OULUN SEUDUN AMMATTIKORKEAKOULU

Leena Kuukasjärvi

CONSTRUCTION AND EVALUATION OF MEASUREMENT SYSTEM OF WCDMA BASE STATION

CONSTRUCTION AND EVALUATION OF MEASUREMENT SYSTEM OF WCDMA BASE STATION

Leena Kuukasjärvi
Bachelor´s Thesis
Spring 2013
Degree Programme in Medical Engineering
Oulu University of Applied Sciences

ABSTRACT

Oulu University of Applied Sciences
Medical Engineering

Author: Leena Kuukasjärvi
Title of thesis: Construction and evaluation of measurement system of WCDMA base station
Supervisors: Kari Jyrkkä (OUAS) ja Antti Ropponen (Elektrobit Wireless)
Spring 2013 Pages: 50 + 0

Base station is a complex embedded device. The hardware as well as software have become increasingly sophisticated. Debugging the software of embedded device, especially at runtime, is in key role when developing the software. From the environment used in development work of complex software is demanded certain requirements. Different kind of measurements are also needed when performance of a software is investigated. Measurement should be done in a way they do not interfere the normal functionality of the device.

The primary objective of this thesis was to construct a measurement system to evaluate the performance of WCDMA base station software. The thesis has been carried out for Elektrobit Wireless.

This thesis introduces a measurement system where different aspects of performance of WCDMA software can be evaluated. The constructed measurement system can be used to measure different kind of performance aspect of software. The wanted measurement features are implemented with software instrumentation. The instrumentation is done with print statements that are part of code. An example of how the instrumentation of software is done within this environment is given. Extra attention is paid to a measurement tool called System Analyzer. System Analyzer provides a portable way to instrument software. It helps the re-usage of software by not being tied to particular hardware or operating system. Also a brief introduction to WCDMA technology is presented. At the end of the thesis the goodness of measurement system is assessed.

Keywords:
WCDMA, base station, measurement system, System Analyzer, software instrumentation

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Hyvinvointiteknologia

Tekijä: Leena Kuukasjärvi
Opinnäytetyön nimi:
Työn ohjaajat: Kari Jyrkkä (OAMK) ja Antti Ropponen (Elektrobit Wireless)
Työn valmistumislukukausi ja -vuosi: 05/2013 Sivumäärä: 50 + 0

Tukiasema on monimutkainen sulautettu laite. Niin laitteisto kuin ohjelmistokin on aina vain monimutkaisempi. Sulautetun laitteen ohjelmiston virheen jäljitys, etenkin laitteen ollessa käynnissä, on isossa roolissa ohjelmiston kehitystyössä. Monimutkaisen ohjelmiston kehitys, kuten tukiaseman, asettaa kehitysympäristölle tiettyjä vaatimuksia. Ympäristössä tehdyt mittaukset eivät saa haitata laitteen normaalia toimintaa. Erilaisia mittauksia tarvitaan tutkittaessa ohjelmiston suorituskykyä.

Tämän lopputyön tavoitteena oli kuvata mittausympäristön luominen WCDMA tukiaseman ohjelmiston suorituskyvyn mittaamiseen. Lopputyö on tehty Elektrobit Wirelessille.

Lopputyössä kuvataan mittausympäristö, jossa voidaan arvioida WDCMA ohjelmiston suorituskyvyn eri ominaisuuksia. Halutut mittausominaisuudet saadaan ohjelmistoon varustamalla ohjelmisto erityisillä mittausinstrumenteilla. Mittausinstrumentit ovat tulostuskomentoja, jotka ovat osa ohjelmiston koodia. Esimerkki tämänlaisesta ohjelmiston mittausinstrumenteilla varustamisesta on esitetty lopputyössä.

Työssä kuvataan myös mittaamisessa käytettyä työkalua System Analyzeriä. System Analyzer mahdollistaa liikuteltavan tavan instrumentoida ohjelmisto. Se parantaa ohjelmiston uudelleen käytettävyyttä, koska sitä voidaan käyttää useissa eri laitteistoissa ja erilaisilla käyttöjärjestelmillä. Lisäksi työn alussa esitellään lyhyesti WCDMA teknologiaa ja lopussa arvioidaan rakennetun mittausympäristön luotettavuutta.

Asiasanat:
WCDMA, tukiasema, mittausympäristö, System Analyzer, ohjelmiston varustaminen mittausinstrumenteilla

ACKNOWLEDGEMENTS

This thesis has been done in Elektrobit Wireless Oulu office during winter 2012 and spring 2013. I want to thank Elektrobit Wireless for the opportunity to do this work.

I would like to thank Mirkka Pietiläinen from Esju for valuable explanations and clarifications during the writing of this work. Many questions would still be without answers if she had not had patience to explain them to me in words of one syllable. Humble thanks go also to my supervisors Kari Jyrkkä and Antti Ropponen. Thank you for support and comments for improving the work.

Last but not least thanks go to my family. Thank you for being so understanding and flexible during my studies.

In Oulu 16th of April 2013

Leena Kuukasjärvi

CONTENTS

ABBREVIATIONS

ACK	Acknowledgement
AMC	Adaptive Modulation and Coding
API	Application Programming Interface
ARQ	Automatic Repeat Request
BMC	Broadcast/Multicast Control
CCS	Code Composer Studio
CN	Core Network
CPU	Central Processing Unit
CQI	Channel Quality Indication
CRC	Cyclic Redundancy Check
CT	Core Network & Terminals
BoD	Bandwidth on Demand
DL	Downlink
E-DCH	Enhanced Dedicated Channel
EDGE	Enhanced Data rates for GSM Evolution
ETB	Embedded Trace Buffer
ETSI	European Telecommunications Standards Institute (Europe)
EUL	Enhanced Uplink
GERAN	GSM EDGE Radio Access Networks
GPRS	General Packet Radio Service
GSM	Global System for Mobile communication

HARQ	Hybrid ARQ
HSDPA	High Speed Downlink Packet Data Access
HS-DSCH	High-Speed Downlink-Shared Channel
HSPA	High Speed Packet data Access
HS-PDSCH	High Speed-Physical DL Shared Channel
HSUPA	High Speed Uplink Packet Data Access
IDE	Integrated Development eEnvironment
IMS	IP Multimedia Subsystem
IUB	Interface between RNC and nodeB
JTAG	Joint Test Action Group
L1	Layer 1, physical layer
L2	Layer 2, MAC layer
L3	Layer 3
LTE	Long Term Evolution
MAC	Medium Access Layer
MIMO	Multiple Input Multiple Output
NACK	Negative Acknowledgement
O&M	Operation and Maintenance
PCG	Project Co-ordination Group in 3GPP
PDCP	Packet Data Convergence Protocol
QoS	Quality of Service
R99	Release 99, The first WCDMA specification
RAN	Radio Access Networks
RLC	Radio Link Controller
RNC	Radio Network Controller

RNS	Radio Network Subsystems
RRC	Radio Resource Control
Rta	Real-time analysis
RX	Receiving
SA	Service & Systems Aspects
SF	Spreading Factor
SIR	Signal-to-Interference Ratio
SoC	System on Chip
SYS/BIOS	Real-Time Operating System for TI Devices
TB	Transport Block
TFRI	Transport Format and Resource Indicator
TI	Texas Instrument
TRX	Transmitter–Receiver
TSG	Technical Specification Group in 3GPP
TTA	Telecommunications Technology Association (Korea)
TTC	Telecommunications Technology Committee (Japan)
TTI	Transmission Time Interval
TTP	Traffic Termination Points
TX	Transmitting
UART	Universal Asynchronous Receiver Transmitter
UE	User Equipment
UIA	Unified Instrumentation Architecture

UL	Uplink
UMTS	Universal Mobile Telecommunications System
UTRA	Universal Terrestrial Radio Access
UTRAN	Universal Terrestrial Radio Access Network
USB	Universal Serial Bus
WCDMA	Wide-band Code-Division Multiple Access
XDC	The eXpress DSP Components is a standard for providing reusable software components, called "packages" that are optimized for real-time embedded systems.

1 INTRODUCTION

The primary objective of this thesis was to construct a measuring system to evaluate the performance of WCDMA base station software. The thesis has been carried out for Elektrobit Wireless and in Elektrobit's premises.

A base station is a complex embedded system formed by a set of hardware components and software. The system requires hard real-time constraints which means that timing and scheduling act an important role in performance of a base station. The measurement system is needed to check that software components meet real-time constraints.

Measuring the performance of software always causes at least some interference to the system to be measured. When embedded system with hard real-time constraints like a base station is in a question, about one has to pay extra attention to interference. Measuring is not a straightforward, particularly when it concerns an embedded system. In the embedded system the interfering measurements can also change the function of a device. When developing a base station this has to be taken under consideration as well. Particularly some algorithms and processes responsible for scheduling and congestion control are challenging to measure due to strict timing constraints and because they are delicate to interference.

The reason why this measurement system was constructed was to evaluate that certain functions and algorithms of the WCDMA base station software were executed in given time limits. System was needed to be constructed in a way that measurements did not disturb the normal function of the base station. Texas Instrument's Code Composer Studio and especially the visualization tool called System Analyzer within it was selected as a measurement tool. In order to get the System Analyzer to work the base station SW was needed to be instrumented. The System Analyzer showed results of pre-defined measurement points. The overall working of the base station as well as more precise action of functions and algorithms were verified from log files provided by extra loggers besides the System Analyzer.

To get clear understanding of log files and their meaning concerning the whole picture during the construction of the system, WCDMA technology is presented the briefly in chapter 2. In the second chapter is also introduced the measurement tool System Analyzer. In chapter 3 focus is on basic structure of WCDMA base station’s SW and on measurement process and how the actual measurements are done with this measurement system. Chapter 4 depicts the constructed measurement system. This chapter also presents examples of instrumentation of SW in order to check needed characteristics of algorithms. Measurement results for evaluation of the goodness of the measurement system are also presented in this chapter. Chapter 5 concludes the thesis.

2 TECHNOLOGIES AND TOOLS

Next chapter presents basic features of WCDMA technology. The chapter also includes the brief presentation of WCDMA enhancement HSPA. At the end of the chapter is presented the measurement tool System Analyzer.

2.1 3GPP

The 3rd Generation Partnership Project (3GPP) is a collaboration between groups of telecommunications standard development organizations established in 1998 (1). The scope of 3GPP is to produce, maintain and develop globally applicable Technical Specifications and Technical Reports for a 3rd Generation Mobile System based on evolved Global System for Mobile communication (GSM) core networks and the radio access technologies that they support (2).

