# SS7 SIGNALLING OVER ETHERNET USING IPSL IN A MOBILE COMMUNICATIONS TESTING LABORATORY

**Razvan Andrei VISAN, Marian ALEXANDRU, Dan Nicolae ROBU**

Transilvania University, Brasov, Romania
(razvan.visan1@gmail.com, marian.alexandru@unitbv.ro, dan.robu@unitbv.ro)

DOI: 10.19062/1842-9238.2017.15.3.8

***Abstract:*** *The practical approach to signaling in mobile communications benefits greatly from the special solution presented through this paper, namely the replacement of the MTP-1, MTP-2 inferior levels of the Message Transfer Protocol through Ethernet and IP (Internet Protocols). SS7 over Ethernet signaling in a test laboratory has provided a series of solutions in Adaptation Layer (AL / Middleware) related to virtualization techniques – "cleavage" in the OSI stack means independence from lower levels (keeping any services, programs, protocols, tests, generally any procedures or network plans at higher levels). This paper contributes to the concept of mixed equipment (real infrastructure combined with the emulated one without affecting controllability, testability) using IPSL – Independent protocol simulator language as a chosen solution.*

***Keywords:*** *Middleware, IPSL, signaling*

## 1. INTRODUCTION

One of the most important part of the GSM network is signaling, provided by the SS7 (Signaling System number 7) protocol suite. Operators are deploying SS7 services over a dedicated 56 kbps or 64 kbps bandwidth using Time Division Multiplex (TDM) technology, or high speed T1 (1.5 Mbps) or E1 (2.048 Mbps) streams.

With the introduction of SIGTRAN in GSM network opened up a new perspective in the concept of signaling. SIGTRAN (SIGnalling TRANsport) is a standardized IP protocol developed by the IETF (Internet Engineering Task Force) working group that allows the transport of signaling using the IP protocol. This protocol has been developed to address the SS7 overload problem due to the 16 channel limitation for TDM technology.

Using SIGTRAN protocol IPSL can be “inserted” almost everywhere in the GSM architecture successfully emulating the rest of the network without affecting the overall functionality. This approach can be either used for learning the behavior of a specific equipment in various scenarios or debugging it. One of the easiest ways of learning is the “black box” approach where the functionality of the equipment is totally neglected, the main focus being the input and especially the output.

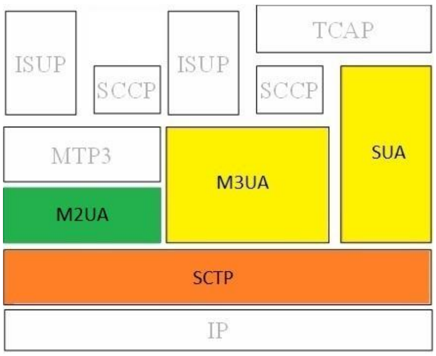



**FIG. 1.** SIGTRAN protocol stack

## 2. IPSL CONCEPT AND WORKFLOW

IPSL is a high level language developed in order to facilitate creations of numerous test suites which verify interoperability cross vendor and the behavior of the network in different scenarios. It is independent of the protocol tested, and the language does not contain specific protocol elements.

All elements, such as message type codes, protocol-specific messages, or message parameters are defined separately and transmitted transparently by the IPSL script to the simulator. IPSL allows multiple scripts or instances of the same script (possibly with different parameters) to run in parallel. An entity of the simulator, which can run a instance of a scenario in terms of IPSL, is a resource (sometimes called port).

A resource also includes a set of data to store all the information needed to direct incoming messages to the correct instance. The required information depends on the protocol used. A context-change between two running scripts is done only at the WaitFor statement or after the script has ended. This can be called a cooperative multitasking.

The function "WaitFor" always specifies which message is expected as the next event. If a different event than expected occurs, it can be interpreted in two ways: as an unexpected event or an asynchronous one. A perfect illustration of the IPSL workflow is presented in Fig. 2.

