A Case Study on Protocol Stack Integration for 3GPP LTE Evolved Node B

Fabbryccio A. C. M. Cardoso, Felipe A. P. Figueiredo, Rafael Vilela and João Paulo Miranda
Centre for Research and Development in Telecommunications (CPqD), Campinas/SP, Brazil
{fcardoso, felipep, rvilela, jmiranda}@cpqd.com.br

***Abstract*—This paper presents a case study of the integration of the Long Term Evolution (LTE) physical layer into the radio protocol subsystem of a macrocell base station. We conduct tests in the laboratory using a modular experimental setup aimed at validating this protocol stack integration. Measurements and analysis are carried out with the aid of Wireshark, a well-known protocol analyzer tool. This choice allows the inspection of data packets at the application program interface between PHY and MAC subsystems. Our results suggest that the LTE protocol stack can operate synchronously, thus supporting message sequences conveyed within both random access and radio-resource-control connection procedures. The proposed experimental setup and test methodology make it possible to validate the control plane operation of the protocol stack without the need of a core network interface.**

I. Introduction

The Long Term Evolution (LTE) technology was standardized in 2008 as the successor of third generation (3G) cellular systems. For the first time, we have technology convergence in one generation of cellular communication systems. This has already been affecting market scale in a number of aspects, *e.g.* the migration to LTE is taking place at a pace considerably faster than that from earlier generation networks to 3G [1].

Higher data rate and growth saturation in existing networks drive the demand for LTE infrastructure, which is currently being installed so as to upgrade the entire ecosystem. This includes base stations, backhaul, and the core network. At the base station, in particular, the radio protocol stack is typically specialized in physical layer and protocol subsystems, each requiring dedicated devices. Examples of such devices include, but are not limited to, multicore general purpose processor (GPP), for the protocol subsystem, while application specific integrated circuit (ASIC), digital signal processor (DSP) and field programmable gate array (FPGA) are some of the options available when it comes to the physical layer subsystem.

However, even when highly integrated system on a chip (SoC) devices are used, system designers face the task of optimizing the interface between subsystems. The synchronized operation and tight timing constraints of LTE require hardware and software optimizations, followed by a careful validation of the radio protocol stack in terms of real time operation.

This paper deals with the latter subject in that it presents a case study of a complete radio protocol integration for modular LTE baseband units [2]. Taking into consideration the high capacity often envisioned for macro cellular base stations, the physical layer subsystem is implemented "in-house", at CPqD, using a Xilinx Virtex 6 FPGA device. In contrast, the protocol subsystem is a commercial software component from Aricent [3] that is optimized for use with Intel core i7 technology. Each device runs on a separated radio advanced mezzanine card (AMC) plugged into a chassis based on the micro telecommunications computing architecture (μTCA). The physical interface between devices is raw gigabit ethernet (GbE), used for validation purposes.

The proposed validation tests focus on the physical layer. Basically, the tests consist of capturing in real time the exchange of message sequences required to establish a radio resource control (RRC) connection with a user equipment (UE) terminal emulator [4]. Besides exercising all physical channels, this approach exploits layer 2-3 interactions regarding transport and logical channels, as well as transparent and acknowledged service modes. It is worth emphasizing that the objective of our experiment is *not* to verify conformance to the LTE standard. Instead, the experiment was designed to support debugging of the physical layer and its corresponding application program interface (API), driven by standard protocol procedures.

The remainder of the paper is as follows. Sections II and III offer some background on the baseband unit in terms of the LTE protocol stack, layer interaction, and connection establishment. Our base station architecture is then presented in Section IV. We carry on with Sections V, VI and VII describing our test methodology, setup, and results, respectively. Limitations of our approach are given in Section VIII, alongside with ways to overcome them. Section IX closes the paper.

II. The LTE Baseband Unit

The physical layer supports single and multiple transmit antennas, using transmission modes TM1-TM4, with up to 20 MHz bandwidth, frequency division duplexing and normal cyclic prefix [11]. For simplicity, packet data convergence protocol (PDCP) functions [5], *e.g.* header compression, cypher and integrity protections, are turned off at the layer 2.

The UE establishes a connection to the base station, referred as evolved node B (eNodeB) in the context of LTE, using procedures of random access [6] and RRC connection establishment [7]. The former is used to achieve time synchronization in the uplink and, if it is the first access, to acquire a 16-bit layer 2 address known as cell radio network temporary identity (RNTI). Dedicated control messages are allowed after the UE has established a RRC connection through the signaling radio bearer (SRB) type SRB1, including initial non-access stratum (NAS) protocol transmission [8], which triggers security activation and network attachment.

