Agilent Technologies

Creating Emulation Tests in a Graphical Environment

Product Note

Introduction

Modern telecommunications are made possible by complex signaling protocols. Designing, testing, deploying, and maintaining these protocols are challenging and time-consuming tasks. To identify and solve problems related to signaling protocols, you will need powerful yet easy-to-use test equipment. Some signaling protocol problems can be identified and resolved using the passive monitoring capability of advanced protocol analyzers. Other, more difficult problems require active transmission of signaling protocols to the network under test. This is accomplished using a protocol analyzer with signaling emulation capability.

Historically, protocol emulation has been a complicated process assigned to only the most senior test engineers. Signaling protocol analyzers have had complex user interfaces that were difficult to learn, difficult to program, and very expensive to operate. Often this has forced test engineers to focus their attention on the test tool rather than the problem to be solved.

This paper introduces a new concept of graphical emulation that dramatically simplifies test programming in the Agilent Signaling Advisor. Using Agilent's new EmuLite software, test engineers can shift their focus away from the complexity of programming to applying their signaling protocol knowledge to resolving the problem at hand.

What do Signaling Engineers Need?

Manufacturers who design and test signaling protocols as well as service providers who deploy and maintain them, perform a variety of tasks that may require protocol emulation:

Engineering team	Needs emulation capability to...
R&D	Develop and test designs
Integration/test	Test and verify complete systems
Service creation	Test new services before they are deployed in a live network
Acceptance test	Verify that new links or interconnections conform to requirements
Maintenance	Identify and solve network problems that monitoring alone can't solve

To operate, telecommunication networks rely on signaling messages being sent between different network elements. These signaling messages contain information relating to a call in progress or a service being used. Therefore, the critical skill needed by all engineering groups is an intimate understanding of signaling protocols and messaging sequences.

Because of the complexity of the signaling itself, engineers want tools that simplify and shorten the test process. Our research shows that they would prefer not to spend excessive time writing programs, and that they would like to be able to create messages easily, perhaps using messages captured from their networks. They would also like the protocol analyzer to notify them which tests have passed and which have failed.

A list of the most common requests that we have received from signaling test engineers includes the following:

- Reduce the amount of programming required.
- Allow us to define tests using message sequences.
- Allow us to do this graphically.
- Provide an intelligent message builder.
- Allow us to use messages captured from the network.
- Provide integrated analysis capabilities that give insight into the cause of problems.
- Reduce the amount of time that testing takes.

Another factor affecting the lives of signaling engineers is the increasing pressure within telecommunication organizations to deliver high-quality new products and services to the market quickly and faultlessly. For example, wireless network operators have made huge investments in licenses for UMTS. It's critical that they deploy their networks and services as quickly as possible so that they can begin to recoup their investments. Signaling test will play a vital role in this activity, and new approaches and test tools are urgently needed.

EmuLite: A Simpler Approach to Protocol Emulation

With the introduction of Agilent's EmuLite software for the Signaling Advisor, engineers can take the first step to simplifying the process of signaling protocol emulation. EmuLite provides a graphical environment that allows engineers to dispense with all programming languages and describe tests as a series of message sequences. Using the software's graphical editor, message sequences are simply drawn on the screen and executed—no compilers or linkers are needed. The message sequences can be created in minutes, and EmuLite's intelligent field-based message editor guarantees that all messages are valid.

In addition, comprehensive sets of sample messages are provided for each signaling protocol, so that engineers need modify only the fields crucial to their test. Messages previously captured from the monitor can be copied into the EmuLite software for future use. For example, all messages associated with a single call can be copied and used to generate test messages and sequences automatically. Simple test execution combined with integrated monitoring capabilities make it easy to identify failed tests and the reasons for the failure.

EmuLite simplifies a complex, time-consuming, expensive activity, thus enabling signaling protocol engineers to focus their skills on improving the signaling performance of networks.

Emulation or Simulation

Before examining in detail how the EmuLite software works, it is important to define the term emulation as it applies to the Signaling Advisor, and to differentiate emulation from simulation, which is a term sometimes used interchangeably.

