesłane do komputera. Koincydencja  $1*2*N3*X$  oznacza, że system wieloparametryczny zarejestruje zdarzenie, gdy w czasie koincydencji na wejściach 1, 2 bloku *paternu* i *koincydencji* pojawią się sygnały, na wejściu 3 sygnał się nie pojawi. Sygnały podawane na inne wejścia mogą mieć dowolną wartość (jest lub nie ma).

#### 3. Typ krotnościowy

Przykład:

Koincydencja  $2<m<5$  oznacza, że system wieloparametryczny zarejestruje zdarzenie, gdy w czasie koincydencji na dowolnych aktywnych wejściach bloku *paternu* i *koincydencji* pojawią się 3 lub 4 sygnały. W innym przypadku zdarzenie zostanie skasowane.

Uwaga:

Jeżeli zdefiniowano więcej niż jedno wyrażenie koincydencyjne, to wynikiem ich działania będzie suma zdefiniowanych w ten sposób zdarzeń logicznych.

Znaczenie użytych symboli:

"\*" - operator logiczny AND

"N" - operator logiczny NOT

"1".."12" - numer wejścia bloku *paternu* i *koincydencji*

"X" - dowolne aktywne wejście bloku *paternu* i *koincydencji* nie wliczone w koincydencji (znak musi znajdować się na końcu wyrażenia)

"<m<" - m oznacza liczbę sygnałów będących w koincydencji

### 3.2.4 System zbierania danych

System zbierania danych odczytuje dane z urządzeń rzeczywistych poprzez bibliotekę urządzeń wirtualnych. System ma za zadanie serializację odczytywanych danych i prezentowanie tych danych wyższym warstwom (głównie systemom archiwizacji i sortowania danych) jako danych zdarzenie po zdarzeniu (event by event - EBE) nazywanymi w tym opracowaniu danymi typu „list”. Dane typu „list” przesyłane są w słowach dwubajtowych i mają następującą postać:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	bit
1	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	Patern
0	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	Parametr i
0	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	Parametr j
0	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	Parametr k
1	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	Patern
0	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	Parametr x
0	d	d	d	d	d	d	d	d	d	d	d	d	d	d	d	Parametr y

Słowo paternu zbudowane jest w następujący sposób:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	bit
1	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	Patern

Na bicie 15 wpisana jest na stałe wartość 1. Wyróżnia to patern od danych, które na bicie 15 mogą mieć jedynie wartość 0. Ustawiona wartość 1 na bitach od 0 do 14 - D1 .. D15 pokazuje numer urządzenia (przetwornika ADC, licznika, itp.) przekazującego dane. Ilość ustawionych bitów na pozycjach D1 .. D15, zawiera informację ile słów z danymi pojawi się za słowem paternu. Po przesłaniu wszystkich danych pojawi się nowy patern.

Przykład:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	bit
1	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	Patern
0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Parametr 2
0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Parametr 5
0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Parametr 9
1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	Patern
0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Parametr 1
0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Parametr 7

„x” oznacza 0 lub 1

## 3.2.5 System prezentacji danych

### 3.2.5.1 Wstęp

System prezentacji danych jest nierozdzielalnym składnikiem systemu prezentacji i analizy danych, który wystawia bogaty interfejs graficzny dla końcowego użytkownika systemu wieloparametrycznego. Jego główna funkcja to prezentacja danych bieżących pochodzących z przeprowadzanego eksperymentu (on-line) lub danych zarchiwizowanych (off-line) w postaci wykresów dwu i trójwymiarowych oraz manipulacja tymi danymi. System ten zasilany jest głównie danymi pochodzącymi z systemu sortowania danych (tablice jedno- i dwuwymiarowe przechowujące informacje o widmach jedno- i wieloparametrycznych oraz wykresach czasowych). Oprócz tego, głównie dla celów diagnostycznych, system ten może pobierać dane w postaci „list” z systemu zbierania danych (tryb on-line) oraz systemu archiwizacji danych (tryb off-line).

### 3.2.5.2 Prezentacja danych

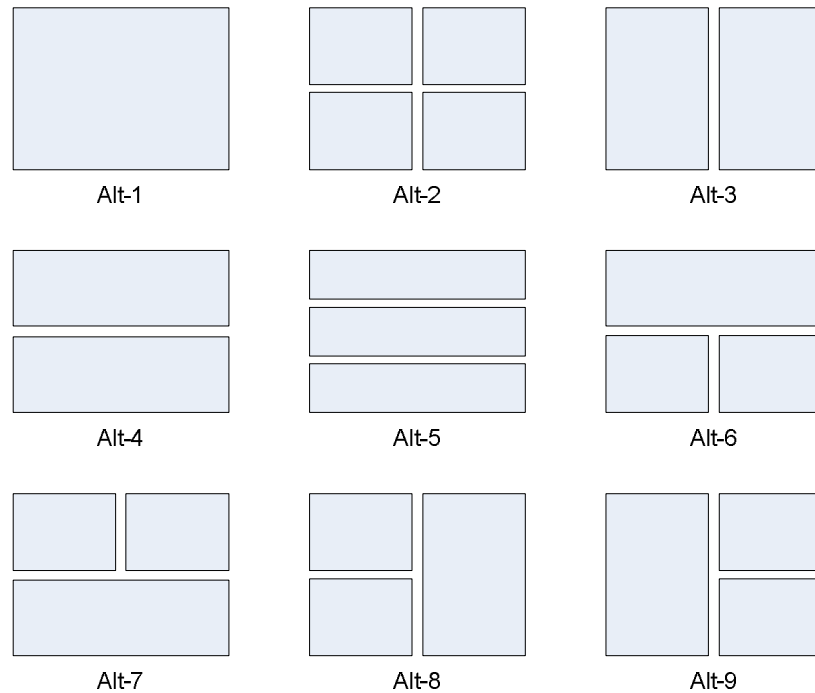
#### 3.2.5.2.1 Tryb graficzny

Podczas pracy programu na ekranie może się pojawić od jednego do czterech okien przeznaczonych do wyświetlania danych zawartych w tablicach jedno- lub dwuwymiarowych reprezentujących widma lub wykresy czasowe. Wyboru tego możemy dokonać naciskając jednocześnie klawisz [Alt] i jeden z klawiszy od [1] do [9]. Wśród wyświetlonych okien zawsze znajduje się jedno okno aktywne. Jest ono wyróżnione np. innym kolorem ramki lub/i specjalnym znakiem w swojej linijce informacyjnej. Pojęcie aktywnego okna zostało wprowadzone dlatego, aby wyróżnić, okno na którym chcemy przeprowadzać operacje dostępne w programie wybierając odpowiednie funkcje z głównego menu. Wielkość okien zależy od rozmiaru głównego okna programu, a ich rozmiar przeskalowywany jest w ten sposób, aby zawsze zachowywać odpowiednie proporcje w stosunku do sąsiadujących okien (w przypadku, gdy na ekranie wyświetlane jest więcej niż jedno okno).