3GPP is responsible for the definition of the functions, requirements and interfaces of the UTRA network. 3GPP specifications are published four times a year. The term 3GPP specification covers all GSM (including GPRS and EDGE), WCDMA and LTE (including LTE-Advanced) specifications. The first full standard of UTRA from 3GPP was Release-99. (3.)

2.2 WCDMA BASE STATION SYSTEM

In order to understand the basic function of nodeB it is important to understand WCDMA technology. Without knowledge of WCDMA it is very hard to interpret the log files used to verify the function of system. Algorithms related for example to the scheduling of user data, HARQ-process, link adaptation or ciphering could not be validated with one specific test result. In most of the cases the successful functionality of a single algorithm can be confirmed by studying of log files as a whole. Because the scope of this thesis is to verify that different software algorithms and functions are executed in the given limits and in right manner, an introduction to the WCDMA technology is presented first.

Universal Mobile Telecommunications System (UMTS) is an umbrella term for the third generation radio technologies developed within 3GPP (4). In 3GPP systems the radio technology used between mobile phones and the base stations is generically known as Universal Terrestrial Radio Access (UTRA) and the access network as Universal Terrestrial Radio Access Network (UTRAN) (5).

Wide-band Code-Division Multiple Access (WCDMA) is a radio technology of UMTS. This technology was chosen as the standard technology for the UTRA air interface by ETSI in 1998. Both variants Frequency Division Duplex (FDD) and Time Division Duplex (TDD) are supported by WCDMA. The first WCDMA specifications were frozen in Release-99 (R99). (6.)

2.2.1 Characteristics of WCDMA

WCDMA is a spread-spectrum modulation technique. This technique uses channels whose bandwidth is much greater than that of the data to be transferred. Instead of each connection being granted a dedicated frequency band just wide enough to match its envisaged maximum data rate, WCDMA channels share a much larger band. User information bits are spread over a wide bandwidth by multiplying the user data with quasi-random bits called chips. Each 10 ms radio frame is divided into 15 slots and every slot includes 2560 chips (7). This leads to a standard chiprate of 3,84 Mcps which in turn leads to a carrier bandwidth of approximately 5 MHz. The carrier spacing can be selected on a 200 kHz grid between approximately 4,4 and 5 MHz, depending on interference between the carriers (figure 1.). In order to support very high bit rates (up to 2 Mbps), the use of a variable spreading factor and multicode connections is supported. The spreading is the multiplication of each user data bit with a sequence of variable amount of code bits, called chips. For example in a case of a sequence of 8 chips it is said that a spreading factor is 8. (6; 8 p.47-48.)

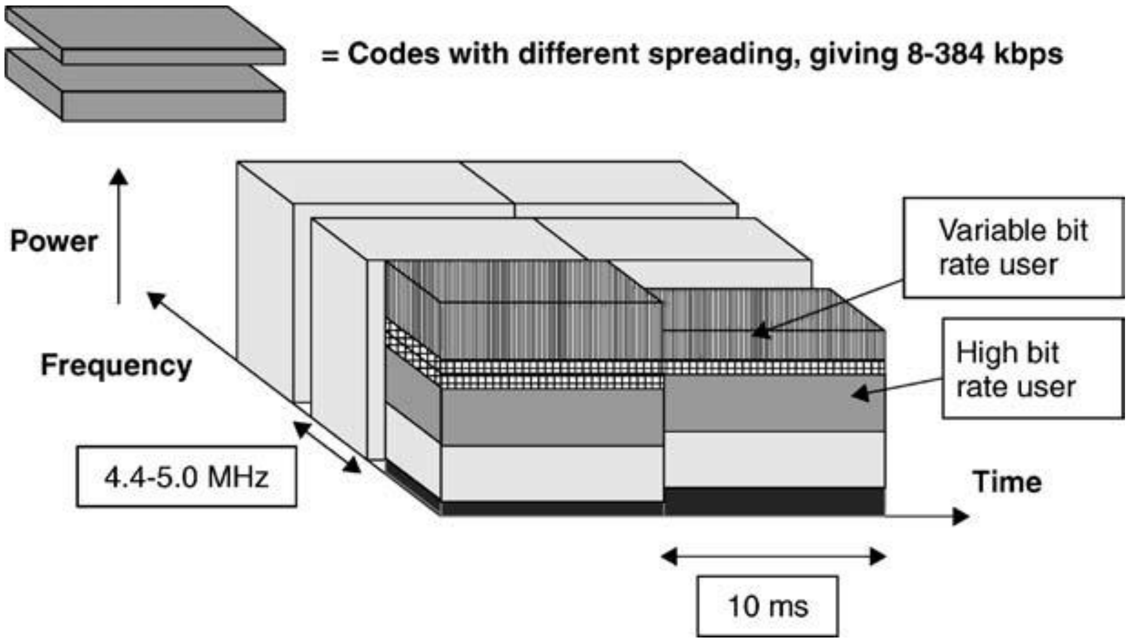


FIGURE 1. Allocation of bandwidth in WCDMA in the time-frequency-code space (8, p.48)

WCDMA supports highly variable user data rates. The concept is called obtaining Bandwidth on Demand (BoD). The user data rate is kept constant during each 10 ms time frame. The data capacity among the users can change from frame to frame. Figure 1. shows an example of this feature. (8. p.48)

WCDMA's basic modes of operations are FDD and TDD. In the FDD mode, separate 5 MHz carrier frequencies are used for the uplink and downlink respectively, whereas in TDD only one 5 MHz carrier is time-shared between the uplink and downlink. Uplink is the connection from the mobile to the base station and downlink is connection from the base station to the mobile. FDD uses frequencies of 2,110 – 2,170 GHz downlink and 1,920–1,980 GHz uplink. TDD uses a frequency band located at both sides of the FDD uplink. (8 p. 48; 9.)

Radio propagation in the land mobile channel is characterized by multiple reflections, diffractions and attenuation of the signal energy. These are caused by natural obstacles such as buildings and hills, resulting in so-called multipath propagation. The two effects resulting from this are that the signal energy may arrive at the receiver across clearly distinguishable time instants and for a certain time delay position there are usually many paths nearly equal in length along which the radio signal travels affecting fast fading. To improve detection of radio signal in WCDMA system rake receiver is used. A rake receiver is a radio receiver designed to counter the effects of multipath fading. Rake receiver uses several "sub-receivers" called fingers, which are correlation receivers each assigned to a different multipath component. (8 p.51-52; 9 p.126.)

Tight and fast power control is perhaps the most important aspect in WCDMA, in particular on the uplink. Without it, a single overpowered mobile could block a whole cell. The solution to power control in WCDMA is fast closed loop power control, shown in figure 2. In closed loop power control in the uplink, the base station performs frequent estimates of the received Signal-to-Interference Ratio (SIR) and compares it to a target SIR. If the measured SIR is higher than the target SIR, the base station will command the mobile to lower the power; if it is too low, it will command the mobile station to increase its power. This measure–command–react cycle is executed at a rate of 1500 times per second (1.5 kHz) for each mobile. (8 p.55.)

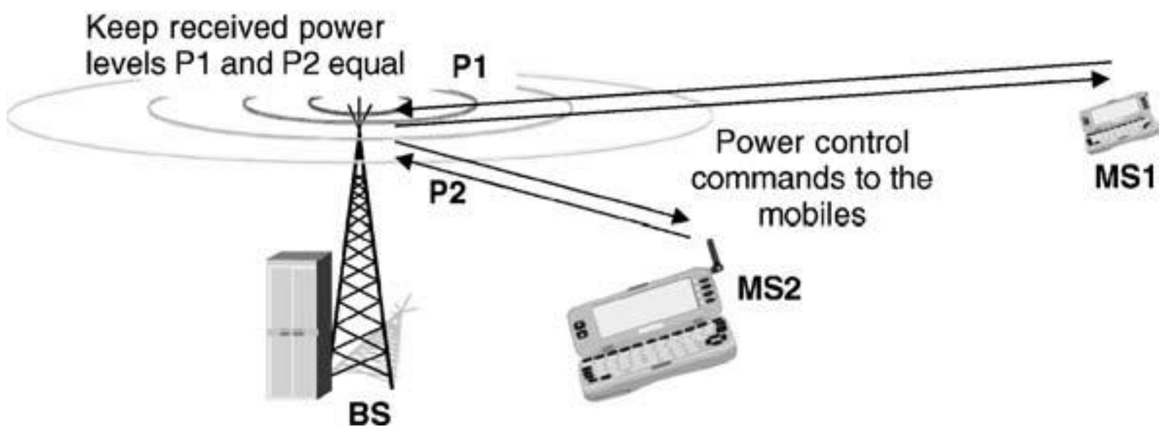


FIGURE 2. Closed loop power control in WCDMA (8 p.55)

WCDMA is designed to be deployed in conjunction with GSM. Therefore, handovers between GSM and WCDMA are supported in order to be able to leverage the GSM coverage for the introduction of WCDMA. WCDMA supports soft and softer handovers. During softer handover, a mobile is in the overlapping cell coverage area of two adjacent sectors of a base station. During soft handover, a mobile station is in the overlapping cell coverage area of two sectors belonging to different base stations (figure 3.). (8 p.57.)

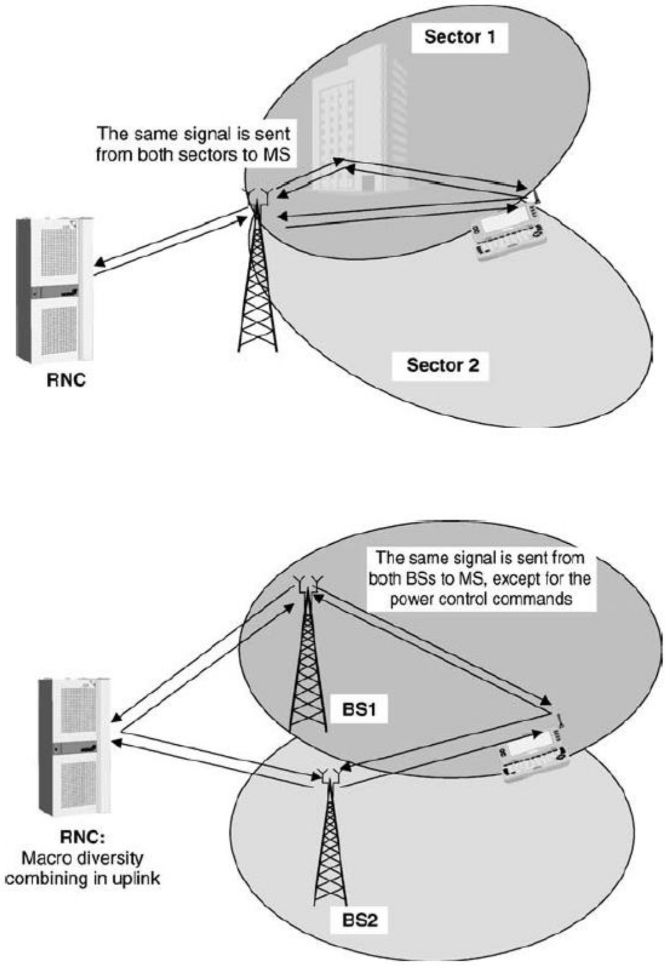


FIGURE 3. Soft and softer handover (8 p.58)

2.2.2 System architecture

The UMTS consists of a number of logical network elements that each has a defined functionality. In the standards, network elements are defined at the logical level, but this quite often results in a similar physical implementation. Functionally, the network elements are grouped into the Universal Terrestrial Radio Access Network (UTRAN) that handles all radio-related functionality, the Core Network (CN), which is responsible for switching and routing calls and data connections to external networks and the User Equipment (UE) that interfaces with the user and the radio interface. (8 p.75.)

