Implementation of Random Access procedure in 5G NR

Internship report submitted in partial fulfilment of the requirements for the degree of B.Tech. + M.Tech ESD

by

Pranav Kumar .S
(Roll No: ESD15I010)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
INDIAN INSTITUTE OF INFORMATION TECHNOLOGY,
DESIGN AND MANUFACTURING, KANCHEEPURAM

October 2019

Certificate

I, **Pranav Kumar.S**, with Roll No: **ESD15I010** hereby declare that the material presented in the Project Report titled **Implementation of Random Access procedure in 5G NR** represents original work carried out by me in **5G Testbed Lab** at the **Indian Institute of Technology, Madras** during the period from **6th May - 6th October 2019**. With my signature, I certify that:

- I have not manipulated any of the data or results.
- I have not committed any plagiarism of intellectual property. I have clearly indicated and referenced the contributions of others.
- I have explicitly acknowledged all collaborative research and discussions.
- I have understood that any false claim will result in severe disciplinary action.
- I have understood that the work may be screened for any form of academic misconduct.

Date: Student's Signature

In my capacity as supervisor of the above-mentioned work, I certify that the work presented in this Report is carried out under my supervision, and is worthy of consideration for the requirements of internship work during the period 6th May 2019 to 6th October 2019.

Advisor's Name: Advisor's Signature

Abstract

The technological advancement of the modern world has enabled continuous connectivity and progress. A key enabler in this achievement is mobile technology. The goal of mobile technologies until the fourth generation(4G) was focused on improving communication between humans. But the definition of the fifth generation of mobile technology 5G encompasses a much broader concept of communication between 'things' such as Vehicle-to-Vehicle (V2V) communication, Machine-to-Machine (M2M) communication.

5G has three major use cases : enhanced mobile broadband (eMBB), massive machine-type communication (mMTC), and ultra-reliable and low-latency communication (URLLC). The radio access interface of 5G called Next Generation Radio Access Network (NG-RAN) has been re-designed keeping in mind the achievability of the use cases and at the same time has been designed to maintain backward compatibility with the legacy LTE Core Network, known as Evolved Packet Core (EPC).

In the design of NR protocols, the initial handshaking process to ensure synchronization between transmitter and receiver occupies a central role. The set of procedures by which devices request access to the network constitute random access procedure and it is one of the foremost considerations in designing a communication system.

The work revolves around the study and implementation of random access procedure, more specifically a component called preamble assignment. In order to understand the design of NR random access procedure, LTE random access procedure was studied and reasons behind the incorporation of new features in NR were understood.

Acknowledgements

I would extend my sincerest gratitude to the Indian Institute of Technology Madras and Dr.Krishna Jagannathan for giving me an opportunity to work for the 5G testbed lab as an intern for completing the five months mandatory internship.

I wish to express my thanks to Dr. Maheshwaran P who supported me for the successful completion of my internship during this tenure.

I would like to express to my gratitude to the people who helped me directly and indirectly to complete this internship successfully.

I thank all the team members of RACH for making me feel like a part of their team. I would personally like to thank all of them for their continued support and guidance.

I am grateful to my college, Indian Institute of Information Technology Design and Manufacturing, Kancheepuram for allowing me to do an internship at 5G testbed lab.

Contents

List of Figures

Abbreviations

C-RNTI	**C**ell **R**adio **N**etwork **T**emporary **I**dentifier
CBRA	**C**ontention **B**ased **R**andom **A**ccess
CFRA	**C**ontention **F**ree **R**andom **A**ccess
CSI-RS	**C**hannel **S**tate **I**nformation **R**eference **S**ignal
gNB	Next **g**enerationNode **B**
IE	**I**nformation **E**lement
MAC	**M**edium **A**ccess **C**ontrol
OFDM	**O**rthogonal **F**requency **D**ivision **M**ultiplexing
PRACH	**P**hysical **R**andom **A**ccess **CH**annel
RA-RNTI	**R**andom **A**ccess textbfRadio **N**etwork **T**emporary **I**dentifier
RACH	**R**andom **A**ccess **CH**annel
RAR	**R**andom **A**ccess **R**esponse
RNTI	**R**adio **N**etwork **T**emporary **I**dentifier
RRC	**R**adio **R**esource **C**ontrol
SSB	**S**ynchronization **S**ignal **B**lock
TC-RNTI	**T**emporary **C**ell textbfRadio **N**etwork **T**emporary **I**dentifier
TS	**T**echnical **S**pecifications
UE	**U**ser **E**quipment