System information acquisition, random access and RRC connection establishment procedures involve practically all interactions between layers in the protocol stack. These procedures can exercise physical, transport and logical channels, as well as NAS and signaling radio bearers SRB0 and SRB1. We therefore deem these procedures as appropriate to validate the interactions between the physical layer subsystem (signal processing) and the protocol subsystem.

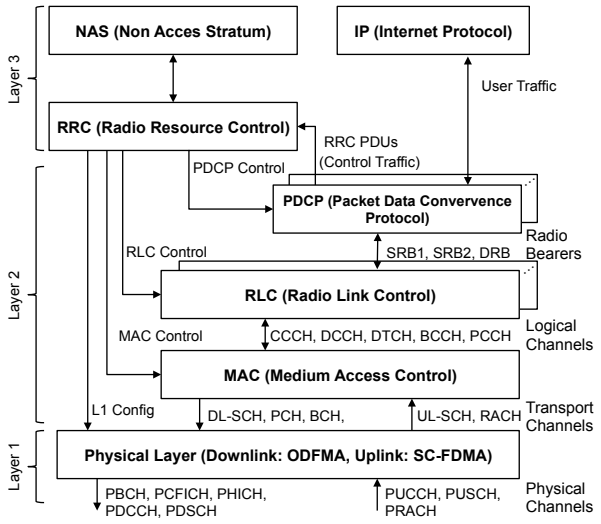


Figure 1: Layer interactions of LTE radio protocol stack.

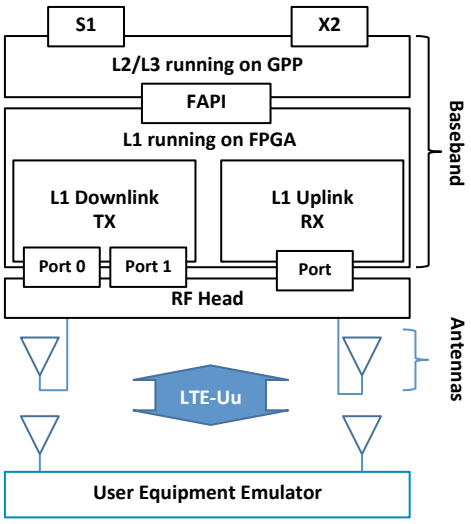


Figure 2: Base station functional architecture.

III. LTE Radio Protocol Stack

The term radio bearer denotes a tunnel segment established between UE and eNodeB with specified quality of service and priority requirements, which are the eNodeB's responsibility to ensure. Depending on the traffic purpose (control/user application data), the LTE radio bearers are classified in SRB or data radio bearers (DRB).

RRC messages convey SRB and can also embed control NAS messages to allow the core network to control mobility and Internet protocol (IP) connectivity functions at the UE [9]. The SRB types supported are SRB0-SRB2. SRB0 is used for RRC messages related to connection requests. It uses the common control logical channel (CCCH). There is no PDCP entity associated. SRB1 is used for RRC messages related to UE resource configuration. It uses the dedicated control logic channel (DCCH) to access services provided by layer 2. SRB2 is used to transmit control messages between the UE and core network mobile management entity. NAS messages embedded into the RRC messages are used in this case.

Figure 1 shows the LTE protocol architecture with emphasis on L2/L3 layers. A UE-eNodeB pair of PDCP entities is associated with each radio bearer. Depending on whether bearer type is unidirectional or bidirectional, each PDCP is further associated with one or two radio link control (RLC) entities [10].

PDCP is the radio protocol entity responsible for integrity and security protection [5]. These services are dedicated per user bearer and made available after the initial security activation and radio bearer establishment that follow the connection establishment procedure. Once security is enabled, all messages carried by SRB1 and SRB2 are encrypted and integrity protected in the PDCP. Control plane messages should always be carried by signaling radio bearers. However, there is an exception for broadcast messages, which carry system information originated in layer 3 (RRC), *e.g.* master information block (MIB) and system information blocks (SIB) of type SIB0, SIB1, SIB2, and paging messages.

RLC is the entity responsible for providing services, such as packet segmentation and concatenation, and for ascertaining the ordered delivery of packets. Data transfer services are provided based on transparent (TM), unacknowledged (UM) or acknowledged (AM) modes [10].

Simply put, there is only one UE-eNodeB pair of MAC entities per UE, which is responsible in one direction for multiplexing logical channels into transport channels and for reversing these operations in the opposite direction [9]. Specialized algorithms for scheduling, hybrid automatic repeat request and link adaptation have an important impact on such multiplex operation. The transport and physical channels determine how bits are encoded and modulated for transmission on both the downlink and uplink of the air interface.

