Delphi Packard Summer Intern 2000
Filtered Header Connector Project

Internship Overview:
Justin DeGuire, Computer Information Systems major, developed a system for Delphi Packard Electric as an internship project. This system automates the testing of components for Power Control Modules. This project and its results are presented on subsequent pages. Justin's faculty advisor on the project is Thomas Bodnovich, Associate Professor of Computer Science and Information Systems.

Department:
187 EMC Technology / Data Communications

Supervisor:
Kin P. Moy

What is EMC Technology?
Electro Magnetic Compatility (EMC) is about the ability of different items of electrical equipment to work together without suffering the effects of interference. Equipment should also operate without interfering with broadcast and communications signals and be immune to normal levels of such signals. EMC implies that a system will not generate unacceptable levels of conducted or radiated signals which could cause interference to other well designed products. Systems should also be designed in such a way that normal ambient levels of electrical noise will not cause degradation of performance they must have an adequate level of immunity.

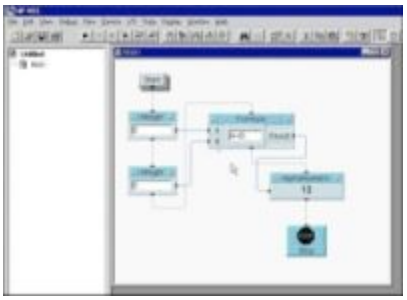
Summer Project Objectives:

Why?
187 needed custom automated test software to increase test efficiency and accuracy from present manual operation. Prior to this software everything was done by hand. The instrument was controlled manually and all data collection was typed manually into an Excel spreadsheet, which left room for error considering the amount of data that needed to be recorded.

How?
The software was written using a visual programming test instrument language called HP VEE 5.0. HP VEE is also used in many existing measurement systems within the department, so it was the language of choice.

Programming Language:
HP VEE stands for Hewlett-Packard Visual Engineering Environment.
HP VEE has two parts; development and run-time environments.
Currently HP VEE is the development standard for the department.
For more infomration about HPVEE go to http://www.agilent.com.
Note: The product is now called Agilent VEE since the instrument division of Hewlett-Packard broke off from the computer hardware division of Hewlett-Packard and changed it's name to Agilent.

HP VEE Development Enviroment Screen Shot:



Here is a snap shot of the development environment of HP VEE. As you can see, writing the actual code is smiliar to creating a flow chart. Writing code in this manner is very different compared to other languages, but this writing style makes debugging easy. Writing code in this manner also gives you the ability to write usable code quickly.

hpvee.jpg [42k]

Filtered Header Measurement System:
Department EMC needed some custom software to efficiently and accurately measure and document the insertion loss measurement of filtered header connectors, which were being used for Power Control Modules (PCM). Originally, measurements of filtered header connectors were all done by hand.

Filtered Header Measurement software will:

- Maintain a record of all measurements
- Keeps track of the current pin that is being measured
- Performs the test with instrumentation control

Result:

- Reduce test time by 75%
- Enhance measurement/data accuracy
- Increase through-put of the lab

Powertrain Control Module:

Harness side of the Powertrain Control Module.

header_harness.jpg [242k]

PCB side of the Powertrain Control Module.

header_pcb.jpg [252k]

What is a Powertrain Control Module?
This mini-computer runs the engine management system and the automatic-transmission electronic controls. The PCM receives input signals from engine sensors, processes the data, then sends output

signals to the ignition system, the fuel injectors and other devices that control engine operation.

On the PCB side of the Powertrain Control Module, there is an intergrated circuit board connected to all 192 pins of the filtered header connector. Manual testing requires de-soldering of this intergrated circuit to measure the insertion loss of the filtered header connector.

Network Analyzer:

Hewlett-Packard 8753E Network Analyzer.

hp_8753e_na.jpg [299k]

I used a Hewlett-Packard 8753E Network Analyzer to measure the insertion loss of filtered header connector. For more information about the Hewlett-Packard 8753E Network Analyzer visit http://www.tm.agilent.com.

