ATSC Candidate Standard: Dedicated Return Channel for ATSC 3.0 (A/323)

Doc. S32-293r9
2 November 2017

Advanced Television Systems Committee
1776 K Street, N.W.
Washington, D.C. 20006
202-872-9160

This specification is being put forth as a Candidate Standard by the TG3/S32 Specialist Group. This document is an editorial revision of the Working Draft (S32-293r8) dated 15 August 2017. All ATSC members and non-members are encouraged to review and implement this specification and return comments to cs-editor@atsc.org. ATSC Members can also send comments directly to the TG3/S32 Specialist Group. This specification is expected to progress to Proposed Standard after its Candidate Standard period.

Revision History

Version	Date
Candidate Standard approved	2 November 2017
Standard approved	Date

Table of Contents

Index of Figures and Tables

ATSC Candidate Standard: Dedicated Return Channel for ATSC 3.0 (A/323)

1. SCOPE

1.1 Introduction and Background

The radical shift towards mobile screens and wireless rich media has posed a pressing need for innovative broadcasting services and a new generation of enabling technologies. To date, terrestrial broadcasting remains one of the most efficient means to deliver massive amounts of information to large numbers of users. On the other hand, conventional linear TV services alone (albeit ultra-high-definition) may not be sufficient to sustain the terrestrial broadcasting business which requires a large amount of highly coveted spectrum resources. Intelligent media delivery and flexible service models that maximize the network Return on Investment (ROI) is of paramount importance to the broadcasting industry in the new era.

Recent studies have shown that interactivity between media customers and service providers and between users themselves will be the most important feature in the next-generation media service [2]. In this document, this unique opportunity is addressed by proposing a Dedicated Return Channel (DRC) system for the next-generation broadcast.

In this document, both the physical layer and MAC (Media Access Control) layer specifications for the ATSC 3.0 DRC (a.k.a uplink) are detailed.

1.2 Organization

This document is organized as follows:

- Section 1 – Outlines the scope of this document and provides a general introduction.
- Section 2 – Lists references and applicable documents.
- Section 3 – Provides a definition of terms, acronyms, and abbreviations for this document.
- Section 4 – System overview
- Section 5 – PHY specification
- Section 6 – MAC specification
- Annex A – Description of PRACH
- Annex B – Synchronization error analysis
- Annex C – Signaling overhead

2. REFERENCES

All referenced documents are subject to revision. Users of this Standard are cautioned that newer editions might or might not be compatible.

2.1 Normative References

The following documents, in whole or in part, as referenced in this document, contain specific provisions that are to be followed strictly in order to implement a provision of this Standard.

[1] IEEE: “Use of the International Systems of Units (SI): The Modern Metric System,” Doc. SI 10-2002, Institute of Electrical and Electronics Engineers, New York, N.Y.

[2] ATSC: “ATSC Standard: Interactive Services Standard,” Doc. A/105:2015, Advanced Television Systems Committee, Washington, D.C., 29 October 2015.

[3] ATSC: “ATSC Standard: Physical Layer Protocol,” Doc. A/322:2017, Advanced Television Systems Committee, Washington, D.C., 6 June 2017.

[4] ATSC: “ATSC Candidate Standard: Scheduler and Studio-Transmitter Link (A/324),” Doc. S32-266r16, Advanced Television System Committee, Washington, D.C., 3 October 2016.

[5] ATSC: “ATSC Proposed Standard: Signaling, Delivery, Synchronization, and Error Protection (A/331),” Doc. S331r0, Advanced Television System Committee, Washington, D.C., 27 September 2017.

[6] ATSC: “ATSC Proposed Standard: ATSC 3.0 Security and Service Protection (A/360),” Doc. S36-086r10, Advanced Television System Committee, Washington, D.C., 3 May 2017.

[7] ATSC: “ATSC Standard: Link-Layer Protocol,” Doc. A/330:2016, Advanced Television System Committee, Washington, D.C., 19 September 2016.

[8] IETF: RFC 3986, “Uniform Resource Identifier (URI): Generic Syntax,” Internet Engineering Task Force, Reston, VA, January, 2005. http://tools.ietf.org/html/rfc3986

[9] W3C: “XML Schema Part 2: Datatypes Second Edition” W3C Recommendation, Worldwide Web Consortium, 28 October 2004. https://www.w3.org/TR/xmlschema-2/

[10] IETF: RFC 6726, “FLUTE – File Delivery over Unidirectional Transport,” Internet Engineering Task Force, Reston, VA, November, 2012. http://tools.ietf.org/html/rfc6726

[11] “Universal Mobile Telecommunications System (UMTS); LTE; Multimedia Broadcast/Multicast Service (MBMS); Protocols and codecs (3GPP TS 26.346 version 13.3.0 Release 13),” Doc. ETSI TS 126 346 v13.3.0 (2016-01), European Telecommunications Standards Institute, 2014.

2.2 Informative References

The following documents contain information that may be helpful in applying this Standard.

[12] ATSC: “ATSC Standard: ATSC 3.0 System,” Doc. A/300:2017, Advanced Television System Committee, Washington, D.C., 19 October 2017.

[13] ATSC: “ATSC Standard: A/321, System Discovery and Signaling,” Doc. A/321:2016, Advanced Television System Committee, Washington, D.C., 23 March 2016.

[14] Digital Video Broadcasting (DVB): “Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2)”, ETSI EN 302 755 V1.4.1, July 2015.

[15] Digital Video Broadcasting (DVB): “Interaction channel for Digital Terrestrial Television (RCT) incorporating Multiple Access OFDM,” ETSI EN 301 958 V1.1.1, March 2002.