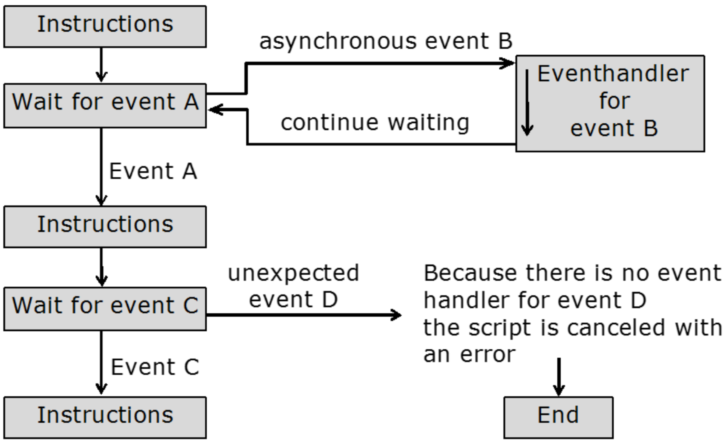



**FIG. 2**. IPSL standard workflow

The internal architecture is defined around two main files: the script and the control file; the first is a sequence of user defined messages which are injected in the network on the interface where IPSL is connected.

The second contains the loop invoking scripts (usually called TestBlock). These loops can expire by time range or by reaching a configurable number of passings.

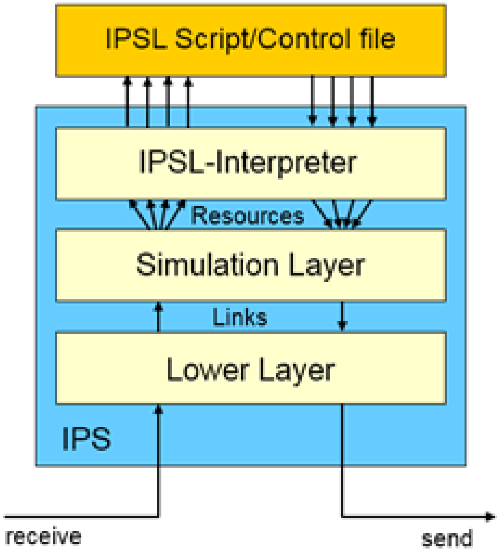



**FIG. 3.** Bottom-up perspective of IPSL functionality

The simulator offers a series of predefined messages grouped in a collection. In the script, a message can be retrieved from that collection via Pool() function. Now the selected message can either be sent with the default values or modified in order to adapt to the networks demands.

There are specific functions which grant access to almost every parameter of a message thus allowing the user either to set the desired value or even not sending the parameter at all. To simplify the IPSL message type model, these are treated as strings. This implies that messages can be stored in variables and can be defined as "string" constants.

## 3. PRACTICAL APPROACH. EMULATION APPLIED IN LOCATION UPDATE SCENARIO

The components used to implement the experiment were the BTS (Base Transceiver System) and BSC (Base Station Controller) representing the access network, these being the real elements of the system. The emulated hardware is made up of the Dialogic® TX 4000 PCI board with the role of interconnecting the computer with the GSM architecture.

The software behind the Dialogic® TX 4000 is IPSL which provides and controls the specific message flow needed by the BSC in order to successfully complete a Location Update scenario.

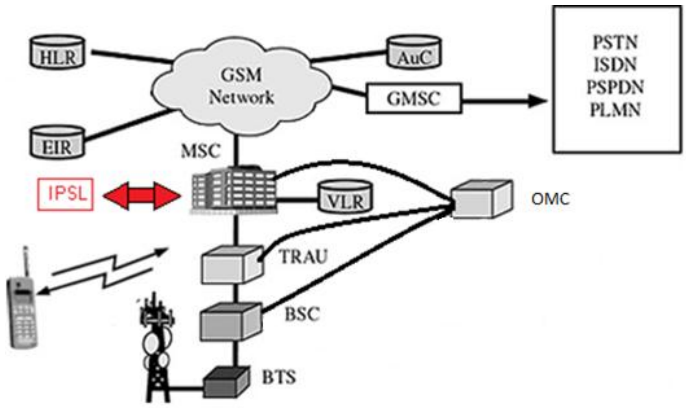



**FIG. 4**. The GSM architecture used in the tested scenario

The results are shown in a graphical user interface which provides the user with the following features: Standard Windows Application; Separate state viewer for each layer; Trace viewer showing messages for each protocol layer; permanent view to the main simulation state (actual count, load and errors).