The UTRAN consists of a set of Radio Network Sub-systems (RNS) connected to the Core Network through the Iu interface (figure 4.). A RNS consists of a Radio Network Controller (RNC) and one or more nodeBs. Inside the UTRAN, the RNCs can be interconnected together through the Iur interface. Iu and Iur are logical interfaces. Iur can be conveyed over direct physical connection between RNCs or virtual networks using any suitable transport network. (10 p.16.)

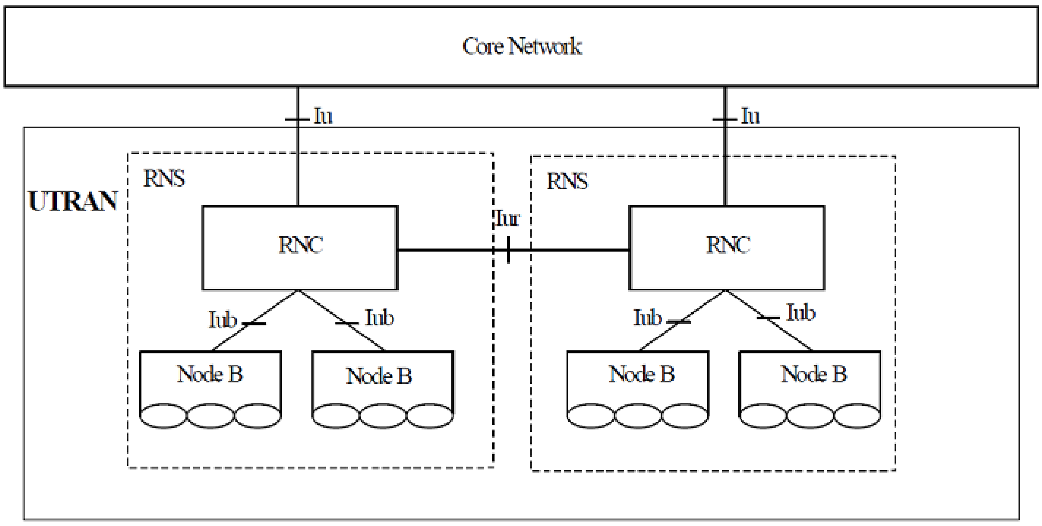


Figure 4. UTRAN architecture (10 p.17)

The nodeB, also called a base station, converts the data flow between the Iub and Uu interfaces. Uu interface is between UTRAN and UE. NodeB also participates in radio resource management. The Radio Network Controller (RNC) is the network element responsible for the control of the radio resources of UTRAN. It interfaces the Core Network (CN) and also terminates the Radio Resource Control (RRC) protocol that defines the messages and procedures between the UE and UTRAN. The RNC is the service access point for all services that UTRAN provides to the CN. (8. p.75;79)

2.2.3 Protocol architecture

All protocol structures in UTRAN interfaces are designed according to the same general protocol model. The UTRAN is layered into a Radio Network Layer and a Transport Network Layer (figure 5). The UTRAN architecture, i.e. the UTRAN logical nodes and interfaces between them, are defined as part of the Radio Network Layer. The layers and planes are logically independent of each other and, if needed, parts of the protocol structure may be changed in the future while other parts remain intact. (8 p.80-81; 10 p.40.)

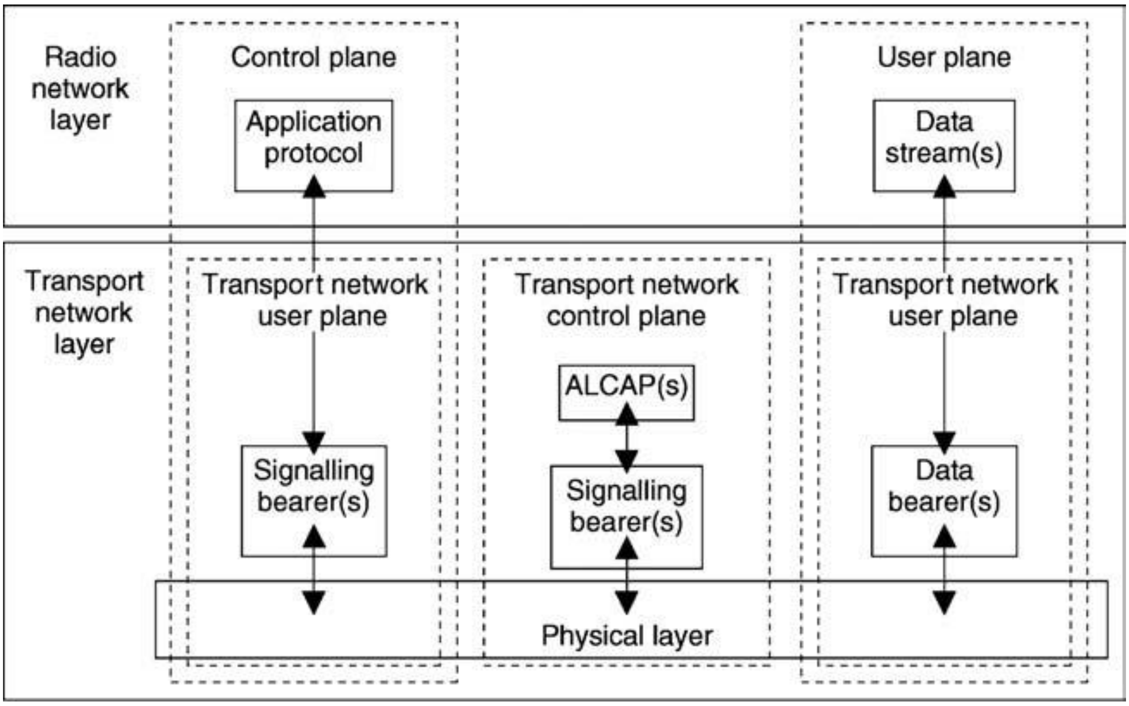


FIGURE 5. General protocol model for UTRAN terrestrial interfaces (10 p.40)

The Control Plane is used for all UMTS-specific control signaling. It includes the Application Protocol and the Signaling Bearer to transport the Application Protocol messages. The Application Protocol is used, among other things, for setting up bearers to the UE. All information sent and received by the user, such as the coded voice in a voice call or the packets in an internet connection, are transported via the User Plane. The Transport Network Control Plane is used for all control signaling within the Transport Layer. The transport network layer provides services for user plane transport, signaling transport and transport of implementation specific Operation and Maintenance (O&M). For each UTRAN interface (Iu, Iur, Iub) the related transport network layer protocol and functionality is specified. (8 p.80-81; 10 p.40.)

2.3 NodeB

The nodeB is a basic network element of UMTS and it is located between the Uu and Iub interfaces. Physically it is between UE and RNC. The nodeB's task is to perform radio signal receiving and transmitting (Rx and Tx), signal filtering and amplifying, signal modulation and demodulation, and interfacing to the UTRAN. The internal structure of the nodeB is very vendor-dependent, but basically consists of the components shown in figure 6. (9 p.101.)

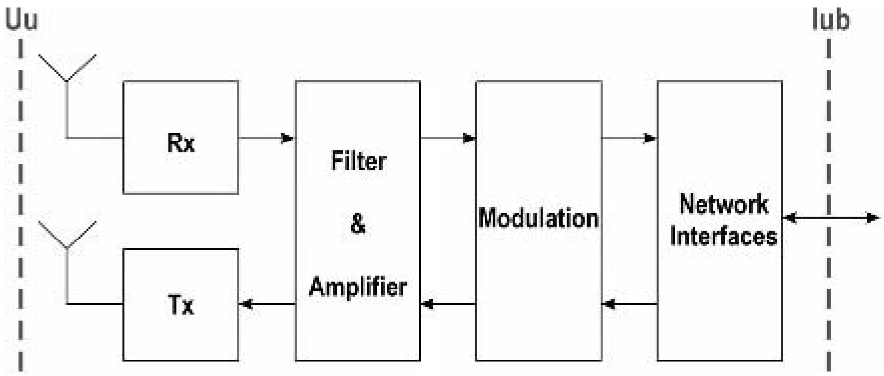


FIGURE 6. The basic structure of nodeB (9 p.102)

NodeB's logical structure is generic. From the network point of view, it can be divided into several logical entities as shown in figure 7. On the Iub side, a nodeB is a collection of two entities: common transport and a number of Traffic

Termination Points (TTPs). Common transport represents those transport channels that are common to all UE in the cell as well as those channels used for initial access. The common transport entity also contains one nodeB control port used for O&M purposes. One TTP consists of a number of nodeB communication contexts. NodeB communication contexts in turn consist of all dedicated resources required when the UE is in dedicated mode. (9 p.102.)