3. DEFINITION OF TERMS

With respect to definition of terms, abbreviations, and units, the practice of the Institute of Electrical and Electronics Engineers (IEEE) as outlined in the Institute’s published standards [1] shall be used. Where an abbreviation is not covered by IEEE practice or industry practice differs from IEEE practice, the abbreviation in question will be described in Section 3.3 of this document.

3.1 Compliance Notation

This section defines compliance terms for use by this document:

shall – This word indicates specific provisions that are to be followed strictly (no deviation is permitted).

shall not – This phrase indicates specific provisions that are absolutely prohibited.

should – This word indicates that a certain course of action is preferred but not necessarily required.

should not – This phrase means a certain possibility or course of action is undesirable but not prohibited.

3.2 Treatment of Syntactic Elements

This document contains symbolic references to syntactic elements used in the audio, video, and transport coding subsystems. These references are typographically distinguished by the use of a different font (e.g., restricted), may contain the underscore character (e.g., sequence_end_code) and may consist of character strings that are not English words (e.g., dynrng).

3.2.1 Reserved Elements

One or more reserved bits, symbols, fields, or ranges of values (i.e., elements) may be present in this document. These are used primarily to enable adding new values to a syntactical structure without altering its syntax or causing a problem with backwards compatibility, but they also can be used for other reasons.

The ATSC default value for reserved bits is ‘1.’ There is no default value for other reserved elements. Use of reserved elements except as defined in ATSC Standards or by an industry standards body is not permitted. See individual element semantics for mandatory settings and any additional use constraints. As currently-reserved elements may be assigned values and meanings in future versions of this Standard, receiving devices built to this version are expected to ignore all values appearing in currently-reserved elements to avoid possible future failure to function as intended.

3.3 Acronyms and Abbreviation

The following acronyms and abbreviations are used within this document.

16 QAM	16-ary Quadrature Amplitude Modulation
ALP	ATSC 3.0 Link layer Protocol
AMC	Adaptive Modulation and Coding
ARQ	Automatic Repeat-reQuest
ARTT	ARQ Retransmission Timer
ATSC	Advanced Television Systems Committee
BAT	Broadcast Access Terminal
BCI	Broadcast Control Information
BEB	Binary Exponential Backoff
BTS	Broadcast Television Station
CID	Connection ID
CP	Cyclic Prefix
CRC	Cyclic Redundancy Check
CRSC	Circular Recursive Systematic Convolutional

CTC	Convolutional Turbo Code
DFT	Discrete Fourier Transformation
DRC	Dedicated Return Channel
FFT	Fast Fourier Transform
GBR	Guaranteed Bit Rate
GP	Guard Period
GT	Guard Time
ID	Identification
IDFT	Inverse Discrete Fourier Transformation
IFFT	Inverse Fast Fourier Transform
LSB	Least Significant Bit
MAC	Media Access Control
MRC	Maximum Retransmission Count
MSB	Most Significant Bit
NAB	National Association of Broadcasters
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
PDU	Protocol Data Unit
PHY	Physical Layer
PLP	Physical Layer Pipe
PLP-R	Physical Layer Pipe for Return Channel
PRACH	Physical Random Access Channel
PUSCH	Physical Uplink Shared Channel
QCI	QoS Class Identifier
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RF	Radio Frequency
RNTI	Radio Network Temporary Identity
RRT	Ranging Response Timer
RTC	Retransmission Count
SC-FDMA	Single Carrier Frequency Division Multiple Access
SINR	Signal to Interference plus Noise Ratio
TB	Transport Block
TS	Transport Stream
TUID	Temporary User ID
UL-MAP	Uplink Resource MAP
ZC`	Zadoff-Chu

3.4 Terms

The following terms are used within this document.

BAT – An ATSC 3.0 receiver with a DRC terminal module in it, or equivalently a DRC-enabled ATSC 3.0 receiver.

BTS – An ATSC 3.0 transmitter with a DRC base station module in it, or equivalently a DRC-enabled ATSC 3.0 transmitter.

Cell – One pair of I/Q components representing a modulated symbol [3].

DRC downlink – Downlink signaling and downlink data transmission of DRC-related information through the PLP for return channel (PLP-R).

DRC uplink – Uplink signaling and uplink data transmission of DRC through the Physical layer and MAC layer specifications defined in this document.

Ranging – User terminals try to access the system.

Paging – The BTS pages for a single terminal in the broadcast network.

Packet – A collection of data sent as a unit, including a header to identify and indicate other properties of the data, and a payload comprising the data actually to be sent, either having a fixed known length or having means to indicate either its length or its end.

Protocol Data Unit – The protocol data unit encapsulated in the DRC Uplink MAC layer. The maximum size of a DRC uplink MAC PDU is limited to 2048 bytes.

reserved – Set aside for future use.

Resource tile – Basic unit in a physical frame. One resource tile occupies 20 continuous subcarriers in the frequency dimension and 2 continuous symbols in the time dimension, which is equal to 40 resource elements. Among the resource elements in one resource tile, 8 of them are used for pilots and 32 are used for data transmission. Each resource tile has an index to represent it, which is numbered according to the position of the resource tile in a frame.

Transport Block –The minimum channel coding block. The size of a transport block is determined by the number of allocated tiles for a user and the coding scheme. Referring to Table 5.3, the maximum effective bits transmitted by a transport block can be 80, 128, 176, 368, 416, 560, and 848. If the size of a MAC PDU is less than or larger than this limitation, padding or segmentation, respectively, shall be used.