#### 3.2.5.2.2 Tryb tekstowy

Oprócz prezentacji danych w postaci graficznej, wyświetlane są także dane w postaci tekstowej. Taki tryb pracy może być wykorzystywany np. w celach diagnostycznych do sprawdzania czy dobrze odczytywane są dane napływające z systemu zbierania danych. Może mieć to znaczenie w przypadku bardzo niskiej częstotliwości rejestrowanych zdarzeń. W wybranym oknie możemy obserwować w postaci tabelarycznej albo dany parametr (parametry), albo wszystkie napływające dane. Pierwsza kolumna zawiera numer zdarzenia, a następne, w zależności od wybranej opcji, dane z przetworników ADC, liczników itp. lub wszystkie rejestrowane dane. Dane w oknie, po całkowitym ich wypełnieniu na ekranie, są automatycznie przewijane bez możliwości powrotu do wcześniej prezentowanych wartości, które zostały usunięte z okna.

Każdy z wyświetlanych układów okien może prezentować dane zarówno w trybie graficznym jak i tekstowym



Rysunek 2: Układy okien do wyświetlania tablic jedno- lub dwuwymiarowych

### 3.2.5.3 Rodzaje prezentowanych wykresów

#### 3.2.5.3.1 Wykresy jednoparametryczne

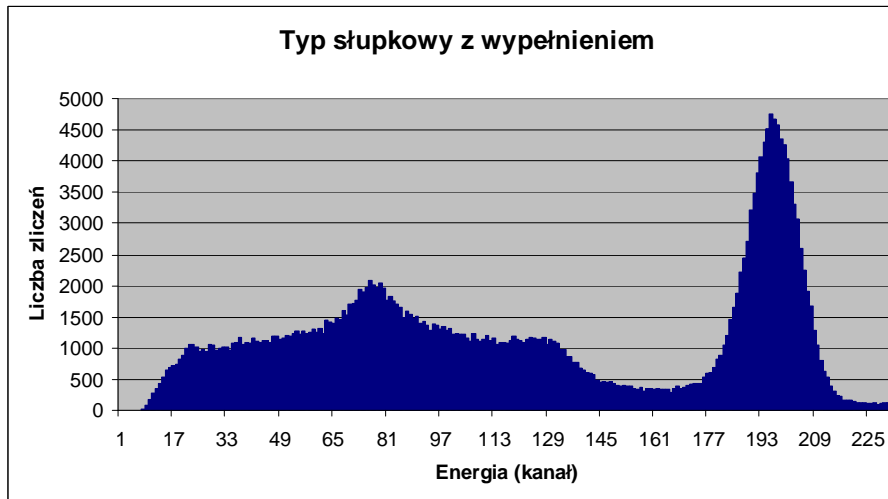
##### 3.2.5.3.1.1 Wstęp

Przy prezentacji wykresów jednoparametrycznych mamy dwa typy danych do wyświetlania: widma jednoparametryczne i wykresy czasowe. W pierwszym przypadku oś X reprezentuje numer kanału odpowiedniego przetwornika ADC, a oś Y liczbę zliczeń w danym kanale. W przypadku wykresów czasowych oś X reprezentuje oś czasu, natomiast oś Y amplitudę sygnału zmierzoną przez przetwornik ADC lub wartość rejestru z licznika.

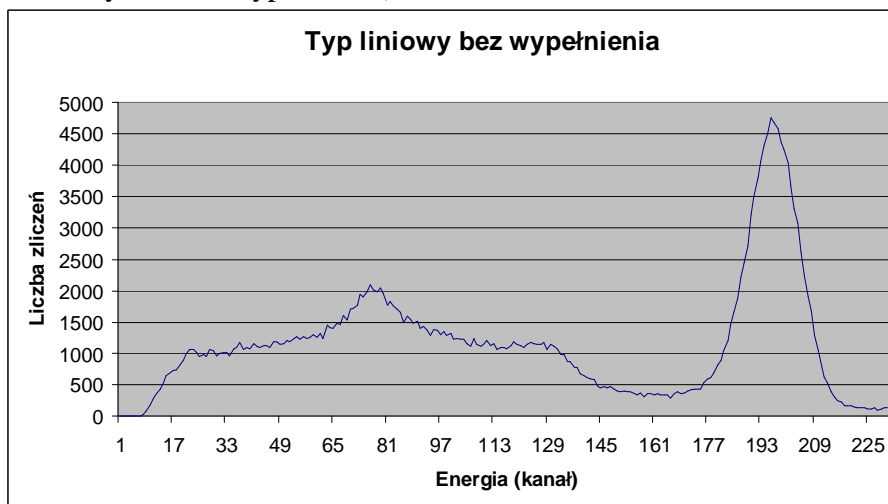
##### 3.2.5.3.1.2 Rodzaje wykresów

Dla wykresów jednoparametrycznych mamy następujące sposoby prezentacji danych (dla uproszczenia przykłady graficzne ograniczają się do wyświetlania widm jednoparametrycznych, pominięte zostały wykresy czasowe):

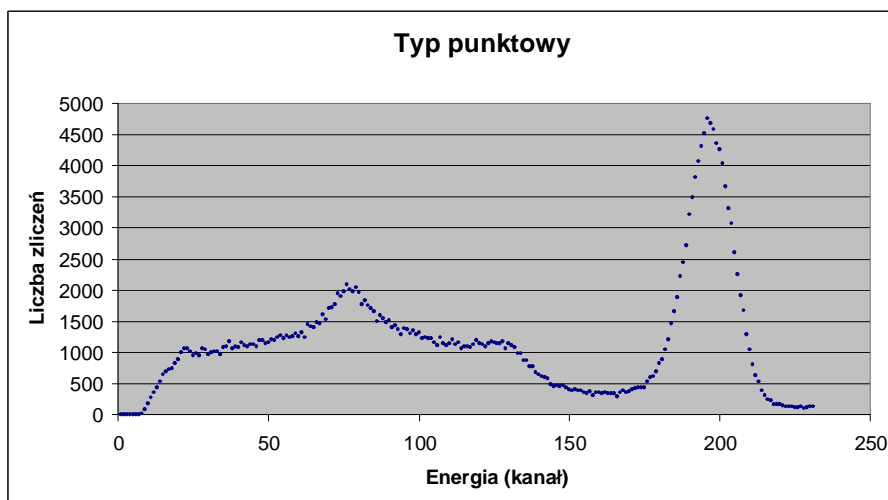
- słupkowy (2 warianty: z i bez wypełnienia)



- liniowy (2 warianty: z i bez wypełnienia)



- punktowy



### 3.2.5.3.1.3 Atrybuty wykresów (opis najważniejszych atrybutów)

#### **Oś X minimum**

Numer pierwszego kanału dla wyświetlanego widma lub czas od którego rysowany będzie wykres czasowy. Domyślnie – 0.

