

Project 02 description

The main goals of this project are to automate generation and testing of the restbus simulation. The materials for a tart are available on Canvas platform. As a guideline, the restbus simulation made in M04 exercise can be used. The final script should be able to automatically generate the restbus simulation. All networks, buses, nodes, databases, panels, and system variables should be automatically added. Also, a proper test environment should be created. Each automatically added part of restbus configuration should be covered by the relevant test cases.

Requirements to be implemented:

Run a Python script that should open an empty configuration. The script should generate content for the restbus simulation. After generation, the script should start a measurement and run tests. After successful testing, the simulation can be stopped and CANoe closed.

A successfully implemented project would have the same content as in the materials and test report should be available.

Regular points

1. The restbus simulation from materials should be extended to contain a test environment and proper test modules (10 pts.)

Req: Test environment and test modules can be manually created. The following functionalities should be covered:

- Verify that signal *Engine::EngineState::OnOff* changes value when system variable *Engine::EngineStateSwitch* (Ignition button) changes its value
- Verify that signal *Engine::EngineState::EngineSpeed* changes value when system variable *Engine::EngineSpeedEntry* (Speed slider) changes its value
- Verify that system variable *Light::LightDisplay* (Lights) changes value when signal *Light::LightState::HeadLights* changes its value

2. Create a Python script that can run CANoe, and create a new configuration (2 pts.)

Req: The script should be written in Python and use COM objects. After execution of the script, CANoe should be opened and a new configuration saved.

3. Extend a Python script to generate new networks and buses like in restbus simulation from materials (2 pts.)

Req: In case an empty configuration is used all needed networks and buses should be automatically added. It's not so crucial to take care of the naming or the order.

4. Extend the Python script to assign existing nodes and database from the materials (2 pts.)

Req: Following nodes from materials should be added:

- Engine
- Light
- Display

Req: The database „*Easy.arxml*“ from materials should be added.

5. Extend the Python script to assign existing test nodes, test environment or test modules from the first part of the project (2 pts.)

Req: All available test environments, test modules or test nodes written in the 1st bullet should be automatically added.

6. Extend the Python script to add existing panels and system variables from the materials (2 pts.)

Req: Following panels from the materials should be added:

- Display
- Control

Req: System variables „*Engine_and_Lights.vsysvar*“ should be automatically added to the configuration.

7. All network nodes are ready for use (2 pts.)

Req: All nodes are compilable and measurement can be started with no errors

8. All test nodes or test modules are ready for use (2 pts.)

Req: All test nodes or modules are compilable and measurement can be started with no errors

9. Test report should be available (1 pts.)

Req: At the end, test report should be available.

Negative points

1. The same segment of the code was used between the two groups. (10 pts.)
2. The same tests were used between the two groups. (10 pts.)