4. SYSTEM OVERVIEW

Dedicated Return Channel (DRC) supports interactive services in ATSC 3.0 without dependence on other non-ATSC e.0 network infrastructure. In ATSC 3.0, downlink broadcast channel and dedicated return channel for interactive services use different RF frequencies (i.e. Frequency Division Duplexing). The PHY layer and Media Access Control (MAC) layer for DRC are defined in the specification.

4.1 Typical Application

A Broadcast Television Station (BTS) transmits DRC system required downlink information to Broadcast Access Terminals (BATs) using the specific PLP called PLP-R, while BATs transmit uplink data to the BTS by using DRC uplink on a separate RF frequency.

When a line of sight path between BAT and BTS exists, the coverage range of DRC uplink can be up to 100 km (see Annex A). However, in urban areas the coverage range is reduced due to larger path loss. Taking different path losses into account, DRC uplink with relays is viable in DRC system operation. Two scenarios with and without relays between BTS and BATs are shown in Figure 4.1 and Figure 4.2, respectively.

In case of relay station usage, all relay stations and the BTS shall be connected by high-speed wired or wireless networks with low latency. All relay stations shall synchronize to the BTS. The frequency of each relay station shall be carefully assigned to avoid interference and to guarantee spatial frequency reuse. It is determined by the service operator as which frequency bands are used for particular relays.

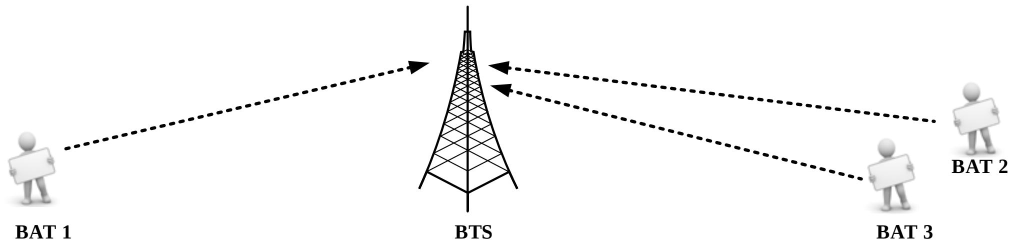

Figure 4.1 Direct uplink transmission from BATs to BTS.

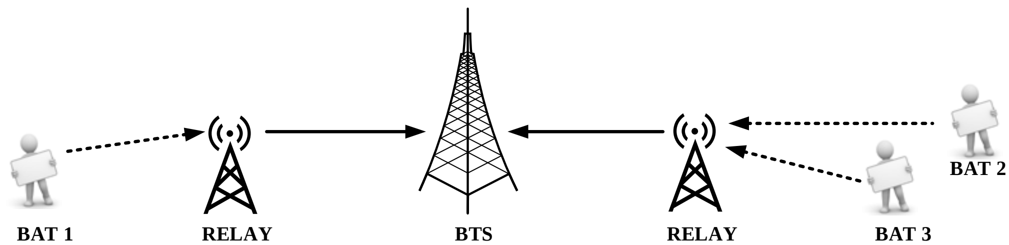

Figure 4.2 Uplink transmission from BATs to BTS through relays.

4.2 System Architecture

The system architecture of ATSC 3.0 with DRC is shown in Figure 4.3. BTS and BATs communicate through wireless channels. BTS transmits downlink payload on frequency f_0 and receivers DRC uplink payload on frequency f_1. In contrast, BAT receives downlink payload on frequency f_0 and transmits DRC uplink payload on frequency f_1.

In BTS, there is a link from the studio(s) to the ATSC 3.0 transmitter through an ATSC 3.0 Downlink Gateway [4]. This Gateway encapsulates ATSC 3.0 Link layer Protocol (ALP) packets from studios into BaseBand Packets (BBPs) and sends them to the ATSC 3.0 transmitter. Each port of the internet protocol (IP) connection used at the Gateway is mapped to a different PLP. DRC downlink signaling and data are also transferred to the Gateway in ALP packets with a specific IP port, and are mapped to the designated PLP called PLP-R for Return channel application.

The ATSC 3.0 receiver in a BAT receives PLPs and separates DRC synchronization, signaling and DRC related data from traditional broadcast service data. DRC synchronization and signaling data are sent to the DRC uplink gateway for processing and maintaining system operation.

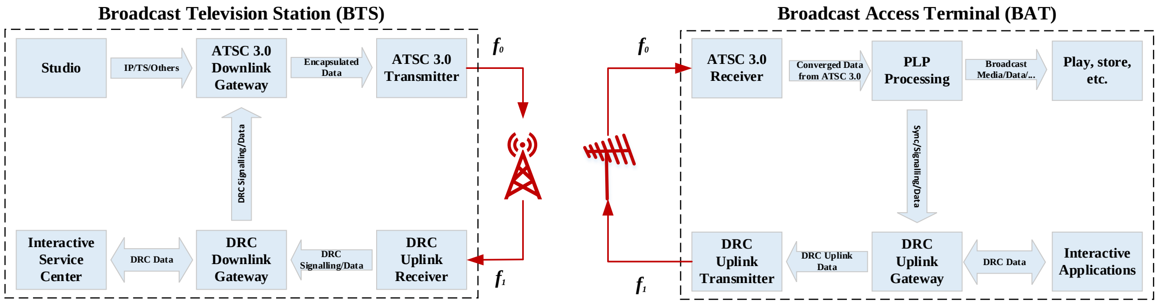

Figure 4.3 System Architecture of ATSC 3.0 with DRC.

4.3 DRC Uplink

The system architecture of the DRC uplink terminal is shown in Figure 4.4. The following modules are included in the DRC uplink terminal: ARQ, Link Adaption, Ranging, and Random Access. The function of each module is described below.