#### **Oś X maksimum**

Numer ostatniego kanału dla wyświetlanego widma lub czas do którego rysowany będzie wykres czasowy. Domyślnie – rozmiar tablicy (ilość kanałów).

#### **Oś Y minimum**

Wartość zliczeń od której rysowane będzie widmo lub minimalna wartość amplitudy prezentowana dla wykresów czasowych. Domyślnie – 0.

#### **Oś Y maksimum**

Wartość zliczeń do której rysowane będzie widmo (większa liczba zliczeń będzie obcinana) lub maksymalna wartość amplitudy prezentowana dla wykresów czasowych (większe amplitudy będą obcinane). Domyślnie – auto, wszystkie wartości prezentowane są bez obcięcia.

#### **Oś Y skalowanie**

Określa sposób skalowania osi Y:

- automatyczny, wszystkie wartości prezentowane są bez obcięcia przy maksymalnym rozciągnięciu wyświetlanego wykresu na osi Y. Atrybut *oś Y maksimum* wyliczany jest na podstawie wzoru:

$$oś\ Y\ maksimum = Y_{max} + 0,1 * (Y_{max} - oś\ Y\ minimum)$$

gdzie  $Y_{max}$  oznacza maksymalną wartość zliczeń (dla widm) lub maksymalną wartość amplitudy (dla wykresów czasowych) dla przedziału [*oś X minimum*, *oś X maksimum*]

- ręczny, wszystkie wartości prezentowane są w zakresie od *oś Y minimum* do *oś Y maksimum*.

#### **Oś Y typ skali**

Określa rodzaj użytej skali: liniowa lub logarytmiczna.

#### **Oś X opis osi**

Pole tekstowe opisujące nazwę wyświetlanego parametru na osi X.

#### **Oś Y opis osi**

Pole tekstowe opisujące nazwę wyświetlanego parametru na osi Y.

#### **Marker główny pozycja**

Określa pozycję głównego markera poruszającego się wzdłuż osi X.

#### **Marker główny aktywny**

Określa czy główny marker wyświetlany jest na wykresie. Przyjmuje wartości logiczne prawda lub fałsz.

### Marker lewy pozycja

Określa pozycję lewego markera poruszającego się wzdłuż osi X.

### Marker prawy pozycja

Określa pozycję prawego markera poruszającego się wzdłuż osi X.

### Marker lewy i prawy aktywny

Określa czy lewy i prawy marker wyświetlane są na wykresie. Przyjmuje wartości logiczne prawda lub fałsz.

### Marker lewy i prawy zablokowany

Określa czy odległość pomiędzy lewym i prawym markerem jest zablokowana. Przyjmuje wartości logiczne prawda lub fałsz.

### Wykres typ

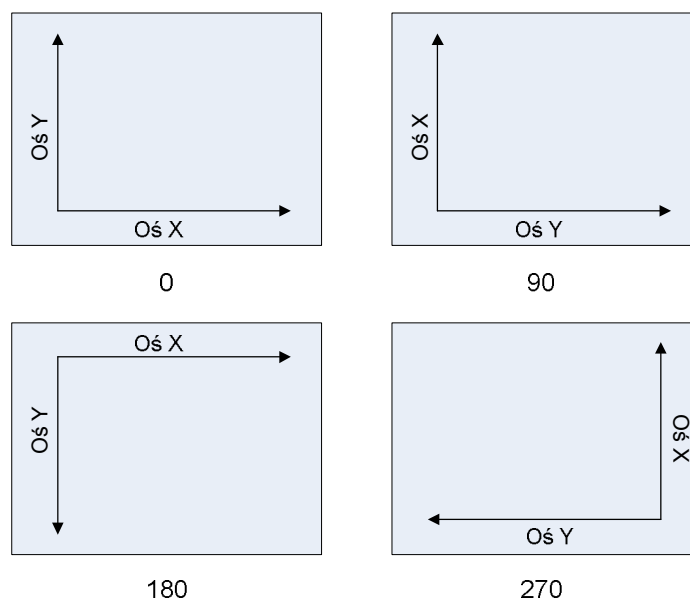
Określa typ wyświetlanego wykresu: słupkowy, liniowy lub punktowy

### Wykres wypełnienie

Określa czy wyświetlany wykres prezentowany jest w formie wypełnionej czy nie. Przyjmuje wartości logiczne prawda lub fałsz.

### Wykres obrót

Określa orientację wyświetlanego na ekranie wykresu. Przyjmuje wartości: 0, 90, 180, 270 (szczegóły na rysunku poniżej).



Rysunek 3: Orientacja wyświetlanych wykresów jednoparametrycznych

#### 3.2.5.3.1.4 Operacje możliwe do przeprowadzania na wykresach

##### Zmiana zakresu wyświetlanego widma

##### - przywrócenie pełnego zakresu wyświetlania

powoduje, że niezależnie od ustawień wykres prezentowany jest w pełnym zakresie dla wartości osi X i osi Y



**- ręczne ustawienie zakresu na osi X**

powoduje, że wykres prezentowany jest w zakresie od ręcznie ustawionych wartości oś *X minimum* i oś *X maksimum*

**- ręczne ustawienie zakresu na osi Y**

powoduje, że wykres prezentowany jest w zakresie od ręcznie ustawionych wartości oś *Y minimum* i oś *Y maksimum*

**- automatyczne ustawienie zakresu na osi Y**

powoduje, że wykres prezentowany jest w zakresie od 0 lub ręcznie ustawionej wartości oś *Y minimum* do automatycznie wyliczanej wartości oś *Y maksimum*

**- zwiększenie dwukrotne skali osi Y**

powoduje, przy ręcznym ustawieniu zakresu na osi Y, dwukrotne zwiększenie wyświetlanego zakresu na osi Y

**- zmniejszenie dwukrotne skali osi Y**

powoduje, przy ręcznym ustawieniu zakresu na osi Y, dwukrotne zmniejszenie wyświetlanego zakresu na osi Y