Chapter 1

Introduction

The fifth generation of mobile communication technologies is under development, keeping in mind the ever growing demand for higher data rates. Understanding the high-level architecture of 5G is necessary for the work. The overall system can be broken down into two components :

- Core Network (CN)
- Radio Access Network (RAN)

Core Network (CN) is a part of the network that provides functionalities such as authentication, charging functionality and end-to-end connection setup.

Radio Access Network(RAN) is the part of the network that is responsible for all radio access related functionalities including connection establishment, routing user plane and control plane data, retransmission protocols, radio-resource handling and QoS flow management.

A device endowed with communication capabilities that wants to access the resources of a network is termed as an **User Equipment(UE)**. The entity in RAN which is responsible for connecting the UE to the network and providing service is **Next generation Node B (gNB)**.

The protocol stack can be split into user-plane stack and control-plane stack, depending upon the functionalities provided by the modules. The physical layer (PHY) constitutes the **L1** stack, whereas SDAP, PDCP, RLC and MAC modules constitute **L2** stack.

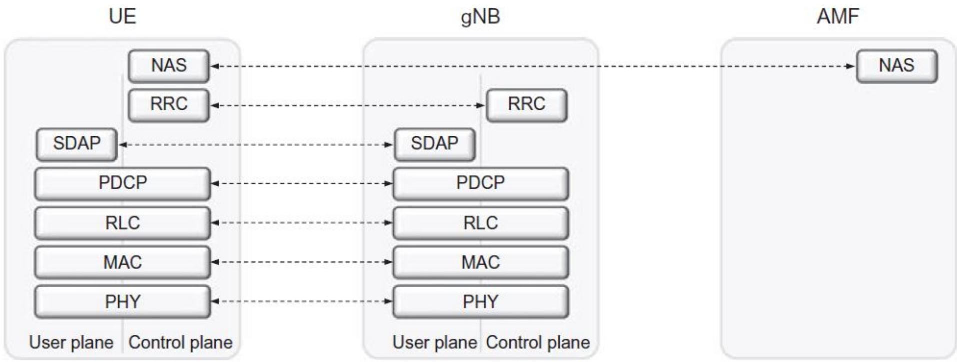

Figure 1.1: User-plane and control plane protocol stack

Our primary focus was on random access procedure, in which MAC module plays the major part. MAC is responsible for scheduling and scheduling-related functions, logical-channel multiplexing, hybrid-ARQ retransmissions, and handling of different numerologies. The overview of MAC structure is given in figure 1.2.

MAC provides services to higher layers in the form of *logical channels*. A logical channel is defined by the type of information it carries and is generally classified as a control channel, used for transmission of control and configuration information necessary for operating an NR system, or as a traffic channel, used for the user data.

From the physical layer, the MAC layer uses services in the form of transport channels. A transport channel is defined by how and with what characteristics the information is transmitted over the radio interface. The transport channel which plays a major role in random access is the Random Access Channel(RACH).