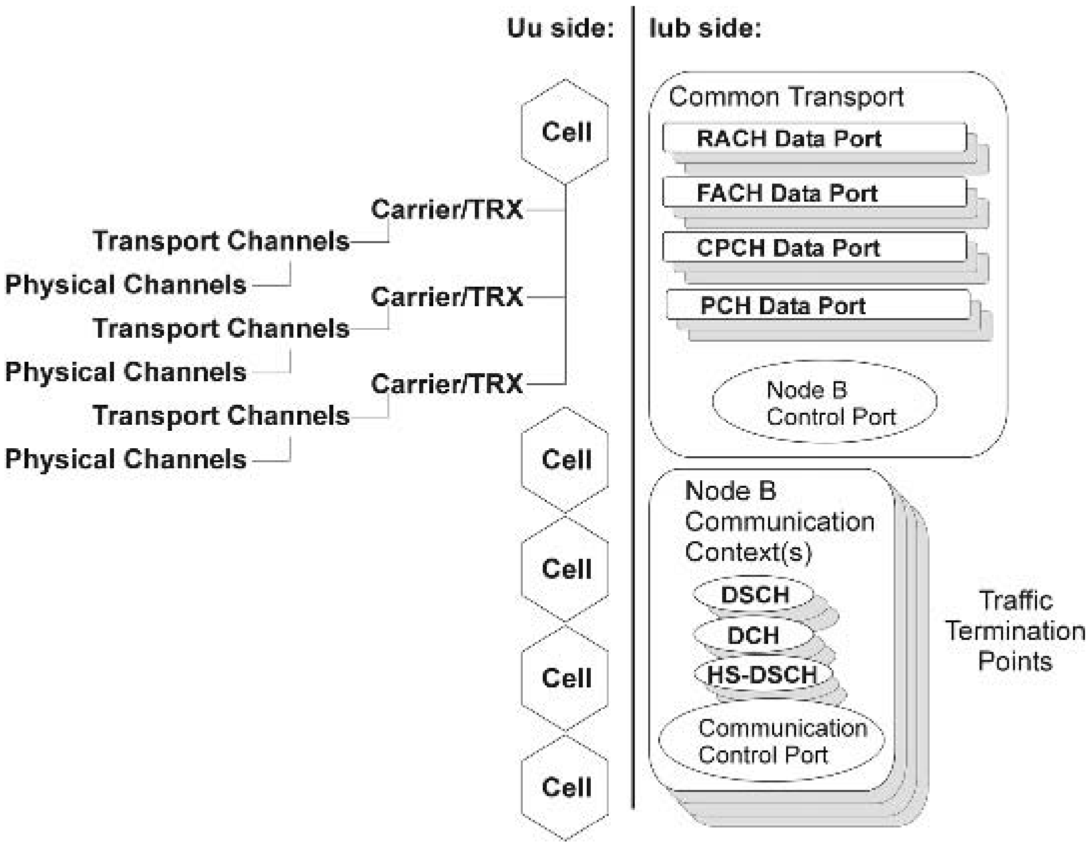


FIGURE 7. NodeB logical structure (9 p.102)

From the point of view of the radio network, the nodeB consists of several other logical entities called “cells”. One cell may have several Transmitter–Receivers (TRXs) under it. One TRX delivers physical channels through the Uu interface and these channels carry the transport channels containing actual information, which may be either common or dedicated on nature. The TRX is a physical part of the nodeB performing various functions, such as the conversion of data flows from the terrestrial Iub connection to the radio path and vice versa. (9 p.103.)

The channel organization used by WCDMA comprises three layers: logical channels, transport channels and physical channels. Logical channels describe the types of information to be transmitted, transport channels describe how the logical channels are to be transferred and physical channels are the ‘‘transmission media’’ providing the radio platform through which the information is actually transferred. (9 p.65)

One of the nodeB’s task is modulation. The used modulation method has a close relationship to overall system capacity and performance. It is up to vendor what modulation methods are used in WCDMA based nodeB. Release 99 introduced methods like QPSK and later releases introduced 16QAM and 64QAM. Also Multiple Input Multiple Output (MIMO) is used to increase bitrates. MIMO is used to increase the overall bitrate through transmission of two different data streams on two different antennas. (11.)

WCDMA utilizes multipath propagation. This requires a special type of receiver. An example of this kind of arrangement is called rake. The purpose of the rake receiver is to improve the level of the received signal by exploiting the multipath propagation characteristics. A basic rake receiver consists of a number of fingers which can receive parts of the transmitted signal and combine them to a sum signal with better quality than individual parts of the signal. (9 p.107.)

2.4 HIGH SPEED PACKET DATA ACCESS

High Speed Packet data Access (HSPA) has been an upgrade to WCDMA networks used to increase packet data performance. The introduction was done in steps. High Speed Downlink Packet data Access (HSDPA) was introduced in 3GPP Release 5 in 2002, and Enhanced Uplink (EUL), also referred to as High Speed Uplink Packet data Access (HSUPA), came in Release 6 in 2004. The combination of HSDPA and HSUPA is referred to as HSPA. HSPA as well WCDMA mobiles should be served in the same network. (11.)

2.4.1 HSDPA

The key idea of the HSDPA concept is to increase packet data throughput. In order to achieve higher throughput, high-peak rates and reduce delay, HSDPA employs such techniques as Adaptive Modulation and Coding (AMC) and Hybrid Automatic Repeat Request (HARQ) combined with a fast-scheduling and cell change procedure. The primary benefits of HSDPA, as perceived directly by the end-user, are the theoretical data throughput up to the 168 Mbps peak rate with modulation of 64QAM, 4-carrier and 2x2 MIMO (3GPP, Rel. 10) (12 p.31). These benefits depend on the modulation used for the resource configuration. Because other factors, such as cell coverage, UE mobility, UE distance from the nodeB and the number of simultaneous users, will also affect the achievable peak rate, the practical maximum peak data rate might lag far behind the target. Other highlights of HSDPA enhancements are significant lower end-to-end latency and improved cell capacity. As a trade-off, HSDPA also has its downside. Though different releases are kept backward-compatible, upgrades and enhancements to Release 99 air interface and architecture are required. (9 p.79-77.)

Theses upgrades and enhancements can be summarized as follows (9 p.77):

- Network architecture: HSDPA requires a significant part of the packet-handling function to be added closer to the radio interface, i.e. nodeB, leading to a more distributed architecture than Release 99.
- Physical layer: new adaptive modulation and coding methods pose significant modifications to the physical layer architecture in terms of channel structure, multiplexing, timing and procedures required for HSDPA operation.
- Fast scheduling means more efficient operation of Medium Access Control (MAC) and it is closer interaction with the physical layer. Short-

framing may also require more processing capacity from the nodeB and, to some extent, from the UE.

- Fast retransmission can be realized by employing more control signaling and an advanced retransmission mechanism.

What HSDPA principally does is that it simply employs a time-multiplexing approach to transfer data packets on a single shared channel while it uses a multicode with a fixed SF. This implies a certain functionality and a set of procedures to make it practical over the air interface. Multiplexed data should be efficiently scheduled, modulated, encoded and conveyed over the air interface and the radio link should be adapted. (9 p.78.)

With HSDPA two of the most fundamental features of WCDMA, fast-power control and variable spreading factor mechanisms, have been replaced by AMC, HARQ and fast-retransmission procedures (8 p.355). These main functional entities are described in the following paragraphs.

The main objective of adaptive modulation and coding (AMC) is to compensate for radio channel instability by fine-tuning the modulation and coding parameters of the physical layer. This is basically done by making use of the radio channel measurements extracted by the UE and, for HSDPA in particular, using the Channel Quality Indication (CQI) and retransmission procedure. Armed with these and traffic-related information, such as Quality of Service (QoS) and the state of radio and physical resources, AMC enables the network to select the most suitable modulation and coding methods. (9 p.78.)

Although the benefits of AMC are well known, its performance is vulnerable due to the radio channel measurements extracted by the UE. The measurements are not error-free. Therefore complementary mechanisms are needed. Hybrid Automatic Repeat Request (HARQ) helps to compensate for the vulnerability of AMC by bringing link-layer information into the process. HARQ messages are sent between nodeB and UE and they allows the nodeB to detect errors and when necessary to request retransmissions. HARQ retransmission process uses an acknowledgement mechanism and an acknowledgement message to confirm successful transmission of a data packet while avoiding retransmission. Successful transmission is reported with ACK (acknowledgement) message and failed transmission with NACK (negative acknowledgement) message. To avoid the additional delay due to waiting HARQ employs N channel to make the retransmission process parallel. Figure 8 depicts HARQ and ARQ processes. ARQ is retransmission process, relying only on error detection, on the RLC protocol layer between UE and RNC. The number of possible HARQ retransmissions can be configured. If the maximum number of HARQ retransmissions are done without successful decoding of the contents, then the ARQ procedure will take over. (9 p.79-80; 11.)

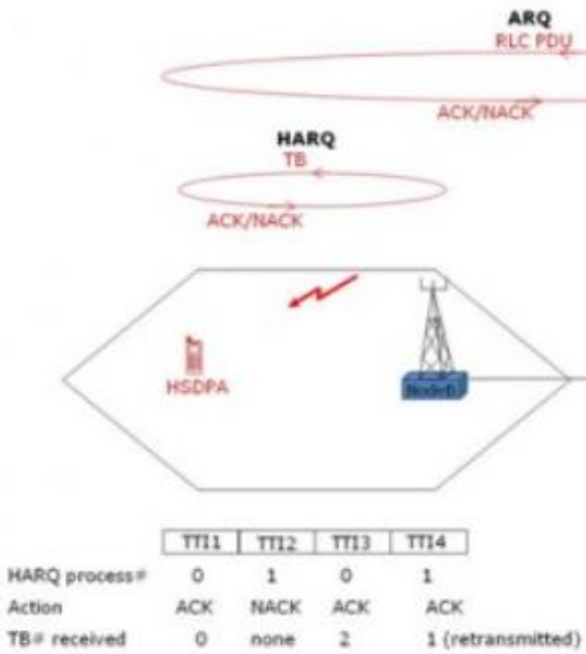


FIGURE 8. HARQ and ARQ processes (11)

The efficient operation of HSDPA with regard to AMC and HARQ implies that the packet-scheduling cycle has to be fast enough to track short-term variations in a UE fading signal. This is the main reason for having the scheduler in the nodeB rather than in the RNC. In this way the delay in the scheduling process is minimized and the radio measurements also better reflect the radio channel condition, leading to more reliable and fairer scheduling decisions. This, combined with the fixed code (=16) allocation strategy and the reduction in the Transmission Time Interval (TTI) from 10 or 20 ms to a fixed slot of 2 ms in HSDPA, enables the scheduler to undertake fast scheduling and frame formation. The implementation of scheduler is vendor-dependent. (9 p.80-81.)

Seamless cell change enables the UE to connect to the best cell available to serve it on the downlink. This will reduce undesirable interference, particularly in the case of soft handover. Cell change ensures UE mobility in association with high-speed data connection. (9 p.81.)

Figure 9 illustrates the basic procedure and functional allocation of HSDPA entities and functions.

