An Automated Test System for RF Transceivers and Receivers

Speaker: Grace Zou
Product Design Group - RF Test Development Unit
Hughes Network Systems
11717 Exploration Lane
Germantown, MD 20876
Phone: (301) 428-5500
Fax: (301) 428-2835
Email: gzou@hns.com

Authors: Grace Zou, Ade Adetuyibi

Product Design Group
Hughes Network Systems
11717 Exploration Lane
Germantown, MD 20876

Abstract: This paper describes an automated RF (Radio Frequency) test system developed by Product Design Group - RF Test Development Unit at Hughes Network Systems. The challenge of this project was to devise a test system that could call multiple test modules written in different computer languages to test GSM basestation transceivers. Moreover, it should be easily scriptable to accommodate changes to the test environment, and it should be flexible enough that a novice programmer or someone with no programming experience could edit it. TestStand, a packaged test executive, met these requirements and dramatically reduced our test system development time.

1. Introduction

Reduced cost and time-to-market are the driving forces behind the virtual instrumentation and test automation revolution. One of the key players in this field is National Instruments. National Instruments developed LabWindows/CVI and LabVIEW (Laboratory Virtual Instrument Engineering Workbench) for instrument control and for building data acquisition and virtual instrumentation systems. The company also produces TestStand - a packaged test executive that can be tailored to manage and control the execution and interaction of multiple test modules that are written in different computer languages including modules written in LabVIEW and LabWindows/CVI.

The Product Design Group - RF Test Development Unit at Hughes Network Systems harnessed the power of TestStand, to test multiple GSM base station transceivers subjected to thermal cycling, via the controlled and managed execution of a LabVIEW application that was developed in-house, a third party DOS-based executable file, and a third party application that was written in LabWindows/CVI. The TestStand approach to 'test module' control was chosen over DOS batch file processing and Perl scripting methods because of its ability to handle the language barriers that existed when instructions were passed between the component test modules.

TestStand also excelled in its ability to create flexible test reports with relative ease. Moreover, the GSM transceiver test application created with TestStand, can be modified by a novice programmer or someone who is with a little or no programming experience, thereby meeting the projects design objectives.

2. Why GSM base station transceivers are subjected to thermal cycling

A GSM (Global System for Mobile Communications) base station is comprised of a set of base transceiver stations (BTS's) and a base station controller (BSC), as shown in Figure 1 below, and it is often installed in an open air environment. Consequently, the base station is often subjected to wide temperature fluctuations during normal operation depending on the part of the world where it is situated. Since the malfunction of any one of the base station components will affect the operation of that particular cell site, manufacturers must design systems that work within design specifications even in the worst environmental conditions that the base station could be subjected to. To this end, base station manufacturers perform quality and reliability (Q&R) tests under changing test environments as the product evolves from the prototype phase to its production phase.

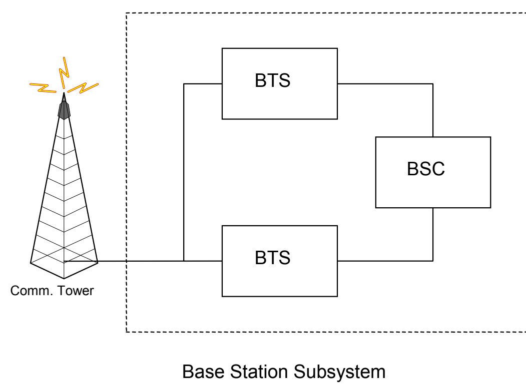

Figure 1 Basestation subsystem.

One of the Q&R tests that manufacturers perform is thermal cycling, which is the process of subjecting a unit under test (UUT) to harsher environmental conditions than it would normally be subjected to in the real world. This process is an accelerated life test to determine the fundamental limits of the UUT. By testing several prototype units, a statistical behavioral model can be developed based on measurement data that can be used to improve the design of subsequent units.

3. The initial test setup

A GSM BTS consists of a baseband section and a RF section. The RF section (transceiver) is further sub-divided into a transmitter section and a receiver section. Though this paper is focused on the transmitter and receiver sections of the BTS, we shall briefly mention the significance of the baseband section in the operation of the BTS. The baseband section of the BTS consists of digital circuitry that controls the operation of the RF section. It is the ‘brain’ of the BTS and is also the means by which the BTS interfaces with baseband peripherals such as the BSC.