1.1 Background

In a mobile communication system, the fundamental requirement is transmission of data packets between Transmitter(Tx) and Receiver(Rx). For this to happen, an initial handshaking procedure has to occur between the Tx and Rx . This handshaking

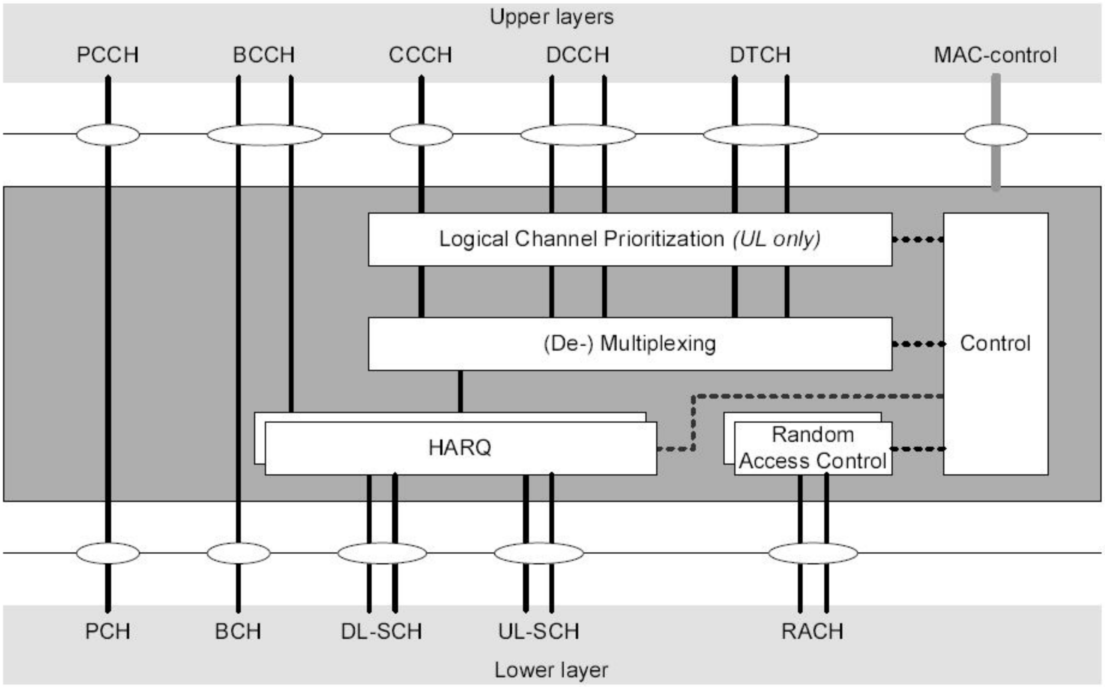

FIGURE 1.2: MAC structure overview

procedure is known as **initial access** in cellular communication. Initial access further consists of two sub-procedures:

- Cell search procedure
- Random Access procedure

Cell search procedures refer to the set of procedures by which a UE finds a cell when entering the coverage area of the cell. Downlink synchronization is achieved as a part of this procedure.

Random access procedures refer to the set of procedures by which a UE requests setting up of connection with the radio access network. Uplink synchronization is achieved as a part of this procedure.

The present work focuses on studying and implementing Random Access procedure at gNB side.

Whenever multiple nodes are involved in accessing a channel, there is always a possibility of collision. There is an inherent need for a mechanism to resolve the conflict. In NR, there are two types of Random Access procedures : Contention-Based Random Access(CBRA) and Contention-Free Random Access(CFRA).

1.1.1 Contention-Based Random Access(CBRA)

Every preamble has a unique signature. Each NR cell has 64 unique preamble signatures. In CBRA, each UE randomly selects one of the signatures for preamble transmission. Multiple UEs may transmit same signature, leading to contention. An additional process at later step is designed to resolve this contention called "Contention Resolution" step.

1.1.2 Contention-Free Random Access(CFRA)

In CFRA, the network decides the preamble assignment for each UE and the transmit timing. Since the preamble allotment for each UE is unique, there is no possibility of contention and hence the name.

