

# Diagnostics Tool for Automated Locomotive Testing

By

Brandon Dineff  
Engineer, Measurement and Automation  
Data Science Automation  
USA

And

Gregory C. Cala, Ph.D.  
Vice President, Operations  
Data Science Automation  
USA

## Category

Manufacturing Functional Test

## Products Used

National Instruments LabVIEW 7.0  
NI VISA 3.0  
TestStand 2.0

## The Challenge

A leading manufacturer of locomotive engines required a tool to improve productivity during the testing of locomotive control system software modifications. The solution was required to fully emulate the custom Diagnostics Information Display (DID) onboard the locomotive. Furthermore, the solution was required to be usable by both field engineers and locomotive simulation engineers.

## The Solution

Data Science Automation (DSA) developed a system to emulate a DID and any power up communications between the DID firmware and the locomotive controller. This system was named the vDID for Virtual Diagnostics Information Display. This system fully emulated the DID communications and was a seamless solution to communicate with a large class of locomotive controllers. The vDID could be automated and the responses from the controller were recorded electronically and compared to expected values using TestStand.

## Abstract

In order to improve productivity during the testing of locomotive control system software modifications, a software emulator of a critical custom locomotive component – the Diagnostic Information Display (DID) was developed. Not only did this Virtual Diagnostic Information Display (vDID) eliminate the need for the custom DID hardware during testing of modifications to the locomotive’s control system software, the vDID could be automated to sequence multiple information requests from the controller and electronically record the response. This eliminated hundred of man hours of effort and eliminated all nuisance transcription errors.

### The Locomotive System

The locomotive is a complex system with an extraordinary number of parts. The locomotive control system has two key components: the controller itself and the Diagnostic Information Display (DID). The controller is the main nerve center that processes all communications and controls operations of the locomotive. The DID is used to annunciate alarms and allow the operator to enter parameter ID numbers in order to display the corresponding values of those parameters. Examples of some of the parameters include temperatures, currents, and speeds. The DID communicates with the controller over an RS232 serial interface. To allow the user the ability to automate sequences of multiple information requests from the controller the vDID was developed using LabVIEW. The vDID was a seamless replacement of the DID which is located in the cab of a locomotive. To allow for the use of the vDID in conjunction with the DID for developmental purposes, a custom cable was created. The cable placed the vDID inline with the controller and the DID. The cable schematic is shown in Figure 1.