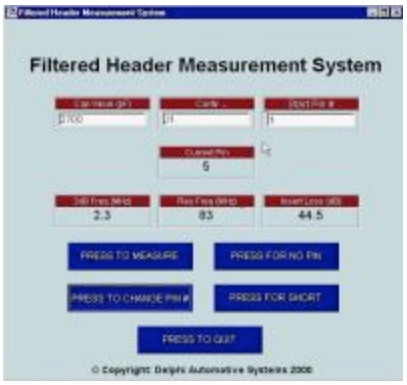
Test Procedure:

Test procedure with the Hewlett-Packard 8753E Network Analyzer and the Filtered Header Connector.

test_setup.jpg [275k]

The Network Analyzer has two identical cables that connect to each side of a pin on the filtered header connector, which then measures the insertion loss of that pin. I used clamps to secure the cables to the pin by clamping it to the sides of the filtered header connector. By using clamps and some custom made metal brackets, the pin is grounded to the side of the filtered header connector, which is necessary to get vaild results. Once the pin is secure, you simply execute my Filtered Header Measurement System, which measures Cap Value (pF), 3dB Frequency (MHz), Resonant Frequency (MHz), and Insertion Loss (dB) to a data file. Then you have to loosen up the clamps and move the connectors on each end of the cables to a new pin for measuring. Simply repeat this process on each pin to measure all the pins on the filtered header connector.

Filtered Header Measurement Software:



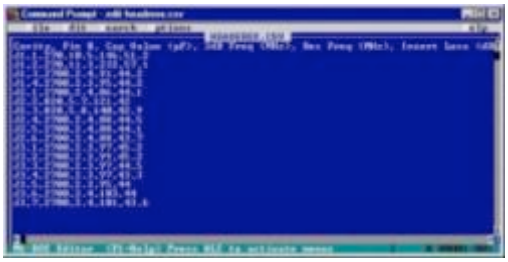
Filtered Header Measurement Software snap shot.

fhcms_snap.jpg [42k]

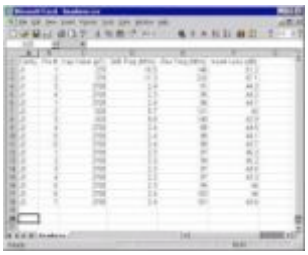
Filtered Header Measurement Software has three data inputs and five command buttons. These input are the Cap Value (pF) of the pin, the cavity number of the filtered header connector where the pin is located, and pin number within the cavity that is initially measured. The total number of pins on each PowerTrain Control Module differs between models, but there were 192 pins (3 cavities, each containing 64 pins) on the model I was measuring. Imagine manually setting up the Network Analyzer and typing the data for each pin. It took approximately ten command buttons on the Network Analyzer to set it up for each pin. That's 1,920 (10 X 192 = 1,920) button strokes to measure one PowerTrain Control Module! Additionally you had to manually log six values for each pin. That's 1,152 (6 X 192 = 1,152) values you had to type for each pin. Doing anything that repetitive will cause some error in logging the data.

With my software, all you have to do is press the MEASURE button to measure each pin, and the software logs those six values automatically. Occasionally there would not be a pin to measure, in which case you would then hit the NO PIN button. Sometimes there would be a pin but it was only an electrical short, which that pin's jobs was to only ground out any RF interference and contained no measurement data, so you would hit the SHORT button instead. Also if you made a mistake in measuring a certain pin, their was a button there, that would change what pin was currently being measured, called CHANGE PIN. All you had to do was type in the pin number, for that incorrectly measured pin, in the "pin input field" and hit the CHANGE PIN button. Finally, there was a QUIT button, which simply quit the program.

Data Acquisition:



DOS edit view of how my program logs the data.
dos_edit.jpg [51k]



How CSV looks in a typical spreadsheet program.

excel_data.jpg [67k]

How does the software log the data?

Data is saved using CSV (comma separated variable) format so that it can be read with many spreadsheet applications. All data measurements are written as they are taken from the analyzer. All data is appended for multiple tests.

Four parameters are logged in the data file based on customer specification:

- Cap Value (pF)
- 3dB Frequency (MHz)
- Resonant Frequency (MHz)
- Insertion Loss (dB)

DELPHI
Automotive Systems

One University Plaza
Youngstown, OH 44555
Phone toll-free 877-GO-TO-YSU (877-468-6978)