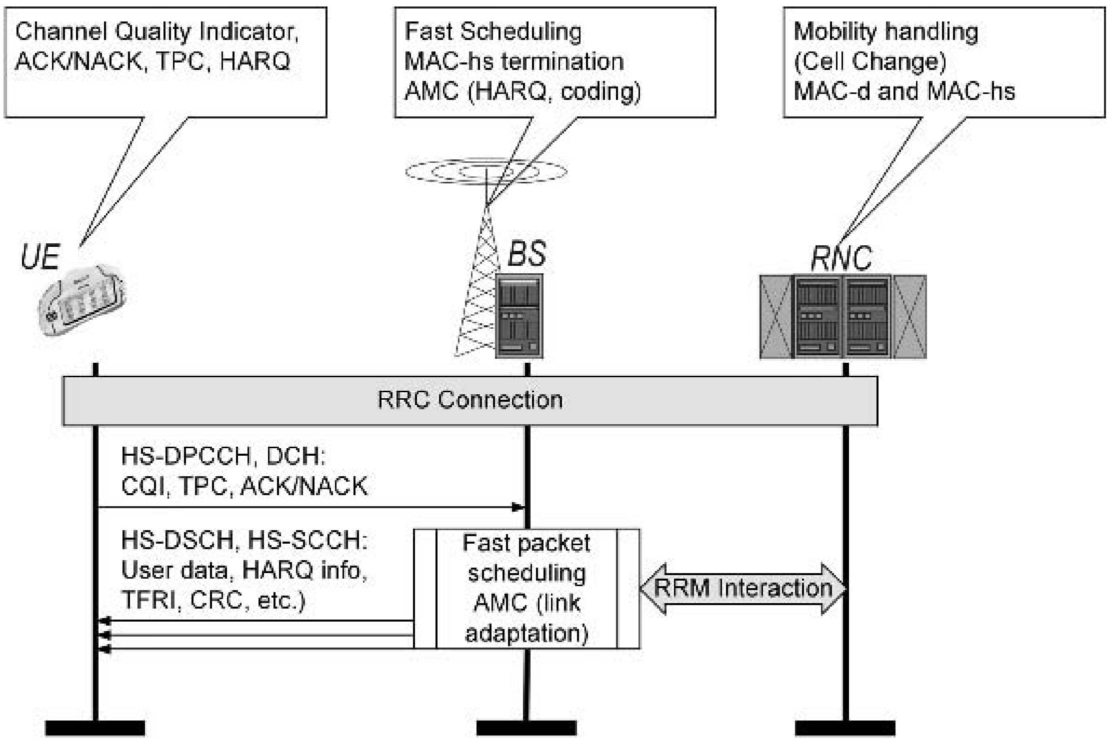


FIGURE 9. The basic procedure and functional allocation of HSDPA (9 p.83)

In a basic HSDPA operation once the RRC connection is in place the UE provides the nodeB channel-quality-related and controlling information including UE capability and requested capacity. Based on this information combined with scheduling-related information, the nodeB may choose the HS-DSCH set and parameters, modulation, etc. and start HS-SCCH transmission two timeslots prior to HS-DSCH transmission. Upon reception, the UE decodes the HS-SCCH information. Based on the information (e.g., extracted from TFRI) it will obtain the necessary parameters, such as the dynamic part of the HS-DSCH transport format, including the transport block set size and modulation scheme as well as the channel-mapping scheme in the corresponding HS-DSCH TTI. Once the UE has decoded all the necessary parameters, it can start to engage in the data processing and HARQ process and return ACK/NACK to the nodeB. After completing the process, the timing between HS-SCCH, HS-DSCH and ACK/NACK will play an essential role during the data connection and, therefore, it should be strictly followed by the UE. (9 p.82-83.)

2.4.2 HSUPA

The main idea of the HSUPA concept is to increase uplink packet data throughput with methods similar to HSDPA. While similar techniques have been applied for HSDPA, there are some fundamental differences between the uplink and the downlink. HSUPA supports several new features (15):

- Multi code transmission
- Short Transmission Time Interval
- Fast hybrid Automatic Repeat reQuest
- Fast scheduling.

In HSUPA the shared resource in the uplink is the interference at the base station. In downlink the shared resource consists of transmission power and channelization codes. The new uplink channel that is introduced for HSUPA is called Enhanced Dedicated Channel (E-DCH) and it is dedicated in nature. HSUPA operates with a TTI of 2 ms or 10 ms in the uplink. A short TTI enables significant reduction in overall latency and provides the means for the other features to adapt rapidly. (13 p.15-16.)

The fast hybrid ARQ protocol used is similar to the one used for HSDPA. Fast scheduling enables rapid resource reallocation between UEs. The scheduling algorithm is not standardized and different scheduling strategies can be implemented. (13 p.16.)

Figure 10 illustrates the basic function of HSUPA. The scheduler in nodeB tracks the instantaneous transmission needs and capabilities of each device and then allocates such a data rate really needed by the device. Tracking is based on the device-specific feedback. A UE can be scheduled from just one nodeB or from several nodeBs at the same time. (13 p.11.)

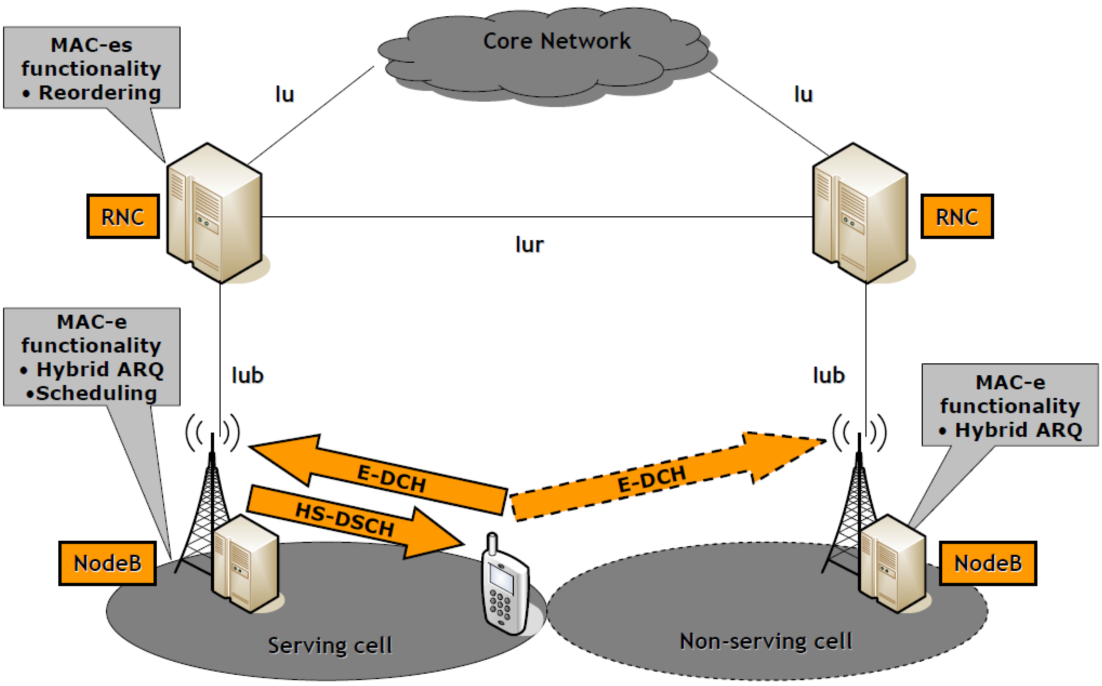


Figure 10. Basic functions of HSUPA (14)

2.5 CCS AND SYSTEM ANALYZER

Code Composer Studio (CCS) is an integrated development environment (IDE) for Texas Instruments (TI) embedded processor families. CCS consists of a suite of tools used to develop and debug embedded applications. It includes among other things compilers for each of TI's device families, source code editor, project build environment, debugger, profiler, simulators and real-time operating system. CCS is based on the Eclipse open source software framework. Eclipse offers a software framework for building software development environments and it is becoming a standard framework used by many embedded software vendors. System Analyzer within CCS is particularly of interest in this thesis. (15.)

System Analyzer is a suite of tools that provides real-time visibility into the performance and behavior of application code and allow for analysis of information that is collected from software and hardware instrumentation. System Analyzer enables benchmarking, CPU and task load monitoring, operating system execution monitoring and multi-core event correlation. Data is collected via software instrumentation using the UIA (Unified Instrumentation Architecture) target packages and can be transported via Ethernet, run-mode JTAG, stop-mode JTAG, USB/UART to the host PC for analysis. UIA is the target-side package which defines the APIs, transports, interfaces and guidelines that enables the instrumentation of embedded software. It defines an instrumentation framework that enables instrumented target content. The framework includes e.g. a messaging infrastructure that allows instrumentation clients to interact with instrumentation endpoints on a target device while the target is running and a set of commands and messages that can be sent over the messaging infrastructure. In figure 11 is depicted the basic idea of tracing with System Analyzer. (15; 16; 17)

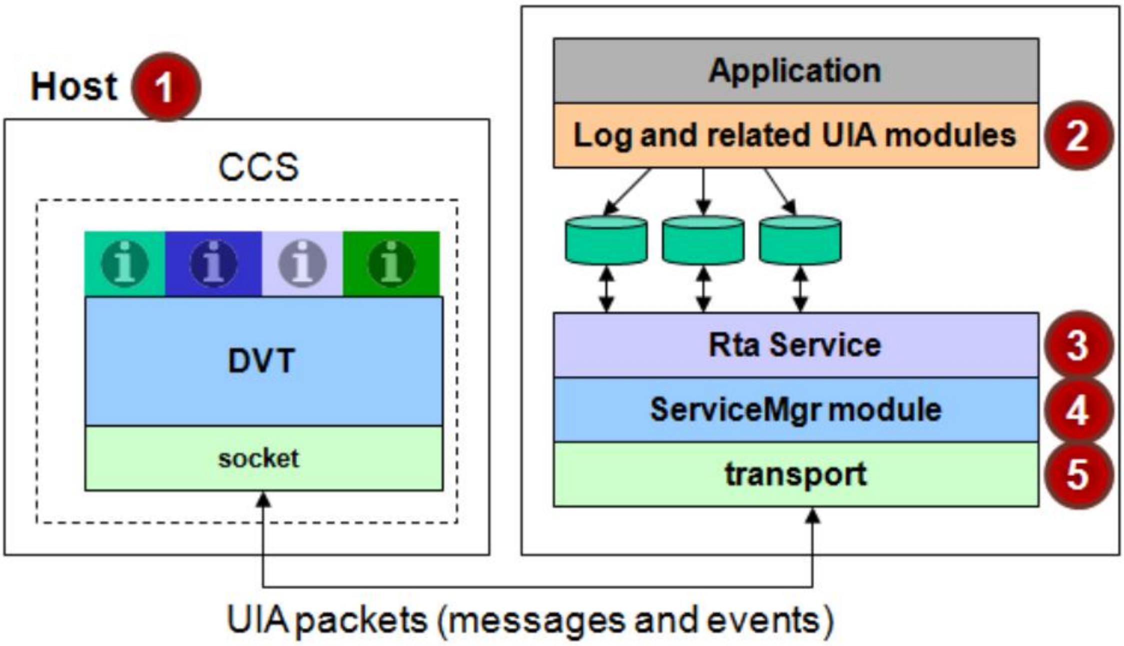


Figure 11. The basic idea of tracing with System Analyzer (16)