Some engineers expect a protocol analyzer to perform all the functions of a switch, while others require only the ability to send messages and receive messages to and from a network element (a switch or other device under test).

In this paper, we define simulation as the more limited ability of the protocol analyzer to execute the test sequences that transmit signaling messages, receive responses, and take appropriate actions (Figure 1). Several protocol analyzers on the market today provide simulation capability. However, only the Signaling Advisor with EmuLite software does not require that test engineers also be expert programmers.

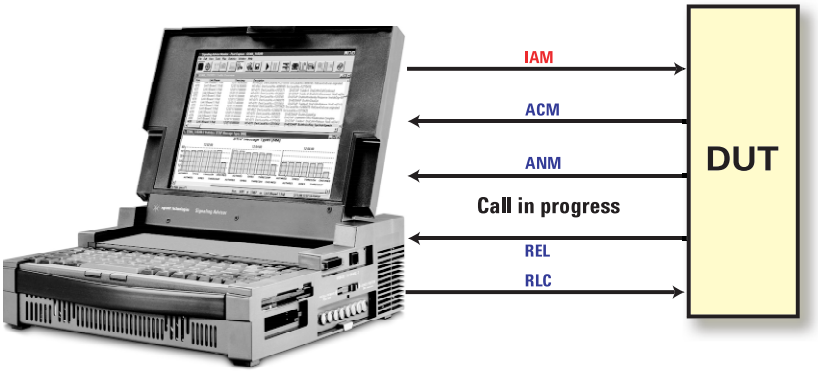

Figure 1. Simulation of an ISUP call

Emulation is defined in this paper as the protocol analyzer's enhanced ability to provide either full or partial functionality of the network element's protocol stack. A simple example of this definition is the ability of the analyzer to automatically emulate the level 2 functions of SS7 MTP2. With this degree of capability, the analyzer can fulfill the point-to-point communication and retransmission needs of a signaling link, thus freeing the test engineers to focus on simulating the higher layer protocols—for example, MTP3, ISUP, SCCP, and TCAP. The Signaling Advisor provides emulation at various layers and generally assumes that all the lower layers have been tested and are functioning correctly. Simulation occurs at the layers above those being emulated (Figure 2).

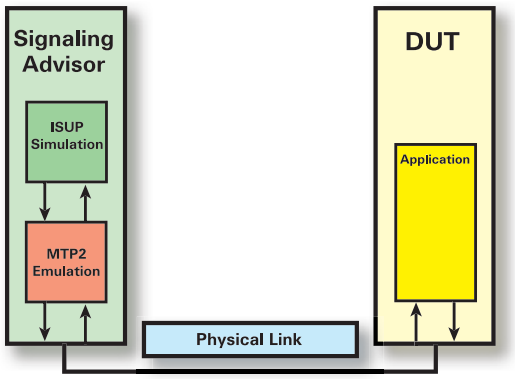

Figure 2. Automatic MTP layer 2

Benefits of a Graphical Environment

To understand how a graphical environment improves testing, we will look at three different engineering groups and the types of problems that they are trying to solve today.

- A GSM operator's engineering group must perform specified tests on newly installed GPRS equipment in order to get the network operational as quickly as possible. This scenario provides an opportunity to look in detail at the benefits of using the EmuLite software's graphical environment.
- A traditional fixed-line operator's SS7 test engineers must perform interoperability testing to verify that a new interconnect operator's ISUP protocol conforms to their network requirements.
- An advanced-service creation group need an effective way to test new services being deployed on their SCP. The ability to quickly create and send messages to test the service is crucial.

GPRS Network Engineering

General Packet Radio Service (GPRS) networks are being rolled out aggressively as a first step toward realization of 3G networks. Speed of deployment coupled with confidence in the functionality of the signaling elements is of paramount importance. As part of the deployment process, tests are required on all of the GPRS network interfaces. In this example, an engineering group must test the Gr link between the Serving GPRS Support Node (SGSN) and the Home Location Register (HLR). The EmuLite software offers many tools to make this job much easier.

