Journal of Advanced Research in Computer Graphics and Multimedia Technology
Volume 1, Issue 1 - 2019, Pg. No. 5-7
Peer Reviewed Journal

Article

Application of Control Desk Test Automation Graphical Tool for Quality Assurance in the Development of Advanced ECUs

Abhishek Shukla[1], Swati Darbari[2]

I N F O

Corresponding Author: Abhishek Shukla

E-mail Id:abhishek@gmail.com

How to cite this article:
Shukla A, Darbari S. Application of Control Desk Test Automation Graphical Tool for Quality Assurance in the Development of Advanced ECUs. *J Adv Res Comp Graph Multim Tech* 2019; 1(1): 5-7.

Date of Submission: 2019-02-02
Date of Acceptance: 2019-03-04

A B S T R A C T

Control Desk Test Automation is a tried and tested tool for quality assurance in the development of advanced ECUs. Control desk is a complete graphical tool which provides GUI for different input/output variables. One can create graphical panels for power supply, analog/digital inputs/outputs; CAN messages from different controllers, Interactive calibration, FIUs, diagnostic trouble codes and so on.Control Desk can be operated in two modes: The Developer mode gives you the full functionality, and the Operator mode protects your experiments against unauthorized changes.

Keywords: Control Desk, Engine Control Unit, Industry Automation, GUI, CAN, Python, Quality Assurance

Control Desk

ControlDesk Test Automation provides the features that allow the remote control of complete hardware-in-the-loop environment as well as automated operation of test benches including external laboratory devices.

A key feature of Control Desk Test Automation is its ability to build up and generate flexible sequences of signals that can be executed in real time. These signals are used as test stimuli, references, or to define the expected behavior. Import of measurement data recorded by external data loggers and replay it in real time can be done. Test automation is based on the modular scripting language Python.

The Specific functions of Python Modules in Control Desk test automation includes

- Generate test stimuli
- Access simulator variables
- Access the serial interface
- Access the ECU's diagnosis line
- Exchange data with MATLAB or invoke MATLAB commands
- Generate test reports using Microsoft Excel and Microsoft Word

As a result, the test operation can be automatically adjusted and analyzed while the application is running.

The Automated test procedure is as follows: At the beginning of a script,

1. The Python environment is initialized

- The required Python modules are imported.
- The script variables are defined. This includes the definition of parameters to adjust the operating point of the simulation, and complete descriptions of signal sequences used as test stimuli.

2. The simulation is started.
3. Perform real-time simulation the application to be tested is downloaded to the real-time hardware and started on the specified real-time processor.

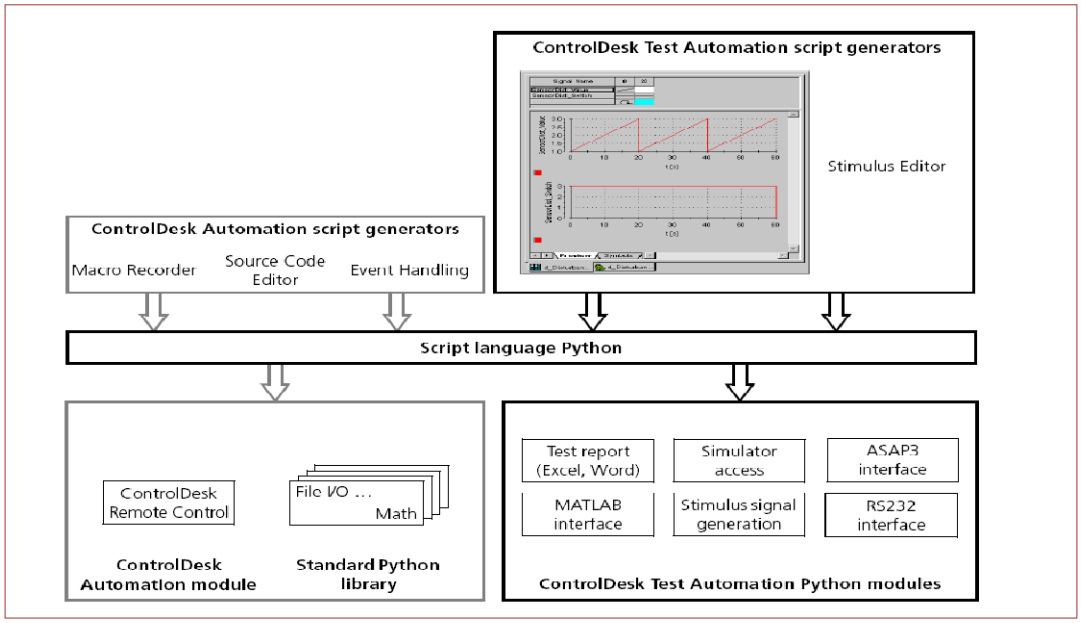

Figure 1.Overview of test automation using Control DeskCourtesy: dSPACE Inc (www.dspace.com)

4. Different test stimuli can be supplied simultaneously or one after another.
5. After test operation has been adjusted, measurement takes place by capturing variables (the signals to be measured).
6. The captured data is analyzed and evaluated according to the expected behavior.
7. The results can be reported in ASCII, Microsoft Word, or Microsoft Excel formats.