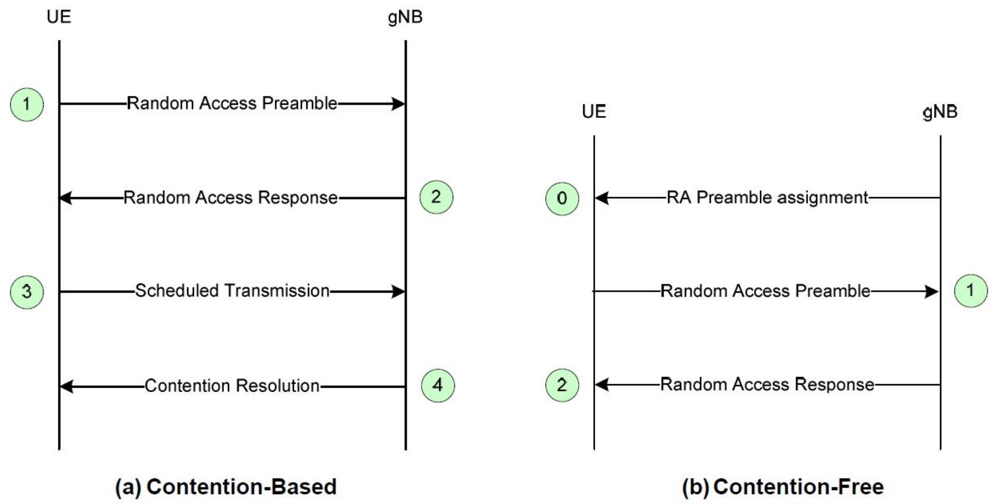

FIGURE 1.3: Two types of Random Access

1.2 Motivation

Synchronization is one of the most important aspects of any digital communication system. The main aim of Random Access procedure is to achieve *uplink synchronization* between transmitter(UE)and receiver(gNB). The secondary purpose of Random Access procedure is to obtain resources Message 3 (such as RRC Connection Request). Therefore Random Access procedure plays an important role in protocol design for 5G wireless systems.

1.3 Objectives of the work

- To understand the functionalities of MAC sublayer in 5G New Radio(NR)
- To implement Random Access procedure at (gNB)
- To simulate Random Access procedure with multiple UEs

Chapter 2

Methodology

The sublayer involved in implementation of major portions of the Random Access procedure is the MAC sublayer.

RRC is responsible for configuring the parameters required for the various sublayers. The parameters are encapsulated under structures which are provided in ASN syntax in TS 38.331. More details on ASN1 can be found in Appendix A.

2.1 Pre-requisites

- ASN1 compiler
- ASN1 header files

2.2 Configuration files

The ASN structures required for Random Access procedure should be identified in RRC specifications and parsed in a ".asn" file. Once the asn file is created, the ASN1 compiler should be used to convert the elements present in ASN format to conventional structures and data types in C programming language and store the created structures in separate header files.

Whenever a particular parameter is needed, the header file corresponding to that parameter has to be included and a structure object or a pointer to a structure has to be created in order to access the parameter.

2.3 Transmission and reception of data

During simulation, sockets were used to model the functionalities provided by PHY, in other words, sockets are used as a substitute for PHY. A client-server model was used to mimic the transmission between UE and gNB, with UE acting as client and gNB acting as server.

Ideally MAC PDUs have to be stored in a buffer which has to be accessed by lower layers. The PDUs to be transmitted to(or received from) lower layer were sent(or received) through sockets for testing purposes.

Chapter 3

Work Done

3.1 Contention Free Random Access

In CFRA, the primary objective is to avoid contention. Contention occurs when same preamble sequence are transmitted by different users at different times. The gNB solves this problem by assigning unique preamble sequences to each of the UEs involved in Random Access procedure. CFRA is used for delay-constrained access requests, because the bottleneck, the contention resolution step is absent in CFRA.

3.1.1 gNB side

Preamble assignment is the most important step in CFRA. The 64 unique preambles are partitioned into two major categories : preambles reserved for CFRA and preambles reserved for CBRA. The idea behind this partitioning is to allow for simultaneous occurrence of CFRA and CBRA in the same gNB. In general, the number of preambles reserved for CFRA is almost always lesser than those reserved for CBRA.

The number of preambles reserved for CFRA is decided from a parameter given in RRC specifications. Outline of preamble assignment algorithm :

- Create a flag array of size 64 for the preamble indices.

- Initialise all values inside to zero.
- If a preamble index is assigned, set the flag of that index to one.
- If Random Access procedure is completed for a particular UE, then set the flag of the corresponding preamble index to zero.