In the early phases of the GSM transceiver’s system design cycle, it was more cost effective to emulate the baseband section with a DSP processor suited to digital communications applications, rather than wait for the actual BTS baseband section to be built. The DSP 16000 Core made by Agere Systems (formerly Lucent Technologies) was the DSP of choice for prototype testing. A TargetView communication system (TCS) board (from Agere Systems) and a Personal Computer (PC) running the LuxWorks code generator (also from Agere Systems) programmed the DSP 16000 Core to receive code generated by LuxWorks at runtime and subsequently send configuration information to the GSM transceiver, before each thermal cycle. Figure 2 illustrates the previous test system.

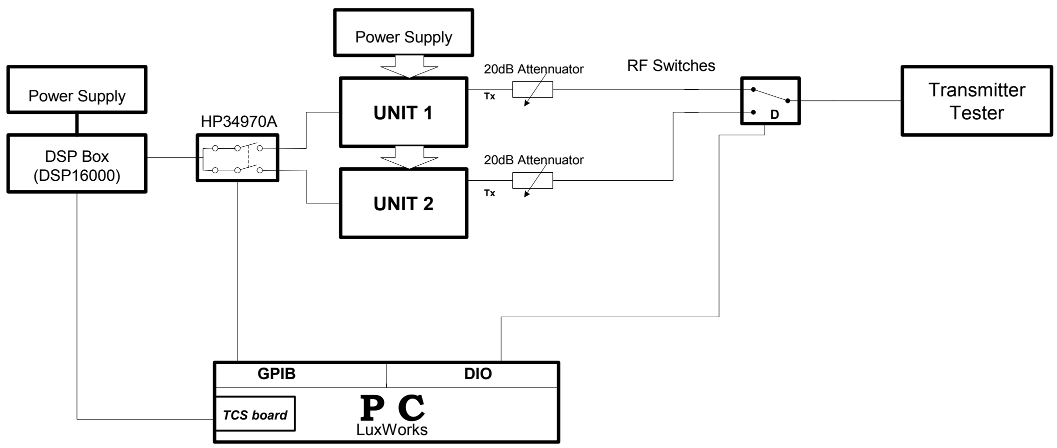

Figure 2 Previous test system capable of testing transmitters only.

This arrangement made it possible to switch the RF connections to the transmitter tester from Unit 1 (transmitter 1) to Unit 2 (transmitter 2) automatically without operator intervention. However, each time this was done, LuxWorks had to reinitialize the new UUT, thereby adding a substantial overhead to the test time. Moreover, it was impossible to test the receiver section of the transceiver with this arrangement due to Luxworks receiver limitation..

The need for an improved test methodology that could test both the transmitter and receiver sections of the transceiver drove the development of a test system that exchanged the LUxWORKS – TargetView combination for a DOS based terminal emulation software package that used a scripting technique to automate the testing process. Also, by this time, the first iteration of GSM baseband boards had been built, so a switch was made from the DSP baseband simulator to a GSM baseband card.

4. The latest test process

We now describe the latest test system that is both faster and supports tandem testing of transmitters and receivers. It employs the aforementioned DOS based terminal emulator, LabWindows/CVI, LabVIEW, and TestStand. All these applications run on the PC illustrated in Figure 3 below.