Transaction scenarios defined in the GPRS specifications describe how the HLR should react to certain message-transaction sequences. In order to test how the HLR will react to those messages, a test instrument acting as the SGSN must send messages to the HLR and analyze the responses.

Let's examine one of the tests in detail. The test engineer must confirm that the GPRS Location Update procedure is performing according to specification. Figure 3 shows the portion of the signaling messages related to this procedure (ETSI specification 09.02, Figure 19.1.1/14: Interface and services for GPRS location updating).

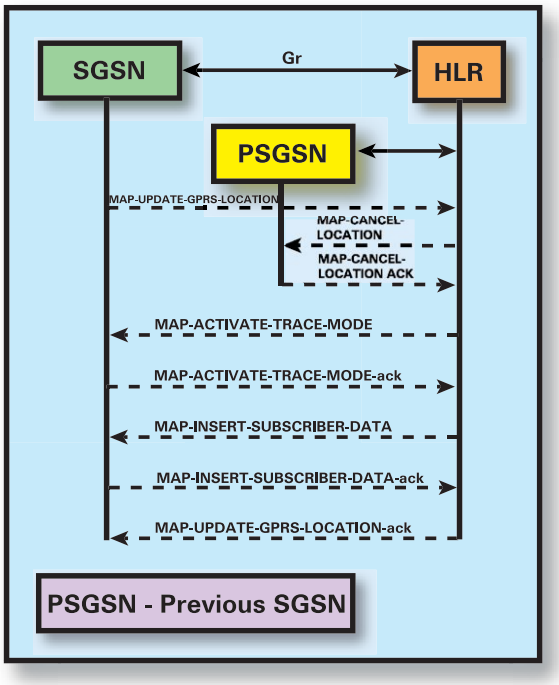

Figure 3. GPRS Location Update Procedure

To see how the graphical environment of the EmuLite software simplifies the development of this test, we'll first take a look at a section of the Specification and Description Language (SDL) code that is typically required. The set of instructions shown in Figure 4 (taken from the HP 37900D protocol analyzer) provides the functionality of the first four messages of the GPRS Location Update procedure previously illustrated. Note that the SDL in this case is entered using a text editor, a manual process that is prone to error. Before this code can be executed, it must be compiled, reedited, and recompiled until all errors are removed. A quick look at SDL suggests how difficult programming environments can be; other protocol analyzers use languages such as C or FORTH, which are even more complex.

```
PROCESS GPRS_Location_Update;

SET ie_0;

STATE S_1;
    INPUT ie_0;
        OUTPUT '     >>> GPRS_Location_Update; <<<';
        OUTPUT '                              ';
        OUTPUT '_____________________________________';
        OUTPUT Update_GPRS_Location;;
        OUTPUT '> Update_GPRS_Location;';
        SET_TIMER  6000;
        NEXTSTATE S_2;
    INPUT UNDEFINED;
        OUTPUT 'Undefined input while waiting for internal flag';
        OUTPUT 'Test Stopped';
        STOP;

STATE S_2;
    INPUT Cancel_Location;;
        OUTPUT '< Cancel_Location;';
        RESET_TIMER;
        OUTPUT Cancel_Location_ack;;
        OUTPUT '> Cancel_Location_ack;';
        SET_TIMER  6000;
        NEXTSTATE S_3;
    INPUT TIMEOUT;
        OUTPUT 'Expected Cancel_Location; NOT received - TIMEOUT';
        OUTPUT 'Test Stopped';
        STOP;
    INPUT UNDEFINED;
        OUTPUT 'Expected Cancel_Location; NOT received';
        OUTPUT 'Test Stopped';
        STOP;

STATE S_3;
    INPUT Activate_Trace_Mode;;
        OUTPUT '< Activate_Trace_Mode;';
        RESET_TIMER;
        OUTPUT 'Test Successfully Terminated';
        STOP;

    INPUT TIMEOUT;
        OUTPUT 'Expected Activate_Trace_Mode; NOT received- TIMEOUT';
        OUTPUT 'Test Stopped';
        STOP;
    INPUT UNDEFINED;
        OUTPUT 'Expected Activate_Trace_Mode; NOT received';
        OUTPUT 'Test Stopped';
        STOP;
    ENDPROCESS;
```

Figure 4. GPRS Location Update Procedure written in SDL code

Now let's look at the graphical representation of this entire sequence shown in Figure 5. With no syntax errors and no compilation or linking needed, the simplicity of the graphical approach seems clear.