Two functions were created to handle the preamble assignment process, namely : `preamble_tracker()` and `preamble_assigner()`. The `preamble_assigner()` function is responsible for assigning a preamble index to a UE from the available set of preambles. Inside this function, `preamble_tracker()` is called which is responsible for tracking the resource pool for identifying the preambles which are free and preambles that are under use. The preamble index assigned is used in `RACH_ConfigDedicated` IE and is encoded using BER encoding and sent through socket.

The function `rar_fill` is responsible for configuring the RAR PDU. The RAR PDU has two parts : sub-header and payload. The format for the payload is given in fig **Insert RAR PDU fig**. To store the contents of RAR PDU according to the given format, the concept of **bit-fields** and **bit-shifting** were used.

Sequence of steps involved :

1. Assign preamble index and transmit to UEs(Downlink)
2. Receive preamble through PRACH and compute RA-RNTI(Uplink)
3. Prepare the contents for RAR and transmit it (Downlink)

3.1.2 UE side

Each UE receives the preamble index allotted by the gNB along with the system information. The system information contains the initialization values for the parameters involved in random access procedure. Resource selection follows initialization, in which appropriate SSBs or CSI-RS elements are chosen based on the conditions given under section 5.1.2 in MAC protocol specifications (TS 38.321). A pointer to the structure

`RACH_ConfigDedicated` is created and used to receive the `RACH_ConfigDedicated` pointer sent from gNB. The information element is decoded using BER decoding.

The `preamble_transmit()` function is designed to track the number of attempts and to specify PHY parameters for preamble transmission. After preamble transmission, UE starts a timer and waits for RAR. Upon receiving RAR, the function `rar_reception` is invoked in which the contents are extracted again using bit-shifting and stored in a buffer.

Sequence of steps involved :

1. Receive preamble index (Downlink)
2. Transmit preamble through PRACH (Uplink)
3. Receive RAR and extract TC-RNTI, uplink timing alignment etc.(Downlink)

3.2 Contention Based Random Access

In CBRA, each UE randomly chooses a preamble from a given set of reserved preambles. The possibility of two different UEs choosing the same preamble and transmitting at the same time leads to contention. CBRA is used for delay-tolerant access requests.

3.2.1 UE side

Each UE is assumed to pick a preamble index that is uniformly defined over a pre-defined interval. The functions `uniform_distribution()` and `choosing_preamble()` are used for this purpose. Once the preamble index has been chosen, the physical layer parameters required for preamble transmission such as initial power, power ramping step, time and frequency domain locations of the preamble in the Resource grid, are determined from `RACHConfigGeneric` IE.

The UE MAC selects the SSB or CSI-RS elements depending on RSRP measurements received from lower layers. Sequence of steps involved :

1. Randomly choose a preamble among the set of preambles reserved for CBRA and transmit through PRACH (Uplink)

2. Receive RAR and extract TC-RNTI, uplink timing alignment etc (Downlink)

3. Transmit Message 3 (MSG3)(Uplink)

3.2.2 gNB side

The initial operations performed at gNB side in CBRA are very similar to those performed for CFRA. In other words, after receiving the preamble, the steps for creating RAR PDU is same as that used in CBRA.

Sequence of steps involved :

1. Receive preamble through PRACH and compute RA-RNTI (Uplink)

2. Prepare the contents for RAR and transmit it addressed to RA-RNTI (Downlink)

3. Perform contention resolution and send ACK to the selected UE (Uplink)

Chapter 4

Results and Discussions

4.1 Preamble transmission and reception

4.1.1 Preamble transmission

In CFRA, preamble index assignment is done by gNB and conveyed to the UEs in advance. The UEs in turn use the assigned preamble for transmission through RACH.

At gNB side, an algorithm was successfully developed for assigning preambles for UEs participating through CFRA. The working of the algorithm was verified for upto four UEs.

In CBRA, the preamble index selection is left to the UEs. No prior operations have to be performed at gNB to assign preambles to UEs.