**- przesuwanie wykresu wzdłuż osi X**

powoduje przesunięcie wykresu wzdłuż osi X w obu kierunkach tej osi (o ile ustawione jest ręczne ustawienie zakresu na osi X); operację można wykonać za pomocą myszy lub klawiszy kursora

**- powiększenie obszaru pomiędzy markerami na osi X**

powoduje wyświetlenie (rozciągnięcie) wykresu w zakresie wyznaczonym przez pozycję lewego i prawego markera (o ile wartość atrybutu *marker lewy i prawy aktywny* jest logiczną prawdą)

**- rozciąganie obszaru na osi X**

powoduje dwukrotne zmniejszenie ilości wyświetlanych kanałów na osi X; punktem którego położenie przy rozciągnięciu nie zmienia się jest punkt środkowy wyświetlanego wykresu o numerze kanału równym  $\frac{1}{2} * (oś X maksimum - oś X minimum)$

**- ściśnięcie obszaru na osi X**

powoduje dwukrotne zwiększenie ilości wyświetlanych kanałów na osi X; punktem którego położenie przy ściśnięciu nie zmienia się jest punkt środkowy wyświetlanego wykresu o numerze kanału równym  $\frac{1}{2} * (oś X maksimum - oś X minimum)$

**- rozciąganie obszaru na osi X względem markera głównego**

powoduje dwukrotne zmniejszenie ilości wyświetlanych kanałów na osi X; punktem którego położenie przy rozciągnięciu nie zmienia się jest pozycja markera głównego

**- ściśnięcie obszaru na osi X względem markera głównego**

powoduje dwukrotne zwiększenie ilości wyświetlanych kanałów na osi X; punktem którego położenie przy ściśnięciu nie zmienia się jest pozycja markera głównego

**- włączanie i wyłączanie markera głównego dla osi X**

powoduje, że główny marker jest lub nie jest wyświetlany na danym wykresie

**- włączanie i wyłączanie markerów lewego i prawego dla osi X**

powoduje, że lewy i prawy marker są lub nie są wyświetlane na danym wykresie

**- przesuwanie markera (głównego, lewego i prawego) dla osi X**

powoduje, że marker jest przesuwany w wybranym kierunku na osi X;

- w momencie, gdy marker osiąga pozycję równą *oś X maksimum*, próba jego przesunięcia w kierunku wyższych wartości kanału, powoduje przesunięcie całego wykresu w kierunku niższych wartości kanałów;

- w momencie, gdy marker osiąga pozycję równą *oś X minimum*, próba jego przesunięcia w kierunku niższych wartości kanału, powoduje przesunięcie całego wykresu w kierunku wyższych wartości kanałów

**- blokowanie wzajemnej pozycji pomiędzy markerem lewym i prawym dla osi X**

powoduje, że odległość pomiędzy lewym i prawym markerem jest zablokowana, w momencie poruszenia którymś z markerów, automatycznie przesuwa się drugi marker o identyczną wartość kanałów

**- przesuwanie zablokowanych markerów lewego i prawego dla osi X**

powoduje, że markery lewy i prawy przesuwane są jednocześnie w wybranym kierunku na osi X;

- w momencie, gdy prawy marker osiąga pozycję równą *oś X maksimum*, próba przesunięcia markerów w kierunku wyższych wartości kanału, powoduje przesunięcie całego wykresu w kierunku niższych wartości kanałów;

- w momencie, gdy lewy marker osiąga pozycję równą *oś X minimum*, próba przesunięcia markerów w kierunku niższych wartości kanału, powoduje przesunięcie całego wykresu w kierunku wyższych wartości kanałów;

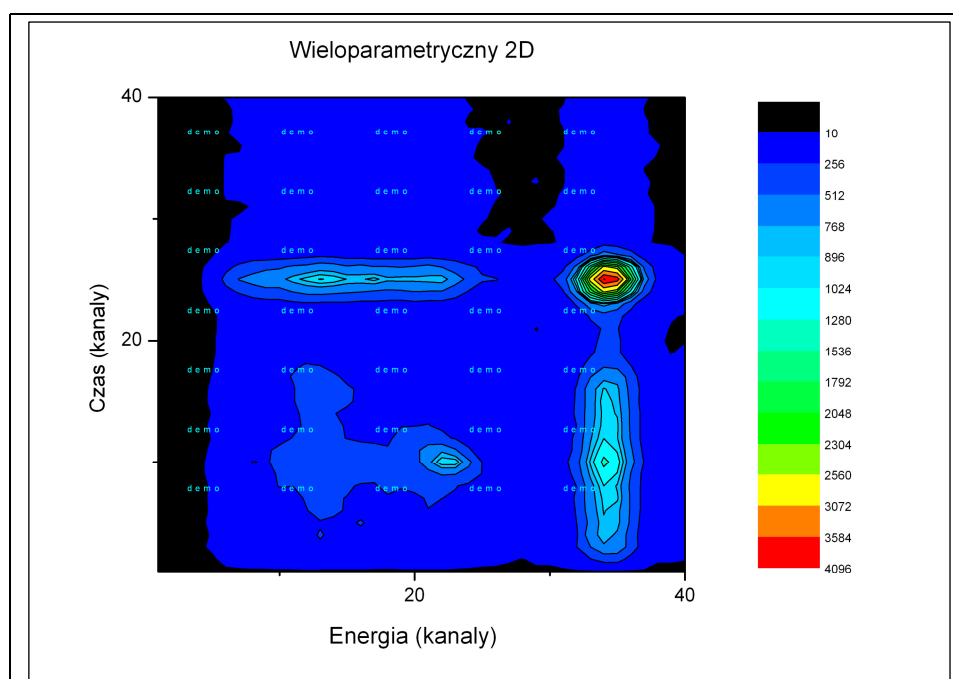
### 3.2.5.3.2 Wykresy wieloparametryczne 2D

#### 3.2.5.3.2.1 Wstęp

Przy prezentacji wykresów wieloparametrycznych 2D (dwuwymiarowych) oś X reprezentuje numer kanału pierwszego wybranego przetwornika ADC, oś Y - numer kanału drugiego przetwornika, natomiast oś Z reprezentowana jest przez skalę barw. Danemu przedziałowi zliczeń odpowiada wcześniej zdefiniowany kolor.

#### 3.2.5.3.2.2 Rodzaje wykresów

Dla wykresów wieloparametrycznych 2D poniższy rysunek pokazuje przykład graficzny takiego wykresu.



#### 3.2.5.3.2.3 Atrybuty wykresów (opis najważniejszych atrybutów)

Jeżeli atrybut występujący dla osi Y jest analogiczny z atrybutem występującym dla osi X to nazwa tego atrybutu została podana w nawiasie

##### **Oś X minimum (oś Y minimum)**

Numer pierwszego kanału dla wyświetlanego widma dla parametru przypisanego do tej osi.