Creating a graphical test sequence to perform the GPRS Location Update is straightforward with the Signaling Advisor's EmuLite software. In this example, the Signaling Advisor assumes the role of the SGSN. The test is created in three simple steps:

- First, a new sequence called GPRS_Location_Update is defined.
- Second, the message sequence is created using the available commands.
- Third, the messages are edited to match the exact requirements of the test.

Using the sequence editor, transmit and receive messages are added to the message sequence by dragging and dropping them from a list into the message sequence. We define the logic of the sequence by using the available commands to exactly match the needs of the GPRS Location Update procedure. In the complete test sequence shown in Figure 5, when the first message MAP_UPDATE_GPRS_LOCATION is transmitted, either the optional MAP_ACTIVATE_TRACE or the MAP_INSERT_SUBSCRIBER_DATA message will be received. The EmuLite software provides exception handling capabilities for receipt of unexpected messages. If the optional AP_ACTIVATE_TRACE message is received, the sequence responds with MAP_ACTIVATE_TRACE_ACK and proceeds to wait for the MAP_INSERT_SUBSCRIBER_DATA message. In this instance, any out-of-sequence messages will cause the test to FAIL.

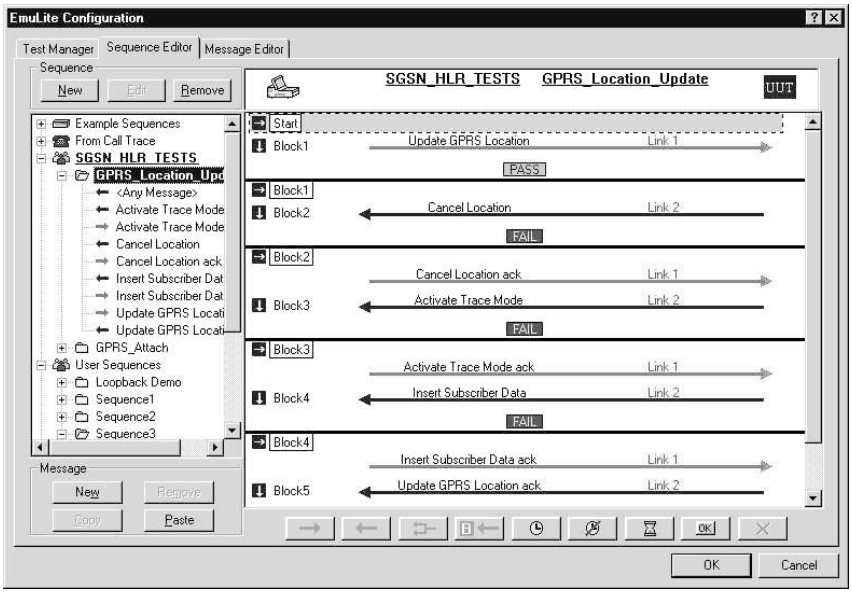

Figure 5. GPRS Location Update Procedure presented in EmuLite's Sequence Editor

It's easy to see how a message sequence representing the GPRS Location Update test procedure can be created in a few minutes. Sequences, as well as individual messages, can be saved on disk and used as templates for future tests. The sequence provides the logic for the test and individual messages provide the commands to the switch (in this case the HLR).

The EmuLite software has message examples for all common protocols, which can be modified using the graphical editor to match the needs of a specific test. In Figure 6, the message editor provides a complete decode of the Update GPRS Location message, which can be modified using the buttons at the bottom of the screen or by highlighting the required field and making appropriate edits.

