|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| logoKM | logoKM | | POLITECHNIKA ŚLĄSKA  WYDZIAŁ ELEKTRYCZNY  KATEDRA MECHATRONIKI | logoKM |
| Instrukcja do ćwiczenia laboratoryjnego | | | | |
| Przedmiot: | | Przetwarzanie i Wizualizacja Danych Pomiarowych | |  |
| Symbol ćwiczenia: | | PiWDP10 | |  |
| Tytuł ćwiczenia: | | **Struktury programistyczne: struktura Event** | | |

**SPIS TREŚCI**

|  |  |  |
| --- | --- | --- |
|  | **SPIS RYSUNKÓW** | 2 |
| 1. | **CELE ĆWICZENIA** | 3 |
| 2. | **WPROWADZENIE** | 3 |
| 2.1. | **Struktura Event** | 3 |
| 2.2. | **Zdarzenia i filtry** | 3 |
| 3. | **LABORATORYJNE STANOWISKO BADAWCZE** | 3 |
| 3.1. | **Obiekt badany** | 3 |
| 3.2. | **Urządzenia dodatkowe** | 3 |
| 3.3. | **Oprogramowanie** | 3 |
| 4. | **PROGRAM ĆWICZENIA – WYKAZ ZADAŃ DO REALIZACJI** | 3 |
| 5. | **PRZYKŁAD REALIZACJI ZADANIA – filtracja zdarzenia zamknięcia aplikacji** | 4 |
| 6. | **RAPORT** | 9 |
| 7. | **PYTANIA** | 9 |

# SPIS RYSUNKÓW

|  |  |  |
| --- | --- | --- |
| 1. | Tworzenie zmiennej lokalnej. | 3 |
| 2. | Widok zmiennej globalnej: powłoka, zmienna i panel zmiennej. | 4 |
| 3. | Widok szablonu zmiennej funkcjonalnej. | 5 |
| 4. | Zmienna funkcjonalna realizująca dwa zadania zapisu i odczytu. | 5 |
| 5. | Przykład pracy równoległej z występowaniem efektu Race Condition. | 6 |
| 6. | Zabezpieczenie przed efektem Race Condition z zastosowaniem zmiennej funkcjonalnej. | 7 |
| 7. | Zastosowanie zmiennej typu FVG do odmierzania przedziałów czasu. | 9 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

1. **CELE ĆWICZENIA**

Celem ćwiczenia jest poznanie się z:

* Obsługą struktury Event do obsługi zdarzeń wywołanych przez obiekty panelu
* Programowaniem zdarzeń oraz filtrów w strukturze Event.

1. **WPROWADZENIE**
   1. **Struktura Event**
   2. **Zdarzenia i filtry**
2. **LABORATORYJNE STANOWISKO BADAWCZE**
   1. **Obiekt badany**

- Środowisko programistyczne LabVIEW,

* 1. **Urządzenia dodatkowe**

- brak,

* 1. **Oprogramowanie**
* LabVIEW 2013 lub nowszy

1. **PROGRAM ĆWICZENIA – WYKAZ ZADAŃ DO REALIZACJI**

Kolejne kroki do wykonania podczas zajęć:

* Zastosowanie struktury Event:

- otworzyć projekt „*System pomiarowy DAQ*”,

- otworzyć plik „*main\_rownolegle.vi*”,

- umieścić na panelu kontrolki: **Temp Limits**, **Zakres uśredniania** (typu **I32**),

- umieścić na diagramie kolejną pętlę **While** (powyżej pętli „*Akwizycja*”),

- Nazwać nową pętlę „*GUI*”: **PPM** 🡪 **Visible Items** 🡪 **Subdiagram Label**,

- wewnątrz pętli umieścić strukturę Event,

- w strukturze Event dodać zdarzenie **stop** 🡪 **Value change**,

- przenieść terminal przycisku **stop** do utworzonego zdarzenia wewnątrz struktury **Event** i podpiąć tak, żeby kończył pętlę,

- uzupełnić aplikację tak, aby przycisk wyłączał także pętlę Akwizycja (za pomocą zmiennej lokalnej – przykład w poprzedniej instrukcji),

- utworzyć obsługę dwóch zdarzeń: „**Temp limits** 🡪 **Value change**, **Zakres uśredniania** 🡪 **Value change**”,

- umieścić odpowiednie kontrolki w zdarzeniu,

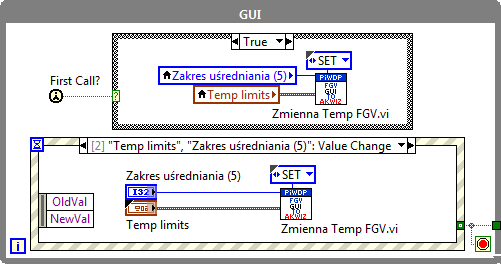
- przygotować zmienną funkcjonalną przenoszącą dane z pętli „*GUI*” do pętli „*Przetwarzanie*” (pamiętać o utworzeniu definicji typu „*Akcje\_rownolegle\_FGV.ctl*”),