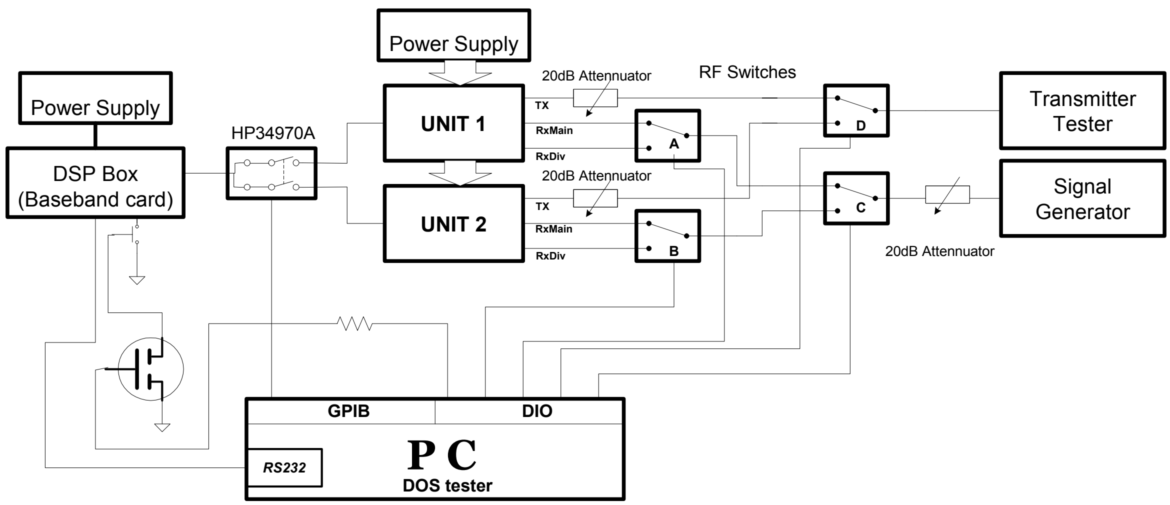

Figure 3 Latest test system capable of testing both transmitters and receivers.

This new arrangement made it possible to switch the RF connections to the transmitter tester and signal generator from Unit 1 (transceiver 1) to Unit 2 (transceiver 2) automatically without operator intervention. The transmitter tester tests the transmitter portion of each transceiver while the signal generator feeds test signals to the receiver section of each transceiver. Compared to the previous test system, there was a significant reduction in the test time.

TestStand managed the flow of control from the software modules written in CVI, LabVIEW, and the DOS executable code. To do this, TestStand passes parameters to and accepts parametric data from any module that it calls. This attribute makes TestStand a very powerful test executive.

The DOS based executable file is a menu driven terminal emulator that can be easily configured at run-time via an ASCII coded scripting language similar to Aspect™. Since Microsoft Windows cannot read a DOS communication port, a separate DOS Command Prompt window is needed to start the DOS Tester software. LabVIEW is used to open and close the DOS Tester software. The CVI code sends scripted ASCII commands to the DOS Tester application via a serial communication port. The DOS Tester software interprets the ASCII coded string and either executes the command or sends out the appropriate proprietary command string to the UUT.

The main code stream for the Q&R test was written in CVI. However, the receiver and transmitter CVI code streams were different. In addition, the receiver and transmitter used different versions of the DOS code, which were incompatible with each other. The initial CVI code that was written did not accept any runtime arguments and therefore it was not immediately usable with TestStand. So, the CVI code was converted to a Dynamic Link Library (DLL) so that TestStand could pass the appropriate parametric data to CVI to test the units.

To begin the testing process, the parametric specifications for the test are input to TestStand from the user interface, from which the user can select the specific test to perform. The user can enter the following parameters: frequency band (800 MHz, 900 MHz, 1800 MHz, or 1900 MHz), Prototypes of the transceiver, test mode(functional test, baseline test, and Thermal cycling test), temperature range, voltage range, input device data such as the serial number of the units to be tested, and the operator's name. When the CVI executable starts, these parameters are passed to each UUT one after the other, which was not very efficient, so the possibility of combining them to improve efficiency was explored. This could have been accomplished with CVI, but it would have taken several weeks to write and debug, so it was decided that the task could be accomplished using LabVIEW in a much shorter period of time, and it would also be much easier to implement.

In addition to the user interface, LabVIEW was used to reset the baseband card and perform RF switching. After each test cycle, the baseband card was reset and the RF path was switched from one transceiver to the other. A matrix of RF switches were designed to accomplish this task. The RF switches and the baseband card reset wire were wired to a Digital I/O (DIO) card controlled by software that resided in the PC. A FET was also used to transform the voltage from 5.0V to 3.3V for the baseband card.