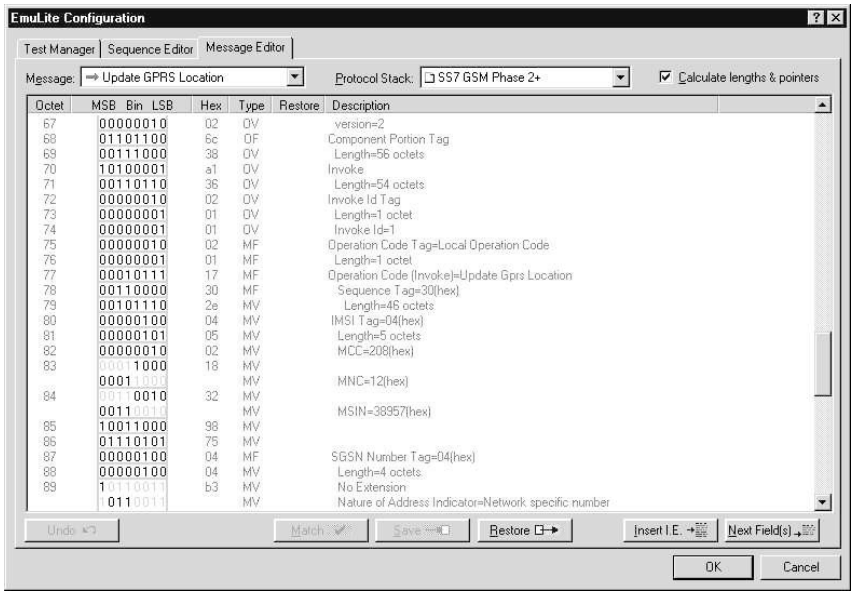

Figure 6. EmuLite Message Editor view of 'Update GPRS Location' message

If test engineers have access to live traffic, they can monitor the signaling transactions and extract messages in sequence that the EmuLite software will use to automatically create a test sequence. (This capability is explained in more detail later.) When the test is complete, it can be saved in a library. Tests can be grouped and named for easy recognition, such as the SGSN_HLR_TESTS illustrated in Figure 7.

Executing tests is simple in a graphical environment. The required tests are selected from the Available Sequences list and moved to the Selected Sequences list. It is possible to change the order in which the selected tests are executed by using the arrows at the foot of the dialog box in figure 7.