All data packets carried by radio bearers are mapped onto transport channels, namely, downlink shared channel (DL-SCH) or uplink shared channel (UL-SCH), and physical data shared channel (PDSCH) or physical uplink shared channel (PUSCH), depending on whether the traffic is conveyed in the downlink or uplink, respectively. The RNTI is used to address PDSCH to the assigned UE. According to this address, it is possible to identify the following messages and logical channels: hexadecimal RNTI address #FFFF is used for SIB; RNTI #FFEE for paging messages; RNTI #0000-#009 for RandomAccessResponse; and the address range #000A to #FFFD for dedicated channels assigned to UEs.

IV. System Under Test

Functionally, a base-station transceiver architecture can be split into radio frequency (RF) head and baseband modules as shown in Figure 2. Although the RF equipment is part of the experiment, only the baseband module is the system under test (SUT) considered in the present paper. We assume that the RF interface card is properly calibrated and tested for the frequency band of interest. The baseband module is responsible for the radio protocol stack (layer 2/layer 3), including the physical layer (layer 1), the network interfaces S1, the interface between an eNodeB and an evolved packet core (EPC), and the logical interface between eNodeBs, X2.

In its current implementation, the baseband processing is distributed over devices that are specialized in protocol processing and signal processing. The chipset employed consists of a high performance FPGA device for the physical layer, and a GPP for the protocol stack.

The logical interfaces S1, X2, femto application platform interface (FAPI), and antenna ports are part of the baseband architecture. S1 and X2 are packet switched interfaces to

the core network and in between eNodeB, respectively. Antenna ports are based on low-voltage differential signaling (LVDS) electrical interface, which transports in-band and quadrature (IQ) signals through an FPGA mezzanine card (FMC) connector between RF card and baseband unit.

The logical interface between the protocol and the physical layer subsystems is based on FAPI over raw GbE [12]. This approach provides a baseline designed to evaluate the correct operation of the baseband unit, integrating protocol and physical layer subsystems.

The experiments described later on in this paper focus on the integration of a physical layer developed "in-house" by CPqD with a third-party protocol subsystem through the FAPI protocol interface. The protocol subsystem was previously tested using a MAC-to-MAC communication approach, while the physical layer was validated in terms of the physical channel using a Signalium UE Emulator (SORBAS) and an Agilent Infinium running VSA89600 [4]. Detailed information about the software and hardware employed in the experiment is provided in Table I. In the sequel, we present procedures and results that validate the integrated subsystem operation in regards to the LTE control plane signaling.

V. Methodology

Consider now the task of integrating the physical layer, which runs on FPGA, and the L2/L3 protocol stack, which runs on GPP. In this context, it is crucial to isolate the problem of validating the interactions between L1 and L2/L3 from other interactions regarding the core network. Bearing these goals in mind, the system information acquisition, random access, and RRC connection establishment procedures can be used to verify the radio protocol stack without being concerned about possible issues that might arise in the interactions with the core network.

Regarding system information acquisition, the eNodeB has to broadcast system information periodically in the cell, $e.g.$ MIB, SIB1, SIB2, etc. Such broadcast messages use only the downlink, but exercise the entire protocol stack in the control plane. System information acquisition is important for the proper operation of the UE, as it configures several essential parameter sets that are common to all UEs.

The random access procedure is performed after the UE has configured (through the SIBs) all the parameters needed for proper operation in the cell. This procedure exercises the radio protocol stack in both downlink and uplink

Table I: Hardware and Software Description

Unit	HW/SW	Description
UE	SORBAS 200/410	Signalion (National Instruments) UE Emulator.
BB	μTCA Platform	Baseband Unit • μTCA Chassis Kontron OM6040; • μTCA Carrier Hub Kontron AM4904-PCI;
L1	AMC card + FMC radio + IP Core	• AMC card Nutaq Perseus 6010 with FPGA Xilinx Virtex 6 LX240T; • FMC card Nutaq Radio421M with two TX and two RX; • CPqD LTE Firmware.
L2/L3	AMC card + Aricent Stack L2/L3	• AMC card Kontron AM4020 with Intel 2.0 GHz Core i7-620LE; • Aricent SW Package for L2/L3.
PA	Wireshark + FAPI Dissector	Protocol Analyzer • Wireshark software with FAPI dissector

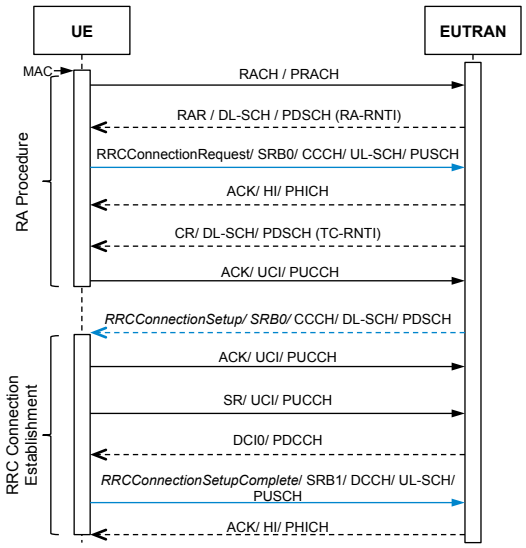


Figure 3: Message sequence for random access and RRC connection establishment.

directions. The message sequence exchanged between UE and eNodeB is shown in Figure 3. After having successfully completed this procedure, the UE acquires its RNTI, and triggers the connection establishment procedure through the message RRCConnectionRequest, also seen in Figure 3.