The Automatic Repeat-reQuest (ARQ) module is used for retransmission of lost packets and is defined in Section 6.2.5.

The link adaptation module is used for adaptive modulation and channel coding of DRC uplink.

The random access module is used for the initial access of the DRC uplink terminal, when the DRC uplink terminal does not have an established connection with the BTS. This process is also initiated when BATs lose synchronization to the BTS.

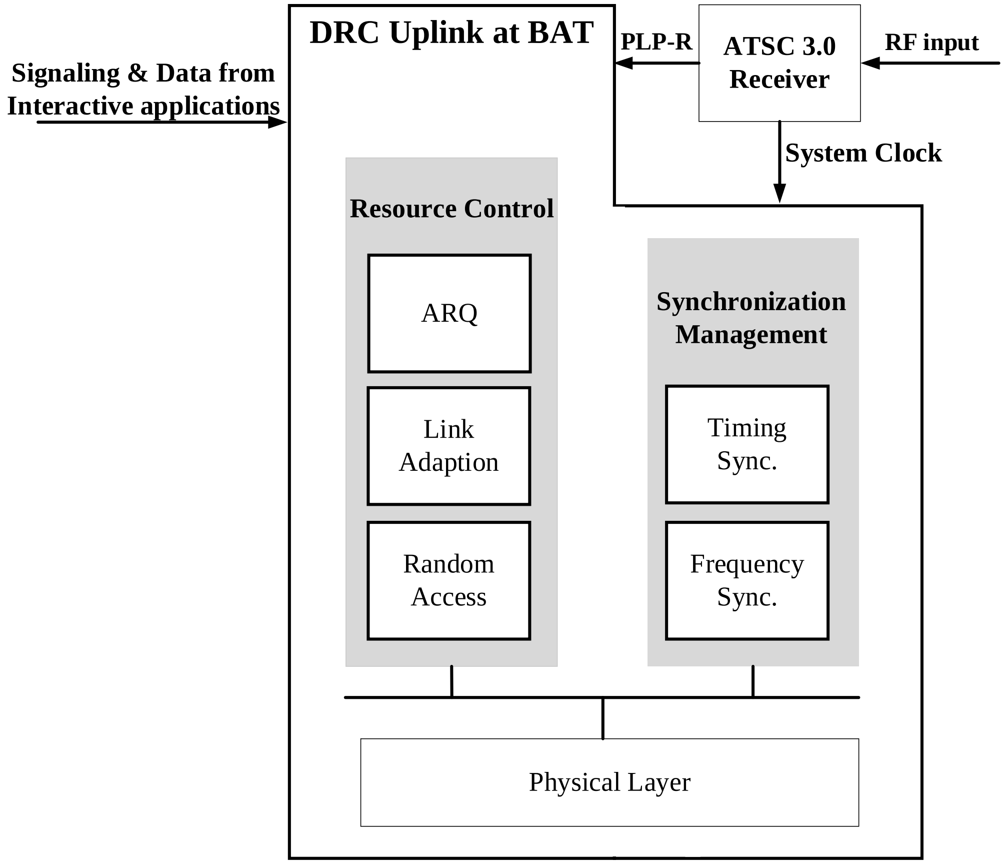

Figure 4.4 System Architecture of the DRC uplink terminal.

Other functions of the DRC uplink system use the existing ATSC 3.0 standards. The transport protocol is as specified in [5]. The encryption protocol is as specified in [6].

The DRC uplink transmitter uses Single Carrier Frequency Division Multiple Access (SC-FDMA) as the multiple access scheme. SC-FDMA is similar to Orthogonal Frequency Division Multiple Access (OFDMA) except for a Discrete Fourier Transform (DFT) operation performed before the Inverse Fast Fourier Transform (IFFT). SC-FDMA is also used by 4G/LTE system.

Binary data flows are input to the uplink transmitter and modulated into complex symbols after going through Cyclic Redundancy Check (CRC) block and Convolutional Turbo Code (CTC) encoder. BPSK, QPSK and 16 QAM are the modulation modes used at the uplink transmitter. N_{DFT} modulated symbols are grouped into symbol blocks. Then an N_{DFT}-point DFT is performed on the symbol blocks. Subcarrier mapping module will map N_{DFT}-point DFT output symbols to N_{IFFT} orthogonal subcarriers, where N_{IFFT} is the total number of orthogonal subcarriers in the frequency domain.

4.4 Interaction between Broadcast and DRC

To be compatible with the ATSC 3.0 traditional broadcast system without DRC, an indication bit (**L1B_return_channel_flag**) of whether a DRC system is associated with the current downlink broadcast system is included in the downlink physical layer (L1) signalling [3] Section 9.2.1. When **L1B_return_channel_flag** = 0, it indicates that DRC is not supported in the current frame of the current frequency band and current broadcast network. When **L1B_return_channel_flag** = 1, it indicates that DRC is supported in the current frame of the current frequency band and current broadcast network.

Synchronization and resource scheduling among all users are required to realize multiple access. Furthermore, the downlink broadcast service and the DRC in ATSC 3.0 use different RF frequencies.

One special downlink PLP, named PLP-R, is defined to carry downlink data and signalling that supports operation of the return channel and is shown in Figure 4.5. A PLP-R has the same physical layer parameters as a PLP in the ATSC 3.0 downlink system. PLP-R in a downlink frame is defined by the System Scheduler according to A/324 [4]. Data in PLP-R is fed into the scheduler in the same way as other ATSC 3.0 downlink traffic. Formats of the data payload in PLP-R are defined in Section 6. Other headers for data transfer, such as IP header and UDP header, shall conform to A/331 [5].