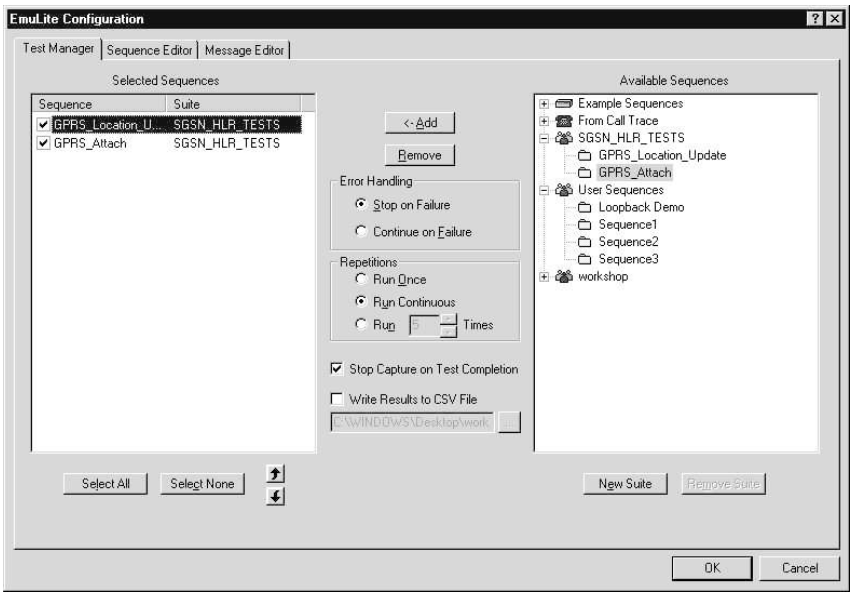

Figure 7. EmuLite Test Manager

An important part of all testing is the analysis of the results. A poor environment for analysis can negate any benefits gained from good test development. With the graphical environment provided by the Signaling Advisor and EmuLite software, executing test sequences and understanding the results is easy and accurate. Figure 8 shows the execution analysis screen, which starts automatically whenever test sequences are run. The screen is divided into three areas. A status view area shows the execution status and result for each sequence, indicating whether that sequence passed or failed the test. Selecting a test from the sequence area displays the message sequence that occurred along with timestamps and the reason for any failure. All messages can be decoded completely for detailed analysis in the decode area.

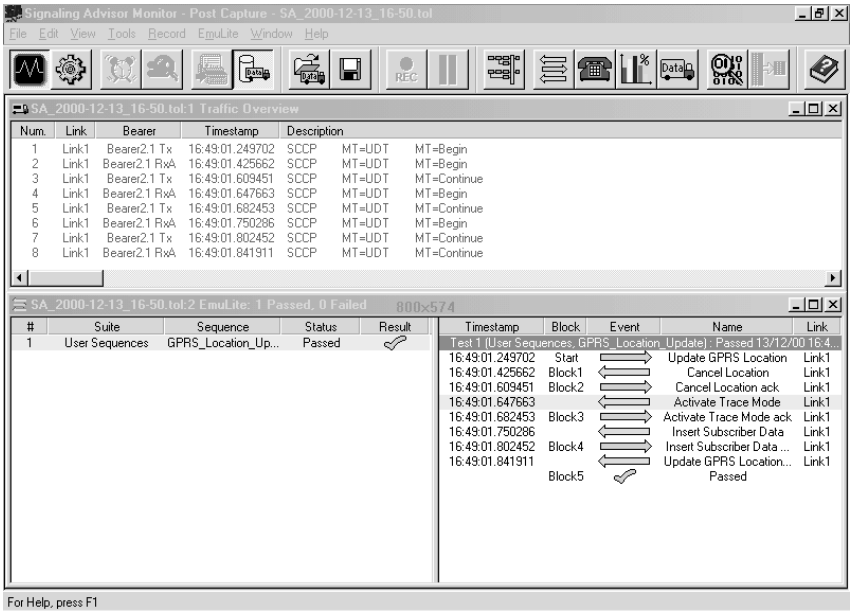

Figure 8. Execution of an EmuLite test sequence

ISUP Interconnecting Testing

Further capabilities of the EmuLite software are illustrated in the following example. Figures 9 and 10 show an originating and a terminating ISUP call. In this case, the sequences are used by a network operator's acceptance test engineers, who need to verify that the interconnecting links from a new operator meet their ISUP requirements and can provide service for both the originating and terminating calls.

The two sequences can be created quickly using the software tools described previously, or they can be created automatically using the advanced extraction capabilities of the Signaling Advisor's advanced call trace application.