The connection establishment starts with the message RRConnectionRequest, but the procedure is triggered only after the eNodeB receives an acknowledgment for the contention resolution message. The messsage RRCConnectionSetup is then used by the UE to configure dedicated radio resources, $e.g.$ SRB addition, MAC main reconfiguration, semi-persistent scheduling reconfiguration and physical channels configuration. To complete the procedure, the UE sends the message RRCConnectionSetupComplete after receiving the scheduling grant from eNodeB in the PDCCH. Such scheduling grant is provided as a response to the UE scheduling request in the PUCCH.

The next protocol procedures after connection establishment would be the initial security activation and radio bearer establishment. However, these procedures require integration to the core network through the S1 interface.

The validation of the eNodeB radio protocol stack is based on laboratory tests, which were designed with the aim of analyzing message exchanges between UE and eNodeB. Message sequences, as shown in Figure 2, are filtered and analyzed to check if the initial connection procedures are correctly implemented in accordance with the LTE standard. The experimental setup described in Figure 4 involves protocol analysis software, μTCA hardware platform and test equipment, which were implemented using the actual components summarized in Table I.

The protocol analyzer Wireshark comes with built-in support for LTE protocols but, as the FAPI protocol is not supported, a third-party FAPI dissector needs to be installed. In our case, an Aricent dissector was installed to guarantee interoperability of the L1 developed in this project with the Aricent L2/L3. Recall that FAPI is not a standard, but a recommendation. The protocols required for analysis are SLL (TCPDump), Ethernet, IPv4, UDP, FAPI, MAC-LTE, RLC-LTE, PDCP-LTE, and LTE-RRC.

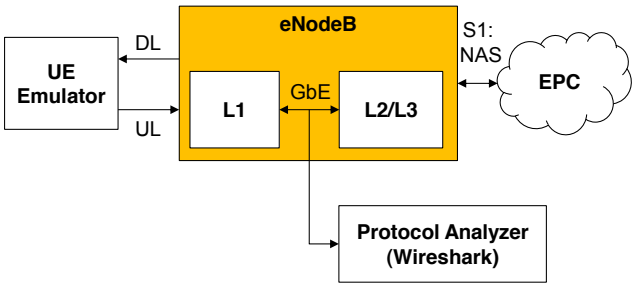


Figure 4: Block diagram of the experimental setup.

The protocol analysis considers the sequence of messages exchanged, along with their timestamps. In fact, the occurrence of a subsequent message in the call flow is an important indication that the previous message has correctly been received and decoded.

The rate of messages exchanged at the FAPI level generates a myriad of packets for timing indication and configuration of physical layer resources. Messages, such as SUB_FRAME_INDICATION, DL_CONFIG_REQUEST and UL_CONFIG_REQUEST, are exchanged every 1 ms. This is a clear indication that it is not possible to analyze message sequences without filtering capabilities in a call flow with, for instance, 90 ms duration.

Wireshark is an excellent tool to capture and analyze packet flows through easily configured filters. We employ the filters described in Table II and III to validate the call flow in Figure 2. Wireshark also enables color representation according to the identified message, or to the values assumed by specified message fields. The set of rules used to color packets according to the filters are based on the same rules used in the filters.

VI. Experimental Setup

The setup used to run the experiment follows the scheme depicted in Figure 4. The eNodeB is modular, as shown in Figure 5, composed from left to right by an Intel core i7 board for L2/L3, an FPGA AMC board with FMC radio for the physical layer, and a network hub for switching capabilities. Detailed information is provided in Table I. Wireshark runs in a notebook with Fedora OS, which is not shown in Figure 5 but can capture packet data through a switch connected to a network hub. The eNodeB TX/RX connects to the UE RF unit through attenuators, and operates with single antenna mode in downlink. Messages between L1 and L2/L3 were opened and analyzed at FAPI, MAC and RRC levels. The eNodeB could be also connected to an EPC through the GbE interface of the L2/L3 board. This interface is *not* used in the experiment.