##### **Oś X maksimum (oś Y maksimum)**

Numer ostatniego kanału dla wyświetlanego widma dla parametru przypisanego do tej osi.

##### **Oś Z minimum**

Wartość zliczeń od której rysowane będzie widmo. Odpowiada minimalnej wartości zliczeń prezentowanej na legendzie z zakresami kolorów. Domyślnie „0”.

### **Oś Z maksimum**

Wartość zliczeń do której rysowane będzie widmo. Odpowiada maksymalnej wartości zliczeń prezentowanej na legendzie z zakresami kolorów. Domyślnie – auto, wszystkie wartości prezentowane są bez obciążenia.

### **Oś Z skalowanie**

Określa sposób skalowania osi Z:

- automatyczny, wszystkie wartości prezentowane są bez obciążenia przy maksymalnym rozciągnięciu wyświetlanego wykresu na osi Z. Atrybut *oś Z maksimum* wyliczany jest na podstawie wzoru:

$$oś\ Z\ maksimum = Z_{max} + 0,1 * (Z_{max} - oś\ Z\ minimum)$$

gdzie  $Z_{max}$  oznacza maksymalną wartość zliczeń (dla widm) dla obszaru [(*oś X minimum*, *oś X maksimum*), (*oś Y minimum*, *oś Y maksimum*)]

- ręczny, wszystkie wartości prezentowane są w zakresie od *oś Z minimum* do *oś Z maksimum*.

### **Oś Z ilość zakresów**

Określa ile zakresów i odpowiadających im kolorów wyświetlanych będzie na wykresie.

### **Oś Z tablica zakresów**

Zawiera tablicę z zakresami zliczeń i odpowiadającymi im kolorami (w formacie RGB). Wielkość tablicy określa wyrażenie: *oś Z ilość zakresów* + 1

### **Oś Z typ skali**

Określa rodzaj użytej skali: liniowa lub logarytmiczna.

### **Oś X opis osi (oś Y opis osi)**

Pole tekstowe opisujące nazwę wyświetlanego parametru na osi danej.

### **Oś Z opis osi**

Pole tekstowe opisujące nazwę wyświetlanego parametru na osi Z. Umieszczony jest nad legendą z zakresami kolorów.

### **Marker główny pozycja**

Określa pozycję głównego markera poruszającego się wzdłuż osi X i osi Y.

### **Marker główny aktywny**

Określa czy główny marker wyświetlany jest na wykresie. Główny marker składa się z dwóch odcinków przecinających się pod kątem prostym. Każdy odcinek reprezentuje odpowiednią współrzędną. Atrybut przyjmuje wartości logiczne prawda lub fałsz.

### **Marker lewy-dolny pozycja**

Określa pozycję lewego-dolnego markera poruszającego się wzdłuż osi X i osi Y.

### **Marker prawy-górny pozycja**

Określa pozycję prawego-górnego markera poruszającego się wzdłuż osi X i osi Y.

### Marker lewy-dolny i prawy-górny aktywny

Określa czy markery lewy-dolny i prawy-górny wyświetlane są na wykresie. Przyjmuje wartości logiczne prawda lub fałsz.

### Marker lewy-dolny i prawy-górny zablokowany

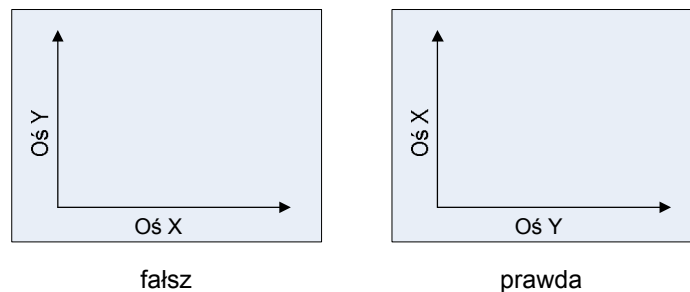
Określa czy odległość pomiędzy lewym-dolnym i prawym-górnym markerem jest zablokowana. Przyjmuje wartości logiczne prawda lub fałsz.

### Proporcje zablokowane

Określa czy operacje powiększania, ściskania i rozciągania mają być wykonywane jednocześnie na osiach X i Y (wartość prawda) czy każda z osi traktowana jest niezależnie (wartość fałsz).

### Wykres 2D zamiana osi

Określa czy nastąpiła zamiana osi na prezentowanym wykresie. Przyjmuje wartości prawda lub fałsz



Rysunek 4: Zamiana osi na wyświetlanych wykresach 2D

#### 3.2.5.3.2.4 Operacje możliwe do przeprowadzania na wykresach 2D

Jeżeli operacja dla osi Y jest analogiczna z operacją występującą dla osi X to nazwa tej operacji została podana w nawiasie.

#### Zmiana obszaru wyświetlanego widma

##### - przywrócenie pełnego zakresu wyświetlania

powoduje, że niezależnie od ustawień wykres prezentowany jest w pełnym zakresie dla wartości osi X, osi Y i osi Z

##### - ręczne ustawienie zakresu na osi X (ręczne ustawienie zakresu na osi Y)

powoduje, że wykres prezentowany jest w zakresie od ręcznie ustawionych wartości *oś X minimum* i *oś X maksimum* (*oś Y minimum* i *oś Y maksimum*)

##### - automatyczne ustawienie zakresu na osi Z

powoduje, że wykres prezentowany jest w zakresie od 0 lub ręcznie ustawionej wartości *oś Z minimum* do automatycznie wyliczanej wartości *oś Z maksimum*

##### - zwiększenie dwukrotne skali osi Z

powoduje, przy ręcznym ustawieniu zakresu na osi Z, dwukrotne zwiększenie wyświetlanego zakresu na osi Z

**- zmniejszenie dwukrotne skali osi Z**

powoduje, przy ręcznym ustawieniu zakresu na osi Z, dwukrotne zmniejszeni wyświetlanego zakresu na osi Z

**- przesuwanie wykresu wzdłuż osi X (przesuwanie wykresu wzdłuż osi Y)**

powoduje przesunięcie wykresu wzdłuż osi X (osi Y) w obu kierunkach tej osi przy ręczne ustawianym zakresie na osi X (osi Y); operację można wykonać za pomocą myszy lub klawiszy kursora