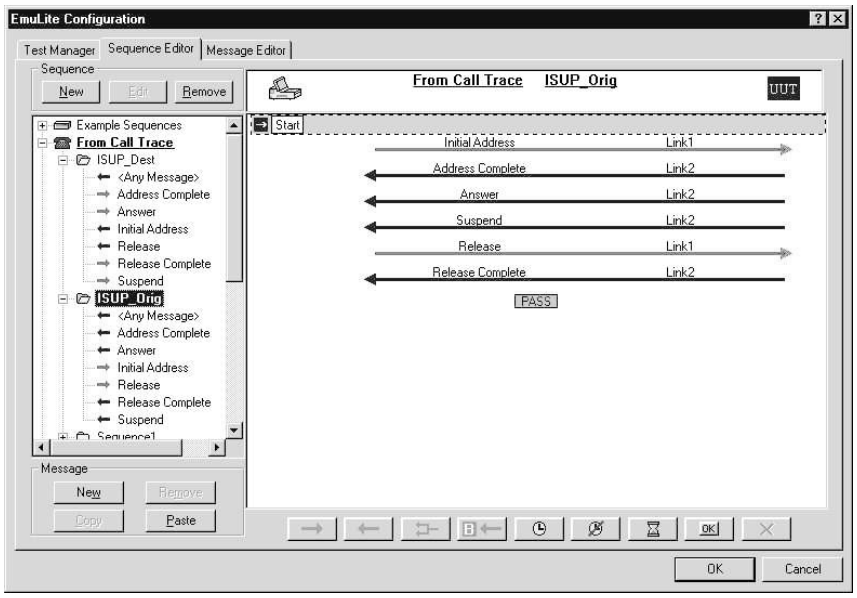

Figure 9. Originating ISUP call

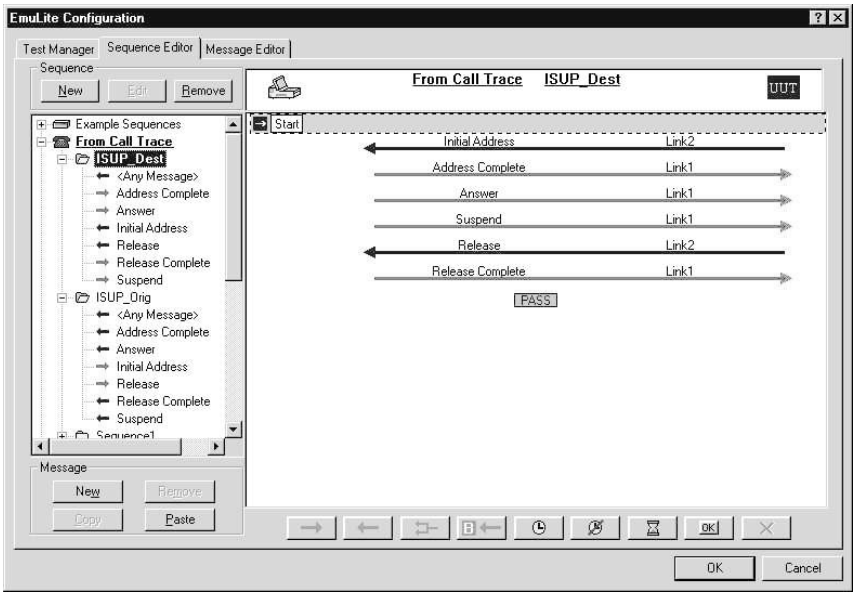

Figure 10. Terminating ISUP call

Testing New Services

Getting new services up and running fast is crucial to maintaining competitiveness. Engineers can test advanced new services such as local number portability (LNP) quickly using the Signaling Advisor and EmuLite software. Figure 11 shows a transaction in which an IAM message has been sent, generating a Query with Permission message from the SCP. It is easy and fast to set up a test that can send the IAM message and verify that the correct Query message, containing the Called Party Digits, has been generated in response. Test engineers also can quickly compare any of the expected fields or create a script that checks the validity of other messages that can be generated as part of the LNP service.