Since the preamble is a sequence of OFDM symbols that are to be transmitted by PHY, the parameters required for its transmission were stored in a buffer and transmitted through client(UE) socket.

4.1.2 Preamble reception

The buffer containing parameters for preamble transmission are received through server (gNB) socket. In both CFRA and CBRA, after receiving the preamble, gNB performs the following functions :

- Computation of RA-RNTI to identify the UE
- Calculation of timing adjustment to be applied by observing uplink subframe boundary
- Assignment of TC-RNTI to act as UE's identity in future transmissions

4.2 Random Access Response

The MAC PDU sub-header can be of three distinct types. All different combinations of sub-headers with RAR payload were transmitted from gNB side and received at UE side.

The sequence of operations performed at gNB to maintain the PDU format is reversed at UE in order to extract the information.

4.3 Encoding and Decoding

Prior to transmission of data through sockets, the information elements are encoded using BER and at the receiver side, BER decoding is used to retrieve the contents of the original message.

Chapter 5

Conclusions and Extensions

The first procedure which is triggered when a UE is turned on, is the Random Access procedure, and therefore plays a critical role during testing and debugging of NR devices. Random access procedure can be triggered in many scenarios :

- Initial access from RRC_IDLE
- RRC Connection Re-establishment procedure
- Downlink or uplink data arrival during RRC_CONNECTED when uplink synchronisation status is "non-synchronised"
- Uplink data arrival during RRC_CONNECTED when there are no PUCCH resources for SR(Scheduling Request) available
- SR failure
- Request by RRC upon synchronous reconfiguration (e.g. handover)
- Transition from RRC_INACTIVE;
- To establish time alignment at SCell addition
- Request for Other SI
- Beam failure recovery

Our work primarily focused on implementation of CFRA procedure at gNB side, leading to testing and verification of gNB and UE code for CFRA. The present implementation works well for those scenarios that are based on CFRA.

The work is done assuming that uplink timing alignment specified in section 5.2 , TS 38.321 is performed by each UE. Further, Bandwidth Part(BWP) operation is assumed to be performed by the MAC entity.

The work can be extended for different trigger cases with varying number of UEs and with features like *dual-connectivity* and *carrier aggregation*. A novel endeavour would be to implement CBRA and CFRA simultaneously on the same gNB.

Appendix A

ASN.1 and Basic Encoding Rules

A.1 ASN.1

Abstract Syntax Notation (ASN.1) is the standard to define specifications of abstract data types. It is also used in 3GPP documents. The main reason to use ASN.1 is it is easy to convert the ASN.1 to any other language. ASN.1 sends information(audio,data,etc) in digital communication. It only covers the structural aspects of the information(there are no operators). Therefore it is not a programming language.

We can convert the ASN.1 language to other languages like C,C++ or Java by using the respective compiler. In ASN.1 there are certain number of pre-defined data types like :

- integers (INTEGER),
- boolean (BOOLEAN),
- character strings (IA5String, UniversalString...)
- bit strings (BIT STRING) and etc.

It is possible to define constructed types such as :

- structures (SEQUENCE),

- lists (SEQUENCE OF),
- choice between types (CHOICE)

The important part of ASN.1 notation is that it is associated with several standardized encoding rules , which can be used in case of bandwidth restriction. ASN.1 is widely used in industry sectors where efficient (low-bandwidth, low-transaction, low-cost) computer communications are needed, but is also being used in sectors where XML- encoded data is required (for example, transfer of biometric information).

A.2 Basic Encoding Rules

The Basic Encoding Rules or BER represents ASN.1 values as an octet string. There are three methods to encode an ASN.1 value under BER. The choice depends on the type of the value and whether length of the value is known.The three methods are primitive, definite-length encoding; constructed, definite- length encoding; and constructed, indefinite-length encoding. Simple non-string types employ the primitive, definite-length method; structured types employ either of the constructed methods; and simple string types employ any of the methods, depending on whether the length of the value is known.