**- powiększenie obszaru pomiędzy markerami na osi X (powiększenie obszaru pomiędzy markerami na osi Y)**

powoduje wyświetlenie (rozciągnięcie) wykresu w obszarze wyznaczonym przez pozycję lewego-dolnego i prawego-górnego markera (o ile wartość atrybutu *marker lewy-dolny i prawy-górny aktywny* jest logiczną prawdą) na osi X (osi Y); jeżeli atrybut *proporcje zablokowane* jest prawdą to operacja dotyczy obu osi jednocześnie

**- rozciąganie obszaru na osi X (rozciąganie obszaru na osi Y)**

powoduje dwukrotne zmniejszenie ilości wyświetlanych kanałów na osi X (osi Y); punktem którego położenie przy rozciągnięciu nie zmienia się jest punkt środkowy wyświetlanego wykresu o numerze kanału równym  $\frac{1}{2} * (oś X maksimum - oś X minimum)$  ( $\frac{1}{2} * (oś Y maksimum - oś Y minimum)$ ); jeżeli atrybut *proporcje zablokowane* jest prawdą to operacja dotyczy obu osi jednocześnie

**- ściśnięcie obszaru na osi X (ściśnięcie obszaru na osi Y)**

powoduje dwukrotne zwiększenie ilości wyświetlanych kanałów na osi X (osi Y); punktem którego położenie przy ściśnięciu nie zmienia się jest punkt środkowy wyświetlanego wykresu o numerze kanału równym  $\frac{1}{2} * (oś X maksimum - oś X minimum)$  ( $\frac{1}{2} * (oś Y maksimum - oś Y minimum)$ ); jeżeli atrybut *proporcje zablokowane* jest prawdą to operacja dotyczy obu osi jednocześnie

**- rozciąganie obszaru na osi X względem markera głównego (rozciąganie obszaru na osi Y względem markera głównego)**

powoduje dwukrotne zmniejszenie ilości wyświetlanych kanałów na osi X (osi Y); punktem którego położenie przy rozciągnięciu nie zmienia się jest pozycja markera głównego; jeżeli atrybut *proporcje zablokowane* jest prawdą to operacja dotyczy obu osi jednocześnie

**- ściśnięcie obszaru na osi X względem markera głównego (ściśnięcie obszaru na osi Y względem markera głównego)**

powoduje dwukrotne zwiększenie ilości wyświetlanych kanałów na osi X (osi Y); punktem którego położenie przy ściśnięciu nie zmienia się jest pozycja markera głównego; jeżeli atrybut *proporcje zablokowane* jest prawdą to operacja dotyczy obu osi jednocześnie

**- włączanie i wyłączanie markera głównego**

powoduje, że główny marker jest lub nie jest wyświetlany na danym wykresie

**- włączanie i wyłączanie markerów lewego-dolnego i prawego-górnego**

powoduje, że markery lewy-dolny i prawy-górny są lub nie są wyświetlane na danym wykresie

**- przesuwanie markera (głównego, lewego-dolnego i prawego-górnego) dla osi X (osi Y)** powoduje, że marker jest przesuwany w wybranym kierunku na osi X (osi Y);

- w momencie, gdy marker osiąga pozycję równą *oś X maksimum (oś Y maksimum)*, próba jego przesunięcia w kierunku wyższych wartości kanału, powoduje przesunięcie całego wykresu w kierunku niższych wartości kanałów dla danej osi;

- w momencie, gdy marker osiąga pozycję równą *oś X minimum (oś Y minimum)*, próba jego przesunięcia w kierunku niższych wartości kanału, powoduje przesunięcie całego wykresu w kierunku wyższych wartości kanałów dla danej osi;

- za pomocą klawiszy kursora możemy sterować kursorem w 8 kierunkach (lewo, prawo, góra, dół + 4 kierunki po skosie) natomiast za pomocą myszy możemy poruszać kursorem w dowolnym kierunku na płaszczyźnie x-y

**- blokowanie wzajemnej pozycji pomiędzy markerem lewym-dolnym i prawym-górnym dla osi X (osi Y)**

powoduje, że odległość na osi X (osi Y) pomiędzy lewym-dolnym i prawym-górnym markerem jest zablokowana, w momencie poruszenia którymś z markerów, automatycznie przesuwa się drugi marker o identyczną wartość kanałów; jeżeli atrybut *proporcje zablokowane* jest prawdą to operacja dotyczy obu osi jednocześnie

**- przesuwanie zablokowanych markerów lewego-dolnego i prawego-górnego dla osi X (osi Y)**

powoduje, że markery lewy-dolny i prawy-górny przesuwane są jednocześnie w wybranym kierunku na osi X (osi Y);

- w momencie, gdy prawy-górny marker osiąga pozycję równą *oś X maksimum (oś Y maksimum)*, próba przesunięcia markerów w kierunku wyższych wartości kanału, powoduje przesunięcie całego wykresu w kierunku niższych wartości kanałów dla danej osi;

- w momencie, gdy lewy-dolny marker osiąga pozycję równą *oś X minimum (oś Y minimum)*, próba przesunięcia markerów w kierunku niższych wartości kanału, powoduje przesunięcie całego wykresu w kierunku wyższych wartości kanałów dla danej osi

## **Operacje na wyświetlanym obszarze widma**

**- rzutowanie pełnego widma 2D**

powoduje, że wyświetlane widmo rzutowane jest na osie X i Y; należy podać nazwy parametrów (reprezentowanych przez tablice jednowymiarowe) przypisanych do tych rzutów; rzutowanie na oś X (oś Y) polega na zsumowaniu wszystkich wartości zliczeń na osi Y (osi X) dla danej wartości kanału na osi X (osi Y)

**- rzutowanie obszaru wyznaczonego przez markery lewy-dolny i prawy-górny**

powoduje, że prostokątny obszar wyznaczony przez markery lewy-dolny i prawy-górny rzutowany jest na osie X i Y; należy podać nazwy parametrów (reprezentowanych przez tablice jednowymiarowe) przypisanych do tych rzutów

**- rzutowanie obszaru nieregularnego**

powoduje, że zaznaczony obszar nieregularny rzutowany jest na osie X i Y; należy podać nazwy parametrów (reprezentowanych przez tablice jednowymiarowe) przypisanych do tych rzutów