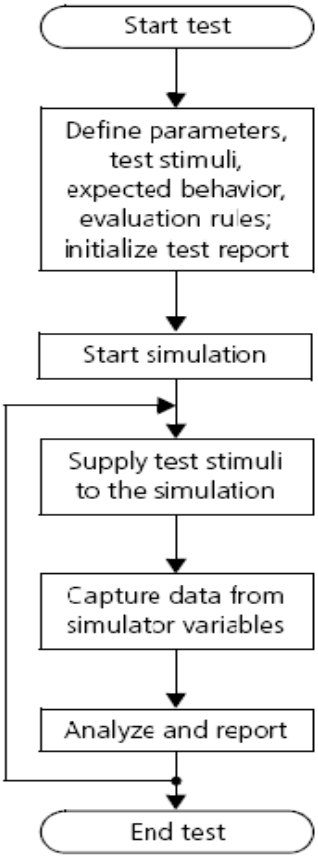

Figure 2.Flowchart for automated test procedure Courtesy: dSPACE Inc. (www.dspace.com)

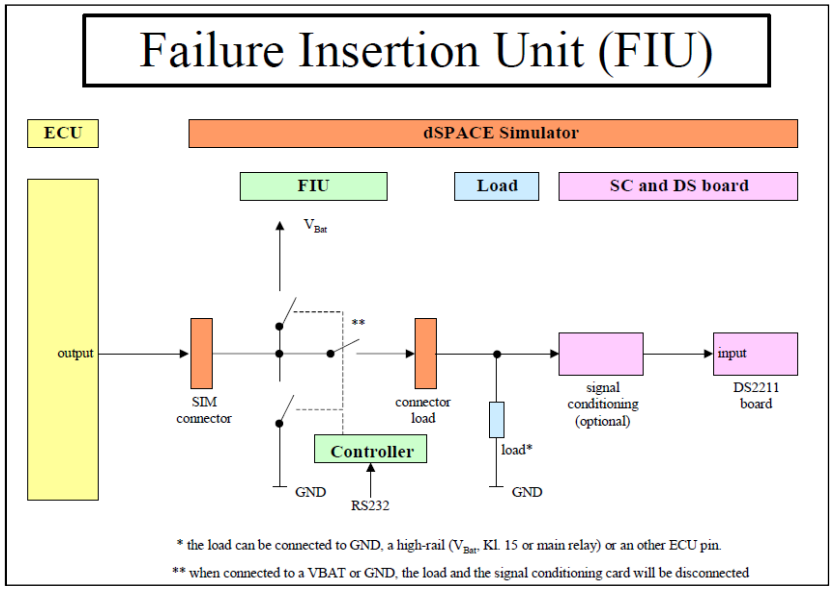

Figure 3.Failure Insertion unit. Courtesy: dSPACE Inc.(www.dspace.com)

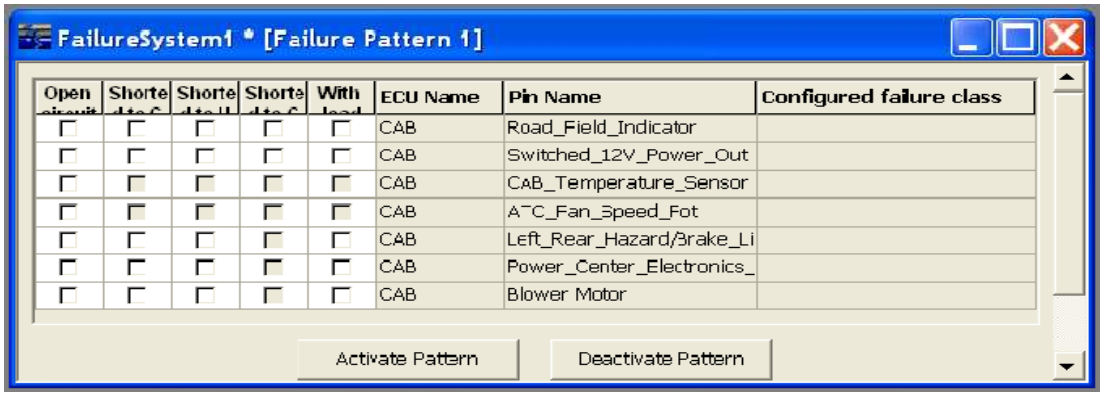

Figure 4.Control Desk GUI for failure system

Failure Simulation

Failure simulation is one of the major features of Control desk which provides platform to test failure conditions.

1. Traditional test methods require the test engineer to manually remove and plug cables using a breakout box to test failure conditions.
2. FIU module is a part of HIL simulator is a Microprocessor controlled device that can physically generate real-time electrical faults between two or more electrical components (E.g.: Open Circuit, Short Circuit or Short to Voltage).
3. The user can control the operation through Control Desk GUI or also through a simple scripting language.
4. The hardware can then modify the electrical signals which are exchanged between the devices.
5. Electrical faults such as cable breakage, loose contacts, corrosion, short circuits and leakage currents can be interactively or programmatically inserted (at a specific time or event) and the behavior of the Electronic control unit is observed.

In Control Desk three different panels e.g. Short to BAT, Short to GND and OPEN CKT are created for each input/ output of ECU. User can mark/unmark individual input/output to test the faults. GUI communicates with FIU module which resides on HIL simulator and activates or deactivates corresponding input or output for corresponding fault condition.

References

1. dSPACE GmbH. Automation Desk Guide" Release 5.0, 2005; 15-212.
2. dSPACE GmbH. Control Desk Experiment Guide" Release 5.0, 17-75, November 2005.
3. Hammond M, Robinson A. Python Programming on Win32. First Edition, O'Reilly, PP 15-45, 67-79, 2000; 351-369.
4. Martelli A, Ascher D. Python CookBook. First Edition, O'Reilly, PP 4.1-4.27, 11.1-11.12, 2002.
5. Society of Automotive Engineers, "J1939_31_SAE_Apr04_NetworkLayer" Rev April 2004, Appendix A.