BER encoding has three parts in each method:

- Identifier octets : These identify the class and tag number of the ASN.1 value, and indicate whether the method is primitive or constructed.
- Length octets : For the definite-length methods, these give the number of contents octets. For the constructed, indefinite-length method, these indicate that the length is indefinite.
- Contents octets. For the primitive, definite-length method, these give a concrete representation of the value. For the constructed methods, these give the concatenation of the BER encodings of the components of the value.

There are three methods of encoding :

- Primitive, definite-length method :This method applies to simple types and types derived from simple types by implicit tagging. It requires that the length of the value be known in advance.
- Constructed, definite-length method:This method applies to simple string types, structured types, types derived simple string types and structured types by implicit tagging, and types derived from anything by explicit tagging. It requires that the length of the value be known in advance.
- Constructed, indefinite-length method: This method applies to simple string types, structured types, types derived simple string types and structured types by implicit tagging, and types derived from anything by explicit tagging. It does not require that the length of the value be known in advance.

Appendix B

Protocol Data Unit Formats

B.1 Introduction

A MAC PDU is a bit string that is byte aligned (i.e. multiple of 8 bits) in length. In the figures that follow, bit strings are represented by tables in which the most significant bit is the leftmost bit of the first line of the table, the least significant bit is the rightmost bit on the last line of the table, and more generally the bit string is to be read from left to right and then in the reading order of the lines. The bit order of each parameter field within a MAC PDU is represented with the first and most significant bit in the leftmost bit and the last and least significant bit in the rightmost bit.

A MAC SDU is a bit string that is byte aligned (i.e. multiple of 8 bits) in length. A MAC SDU is included into a MAC PDU from the first bit onward.

A MAC CE is a bit string that is byte aligned (i.e. multiple of 8 bits) in length.

A MAC subheader is a bit string that is byte aligned (i.e. multiple of 8 bits) in length. Each MAC subheader is placed immediately in front of the corresponding MAC SDU, MAC CE, or padding.

B.2 MAC PDU (Random Access Response)

A MAC PDU consists of one or more MAC subPDUs and optionally padding. Each MAC subPDU consists one of the following:

- a MAC subheader with Backoff Indicator only
- a MAC subheader with RAPID only (i.e. acknowledgment for SI request)
- a MAC subheader with RAPID and MAC RAR

A MAC subheader with Backoff Indicator consists of five header fields E/T/R/R/BI as described in Figure B.1.

MAC subPDU with Backoff Indicator only is placed at the beginning of the MAC PDU, if included. 'MAC subPDU(s) with RAPID only' and 'MAC subPDU(s) with RAPID and MAC RAR' can be placed anywhere between MAC subPDU with Backoff Indicator only (if any) and padding (if any).

A MAC subheader with RAPID consists of three header fields E/T/RAPID as described in Figure B.2.

Padding is placed at the end of the MAC PDU if present. Presence and length of padding is implicit based on TB size, size of MAC subPDU(s).

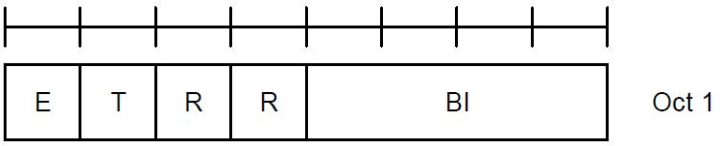

FIGURE B.1: E/T/R/R/BI MAC subheader

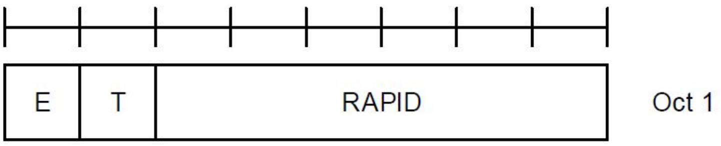

FIGURE B.2: E/T/RAPID MAC subheader

The MAC subheader consists of the following fields:

- E: The Extension field is a flag indicating if the MAC subPDU including this MAC subheader is the last MAC subPDU or not in the MAC PDU. The E field is set to "1" to indicate at least another MAC subPDU follows. The E field is set to "0" to indicate that the MAC subPDU including this MAC subheader is the last MAC subPDU in the MAC PDU;