Number 1 in figure 11 is a host that is a PC running Code Composer. Within CCS the System Analyzer displays and analyzes the received UIA packets. System Analyzer uses TI's Data Visualization Technology (DVT) to provide analysis features. Number 2 is the target application that uses SYS/BIOS and/or XDCtools for configuration and APIs. Internally, the SYS/BIOS modules make UIA API calls to log events related to threading. Number 3 is UIA's Rta module on the target that collects events from the log written by both the pre-instrumented SYS/BIOS threads and any custom instrumentation that have been added. Rta sends events on to the UIA ServiceMgr module (number 4). UIA ServiceMgr module moves data off the target primarily in the background. Module can be configured case by case basis depending on e.g. which kind of core or what kind of transport type is in use. Number 5 depicts transport that can happen in various ways. Regardless of the transport way UIA packets are transported from the target to the host. (16.)

3 SOFTWARE OF A WCDMA NODEB

The software of nodeB consists of different levels. The radio interface is layered into three protocol layers: the physical layer (L1), the data link layer (L2) and network layer (L3). Layer 2 is split into sub-layers Medium Access Control (MAC), Radio Link Control (RLC), Packet Data Convergence Protocol (PDCP) and Broadcast/Multicast Control (BMC). Layer 3 and RLC are divided into Control and User planes. L3 is further partitioned into sub-layers in control plane. The lowest sub-layer is Radio Resource Control (RRC). The general software structure of nodeB is presented in figure 11. (18 p.9-10.)

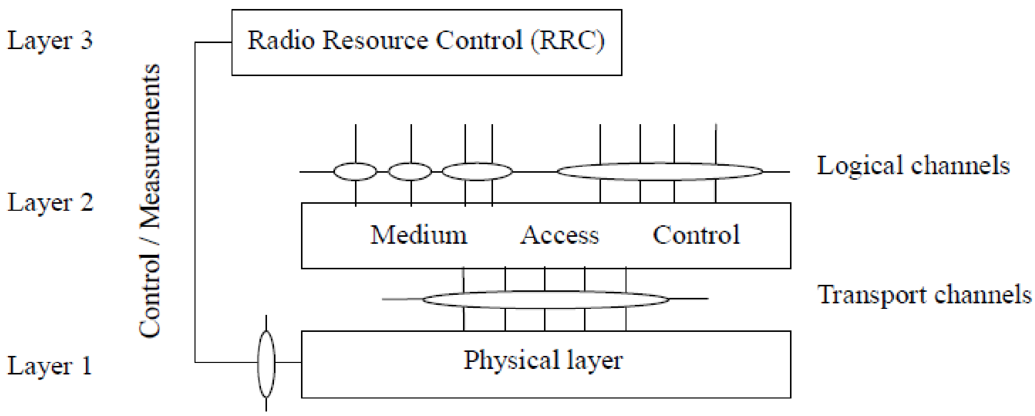


FIGURE 12. The basic software structure of nodeB (19 p.7)

3.1 Requirements of specifications

The 3GPP specifications determine the basic functions of the WCDMA nodeB. Next are presented the main functions of different layers. The physical layer offers information transfer services to MAC and higher layers. The physical layer transport services are described by how and with what characteristics data are transferred. The main functions that physical layer performs are (18 p.12-14):

- Soft handover execution.
- Error detection on transport channels and indication to higher layers.

- Multiplexing and demultiplexing of transport channels.
- Mapping of transport channels on physical channels.
- Modulation and spreading/demodulation and despreading of physical channels.
- Frequency and time (chip, bit, slot, frame) synchronization.
- Closed-loop power control.
- RF processing.

Layer 2 offers services like data transfer, reallocation of radio resources and MAC parameters and reporting of measurements to upper layers. The main functions that L2 performs are (18 p.14; 18-19):

- Mapping between logical channels and transport channels.
- Selection of appropriate Transport Format for each Transport Channel depending on instantaneous source rate.
- Priority handling.
- Identification of UEs.
- Multiplexing/demultiplexing of upper layer PDUs.
- Traffic volume measurement.
- Ciphering.
- ARQ and HARQ functionality.
- Transfer of user data.
- Error correction.
- Flow control and scheduling.
- Acknowledged and unacknowledged data transfer.

The RRC layer handles the control plane signaling of Layer 3 between the UEs and UTRAN. The main functions that RRC offers are (18 p.20-21):

- Broadcast of system information.
- Establishment, re-establishment, maintenance and release of an RRC connection between the UE and UTRAN.
- Establishment, reconfiguration and release of Radio Bearers.

- Assignment, reconfiguration and release of radio resources for the RRC connection.
- RRC connection mobility functions.
- Routing of higher layer PDUs.
- Control of requested QoS.
- Outer loop power control.

3.2 Measurement process

Many of the features of stack software of the WCDMA nodeB have an effect across different layers presented in the previous chapter. Stack software is that part of software of nodeB where the requirements of 3GPP standards and vendor specific solutions to items that relate to protocols and functions of 3GPP specifications are implemented. Stack software is divided into different layers (L1, L2 and L3) as is presented in figure 12. It happens very often when developing software that one has to add a new feature on top of existing software. Adding a piece of your own code is rarely just making one insertion in a certain spot of the stack software.

The software of WCDMA nodeB contains many different algorithms and implementations that are vendor dependent since they are not specified by the 3GPP. Scheduling, parts of link adaptation, ciphering and solutions for QoS to name something. An effective algorithm for example for scheduling user data can improve the performance of the whole nodeB and thus give an advantage in market when competing of customers. Competition of customers is the reason why it is in vendor’s interest to develop all these non-standardized vendor dependent modules. With these modules vendors differ from each other. To develop something always means testing it too. In this thesis the aim is to construct a measurement environment where the performance of functions like mentioned above can be evaluated.

The nodeB is an embedded system. It is a complex device with hard real-time constraints. The measurements of the embedded system need to be done in a way they do not interfere with the normal function of the device. When measurement statements are part of code, other developers figure out more easily what is going on as the software executes. In this thesis visibility into the system was made by instrumenting software with print statements. Visibility into the operation of the software at runtime created possibilities to evaluate the functions of different software algorithms. If function under measurement did not meet the requirements, it was possible to divide it to smaller parts and investigate it further why the requirements were not met. Most of the measurements were done in L2 and the used tool was System Analyzer.

Measurement process used in this thesis followed the pattern:

- Adding target configuration for enabling UIA logging.
- Adding benchmark code to target application.
- Building, loading and running application.
- Checking the log files for conclusions.

Target configuration means that necessary configurations are done in target's configuration file. Configuration enables using of UIA modules needed with System Analyzer and it also determines e.g. how logged events are uploaded to the host and which clock frequency of CPU is set. Benchmark code is added to the target application, to stack software. What kind of events are added to the code, depends on which analysis features are used. Possible types of analysis features are e.g. count analysis, content aware profile and duration analysis. An example is presented in the next chapter.

After instrumentation the made changes need to be built. When the build is ready certain outputs are needed as an input for System Analyzer configuration file. Once again the needed outputs depend on what type of analysis is performed. Loading means that output files are loaded to the application and running that loaded files are put into operation. Data loggers are started at the same time as the application.

After the application has run requisite amount of time, the log files are checked. According to these files the overall function of software is determined. Detailed information is verified from data provided by System Analyzer.

4 MEASUREMENT SYSTEM AND MEASUREMENTS

4.1 Construction of Measurement System

When a measurement system is constructed, it needs to be taken into consideration that measuring itself will not affect the measurement results or at least the impact should be as small as possible. The embedded trace buffer (ETB) is one solution for this. ETB is a buffer area on chip where the UIA trace information is stored and read from the chip later at a slower rate. It is a circular memory buffer and the trace information is stored there in a compressed mode. The size of the buffer depends on the chip implementation. The utilization of buffer can be determined by software. With ETB the state of target can be monitored without disturbing the normal functionality of system on chip (SoC). ETB is a separate module and thus will practically impact nothing to the function of SoC. Figure 13 depicts the architecture of tracing on very high level. (20.)

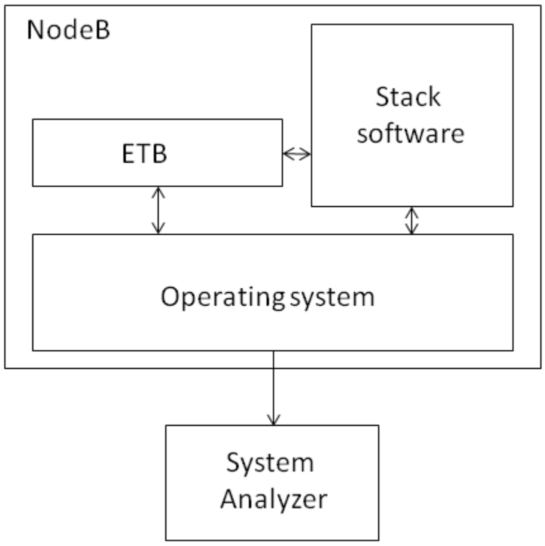


FIGURE 13. Architecture of tracing on high level

ETB communicates with stack software and operating system (OS) of nodeB. Tracing information is transported to the analyzer via OS. The tool used for actual measuring in this work was System Analyzer carried within CCS. System

Analyzer enables visibility into the real-time performance and behavior of software running on TI's embedded devices (21 p.8). In figure 14 the overall structure of measurement system is depicted.

