

---

---

## TECHNICAL REPORT

# Microcontroller Emulation Meets LabVIEW

**Keywords:** Embedded Systems development process, v-model, seamless transition between the different phases, hardware/software integration, In-Circuit Emulator / On-Chip Emulator, On-Chip Debugger, automated test, graphical programming environment, LabVIEW, virtual instruments, remote control of In-Circuit Emulator / On-Chip Emulator, visualization of real-time data, test and diagnostics, CAN Bus, development of an electronic control unit, closed validation process at EDAG for vehicle tests

### **Embedded Systems development and test – a review**

For many years there has been a vision of creating a universal engineering tool for developing, debugging and testing software. Today, we are one step closer to having such a tool that can also be used in the world of embedded systems development.

National Instruments changed the world of programming by introducing their paradigm of virtual instruments and the LabVIEW platform. Today, National Instruments dominates the markets of test, measurement and automation. LabVIEW has grown in use in many environments, but has not yet been widely adopted in the embedded systems area. LabVIEW's use of graphical programming as a basis for building test and measurement applications has made it so successful thus far, and it certainly seems it could be so in embedded systems as well.

Graphical programming of embedded systems and designing test systems, both fully automated, is a challenge for most embedded systems tool vendors in regard to their next-generation products. Not only will tool vendors have a daunting task creating such an interface, but they will have to shift their paradigms and establish new software and hardware development processes to streamline the whole product development cycles. State of the art development and testing in the embedded market is still based on C or C++. Sometimes even assembly language is still employed. Specifications are loosely coupled to the final implementation. Should test suites be created at all, for the most part they are largely non-automated.

During the recent few years, UML or model-based object-oriented design as well as graphical programming approaches have explored the world of embedded development processes. When interviewing users trying out modelling tools, the feedback includes "Yes, we do model our applications, but the implementation is still done by hand in C", and "We are still evaluating such tools; they are partly used to do some specification work", and "We have to be on top of the code to ensure performance and reliability of our software."

The reason for these comments are the very specific requirements of an embedded system in terms of limited resources such as memory, real-time behaviour, the large variety of microcontrollers (8/16/32 bit), and more. As the complexity of embedded software grows, embedded design processes have to be standardized and the use of high-level tools enter this market. Code generation technology will be used more frequently. This is the only opportunity to lower costs and errors within embedded software, to achieve higher acceptance of the products and to be innovative in the future. The automotive manufacturing industry is doing this today, aerospace has already implemented these techniques, and others will follow.

## **Virtual Instruments connect the world of graphical programming to In-Circuit and On-Chip emulation**

Companies developing embedded systems currently use graphical programming environments for building prototypes and complex test systems. LabVIEW and the appropriate National Instruments hardware can support embedded systems development and test. However, the processes of hardware and software development integration as well as further steps of development and test don't merge seamlessly. Knowledge and experience from the early phases of the software development process, as well as prototypes and software already developed, can become lost with most of the work having to be done over again. This is due to the lack of well-defined interfaces between development tools. An automatic transition between the different development steps has not been implemented yet. To provide a graphical development environment for a complete end-to-end embedded systems project, major innovations in embedded tools (e.g., improvement of code generators, graphical debugging mechanisms, ...) would be required.

An intermediate solution, and one which capitalizes on "best of class" technology from both sides - embedded and graphical test - was to interface the graphical programming and visualization of LabVIEW with true in-circuit and on-chip emulation.

A user interface such as winIDEA for an in-circuit emulator is easy to use, but the ability to visualize measured (real-time) data is missing. Connecting an emulator to the front panel of LabVIEW enables a user to use an existing set of graphical objects to analyze the behavior of an embedded application on any target hardware. The In-Circuit Emulator or On-Chip Debugger can then be used for software/hardware integration, driver development and real-time, performance as well as code coverage analysis.

For the LabVIEW programmer, this interface allows use of an In-Circuit Emulator or On-Chip Debugger without deep user knowledge or experience with such tools. They employ the tools they are familiar with, LabVIEW, using graphical programming objects (virtual instruments) to remotely control an emulator. Thus, a LabVIEW program is now able to download embedded software to a target system, to execute the application and read/write data as well as run automated tests. The results of such tests or the behavior of the embedded application may be displayed in an appropriate LabVIEW panel application or can be read into other tools such as test databases and analysis tools.

### **Real-world application: Test automation at EDAG Engineering + Design AG**

At Germany-based EDAG Engineering + Design AG seamless transition between the different phases of an embedded software development process has been a reality for several years.

EDAG has developed a concept and system for the automotive industry that implements automated software-in-the-loop tests, software integration tests and complete vehicle tests. The experts talk about a closed validation process offering software and test engineers the ability to work with a single database and tool chain that comes with a lot of synergy regarding the measurement concepts of each department.

Today, those test concepts include a huge overhead in terms of additional software that has to be developed to do diagnostics. Managing an optimal test execution, different areas of control software must be accessible, including data acquisitions such as program states, EEPROM parameters, control modes etc. The standard way to transport this information is over the industry-standard CAN bus. The software engineer must implement special routines

which are added to the existing control software. The result is an increased bus load and additional work in validation because these functions must be tested as well.

To overcome these barriers and to reduce or eliminate the extra work of code instrumentation, a combination of In-Circuit / On-Chip emulation and CAN diagnostics is the solution.

EDAG Engineering + Design AG has implemented this test automation for a National Instruments based development environment. With the integration of iSYSTEM In-Circuit / On-Chip emulators, using corresponding VIs (Virtual Instruments) to "remote control" those devices, additional diagnostic information is received not only over CAN, but also directly by (real-time) memory access within the electronic control unit.



*LabVIEW and winIDEA running together on a two-screen PC system, interfaced to EDAG's embedded target with the iSYSTEM iC3000 On-Chip Debugger.*

The EDAG-implemented system demonstrates the combined power of a graphical interface (LabVIEW) and non-intrusive, real-time hardware access (in-circuit / on-chip emulator) for embedded systems development.

For further information:

[erol.simsek@isystem.com](mailto:erol.simsek@isystem.com)

iSYSTEM AG

Carl-Zeiss-Str. 1

85247 Schwabhausen, Republic of Germany

Tel. +49-8138-6971-56

Fax +49-8138-6971-46

[jochen.zaepf@edag.de](mailto:jochen.zaepf@edag.de)

EDAG Engineering + Design AG

Reesbergstr. 1

36039 Fulda, Republic of Germany

Tel. +49 661-6000-457

Fax +49 661 6000-113325