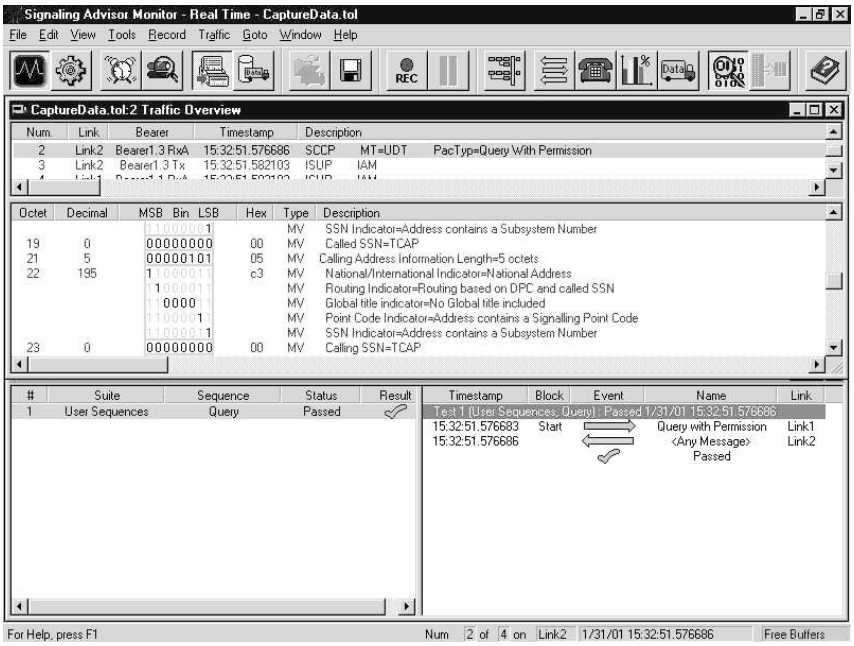

Figure 11. An LNP transaction in EmuLite

Conclusion

Whether you are developing and testing complex or simple signaling procedures, your time is a precious and ever-diminishing resource. EmuLite users tell us that by using the Signaling Advisor's graphical emulation environment, they have significantly reduced the time it takes to develop tests. With the Signaling Advisor's advanced analysis capability, they also have been able to dramatically reduce the time it takes to identify faults.

The days of using complicated programming languages, text editors, hex message builders, and compilers to solve signaling problems are numbered. The Signaling Advisor and EmuLite software are changing the playing field on which signaling testing takes place.

Notes _____

www.agilent.com

Agilent Technologies' Test and Measurement Support, Services, and Assistance

Agilent Technologies aims to maximize the value you receive, while minimizing your risk and problems. We strive to ensure that you get the test and measurement capabilities you paid for and obtain the support you need. Our extensive support resources and services can help you choose the right Agilent products for your applications and apply them successfully. Every instrument and system we sell has a global warranty. Support is available for at least five years beyond the production life of the product. Two concepts underlie Agilent's overall support policy: "Our Promise" and "Your Advantage."

Our Promise
Our Promise means your Agilent test and measurement equipment will meet its advertised performance and functionality. When you are choosing new equipment, we will help you with product information, including realistic performance specifications and practical recommendations from experienced test engineers. When you use Agilent equipment, we can verify that it works properly, help with product operation, and provide basic measurement assistance for the use of specified capabilities, at no extra cost upon request. Many self-help tools are available.

Your Advantage
Your Advantage means that Agilent offers a wide range of additional expert test and measurement services, which you can purchase according to your unique technical and business needs. Solve problems efficiently and gain a competitive edge by contracting with us for calibration, extra-cost upgrades, out-of-warranty repairs, and on-site education and training, as well as design, system integration, project management, and other professional engineering services. Experienced Agilent engineers and technicians worldwide can help you maximize your productivity, optimize the return on investment of your Agilent instruments and systems, and obtain dependable measurement accuracy for the life of those products.

By internet, phone or fax, get assistance with all your Test and Measurement needs.

Online assistance:
http://www.agilent.com/find/assist

United States:
(Tel) 1 800 452 4844

Canada:
(Tel) 1 877 894 4414
(Fax) (905) 282 6495

China:
(Tel) 800-810-0189
(Fax) 1-0800-650-0121

Europe:
(Tel) (31 20) 547 2323
(Fax) (31 20) 547 2390

Japan:
(Tel) (81) 426 56 7832
(Fax) (81) 426 56 7840

Korea:
(Tel) (82-2) 2004-5004
(Fax) (82-2) 2004-5115

Latin America:
(Tel) (305) 269 7500
(Fax) (305) 269 7599

Taiwan:
(Tel) 080-004-7866
(Fax) (886-2) 2545-6723

Other Asia Pacific Countries:
(Tel) (65) 375-8100
(Fax) (65) 836-0252

Printed in U.S.A. April 25, 2002

5988-0944EN

Use this link to go directly to our network troubleshooting solutions:
http://www.agilent.com/comms/onenetworks

Agilent Technologies