- uzupełnić ikonę oraz dokumentację zmiennej funkcjonalnej,

- zapisać zmienną pod nazwą „*Zmienna Temp FGV.vi*”,

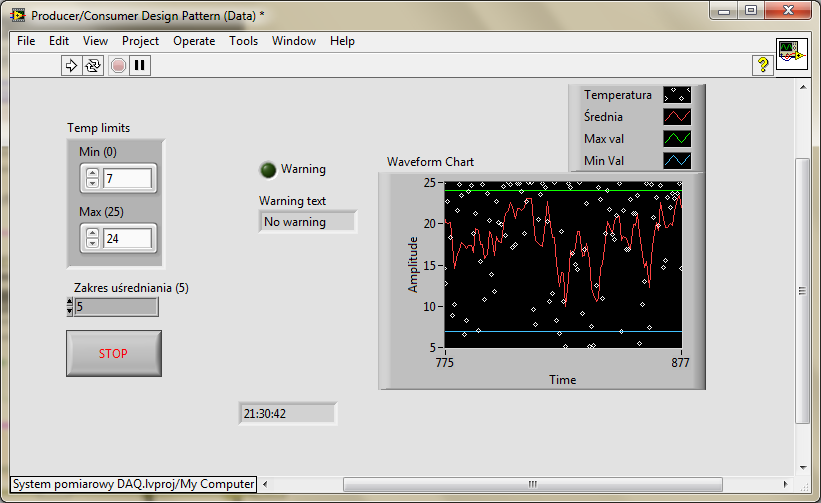
- wprowadzić dane akcją **SET** w zdarzeniu „**Temp limits**, **Zakres uśredniania** 🡪 **Value change**”,

- uzupełnić pętlę „GUI” o startowe przypisanie wartości do zmiennej za pomocą węzła „*First cal?*” jak na rysunku:



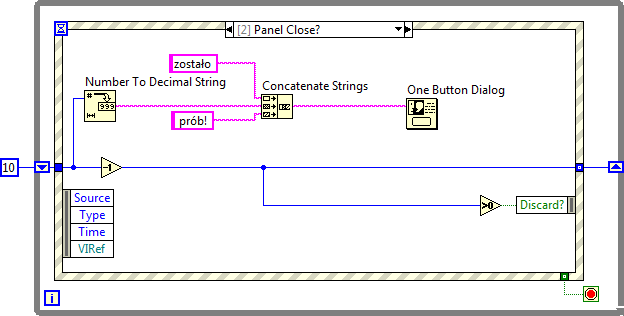
- uzupełnić pętlę „*Wyświetlanie*” tak, aby wskaźnik „*Waveform Chart*” wyświetlał przebiegi temperatury oraz ustawione wartości minimalną i maksymalną (skorzystać ze zmiennej funkcjonalnej w pętli „*Wyświetlanie*”),

- uzupełnić strukturę Event tak, żeby program w dole ekranu wyświetlał aktualną godzinę.

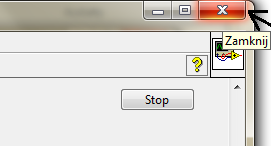


1. Widok panelu programu ze wskaźnikiem aktualnej godziny.
2. **PRZYKŁAD REALIZACJI ZADANIA – filtracja zdarzenia zamknięcia aplikacji**

Zdarzenie struktury Event liczące wyłączenia aplikacji za pomocą przycisku „***X***”. W tym celu zastosowano filtr „*Panel close?*”.



1. Struktura Event pozwalająca na wyłączenie aplikacji po 10 kliknięciach w „**X**”.



1. Miejsce kliknięcia działające na zdarzenie.
2. **RAPORT**

Raport z przeprowadzonego ćwiczenia laboratoryjnego powinien zawierać opis kolejnych czynności wykonywanych w trakcie realizacji ćwiczenia, zrzuty ekranu dokumentujące wykonane kroki oraz zanotowane parametry konfiguracyjne kart DAQ (mogą być zawarte  
w tabeli)

1. **PYTANIA**
2. Omówić metody programowania równoległego.
3. Wymienić metody przesyłania danych między pętlami równoległymi, podać zalety i wady każdej metody.
4. Co to jest „sztuczna zależność struktur”?

# LITERATURA

1. LabVIEW Core 1 Course manual.
2. LabVIEW Core 1 Exercise book.
3. LabVIEW Core 2 Course manual.
4. LabVIEW Core 2 Exercise book.
5. https://www.youtube.com/watch?v=iNm0zWY7o8g&list=PLUnVykytJXxPxm5u0vRKpPRVVg2u\_WgFg&index=3
6. https://www.youtube.com/watch?v=RuIN31rSO2k

Opracowanie: Marek Kciuk

# ZADANIA DO REALIZACJI PRZED PRZYSTĄPIENIEM DO ĆWICZENIA

- ukończyć zadania z instrukcji nr 7.

# ZADANIA DODATKOWE PO WYKONANIU ĆWICZENIA

- uzupełnić strukturę danych oraz działanie programu „*Main\_Maszyna.vi*”, żeby były przechowywane wektory próbek wartości *Temperatura* i *Średnia*. – zadanie ma na celu przygotowanie programu do uzupełnienia stau „*SAVE*”.

- zmodyfikować działanie aplikacji tak, żeby sekwencja: Measure, Analyze, Display była zapętlona przez zadany okres czasu, z możliwością wcześniejszego wyjścia.

- zmodyfikować program tak, żeby przycisk „*Zakończ Pomiar*” znajdował się w strukturze kontrolki „Akcje”.

# ZAŁĄCZNIKI

## Opóźnienia i zarządzanie czasem działania aplikacji