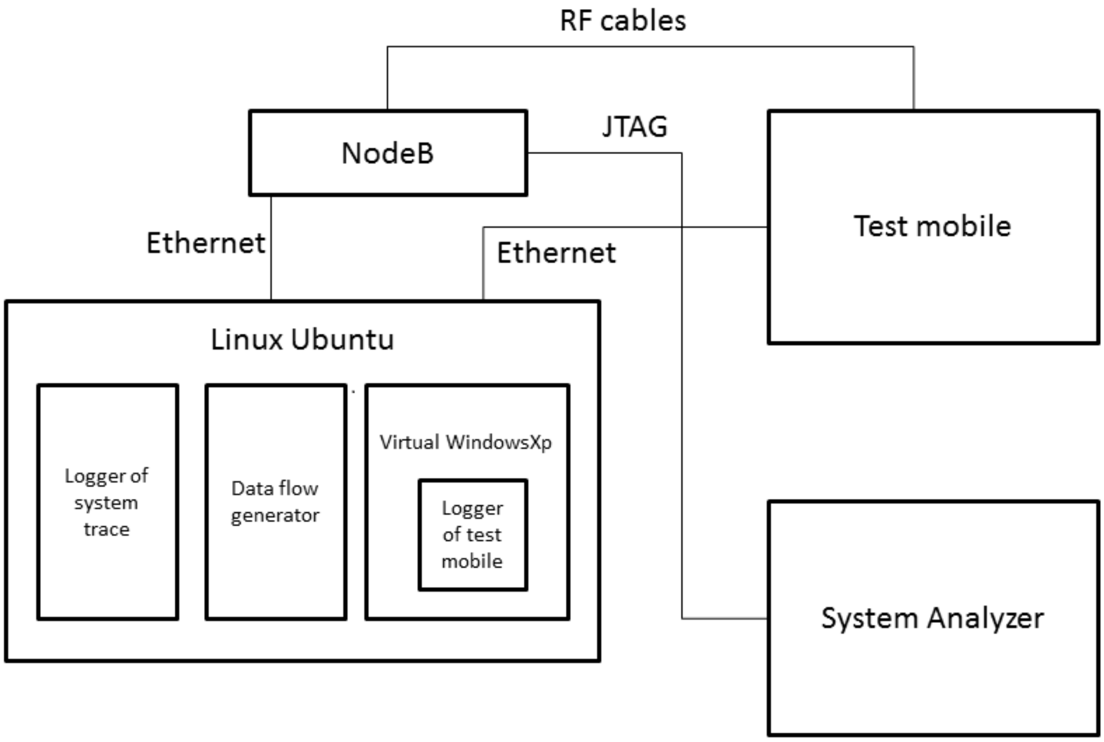


FIGURE 14. Basic structure of measurement system

The nodeB is in a scope of this system. Base station cannot operate on its own but it needs other network equipment like RNC and mobile phones to run in an appropriate way. Different kinds of logging and tracing tools are also needed. The measurement system without any logging and tracing abilities will not give much information about the function of a device.

The constructed measurement system in this thesis consists of nobeB itself, test mobile, data flow generator and a data-capturer System Analyzer. NodeB is commanded via Linux Ubuntu and the communication is handled using telnet protocol via Ethernet cable. Test mobile is commanded through virtual Windows XP running on the same Linux Ubuntu than nodeB is commanded with. The communication between test mobile and Windows XP is handled also via

Ethernet with a special application designed for this purpose. Application contains data logger that captures information from test mobile’s side of network.

The nodeB is “a pipe” for user data to go through in the wireless communication. Its function is to offer an interface between wired telecommunication system and wireless user equipment. NodeB does not provide user data to be transferred. It processes data into suitable form for transmission. The data that measurement system needs is created in an artificial way. Data flow is generated by data flow generator application running on Linux Ubuntu. Data flow can be modified with scripts used by generator application. The purpose of the generator is to create data which is transported by nodeB to the test mobile.

The measurement tool, System Analyzer, is connected to the nodeB with JTAG. Analyzer captures the data events instrumented in the code without disturbing the execution of program. In capturing EMT is used. This ensures that actual program behavior can be observed. The captured data can be analyzed later with visualizing tools of System Analyzer. Overall function of the nodeB can be monitored from log files produced by a logger running on Linux Ubuntu.

The features under investigation are accomplished with software. Because base station is an embedded device hardware always has its own share. To study the characteristics of nodeB it is important to know what kind of data goes in and what kind of data comes out, how long the transmission takes and how much data goes through. Certain features can only be verified with the system containing real or simulated network elements necessary for the use of nodeB. Figure 15 is a photo of a real measurement system in a laboratory.

FIGURE 15. Measurement system in laboratory environment

Inside the upper circle is the nodeB under development and inside the circle on the right is the test mobile. The System Analyzer is running on the laptop. NodeB is commaned with the Pc on the left. With that Pc the test mobile is also commended and logfiles from different sources are collected.

4.2 Example of instrumentation for measurement tool

To log different types of events with the System Analyzer the target code needs to be instrumented with APIs proprietary for Analyzer. By default the UIA configuration itself provides some basic instrumentation data to be sent to the System Analyzer. This requires no target code. Additional instrumentation requires more configuration to the configure file and extra C code to the target application.

UIA provides data to be analyzed when the necessary configuration is done and the code added to target. The target-side APIs and events that are used to instrument application code are provided by UIA packages. UIA event definitions are placed in ’Events’ package, DSP core UIA APIs are placed in ‘Runtime’ package and UIA APIs that depend on SYS/BIOS are placed in ‘Sysbios’ package. The LoggingSetup module from ‘Sysbios’ package automates the process of configuring an application to use UIA events and configures SYS/BIOS modules to capture user-specified information such as CPU Load, Task Load and Task Execution so that it can be displayed by the System Analyzer. It also automates the creation of infrastructure modules such as loggers and upload of the events over a user-specified transport. (22.)

Next the needed configuration and target code is presented in the situation where amount of time spent between two points in program execution can be measured.

Configuring an application to measure times between different points in code requires following statements to XDC-configuration file:

```
var LoggingSetup = xdc.useModule('ti.uia.sysbios.LoggingSetup');
LoggingSetup.eventUploadMode =
LoggingSetup.UploadMode_JTAGRUNMODE
var UIABenchmark = xdc.useModule('ti.uia.events.UIABenchmark');
var Diags = xdc.useModule('xdc.runtime.Diags');
var Defaults = xdc.useModule('xdc.runtime.Defaults');
Defaults.common$.diags_ANALYSIS = Diags.ALWAYS_ON;
```

The LoggingSetup module automates the process of configuring an application to use UIA events and configures SYS/BIOS modules to capture user-specified information. LoggingSetup module is enabled with statement *var LoggingSetup = xdc.useModule('ti.uia.sysbios.LoggingSetup').* SYS/BIOS load and task modules event logging is enabled in order to allow the System Analyzer to display CPU load, task load, and task execution information. By using LoggingSetup module all earlier mentioned information is enabled by default. The LoggingSetup module also creates logger instances needed by UIA and assigns those loggers to the other modules enabling them to provide UIA data.

The Event upload mode is configured for JTAGRUNMODE. This allows events to be uploaded in real-time via the JTAG transport while the target is running.

The UIABenchmark module defines events that that can be used to analyze the performance of the software (processing time, latency, etc.). The generation of UIABenchmark events is controlled by a module's diagnostics mask. Every high level module has a diagnostics mask that allows clients to enable or disable diagnostics on a per module basis. The Diags module manages a module's diagnostics mask. UIABenchmark events are generated only when the Diags.ANALYSIS bit is set in the module's diagnostics mask. By default the LoggingSetup module sets the ANALYSIS bit to on for the Main module. In this example the generation of events is assigned on as a default.

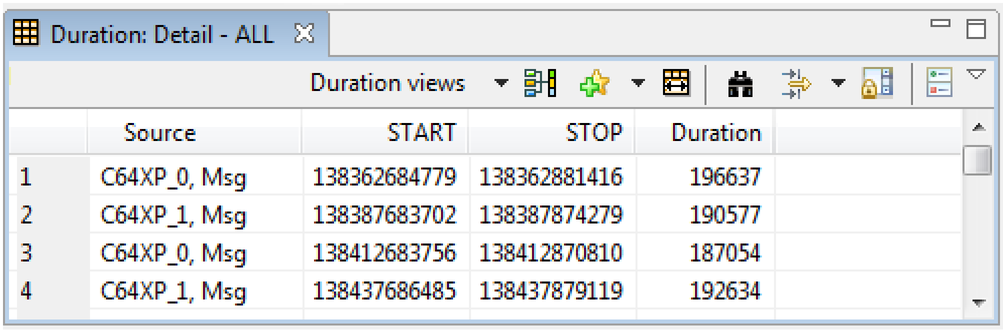
The Duration feature matches pairs of UIABenchmark_start and UIABenchmark_stop events in target code. The following C code is needed to measure duration between two different points. C code is added to the target code in a place where measurements are performed.

```
#include <xdc/runtime/Log.h>
#include <ti/uia/events/UIABenchmark.h>

Log_write1(UIABenchmark_start, (IArg)"Msg");
….
Here comes something to be measured.
….
Log_write1(UIABenchmark_stop, (IArg)"Msg");
```

Log_write1() function is provided by the xdc.runtime.Log module. It expects an event as a first argument. These events are sent to the host via calls to Log_write1(). The ti.uia.events package contains a number of modules that define events that can be passed to the Log_write1() function. One of these modules is UIABenchmark. UIABenchmark includes start and stop events that can be used to identify the instances of the function being benchmarked.

IArg is a XDC specific data type (23 p.2-12). Its counterpart in C language is intptr_t. “Msg" is identification to System Analyzer to combine a certain start and stop events. It is also a message that is conveyed to the Analyzer. In figure 16 is shown results of measurements of duration feature in System Analyzer. In source column can be seen the message “Msg”. C64XP_0 and C64XP_1 are different cores where same measurement (Msg) was done. Measurement was done twice for both of cores. Every measurement shows time instant when measurement started and stopped. And in duration column time between starting and stopping instants is calculated.



	Source	START	STOP	Duration
1	C64XP_0, Msg	138362684779	138362881416	196637
2	C64XP_1, Msg	138387683702	138387874279	190577
3	C64XP_0, Msg	138412683756	138412870810	187054
4	C64XP_1, Msg	138437686485	138437879119	192634

FIGURE 16. System Analyzer results of duration measurements

4.3 Measurements

The goodness of measurement system needs to be assessed. There are many ways to do that. Because we were interested in software execution times we decided to measure known TTI.

A basic unit of data exchanged between L1 and MAC layer is transport block (TB). A set of transport blocks exchanged between L1 and MAC at the same time interval is called transport block set (TBS). Transmission time interval (TTI) is the duration at which a TBS is transferred. TTI can be 2, 10, 20, 40, 80 ms. (11; 24.)