Table II: Capture Filters for L2/L3 Messages

Message	Filter
MIB	lte-rrc.BCCH_BCH_Message
SIB	mac-lte.rnti == 0xffff
SIB1	lte-rrc.systemInformationBlock Type1
SIB (SIB2, SIB3, ...)	lte-rrc.bcch_Config
RACH (preamble)	L1.lte_phy_header.msgId == 0x88 and L1.FAPI_rachIndication _st.numOfPreamble != 0x0000
RAR	mac-lte.rar and mac-lte.rar.t == 0x01
RRCConnectionRequest	lte-rrc.rrcConnectionRequest
CR	mac-lte.dlsch.lcid == 0x1c
RRCConnectionSetup	lte-rrc.rrcConnectionSetup
RRCConnectionSetupComplete	lte-rrc.rrcConnectionSetupComplete

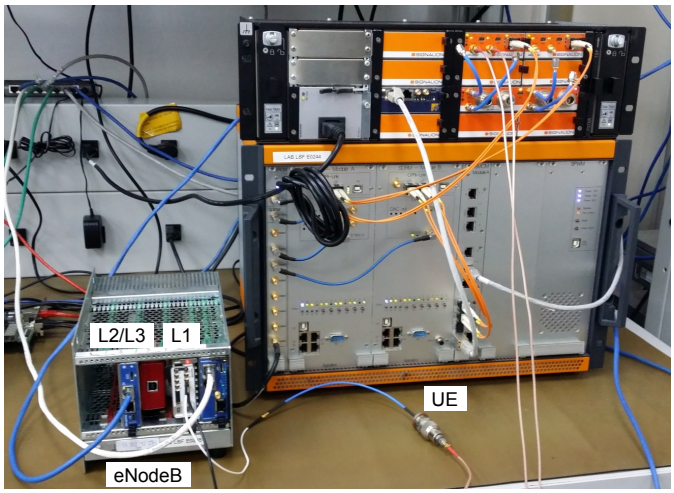


Figure 5: Actual setup used in the experiment.

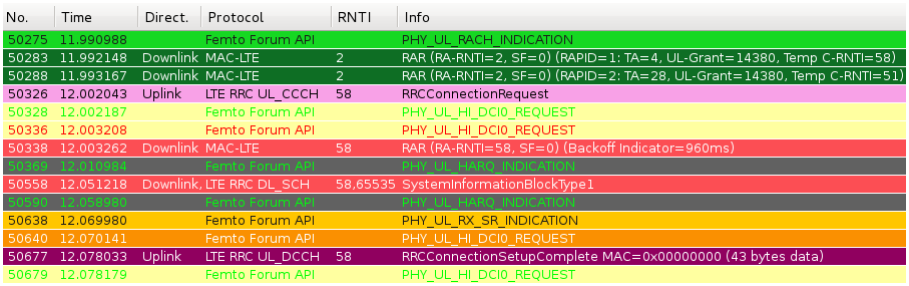
VII. Experimental Results

Exemplary messages captured and filtered using Wireshark are shown in Figure 6. For this specific example, Table IV adds timing information concerning subframe number (SF) and system frame number (SFN). This information was collected at the FAPI level. Every unit increment in SF is equivalent to 1 ms, while in SFN it is equivalent to 10 ms.

Note the second random access response (RAR) in packet number 50288. This is the result of an erroneous preamble detection. This false alarm is inconvenient because it generates an overhead of radio resource usage and eNodeB processing time. However, it is not critical in terms of system functionality, since there is no UE access attempt associated to that RAR. The correct RAR caused the UE to respond with an RRCConnectionRequest, which is correctly detected by the eNodeB in packet 50326 (5 ms after the corresponding RAR). Receipt confirmation with ACK is sent 4 ms after the receipt of the RRCConnectionRequest in packet 50328. As the second RAR is in error, the eNodeB does not receive any connection request within the expected time frame, so it sends back NACK as response in packet 50336. This ends the message sequence started by the false preamble detection. Note that