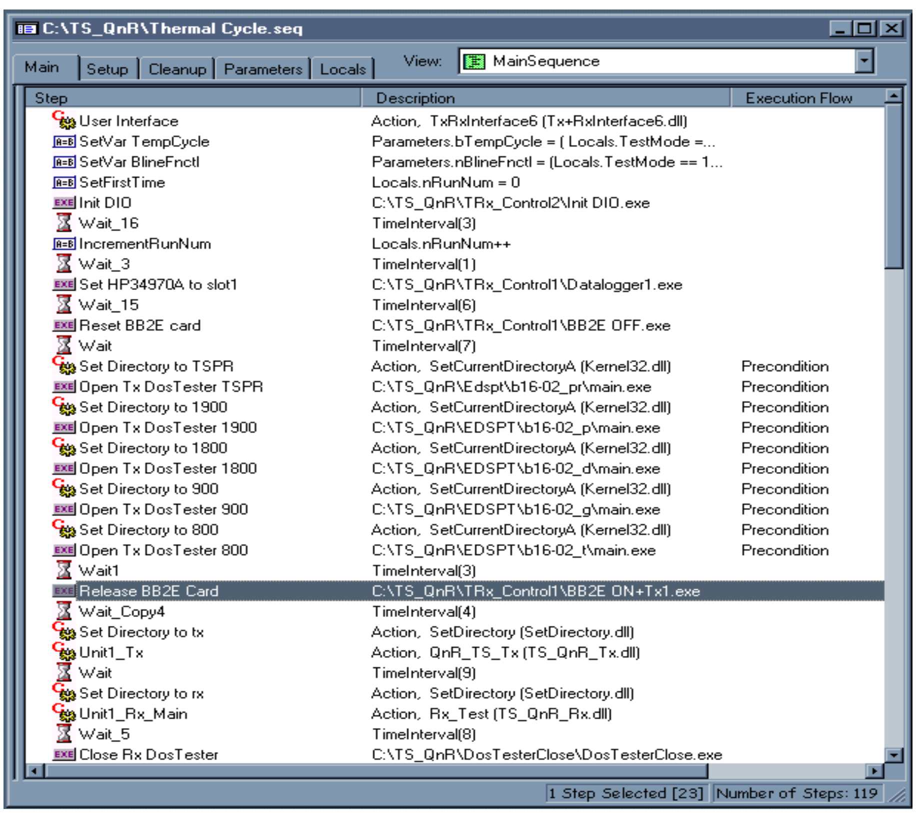

Figure 4 Sample view of TestStand sequence

LabVIEW was also used to open and close the DOS Tester application software. Controlling the DOS Tester application was complicated because there was no source code for the application. To open the DOS Tester application, a DOS Command Prompt Window must be opened and then the executable must be called. This was easily handled in LabVIEW by using a call to the system. Once the DOS Tester application was open, the CVI module was called from TestStand. The CVI module, which was now a DLL object, would then send the appropriate commands to the DOS Tester software to configure the baseband card, and in turn the baseband card would send the appropriate command to the UUT. After the UUT was tested, the next unit would be switched on for test automatically. This procedure included closing the DOS Tester application, which turned out to be a difficult task, as there were no commands within the DOS Tester software to exit out of the application software. So an elaborate method of calling upon the Windows System Manager, which manages various open windows, was used. Within the Windows System Manager a call was made to close the DOS Tester window, just like pressing the window close icon on an open window with the mouse. LabVIEW made system calls much easier with very little system knowledge.

All of the LabVIEW modules, called Virtual Instruments or VI's for short, were built as DLL files to allow easy passing of data from LabVIEW to TestStand. By building all the CVI and LabVIEW modules as DLL files, it became a much easier task to thread all the modules together within TestStand. In this way multiple units could be tested without any software constraints. The current hardware implementation tests two transceiver units in consecutive order.

5. Summary

By using TestStand, the software development time was reduced from a couple of months to just under a few weeks. By combining the transmitter and receiver testing into one test system it reduced the cost of the overall test system from that of the previous hardware architecture. This was a significant saving in equipment costs. By re-using the CVI code from the previous test system we were able to rely on proven test modules without significantly impacting the overall project. By using LabVIEW Active-X components we were able to effectively and effortlessly provide RF switching capability to the test system, and manipulate DOS Tester software so that it could be reconfigured at will. TestStand gave us the flexibility to program the operator user interface in a language we were comfortable with. In addition TestStand allowed us to thread our existing code modules and be able to test units in a shorter time than could have been possible if we had written the test system entirely in C or LabVIEW.

6. Acknowledgements

The author is grateful to God and to the following people:
Merdod Badie (leader of the RF Development unit) for guided the technical direction of this project. His decision to use TestStand paid off with minimal test system development time.

Mark Wickham who proposed the development of the DOS executable for transmitter and receiver testing.
Joyal Jose for making the CVI code more efficient.
Keith Frey who wrote the code that closes the DOS application window.

References:

[1] http://www.ni.com
[2] http://www.gsmworld.com
[3] http://www.agere.com/dsp16000/pdf/TN99002.pdf