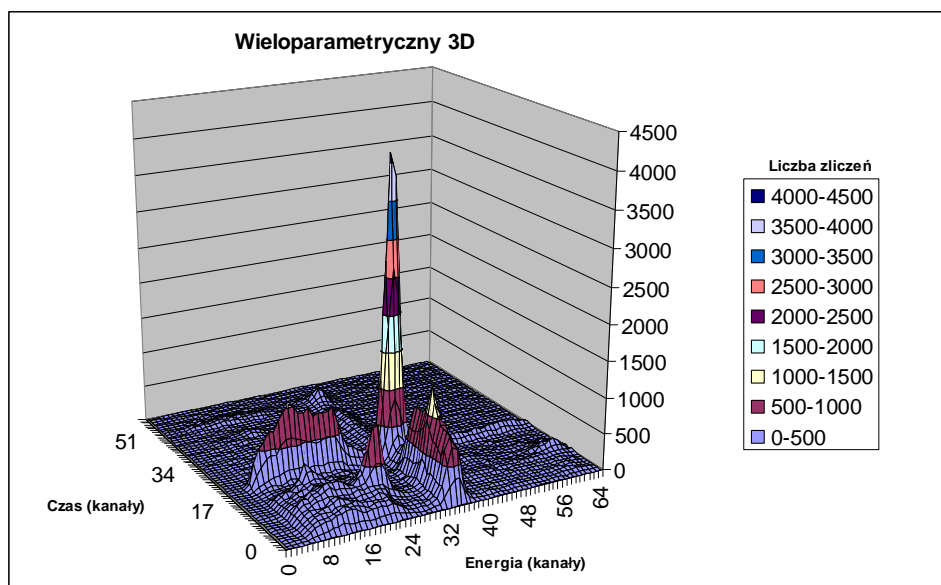
### 3.2.5.3.3 Wykresy wieloparametryczne 3D

#### 3.2.5.3.3.1 Wstęp

Przy prezentacji wykresów wieloparametrycznych 3D (trójwymiarowych) oś X reprezentuje numer kanału pierwszego wybranego przetwornika ADC, oś Y - numer kanału drugiego przetwornika, natomiast oś Z wyświetla ilość zliczeń dla kanałów o współrzędnych (x,y).

#### 3.2.5.3.3.2 Rodzaje wykresów 3D

Dla wykresów wieloparametrycznych 3D mamy dwa rodzaje wykresów bez i z kolorowaniem. Poniższy rysunek pokazuje przykład graficzny wykresu z kolorowaniem. Danemu przedziałowi zliczeń odpowiada wcześniej zdefiniowany kolor.



#### 3.2.5.3.3.3 Atrybuty wykresów (opis najważniejszych atrybutów)

Wszystkie atrybuty są takie jak w przypadku wykresów 2D. Do ustalenia pozostaje kwestia markerów.

Dodatkowo występują następujące atrybuty:

#### Wykres 3D kolorowanie

Określa, czy wykres ma być wyświetlany jako jednokolorowy (wartość fałsz) czy ma wystąpić jego kolorowanie (wartość prawda), w którym kolory odpowiadają liczbie zliczeń przedstawianej prezentowanej w legendzie.

#### Obrót 3D

Określa położenie wykresu w przestrzeni X,Y,Z. Reprezentowany jest przez 3 wartości liczbowe odpowiadające kątom nachylenia osi X, Y i Z.

#### Zbliżenie (oddalenie)

Określa rozmiar wyświetlanego wykresu względem jego oryginalnej wielkości (100%)



#### 3.2.5.3.3.4 Operacje możliwe do przeprowadzania na wykresach 3D

Wszystkie operacje są takie jak w przypadku wykresów 2D. Do ustalenia pozostaje kwestia operacji związanych z markerami oraz rzutowaniem.

Dodatkowo występują następujące operacje:

##### - włączanie i wyłączanie kolorowania wykresu 3D

powoduje, że wykres jest wyświetlany jako jednokolorowy lub kolorowany, w którym kolory odpowiadają liczbie zliczeń przedstawianej prezentowanej w legendzie

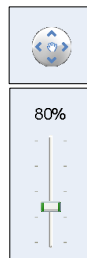
##### - obrót 3D

powoduje, że wykres można dowolnie wyświetlić w przestrzeni X,Y,Z dokonując obrotów we wszystkich osiach, operacja ta możliwa jest jedynie z wykorzystaniem myszy; przecięcie osi obrotu znajduje się w środku ciężkości prostopadłościanu, w którym zawarty jest wykres

##### - zbliżenie (oddalenie)

powoduje, że wykres można dowolnie przybliżyć lub oddalić w przestrzeni X,Y,Z; operacja ta możliwa jest jedynie z wykorzystaniem myszy; punkt stały przekształcenia znajduje się w środku ciężkości prostopadłościanu, w którym zawarty jest wykres

Operację wykonania obrotów 3D i zbliżania (oddalania) wymagają użycia specjalnych elementów interfejsu użytkownika. Ich realizacja może wyglądać jak na rysunku poniżej.



Górny element odpowiedzialny jest za manipulowanie obrotami, dolny – za zbliżanie i oddalanie wykresu.

## 3.2.6 System analizy danych.

### 3.2.6.1 Wstęp

Podobnie jak w przypadku bibliotek graficznych nie należy tworzyć zaawansowanych bibliotek do analizy danych. Stworzenie takich bibliotek wymaga dobrej znajomości zasad pisania algorytmów numerycznych, długiego okresu testowania, weryfikacji poprawności i dokładności obliczeń. Zamiast tego proponuję wykorzystanie sprawdzonej (uznanej jako standard) i kompletnej (spełniającej wymagania systemu wieloparametrycznego) biblioteki numerycznej systemu ROOT stworzonego w ośrodku naukowo-badawczym CERN (<http://root.cern.ch>).

Naturalnym językiem programowania dla biblioteki ROOT jest interpreter języka C++ – CINT. Stworzono również biblioteki dla języka Python – PyROOT. Istnieje także ograniczone wsparcie dla języka Java – javaROOT.

### 3.2.6.2 Lista dostępnych funkcji z zaznaczeniem wykorzystania biblioteki ROOT

#### 3.2.6.2.1 *Widma jednowymiarowe*

- obliczanie parametrów pojedynczego pliku (ROOT)
  - **tło pod pikiem** - wyznaczone jest za pomocą funkcji liniowej prowadzonej pomiędzy interpolowanymi kanałami początku i końca obszaru. Interpolacje wyznacza się z trzech kanałów leżących na zewnątrz obszaru i kanału, na którym stoi marker;
  - **pole całkowite** - obliczane jako suma liczby zliczeń we wszystkich kanałach należących do obszaru;
  - **pole netto** - obliczane jako suma liczby zliczeń netto w pikie:
$$\text{pole\_netto} = \text{pole\_calko} \quad \text{wite} - \frac{(\text{B}_L - \text{B}_P) * N}{2}$$
gdzie:
    - $\text{B}_L$  - średnia liczba zliczeń w 3 kanałach leżących z lewej strony pikie,
    - $\text{B}_P$  - średnia liczba zliczeń w 3 kanałach leżących z prawej strony pikie
    - $N$  - liczba kanałów wewnątrz pikie;
  - **centrum** - położenie pikie (centrum) obliczane z wartości netto zliczeń metodą średniej ważonej;
  - **Fwhm** - (ang. Full Width at Half Maximum) szerokość połówkowa pikie;
  - **Fwtm** – (ang. Full Width at Tenth Maximum) szerokość pikie w 1/10 jego wysokości;
- ręczne zaznaczanie pików na widmie
- automatyczne wyszukiwanie pików na widmie
- identyfikacja nuklidów (ROOT, klasa TGeoElementRN, przykłady \$ROOTSYS/tutorials/RadioNuclides.C)