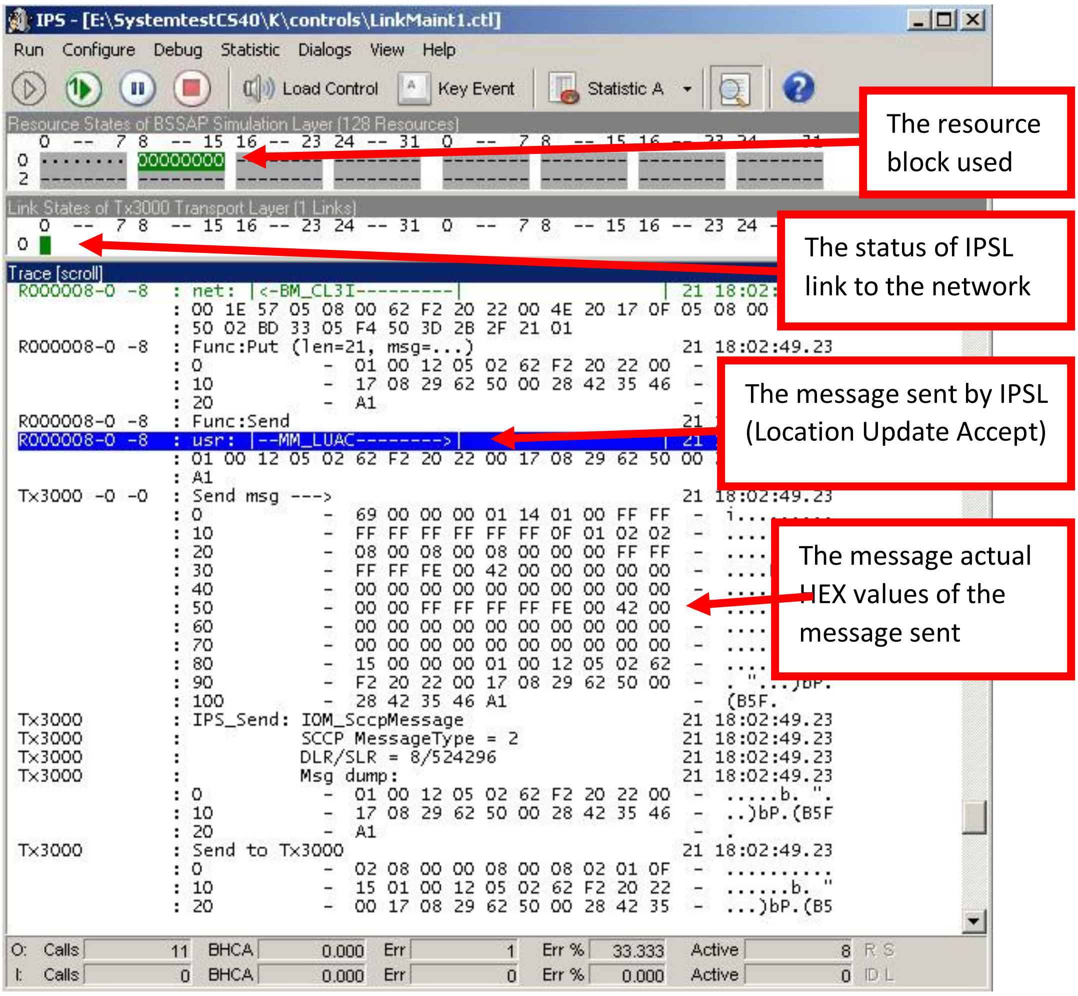


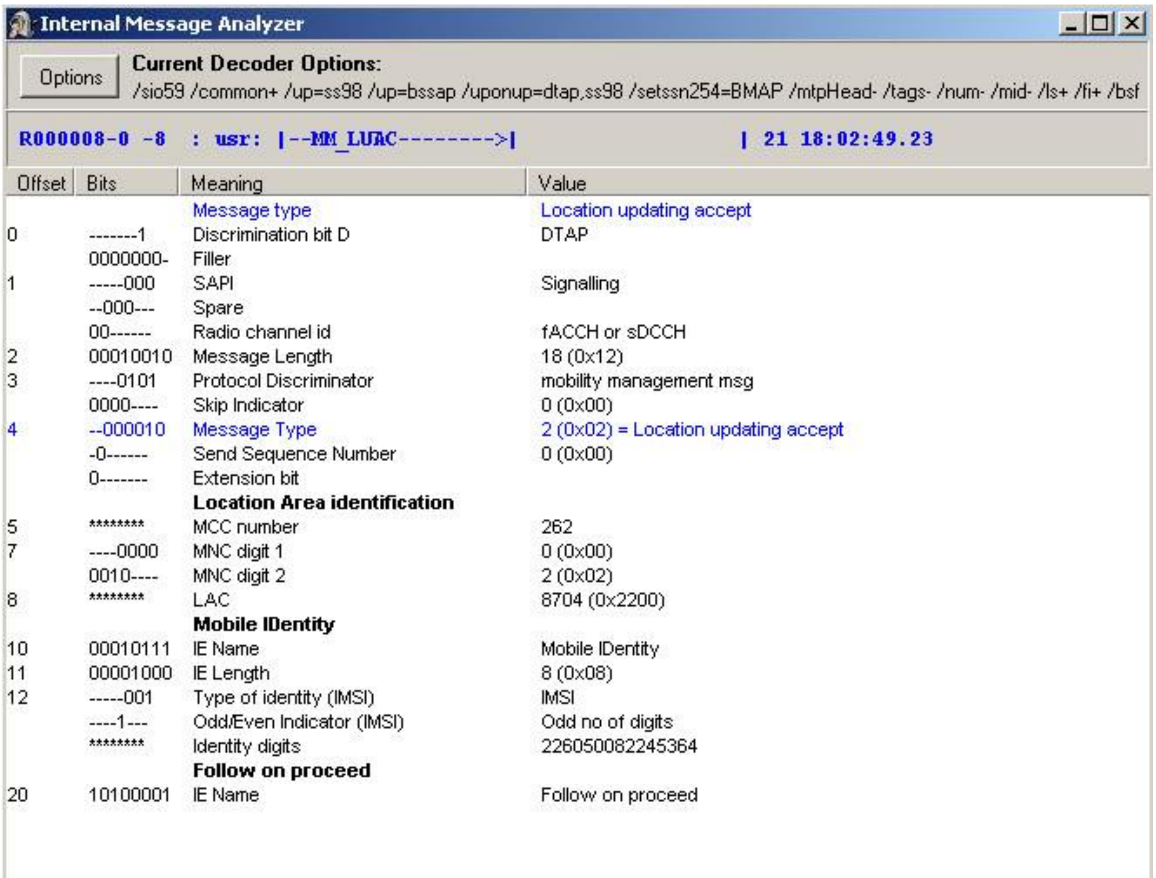

**FIG. 5.** IPSL GUI in live scenario

Every message from the GUI (Graphic User Interface) can be further analyzed using the Internal Message Analyzer; this tool provides the user with: online decoding of sent or received messages; decoding of messages on different protocol layers; discovering of wrongly structured messages.

The Internal Message Analyzer decodes the HEX values sent in a user interpretable manner. The messages name use the following format: <GSM_function>_<Message_Abreviation>. In Fig.. 6 is presented the actual "MM_LUAC" – (Mobility Management Location Update Accept) used in the tested scenario. Every HEX value presented in Fig. 5 is translated into a pair meaning – value.

These values were set in the IPSL script in order to match the Location Update request thus granting the BSC the needed parameters in order to authorize the whole process. With real-world equipment, the response would have been built at the MSC level by successive interrogations on the VLR and HLR databases.