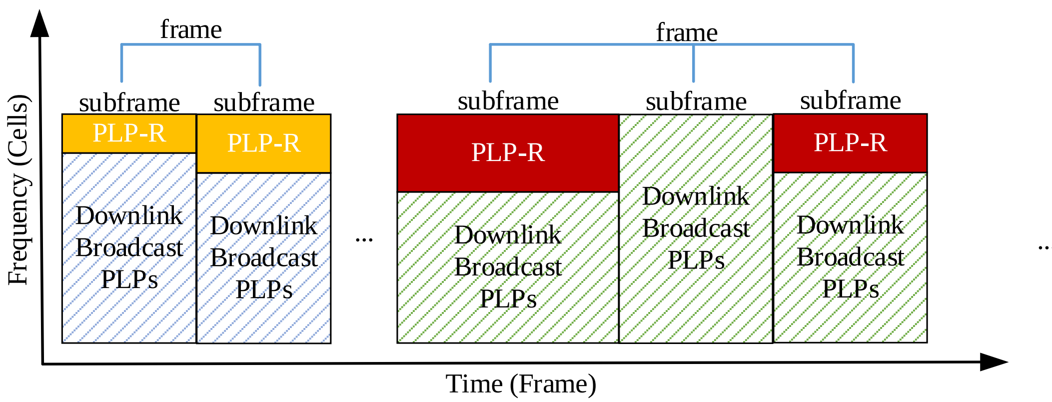

Figure 4.5 Mapping of PLP-R.

A possible resource allocation scheme of PLP-R is shown in Figure 4.6. Three kinds of information can be sent in PLP-R, i.e., Broadcast Control Information (BCI), Uplink MAP (UL-MAP), and DRC Downlink MAC Protocol Data Unit (PDUs).

Considering the tradeoff between resource granularity and scheduling overhead, the frame length for the DRC uplink is defined as 10ms. The response time from the BTS following an uplink transmission by a BAT depends on how PLP-R is reserved in the ATSC 3.0 transmitter. If the resource for PLP-R is reserved along the entire frame, BATs can obtain a response quickly. Otherwise, if PLP-R is only allocated in partial subframes in a frame, the response time is at least the time duration between two successive subframes with allocated PLP-R.

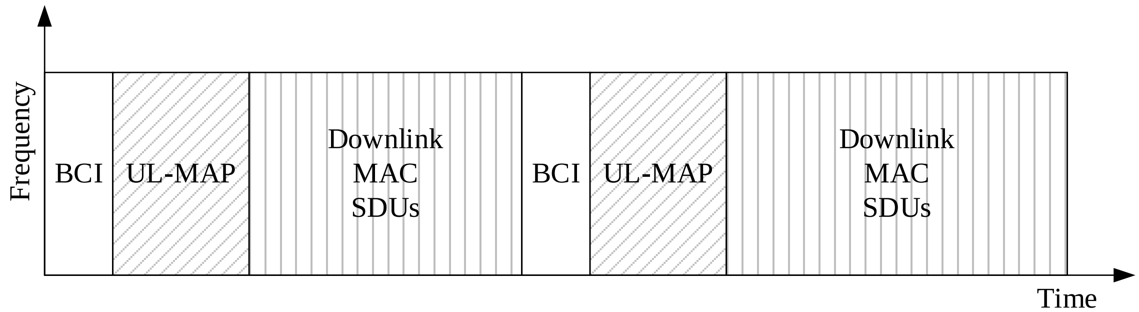

Figure 4.6 Illustration of resource allocation in PLP-R.

The structure of the Studio to Transmitter Link (STL) and the physical layer of the downlink system with DRC is the same as that without DRC. The only difference is that, with DRC, a DRC-related downlink signaling is fed into the ATSC Link-Layer Protocol (ALP) module besides existing traditional broadcasting service data sources. The high level overview of STL with DRC is shown in Figure 4.7.

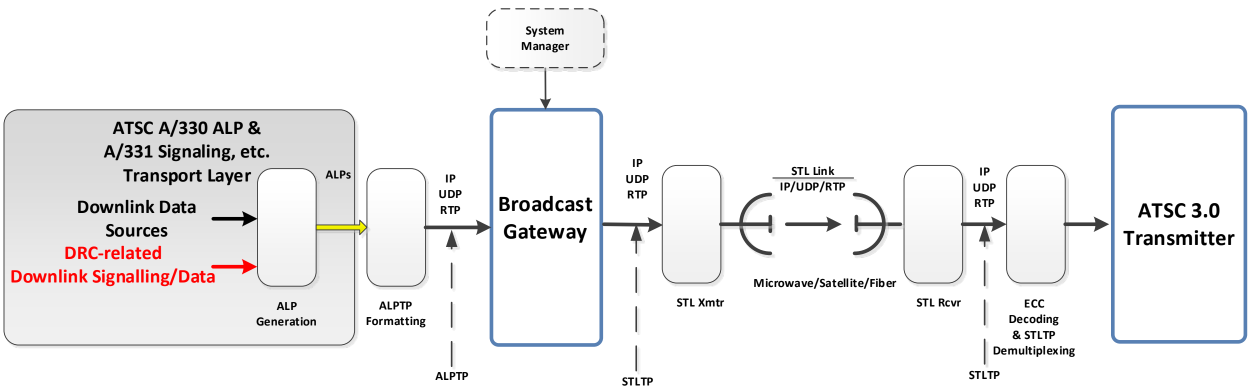

Figure 4.7 High level overview of Downlink STL with DRC.