Table III: Capture Filters for Basic L2 and FAPI Messages

Message	Filter
PDSCH HARQ	L1.lte_phy_header.msgId == 0x85
PDSCH HARQ (ACK)	L1.lte_phy_header.msgId == 0x85 and L1.FAPI_fddHarqPduIndication_st.harqTB1 == 0x01
PDSCH HARQ (NACK)	L1.lte_phy_header.msgId == 0x85 and L1.FAPI_fddHarqPduIndication_st.harqTB1 == 0x02
PUSCH HARQ	L1.lte_phy_header.msgId == 0x83 and L1.FAPI_dlHiDCIPduInfo_st.numOfHI != 0x00
PUSCH HARQ (ACK)	L1.lte_phy_header.msgId == 0x83 and L1.FAPI_dlHiDCIPduInfo_st.numOfHI != 0x00 and L1.FAPI_dlHiPduInfo_st.hiValue == 0x01
PUSCH HARQ (NACK)	L1.lte_phy_header.msgId == 0x83 and L1.FAPI_dlHiDCIPduInfo_st.numOfHI != 0x00 and L1.FAPI_dlHiPduInfo_st.hiValue == 0x00
SR	L1.lte_phy_header.msgId == 0x8a and L1.FAPI_rxSRIndication_st.numOfSr != 0x0000
UL Scheduling Grant	L1.lte_phy_header.msgId == 0x83 and L1.FAPI_dlHiDCIPduInfo_st.numOfDCI != 0x00
UL CRC	L1.lte_phy_header.msgId == 0x86
UL CRC (OK)	L1.lte_phy_header.msgId == 0x86 and L1.FAPI_crcPduIndication_st.crcFlag == 0x00
UL CRC (NOK)	L1.lte_phy_header.msgId == 0x86 and L1.FAPI_crcPduIndication_st.crcFlag == 0x01



No.	Time	Direct.	Protocol	RNTI	Info
50275	11.990988		Femto Forum API		PHY_UL_RACH_INDICATION
50283	11.992148	Downlink	MAC-LTE	2	RAR (RA-RNTI=2, SF=0) (RAPID=1: TA=4, UL-Grant=14380, Temp C-RNTI=58)
50288	11.993167	Downlink	MAC-LTE	2	RAR (RA-RNTI=2, SF=0) (RAPID=2: TA=28, UL-Grant=14380, Temp C-RNTI=51)
50326	12.002043	Uplink	LTE RRC UL_CCCH	58	RRCConnectionRequest
50328	12.002187		Femto Forum API		PHY_UL_HI_DCI0_REQUEST
50336	12.003208		Femto Forum API		PHY_UL_HI_DCI0_REQUEST
50338	12.003262	Downlink	MAC-LTE	58	RAR (RA-RNTI=58, SF=0) (Backoff Indicator=960ms)
50369	12.010984		Femto Forum API		PHY_UL_HARQ_INDICATION
50558	12.051218	Downlink,	LTE RRC DL_SCH	58,65535	SystemInformationBlockType1
50590	12.058980		Femto Forum API		PHY_UL_HARQ_INDICATION
50638	12.069980		Femto Forum API		PHY_UL_RX_SR_INDICATION
50640	12.070141		Femto Forum API		PHY_UL_HI_DCI0_REQUEST
50677	12.078033	Uplink	LTE RRC UL_DCCH	58	RRCConnectionSetupComplete MAC=0x00000000 (43 bytes data)
50679	12.078179		Femto Forum API		PHY_UL_HI_DCI0_REQUEST

Figure 6: Exemplary screenshot showing packets captured using filters defined as in Tables II and III.

the correct RAR has assigned the UE with RNTI equal to 58.

The eNodeB has transmitted a contention resolution message in packet number 50338, five subframes after RRC-ConnectionRequest and one subframe after ACK confirmation to the UE. Note that Wireshark does not correctly detect the contention resolution message, which is instead identified as malformed. However, this message is confirmed by the UE with an ACK in packet number 50369 after four subframes. This issue is caused at the FAPI level due to some behavior discrepancies with respect to the reserved bits in the installed FAPI dissector. Whenever such issue occurs, the "info" field at Wireshark display becomes inconsistent and should be disregarded.

Following the message sequence, notice that in Packet Number 50558 FAPI, message TXRequest is carrying two RRC messages, one for SIB and another to RRCConnectionSetup. The latter RRC message is confirmed by the UE within the expected time frame (after four subframes). However, this analysis is not straightforward because of issues in the FAPI dissector regarding TXRequest messages, which can erroneously treat this message as malformed.

After receiving message RRCConnectionSetup, the UE requests resources to transmit the message RRCConnectionSetupComplete after 15 ms. The message Scheduling Request is correctly detected in Packet 50638 and it is answered with a transmission grant in Packet 50640 after four subframes.

The transmission grant is received by the UE in Downlink Control Information Format 0 (DCI0) of PDCCH, which allows the transmission of RRCConnectionSetupComplete. At the eNodeB, this message is received in Packet 50677, four subframes after sending the grant. The eNodeB confirms the correct receipt of that message in Packet 50679, after four subframes, thus finalizing the connection procedure.

The periodic broadcasting of SIB and MIB messages that support the UE procedure of system acquisition is shown in Figure 7. Note that SIB is transmitted every 4 SFN, which corresponds to 40 ms. The SIB1 message is sent every 2 SFN (20 ms) and SIB2, along with the remainder SIBs, is transmitted in SystemInformation with periodicity configured by SIB1 of 16 SFN (160 ms).

