ITASDI PROJECT

Innovative Teaching Approaches in development of Software Designed Instrumentation and its application in real-time systems

Erasmus+ KA2 2018-1-RS01-KA203-000432

Course „Advanced LabVIEW Applications"

Laboratory no. 3 - Producer/Consumer Design Pattern

Leader Partner: [1]Warsaw University of Technology
Authors: Dariusz Tefelski [1]

Circulation: Public
Version: 02
Stage: Final
Date: 10.09.2019

1 Car wash

1.1 Goal

Create an application, which simulate the car wash.

1.2 General requirements

- The application should be hierarchical and scalable. Remember to use subVIs.
- Use the Producer/Consumer Design Pattern.
- Avoid using the local, global or shared variables if it is possible.
- Close all opened references and handles.
- Application shouldn't crash. Inform the user about the errors using the error cluster or a dialog box.
- Remember to prepare well documented code. Especially remember about: labels on long wires, description showing in context help, tip strips of controls and labels of constant values.

1.3 Description

- The front panel contains three sections: wash options, status and car position.
- User may choose the wash options and specify the name of car owner. When user clicks the *Start* button, the wash option and name will be added to queue.
- The status panel will show the actual running program, the Name of owner which car is being washed and the number of cars in queue on slider and numeric control.

Rysunek 1: The *front panel* of car wash.

- The car position should be changed to the appropriate station depending on chosen car options.

Co-funded by the
Erasmus+ Programme
of the European Union

1.4 Realization

- Use the Producer/Consumer Design Pattern, in which producer loop will add the wash options and name of car owner to queue. The washing simulation should be realized in consumer loop.

- Each chosen program should take 7000ms of time.

- Suggestion: you can use the state machine inside the consumer loop.

Please use the front panel from public folder.