The identification of the service type carried in a PLP is defined by Low Level Signalling (LLS) table in A/331 [5]. An ATSC 3.0 receiver identifies the LLS table of a service at first and then process the service carried in the PLP by the corresponding component. In A/331 [5], a service type for DRC, i.e., Dedicated Return Channel Table (**DRCT**) is defined as case 0x[TBD] in Table 6.1. The syntax of **DRCT** table is defined in Section7 of this document.

When PLP-R exists in an ATSC 3.0 system, the DRC-enabled receiver shall identify PLP-R and then process it by the receiver component for DRC, while a conventional receiver without DRC function shall neglect PLP-R. If the conventional receiver is implemented with the definition of DRCT, it shall neglect PLP-R. If the conventional receiver is implemented without the definition of DRCT, it will delete it because DRCT is illegal.

5. PHY SPECIFICATION

In this section, the framing, channel coding, modulation, DFT transform, resource mapping, pilot mapping, signal generation, random access, synchronization, and mapping from MAC PDU to Transport Block (TB) of DRC uplink are defined.

5.1 DRC Uplink Frame and Modulation

Single-Carrier Frequency Division Multiple Access (SC-FDMA) with Cyclic Prefix (CP) shall be used for DRC uplink. The signal generation of SC-FDMA is depicted in Figure 5.1, where P/S means parallel-to-serial conversion of the complex values resulting from the IFFT (this is not a parallel-to-serial bit conversion).

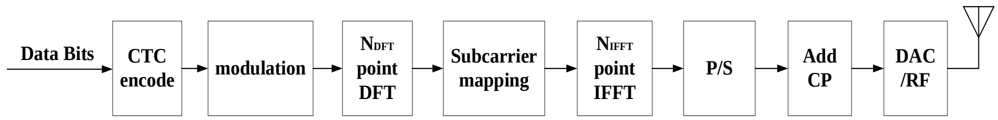

Figure 5.1 SC-FDMA signal generation structure

The uplink frame structure is shown in Figure 5.2. Each uplink frame shall have a time length of $T_F = 10\ msec$ and shall consist of 44 symbols followed by one guard period (GP).

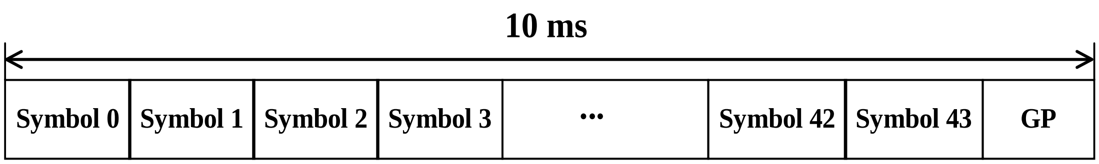

Figure 5.2 Uplink frame structure in time domain.

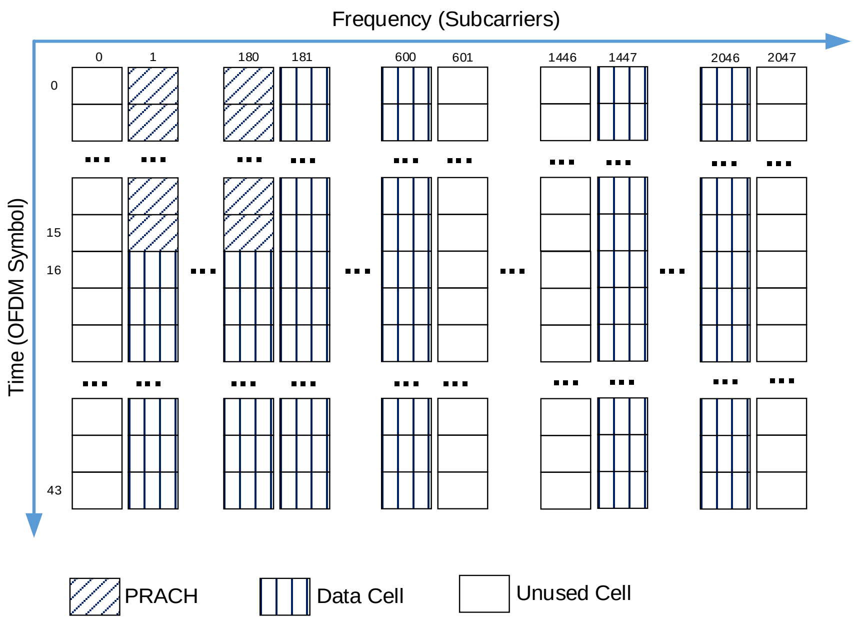

Figure 5.3 Resource map in DRC uplink frame.

The resource map structure in a DRC uplink frame is shown in Figure 5.3. A resource unit consisting of one symbol in the time dimension and one subcarrier in the frequency dimension is defined as a cell. A cell is uniquely indexed by the pair (k, l), where k and l are the indices in the frequency and time dimensions, respectively.

In the frequency dimension, there are 2048 subcarriers, which are indexed from 0 to 2047. Distribution of subcarriers on the frequency dimension is illustrated in Figure 5.4. The virtual Direct Current (DC) subcarrier shall be located in the middle of subcarrier 0 and subcarrier 2047.