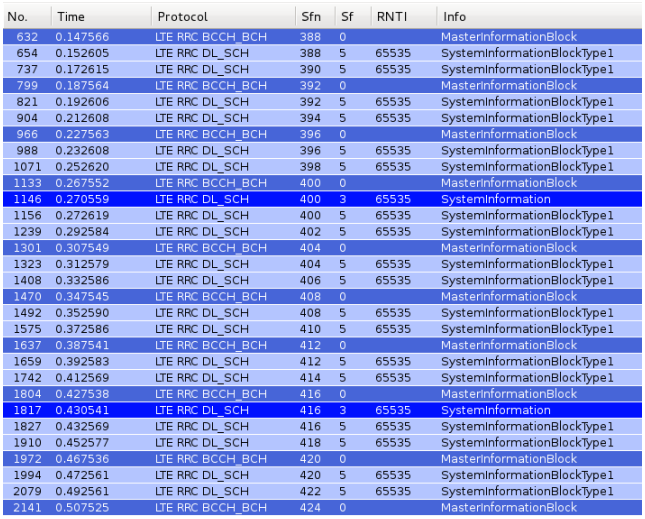
Table IV: Wireshark Results wrt. Figure 6

Message	**No.**	**SFN**	**SF**	**RNTI**	**Info**
PRACH Premble	50275	548	1		
Random Access Response	50283	548	6	2	C-RNTI=58
Random Access Response	50288	548	7	2	C-RNTI=51
RRCConnectionRequest	50326	549	2	58	
UL HARQ (ACK/NACK)	50328	549	6		ACK
UL HARQ (ACK/NACK)	50336	549	7		NACK
Contention Resolution	50338	549	7	58	Msg OK, Dissector NOK
DL HARQ (ACK/NACK)	50369	550	1		ACK
RRCConnectionSetup	50558	554	5	58	
DL HARQ (ACK/NACK)	50590	554	9		ACK
Scheduling Request	50638	556	0		
DCI0 with Scheduling Grant	50640	556	4		
RRCConnectionSetup Complete	50677	556	8		
UL HARQ (ACK/NACK)	50679	557	2		ACK

VIII. Limitations and Discussions

The analysis of only a few procedures is obviously not enough to evaluate the full real-time operation of the radio protocol stack. Besides, the test methodology has taken into account only radio protocol messages from the control plane. Although the physical channel PDSCH has been tested with this approach, further tests are necessary to evaluate PDSCH performance from the user data plane, e.g. to check if the protocol stack operates correctly for this channel under higher data rate scenario. In this case, full experimental setup considering also EPC core network are required for tests involving network attachment and data radio bearers.

The limitation of not testing the radio protocol stack from the user plane perspective was an option to avoid potential issues that might emerge from core network interactions. A second validation phase is then required to validate these interactions, which include, among other procedures, initial security activation, radio bearer establishment and handover. However, recall that the protocol subsystem already is a



No.	Time	Protocol	Sfn	Sf	RNTI	Info
632	0.147566	LTE RRC BCCH_BCH	388	0		MasterInformationBlock
654	0.152605	LTE RRC DL_SCH	388	5	65535	SystemInformationBlockType1
737	0.172615	LTE RRC DL_SCH	390	5	65535	SystemInformationBlockType1
799	0.187564	LTE RRC BCCH_BCH	392	0		MasterInformationBlock
821	0.192606	LTE RRC DL_SCH	392	5	65535	SystemInformationBlockType1
904	0.212608	LTE RRC DL_SCH	394	5	65535	SystemInformationBlockType1
966	0.227563	LTE RRC BCCH_BCH	396	0		MasterInformationBlock
988	0.232608	LTE RRC DL_SCH	396	5	65535	SystemInformationBlockType1
1071	0.252620	LTE RRC DL_SCH	398	5	65535	SystemInformationBlockType1
1133	0.267552	LTE RRC BCCH_BCH	400	0		MasterInformationBlock
1146	0.270559	LTE RRC DL_SCH	400	3	65535	SystemInformation
1156	0.272619	LTE RRC DL_SCH	400	5	65535	SystemInformationBlockType1
1239	0.292584	LTE RRC DL_SCH	402	5	65535	SystemInformationBlockType1
1301	0.307549	LTE RRC BCCH_BCH	404	0		MasterInformationBlock
1323	0.312579	LTE RRC DL_SCH	404	5	65535	SystemInformationBlockType1
1408	0.332586	LTE RRC DL_SCH	406	5	65535	SystemInformationBlockType1
1470	0.347545	LTE RRC BCCH_BCH	408	0		MasterInformationBlock
1492	0.352590	LTE RRC DL_SCH	408	5	65535	SystemInformationBlockType1
1575	0.372586	LTE RRC DL_SCH	410	5	65535	SystemInformationBlockType1
1637	0.387541	LTE RRC BCCH_BCH	412	0		MasterInformationBlock
1659	0.392583	LTE RRC DL_SCH	412	5	65535	SystemInformationBlockType1
1742	0.412569	LTE RRC DL_SCH	414	5	65535	SystemInformationBlockType1
1804	0.427538	LTE RRC BCCH_BCH	416	0		MasterInformationBlock
1817	0.430541	LTE RRC DL_SCH	416	3	65535	SystemInformation
1827	0.432569	LTE RRC DL_SCH	416	5	65535	SystemInformationBlockType1
1910	0.452577	LTE RRC DL_SCH	418	5	65535	SystemInformationBlockType1
1972	0.467536	LTE RRC BCCH_BCH	420	0		MasterInformationBlock
1994	0.472561	LTE RRC DL_SCH	420	5	65535	SystemInformationBlockType1
2079	0.492561	LTE RRC DL_SCH	422	5	65535	SystemInformationBlockType1
2141	0.507525	LTE RRC BCCH_BCH	424	0		MasterInformationBlock