Using IPSL requires prior knowledge of the parameters expected by the BSC to successfully accomplish the emulation of the MSC response such as MCC (Mobile Country Code), MNC (Mobile Network Code), LAC (Location Area Code) etc.


Internal Message Analyzer

Options **Current Decoder Options:**
/sio59 /common+ /up=ss98 /up=bssap /uponup=dtap,ss98 /setssn254=BMAP /mtpHead- /tags- /num- /mid- /ls+ /fi+ /bsf

R000008-0 -8 : usr: |--MM_LUAC-------->| | 21 18:02:49.23

| Offset | Bits | Meaning | Value |
|---|---|---|---|
| | | Message type | Location updating accept |
| 0 | -------1 | Discrimination bit D | DTAP |
| | 0000000- | Filler | |
| 1 | -----000 | SAPI | Signalling |
| | --000--- | Spare | |
| | 00------ | Radio channel id | fACCH or sDCCH |
| 2 | 00010010 | Message Length | 18 (0x12) |
| 3 | ----0101 | Protocol Discriminator | mobility management msg |
| | 0000---- | Skip Indicator | 0 (0x00) |
| 4 | --000010 | Message Type | 2 (0x02) = Location updating accept |
| | -0------ | Send Sequence Number | 0 (0x00) |
| | 0------- | Extension bit | |
| | | **Location Area identification** | |
| 5 | ******** | MCC number | 262 |
| 7 | ----0000 | MNC digit 1 | 0 (0x00) |
| | 0010---- | MNC digit 2 | 2 (0x02) |
| 8 | ******** | LAC | 8704 (0x2200) |
| | | **Mobile IDentity** | |
| 10 | 00010111 | IE Name | Mobile IDentity |
| 11 | 00001000 | IE Length | 8 (0x08) |
| 12 | -----001 | Type of identity (IMSI) | IMSI |
| | ----1--- | Odd/Even Indicator (IMSI) | Odd no of digits |
| | ******** | Identity digits | 226050082245364 |
| | | **Follow on proceed** | |
| 20 | 10100001 | IE Name | Follow on proceed |

**FIG. 6.** Analysis of Location Update Accept message

## 5. CONCLUSION

Based on the practical results obtained in this paper, by integrating high-complex equipment with the possibility of emulating any equipment in the GSM architecture, we have been able to implement the mixed reality concept without affecting the functions of the access network. The scenario developed in the IPSL environment has been designed to meet the test standards.

Based on the protocols used, SS7 and SIGTRAN the used messages have standard template. Their correct parameterization was obtained by repeated tests, the results being collected on the basis of the response of the tested equipment. This has demonstrated the effect of parameterisation of messages on the access network. This kind of software can have a major economic impact because now any equipment can be tested wihtout having the whole architecture at your disposal.

## REFERENCES

[1] M. Sauter, *From GSM to LTE-Advanced: An Introduction to Mobile Networks and Mobile Broadband, Revised Second Edition*, ISBN: 978-1-118-86195-0;

[2] M. Alexandru, G. Morariu, *Comunicatii Mobile Celulare şi Calcul Mobil. Evolutia de la 3G la 4G,* Editura Universității Transilvania din Brașov, 2015, ISBN: 978-606-19-0567-6;

[3] T. Balan, D. Robu, F. Sandu, *Integrarea Sistemelor de Calcul şi Telecomunicaţii*, Editura Universității Transilvania din Brașov, ISBN 978-606-19-0609-3;

[4] J. Eberspächer, H.J. Vögel, C. Bettstetter. *GSM Switching, Services and Protocols*, Second Edition, ISBN: 978-047-14-9903-9;

[5] A. Z. Dodd, *The Essential Guide to Telecommunications*, Fifth Edition, ISBN: 978-0137058914;

[6] M. Sauter, *3G, 4G and Beyond: Bringing Networks, Devices and the Web Together*, Second Edition, ISBN: 978-1-118-34148-3.