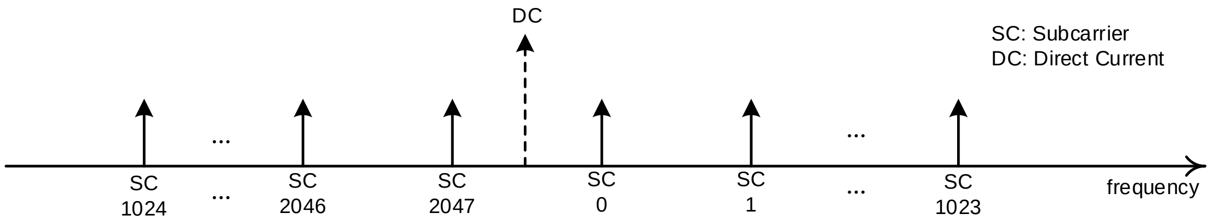

Figure 5.4 Distribution of DRC subcarriers.

Among these subcarriers, subcarriers with indices 0, 2047, and those with indices from 601 to 1446 shall not be used. Thus, the number of used subcarriers is 1200, and the equivalent bandwidth in a system with 6 MHz bandwidth is 5.859 MHz.

Subcarriers indexed from 1 to 180 within symbols indexed from 0 to 15 shall be reserved for use by the Physical Random Access Channel (PRACH) for random access. The other cells are used by the Physical Uplink Shared Channel (PUSCH) for normal transmission.

As shown in Figure 5.5, a tile shall occupy $N_{symb} = 2$ symbols in the time dimension and $N_{SC} = 20$ subcarriers in the frequency dimension, i.e. 40 cells in total. Among the N_{SC} subcarriers in each symbol, there exist $N_{SC\_data} = 16$ data cells and 4 pilot cells.

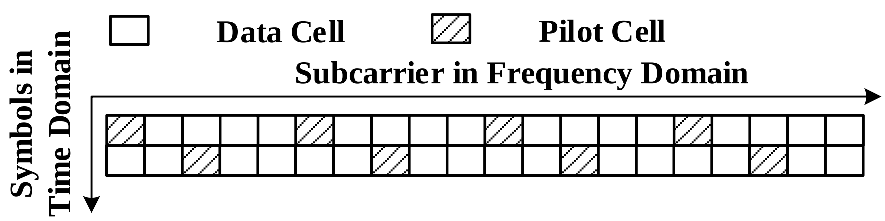

Figure 5.5 Tile pattern.

A tile in a frame is indexed from 0 to 1247. Let n_{symb} and m_{subc} denote the starting symbol and the starting subcarrier of a tile, respectively. Let q_{tile} denote the index of the tile, then the starting position of the tile with index q_{tile} shall be:

$$n_{symb} = \begin{cases} (\lfloor q_{tile} / 51 \rfloor \times 2, & 0 \le q_{tile} \le 407 \\ 16 + \lfloor (q_{tile} - 408) / 60 \rfloor \times 2, & 408 \le q_{tile} \le 1247 \end{cases} \tag{5.1}$$

$$m_{subc} = \begin{cases} 181 + (q_{tile} \bmod 51) \times 20, & (0 \le q_{tile} \le 407) and [(q_{tile} \bmod 51) < 21] \\ 1447 + [(q_{tile} \bmod 51) - 21)] \times 20, & (0 \le q_{tile} \le 407) and [(q_{tile} \bmod 51) \ge 21] \\ [(q_{tile} - 408) \bmod 60] \times 20, & (408 \le q_{tile} \le 1247) and [(q_{tile} - 408) \bmod 60 < 30] \\ 1447 + \{[(q_{tile} - 408) \bmod 60] - 30\} \times 20, & (408 \le q_{tile} \le 1247) and [(q_{tile} - 408) \bmod 60 \ge 30] \end{cases} \quad (5.2)$$

where $\lfloor x \rfloor$ means truncation of x toward zero.

The resource allocated to a BAT is identified by the index of the start tile and the length of the allocated resource in tiles. The starting tile is indicated by q_{tile}, while the allocated tiles are allocated in two optional ways, i.e., time-frequency dimension and frequency-time dimension.

In the time-frequency dimension, the allocated tiles are counted as:

1) Find the minimum subcarrier index with available tiles;
2) Allocate tiles of the subcarrier along the time dimension from smaller symbol index to larger symbol index;
3) When there is no available tiles in the subcarrier, increase the subcarrier index by 1;
4) Repeat step 2 until the resource allocation to a BAT is finished.

In the frequency-time dimension, the allocated tiles are counted as:

1) Find the minimum symbol index with available tiles;
2) Allocate tiles of the subcarrier along the frequency dimension from smaller subcarrier index to larger subcarrier index;
3) When there is no available tiles in the symbol, increase the symbol index by 1;
4) Repeat step 2 until the resource allocation to a BAT is finished.

5.2 Channel Coding

Cyclic Redundancy Check (CRC) and Convolutional Turbo Coding (CTC) are used for error detection and channel coding, respectively, in the DRC uplink. First, the CRC bits are appended after the data bits, and then the resulting bits are processed by CTC encoding.

5.2.1 CRC Code Generation

The CRC can be computed using a shift register circuit as illustrated in Figure 5.6.

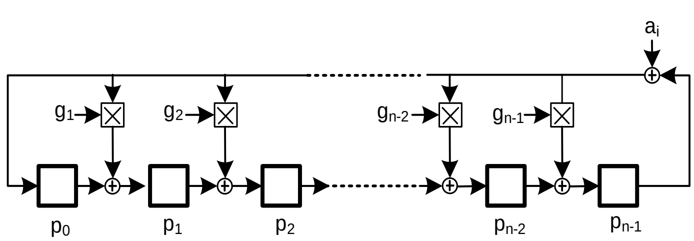

Figure 5.6 Shift register for CRC.