Figure 7: Message sequence of broadcasted system information.

mature commercial product. Considering that the physical layer is a new development, we have focused the validation test on message sequence analysis deemed relevant to exploit all physical channels available in the physical layer.

It is a matter of fact that the FAPI protocol, used to integrate layer 1 with layers 2-3, is rather a recommendation than a standard. Consequently, vendor-specific implementations typically deviate from the original FAPI recommendation due to performance requirements. This affects the actual FAPI implementation differently, depending on the hardware platform and the application scenario for the base station. The protocol analyses is also affected if these specificities are not met. However, the repeatability of the experimental results is assured when the appropriate parser is installed in Wireshark, since the remainder of the protocol stack follows the LTE standard.

Test automation is another limitation, even with the series of filters designed specifically to capture the message sequences from the expected call flows. In this case, inspection and analysis are conducted manually with the Wireshark aid. Additional work would demand considerable effort if the required script was implemented from the scratch, without an appropriate message parser like Wireshark. Currently, we still do not have the best solution to perform these tasks automatically. This is object of our future work.

IX. Conclusions

This paper has described the experimental validation of a radio protocol stack designed for the baseband unit of LTE base stations. Procedures for system acquisition, random access and RRC connection establishment were considered as a way to demonstrate the joint operation between physical layer and the remaining protocol stack without core network interactions.

Our results have also confirmed that the UE emulator SORBAS successfully executes essential tasks during these procedures, e.g. to configure its protocol stack on the basis of received system information. RNTI identification was obtained during random access procedure as expected. Further, the radio bearers between UE and eNodeB were also successfully established.

Finally, the proposed test methodology has allowed the complete radio protocol validation from the control plane perspective. This approach exercises all physical channels, transport channels, logic channels, RLC services, as well as layer 3 control tasks, without involving any interfaces with the core network.

Acknowledgment

This work was supported by Funttel project “RASFA-4G” at CPqD Telecom and IT Solutions.

References

[1] GSMA Intelligence, “*Global LTE network forecasts and assumptions, 2013–17*”, Nov. 2013.

[2] J. Perala, P. Jurmu and J. Pinola , “*Hybrid Approach for Protocol Testing of LTE System*”, In Proc. IEEE Int. Conf. on Advances in System Testing and Validation Lifecycle, pp. 32-36, Aug. 2010.

[3] The Aricent Group, “*eNodeB Protocol Stacks and Software Frameworks*”, [Online] Available: http://www.aricent.com/software/lte-enodeb-software-framework.html.

[4] Signalion GmbH. [Online] Available: http://www.signalion.com. (This product was discontinued).

[5] 3rd Generation Partnership Project. 3GPP TS 36.323: Packet Data Convergence Protocol (PDCP) specification, V9.0.0, Sep. 2012.

[6] 3rd Generation Partnership Project. 3GPP TS 36.300: Overall description - Stage 2, V9.10.0, Dec. 2012.

[7] 3rd Generation Partnership Project. 3GPP TS 36.331: Radio Resource Control (RRC) Protocol specification, V.9.16.0, Sept. 2013.

[8] 3rd Generation Partnership Project. 3GPP TS 24.301: Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS), V9.11.0, Mar. 2013.

[9] 3rd Generation Partnership Project. 3GPP TS 36.321: Medium Access Control (MAC) protocol specification, V9.6.0, Mar. 2012.

[10] 3rd Generation Partnership Project. 3GPP TS 36.322: Radio Link Control (RLC) protocol specification. V9.3.0, Sep. 2010.

[11] 3rd Generation Partnership Project. 3GPP TS 36.213: Physical layer procedures, V9.3.0, Sept. 2010.

[12] FemtoForum, LTE eNB L1 API Definition v1.1. 2010.