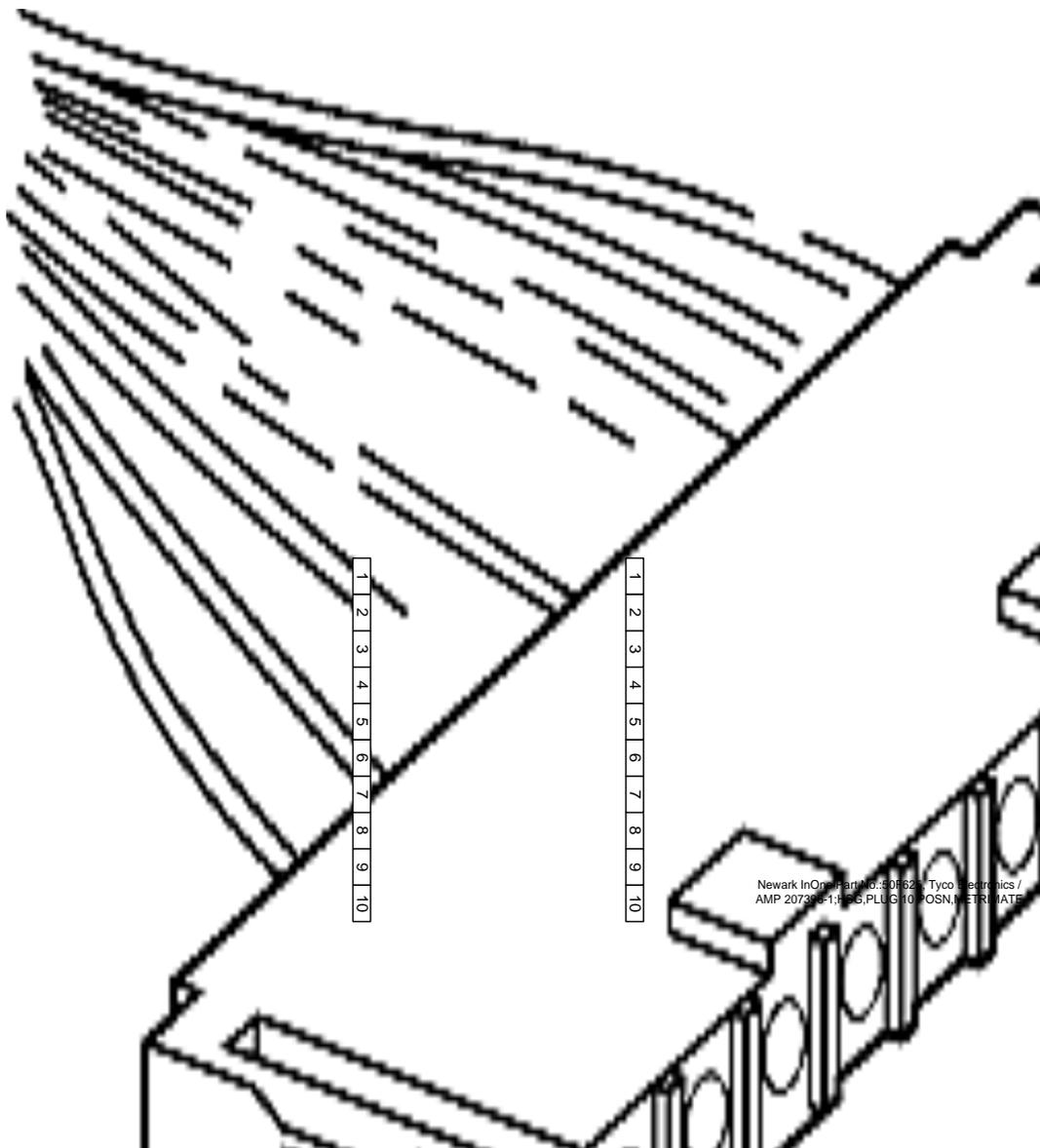


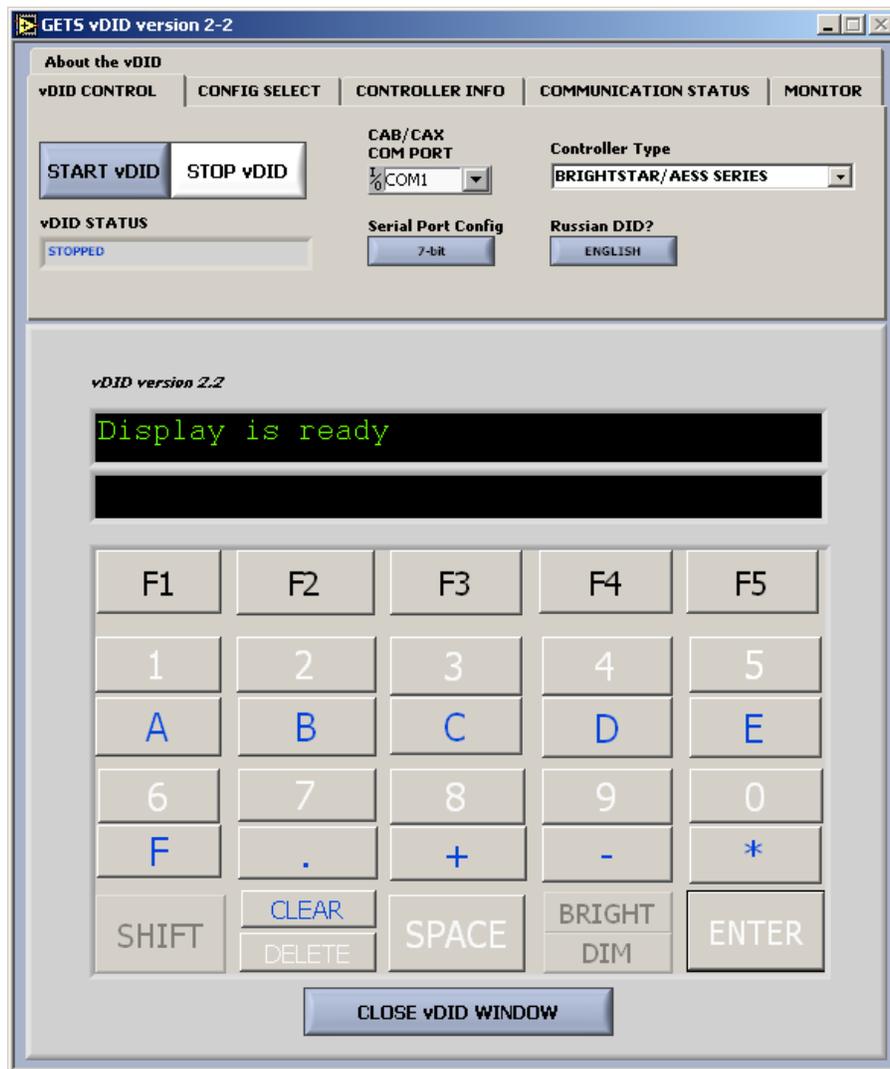
Figure 1.

This cable allowed sharing of the receive line for displaying controller responses to requests on both the vDID and the DID. The toggle switch routed the transmit line between either the DID and the controller or between the vDID and the controller.

### The vDID

The vDID was originally designed for use in the locomotive simulation lab. It was used to retrieve parameter values, electronically record the values and compare them to expected values based upon the known test conditions. Later the vDID was used as a tool by field engineers to electronically record the operating conditions of the locomotive during field testing. The vDID user interface is shown in Figure 2.

Figure 2.



The vDID has many capabilities including the support for a Russian controller with Cyrillic fonts. The vDID was designed to support multiple controller types and be able to save configurations for specific controllers. This allowed users to store configurations and easily connect for productive testing.

### Testing using the vDID

The vDID has uses in both the field and in the simulation laboratory. In both cases, the vDID handled the handshaking communications with the controller during power up. This power up sequence included the exchanging of information stored in an (EEPROM) Electrically Erasable Programmable Read Only Memory ordinarily held on the DID. Two types of commands are possible: one for reading from the EEPROM and the other for writing to the EEPROM. This handshaking started with a request by the controller for configuration information from the DID. An example command would contain the memory location of the data (ex. 00x is configuration data), what command (11 for an EEPROM read, 12 for an EEPROM write), and a checksum and (CRC) Cyclical Redundancy Check. These messages were parsed to separate each component to build a reply string. When using the DID, the reply was generated automatically by the processor embedded in the DID but when using the vDID a response string had to be programmatically constructed. Two types of replies to the power up request for configuration information from the controller are possible. The first type applied to a “virgin” configuration. This configuration intentionally replied with invalid information to force the controller to see the vDID as a new DID that hadn’t been previously configured. This forced the controller to send DID configuration information to the vDID in the form of EEPROM writes. The vDID then wrote that information to a configuration file. That configuration file could be used later when using this controller. The user could then set an option to use a specified configuration file. Instead of the vDID replying with virgin information, the vDID would now construct an appropriate reply string based on the configuration file contents. An example of a configuration file is shown in Figure 3.