- T: The Type field is a flag indicating whether the MAC subheader contains a Random Access Preamble ID or a Backoff Indicator. The T field is set to "0" to indicate the presence of a Backoff Indicator field in the subheader (BI). The T field is set to "1" to indicate the presence of a Random Access Preamble ID field in the subheader (RAPID)

- R: Reserved bit, set to "0"

- BI: The Backoff Indicator field identifies the overload condition in the cell. The size of the BI field is 4 bits

- RAPID: The Random Access Preamble IDentifier field identifies the transmitted Random Access Preamble. The size of the RAPID field is 6 bits. If the RAPID in the MAC subheader of a MAC subPDU corresponds to one of the Random Access Preambles configured for SI request, MAC RAR is not included in the MAC subPDU.

B.3 MAC payload for Random Access Response

The MAC RAR is of fixed size as depicted in Figure B.3, and consists of the following fields:

- R: Reserved bit, set to "0"

- Timing Advance Command: The Timing Advance Command field indicates the index value TA used to control the amount of timing adjustment that the MAC entity has to apply in TS 38.213. The size of the Timing Advance Command field is 12 bits

- UL Grant: The Uplink Grant field indicates the resources to be used on the uplink in TS 38.213. The size of the UL Grant field is 27 bits
- Temporary C-RNTI: The Temporary C-RNTI field indicates the temporary identity that is used by the MAC entity during Random Access. The size of the Temporary C-RNTI field is 16 bits

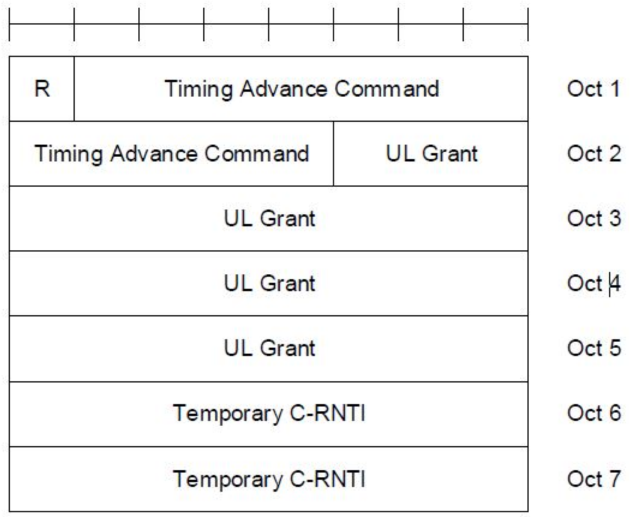

Figure B.3: MAC RAR

The MAC RAR is octet aligned.

Bibliography

[1] 3GPP TS 38.321 version 15.5.0 Release 15 "NR : Medium Access Control (MAC) protocol specification". May 2019

[2] 3GPP TS 38.331 version 15.5.1 Release 15 "NR : Radio Resource Control (RRC) Protocol specification" May 2019

[3] E.Dahlman, S.Parkvall, J.Skold "5G NR: The Next Generation Wireless Access Technology", Elsevier, 2018

[4] A.Omri, M.Shaqfeh, A.Ali and H.Aluneweri, "Synchronization Procedure in 5G NR Systems",IEEE, 2018

[5] M.Hasan and E.Hossain, "Random Access for Machine-to-Machine Communication in LTE-Advanced Networks: Issues and Approaches", IEEE Communications Magazine, 2013

[6] L.T.Oquendo, I.L. Mayorga, V. Pla, J.M. Bauset and V.C.Giner, "Analysis of LTE-A Random Access Procedure: A Foundation to Propose Mechanisms for Managing the M2M Massive Access in Wireless Cellular Networks", ITACA-WIICT 2015

[7] 5G/NR - Initial Access/RACH `https://www.sharetechnote.com/html/5G/5G_RACH.html#TimeDomain_RO_Ex`