In the software of a nodeB, there are different tasks and processes that are done periodically in every TTI. Into one of these tasks Benchmark points was put in order to measure the 2 ms TTI itself. With TTI measure it was able to confirm if the measurement system worked properly. In other words it was able to check if the time measured really was the TTI as it should be.

Figure 17 depicts TTIs captured with the measurement system. The figure proves that measurement system actually measures what it should measure. It impacts no effects to the measurements. As can be seen from values in the figure, measurements are approximately normally distributed. The average of values is 1,999983 ms and standard deviation is 0,000115 ms. Sample size was 109 measurements.

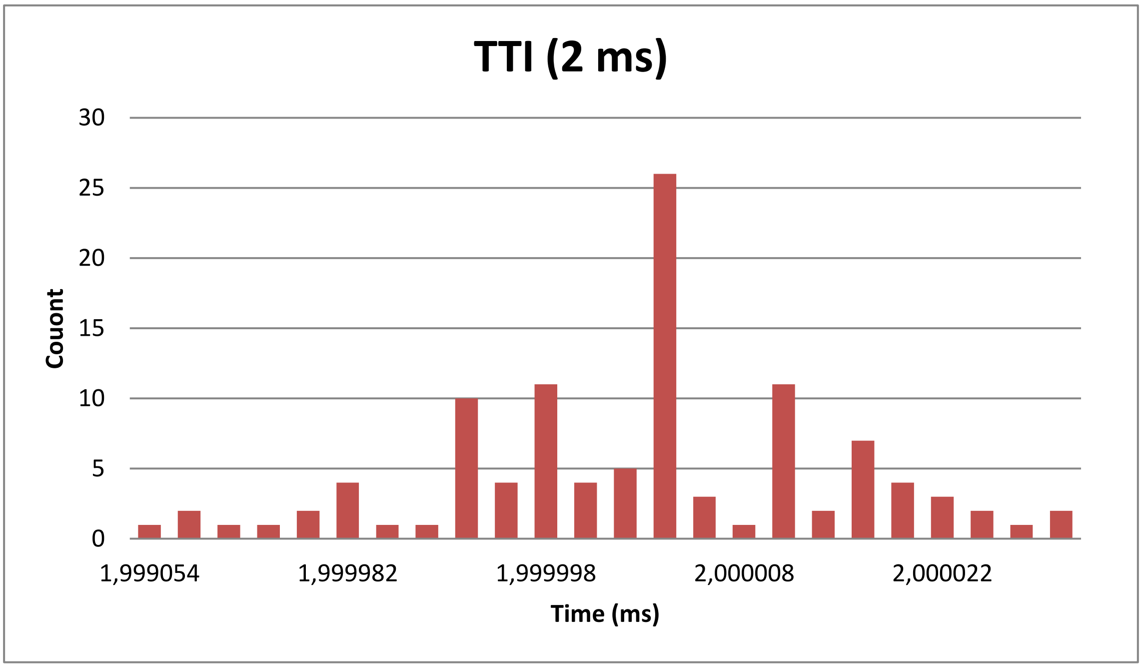


FIGURE 17. Measured durations of 2 ms TTI

The measured TTI is from process that is not on a highest priority level. The priority level of process or function of embedded software determines when the process gets runtime. The higher the priority level is, the less the process has to wait for runtime. The received results can be considered as a good evidence that measurement system has only minor impact or practically no impact at all to the functionality of nodeB. The received results also confirm that measurement from system can be trusted. Measured times are real times measured in nano seconds not only comparable times between each other in some artificial unit.

5 CONCLUSION

The main target of this thesis was to construct a measurement system that can be used to investigate the performance of software of a base station. The main reason why the system was needed is caused by the demand of software development. During software development it is essential to test the code along the evolution work. And when embedded software is dealt with, like base station's software, the testing in real and simulated environment is needed.

Development work of different algorithms is good to be done in phases. When one phase is done it is implemented to the whole software and tested. The measurement system produced in this thesis was done to help implementation work. This system allows the software developer add an benchmark point into the code and check the different features of implemented algorithms. The system was constructed mainly for the purposes of software developers.

Measurement system has no value if it is not proved to work properly. The goodness of measurement system was evaluated by measuring how long it last to perform a certain piece of code. In base station software there are different processes that are periodic. By adding benchmark points to one of these processes and measuring its duration it was possible to confirm if the system was working in a wanted way. By reviewing results received from the System Analyzer it was possible to make a conclusion that measurement system did not interfere with the function of base station. The measurement results of the measured TTI were not exactly 2 ms, but the difference was so little that it might be random in nature. The used hardware was a prototype and not the final product so it might have some effect on the results as well. Over all the gained results showed that the measurement system was good enough to be used in software development work.

As my daily work at Elektrobit I have constructed and evaluated this measurement system. After this system was done I have performed different measurements concerning different algorithms and tasks with it. The daily work contains not only making measurement but also checking from log files that the wanted functionalities happen and that they happen in right order. Log files are provided by several different components of the measurement system. The structure of my thesis followed the pattern how I have done the work on daily bases. First I had to learn the basics of WCDMA technology to understand something from log files. After that I explored different choices for measurement tool and then learned how to use selected one. In this way I discovered that a base station is a very complicated device and its thorough understanding requires lots of experience.

REFERENCES

1. 3GPP - Third Generation Partnership Project. 2011. ETSI. Available: http://www.etsi.org/website/aboutetsi/globalrole/3gpp.aspx. Site accessed on 26.11.2012.

2. About 3GPP. 2013. 3GPP. Available: http://www.3gpp.org/About-3GPP. Site accessed on 03.01.2013.

3. 3GPP - Spesifications. 2013. 3GPP. Available: http://www.3gpp.org/specifications. Site accessed on 03.01.2013.

4. UMTS. 3GPP. 2013. Site accessed on 03.01.2013.http://www.3gpp.org/Technologies/Keywords-Acronyms/article/umts. Site accessed on 03.01.2013.

5. UTRA – UTRAN. 2012. ETSI. Available: http://www.etsi.org/technologies-clusters/technologies/mobile/utran. Site accessed on 03.01.2013.

6. W-CDMA. 3GPP. 2013. Available: http://www.3gpp.org/Technologies/Keywords-Acronyms/article/w-cdma. Site accessed on 13.1.2013.

7. UMTS. 2012. ETSI. Available: http://www.etsi.org/technologies-clusters/technologies/mobile/umts. Site accessed on 03.01.2013.

8. Holma Harri & Toskala Antti. 2010. WCDMA for UMTS – HSPA Evolution and LTE, Fifth edition. John Wiley & Sons.

9. Kaaranen Heikki, Ahtiainen Ari, Laitinen Lauri, Naghan Siamäk, Niemi Valtteri, 2005. UMTS Networks: Architecture, Mobility and Services. Second edition. John Wiley & Sons.

10. 3GPP TS 25.401 version 9.1.1 Release 9; Universal Mobile Telecommunications System (UMTS); UTRAN overall description. 2011. Available: http://www.etsi.org/deliver/etsi_ts/125400_125499/125401/09.01.01_60/ts_125401v090101p.pdf. Site accessed on 30.1.2013.

11. HSPA. 3GPP. 2013. Available: http://www.3gpp.org/HSPA. Site accessed on 8.2.2013.

12. White paper. 2011. The Evolution of HSPA – The 3GPP standards progress for fast mobile broadband using HSPA+. 4g Americas. Available: http://www.4gamericas.org/documents/4G%20Americas%20White%20Paper_The%20Evolution%20of%20HSPA_October%202011x.pdf. Site accessed: 16.4.2013.

13. White Paper. 2007. Basic Concept of HSPA. Ericsson AB. Available: http://netlab.ulusofona.pt/cr/HSPA-Concepts.pdf. Site accessed: 22.2.2013.

14. Marcin Mikuszewski. HSDPA & EUL. Available: http://kolo.kt.agh.edu.pl/wp-content/uploads/2010/11/HSDPA-_-EUL.pdf. Site accessed: 22.2.2013.

15. Code Composer Studio (CCStudio) Integrated Development Environment (IDE) v5. Texas Instruments. Available: http://www.ti.com/tool/ccstudio. Site accessed: 1.3.2013.

16. Multicore System Analyzer. Texas Instruments. Available: http://processors.wiki.ti.com/index.php/Multicore_System_Analyzer. Site accessed: 4.3.2013.

17. UIA. Texas Instruments. Available: http://processors.wiki.ti.com/index.php/Category:UIA. Site accessed: 12.3.2013.

18. 3GPP TS 25.301 version 11.0.0 Release 11; Universal Mobile Telecommunications System (UMTS); Radio interface protocol architecture.

2012. Available: http://www.etsi.org/deliver/etsi_ts/125300_125399/125301/11.00.00_60/ts_125301v110000p.pdf. Site accessed on 4.3.2013

19. 3GPP TS 25.201 version 11.1.1 Release 11; Universal Mobile Telecommunications System (UMTS); Physical layer - general description. 2013. Available: http://www.etsi.org/deliver/etsi_ts/125200_125299/125201/11.01.00_60/ts_125201v110100p.pdf. Site accessed on 4.3.2013.

20. Embedded Trace Buffer. Texas Instruments. http://processors.wiki.ti.com/index.php/Embedded_Trace_Buffer. Site accessed : 2.4.2013.

21. System Analyzer 1.1 User’s Guide. 2012. Texas Instruments. Literature Number: SPRUH43D. Available: http://processors.wiki.ti.com/images/b/bc/MCSA1.1_User_Guide.pdf. Site accessed on 5.3.2013.

22. System Analyzer (UIA Target) 1.02.00.07. Cdoc. Available: http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/uia/1_02_00_07/exports/docs/docs/cdoc/index.html. Site accessed on 12.3.2013.

23. XDC Consumer User’s Guide. 2007. Texas Instruments. Literature Number: SPRUEX4. Available: http://www.ti.com/lit/ug/spruex4/spruex4.pdf. Site accessed: 15.3.2013.

24. Chockalingam A. WCDMA Physical Layer Design. Indian Institute os Science, Bangalore-12. Available : http://www.google.fi/url?sa=t&rct=j&q=&esrc=s&source=web&cd=5&ved=0CFEQFjAE&url=http%3A%2F%2Fwww.pal.ece.iisc.ernet.in%2FPAM%2Fdocs%2Fworkshops%2Fwork03pres%2Fiisc-drdo-ac-talk3.ppt&ei=2C9IUaTYCOOs4ASanoCYCQ&usg=AFQjCNG1yXnQ7NcHnu

HQTBcyRMKU2zuizQ&bvm=bv.43828540,d.bGE. Site accessed : 19.3.2013.