- fitowanie pojedynczych pików

tworzenie tablicy ROI (ang. Region Of Interest)

### 3.2.6.2.2 *Widma dwuwymiarowe 2D*

- obliczanie parametrów pojedynczego pliku (ROOT)

- ręczne zaznaczanie obszarów na widmie

- poszukiwanie automatyczne obszarów (ang. Range Searching) (ROOT)

- rzutowanie wybranych obszarów na osie X i Y

## 3.2.7 System sterowania.

### 3.2.7.1 Wstęp

System sterowania poprzez bibliotekę urządzeń wirtualnych zapewnia interakcję ze sprzętem (zapis parametrów, odczyt stanu, inicjalizację urządzeń, ich kasowanie, itp.). Dodatkowo jego nadrzędną funkcją jest sterowanie pomiarem (eksperymentem) i kontrola jego przebiegu.

### 3.2.7.2 Sterowanie pomiarem

Do ręcznego sterowania pomiarem mamy następujące funkcje:

#### - **Start**

rozpoczyna zbieranie danych rzeczywistych lub odczyt danych z dysku zapisanych w formacie „list”. Przy obsłudze systemu on-line w pierwszej kolejności testowane są przetworniki, liczniki i inne urządzenia wchodzące w skład systemu wieloparametrycznego (w przypadku wykrycia błędu, odpowiedni komunikat o ich stanie pojawia się na ekranie). Równocześnie następuje faza inicjalizacji części pomiarowej systemu według wcześniej wprowadzonych danych z formularza opisu eksperymentu. Ponowne wybranie funkcji **start** możliwe jest dopiero, gdy użytkownik uruchomi funkcję **koniec**.

#### - **Pauza**

czasowo przerywa zbieranie danych. Ponowne uruchomienie zbierania następuje po wybraniu funkcji **kontynuacja**.

#### - **Kontynuacja**

powoduje ponowne uruchomienie zbierania danych, przerwane wcześniej przez funkcję **pauza**. Wznowienie zbierania odbywa się bez inicjalizacji systemu.

#### - **Koniec**

kończy zbieranie danych. Ponowne uruchomienie zbierania jest możliwe tylko przy użyciu funkcji **start**, ale wiąże się to z powtórnią inicjalizacją systemu. Koniec zbierania danych może nastąpić automatycznie zgodnie z ustawionymi opcjami na formularzu opisu eksperymentu.

### 3.2.7.3 Kontrola przebiegu pomiaru

System wieloparametryczny zapewnia różne możliwości automatycznej kontroli przebiegu pomiaru. Poniżej przedstawiono najważniejsze z nich:

#### - Pomiar pojedynczy

Powoduje, że system zostanie uruchomiony zgodnie z **kryterium startu pomiaru** i zakończy pracę zgodnie **kryterium końca pomiaru**.

#### - Seria pomiarowa

Powoduje, że system zostanie uruchomiony zgodnie z **kryterium startu pomiaru** i zakończy pracę zgodnie **kryterium końca pomiaru**. W czasie pomiaru system wykona **n** cykli pomiarowych z zadanymi parametrami: **kryterium końca cyklu** i **czasie przerwy**.

#### - Kryteria startu pomiaru

ustawienie czasu startu pomiaru (data i czas zegarowy)  
wyzwolenie startu pomiaru zewnętrznym sygnałem

#### - Kryteria końca pomiaru

ustawienie czasu końca pomiaru (data i czas zegarowy)  
wyzwolenie końca pomiaru zewnętrznym sygnałem  
przekroczenie zadanego całkowitego czasu żywego<sup>1</sup> (interwał czasu)  
przekroczenie zadanego czasu rzeczywistego (interwał czasu)  
przekroczenie zadanej liczby zliczeń dla danego parametru  
przekroczenie zadanej liczby zarejestrowanych zdarzeń

#### - Kryteria końca cyklu

przekroczenie zadanego całkowitego czasu żywego (interwał czasu)  
przekroczenie zadanego czasu rzeczywistego (interwał czasu)  
przekroczenie zadanej liczby zliczeń dla danego parametru  
przekroczenie zadanej liczby zarejestrowanych zdarzeń

#### - Czas przerwy

Określa czas przerwy pomiędzy cyklami pomiarowymi dla danej serii pomiarowej.

#### - Kasowanie danych

Określa czy po wykonaniu cyklu pomiarowego i ewentualnym zapisaniu danych, dane z tablic zawierających posortowane zdarzenia mają zostać skasowane. Przyjmuje wartość prawda lub fałsz.

#### - Zapisywanie danych

Po wykonaniu pomiaru lub cyklu pomiarowego możliwe jest automatyczne zapisanie posortowanych danych. Przyjmuje wartość prawda lub fałsz.

---

<sup>1</sup> Czas żywy jest to czas rzeczywisty pomniejszony o czas, jaki zajmuje systemowi obróbka poszczególnych impulsów z detektora.

### 3.2.7.4 Formularz opisu eksperymentu

Formularz opisu eksperymentu służy do przygotowywania danych dotyczących inicjalizacji systemu wieloparametrycznego oraz sposobu prowadzenia pomiarów. Na formularzu tym zapisuje się następujące dane:

- **imię i nazwisko** osoby prowadzącej pomiar (eksperyment)
- **data i czas** utworzenia formularza
- **komentarz** opisujący pomiar (eksperyment)
- przypisanie numerów parametrów (używanych przy wyświetlaniu i analizie danych) do fizycznych urządzeń
- ustawienie parametrów dla wirtualnych urządzeń występujących w danej konfiguracji systemu wieloparametrycznego (patrz: 3.2.3 Biblioteka urządzeń wirtualnych)
- ustalenie rodzaju i czasu koincydencji (patrz: 3.2.3.5 Blok paternu i koincydencji)
- ustalenie parametrów przebiegu pomiaru (patrz: 3.2.7.3 Kontrola przebiegu pomiaru)