The generator polynomial of *n* bit CRC can be expressed as:

$$G_{crc-n}(D) = D^n + g_{n-1}D^{n-1} + g_{n-2}D^{n-2} + ... + g_2D^2 + g_1D + 1 \quad (5.3)$$

16-bit CRC coding shall be used in DRC uplink. The CRC code generator polynomial shall be as presented in the equation below:

$$G_{CRC-16}(D) = \left[D^{16} + D^{12} + D^{5} + 1\right] \tag{5.4}$$

All coefficients of the 16-bit CRC generator polynomial are 0 except g_5 and g_{12}.

The input bit sequence to the CRC operation is written as $a_0, a_1, a_2, a_3, \ldots, a_{L_I-1}$, where L_I is the length of the input sequence. The check bits are written as $p_0, p_1, p_2, p_3, \ldots, p_{L_c-1}$, where $L_c = 16$ is the length of the CRC check. The combined output of the CRC operation is written as $c_0, c_1, c_2, c_3, \ldots c_{L_O-1}$, where $L_O = L_I + L_C$ and

$$c_k = \begin{cases} a_k, & k = 0,1,2,\cdots,L_I - 1, \\ p_{k-L_i}, & k = L_I, L_I + 1, \cdots, L_O - 1. \end{cases} \tag{5.5}$$

At the receiver side, if the CRC check of a MAC PDU fails, the MAC PDU shall be dropped and the transmitter be notified. The transmitter can retransmit the lost MAC PDU.

5.2.2 CTC Encoding

5.2.2.1 CTC Encoder

The output of CRC encoding shall be divided into two parts A and B.

$$\begin{aligned} A_i &= c_{2i}, \quad i = 0,1,2,\cdots,L_O / 2 - 1, \\ B_i &= c_{2i+1}, i = 0,1,2,\cdots,L_O / 2 - 1. \end{aligned} \tag{5.6}$$

Every bit pair, (A_i,B_i), constitutes a couple and shall be fed into the Convolutional Turbo Coding (CTC) module, which shall generate outputs as shown in Figure 5.7. The CTC encoder uses Circular Recursive Systematic Convolutional Codes (CRSC) as component codes with double-binary input. As required by CRSC, the length of input sequence must be L_O bits or N_{CTC_in} couples, where $L_O = 2\ x\ N_{CTC\_in}$ is satisfied.

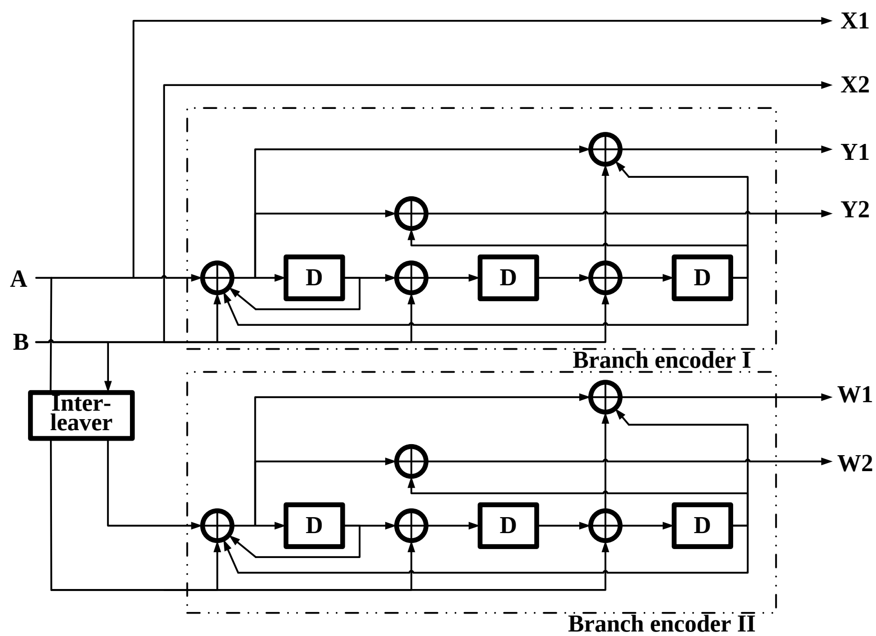

Figure 5.7 Duo-binary Convolution Turbo Coding (CTC).

As shown in Figure 5.7, the CTC encoder generates six output bits for each pair of input bits.

Suppose the input sequences A and B are written as $A_0, A_1, A_2, A_3, \ldots, A_{N_{CTC\_in}-1}$ and $B_0, B_1, B_2, B_3, \ldots, B_{N_{CTC\_in}-1}$, respectively. The output sequences of the CTC are $X1$, $X2$, $Y1$, $Y2$, $W1$, and $W2$. $X1$ and $X2$ correspond exactly to the input sequences A and B, respectively (i.e. $X1$ and $X2$ represent the systematic bits). $Y1$ and $Y2$ represent the output of branch encoder I. $W1$ and $W2$ represent the output of branch encoder II. Sequences $Y1$ and $Y2$ can be written as $Y1_0, Y1_1, Y1_2, Y1_3, \ldots, Y1_{N_{CTC\_in}-1}$ and $Y2_0, Y2_1, Y2_2, Y2_3, \ldots, Y2_{N_{CTC\_in}-1}$, respectively. Sequences $W1$ and $W2$ can be written as $W1_0, W1_1, W1_2, W1_3, \ldots, W1_{N_{CTC\_in}-1}$ and $W2_0, W2_1, W2_2, W2_3, \ldots, W2_{N_{CTC\_in}-1}$, respectively.

Based on the CTC encoder given in Figure 5.7, the final output sequence of the CTC encoder can be written as $z_0, z_1, z_2, z_3, \ldots, z_{6N_{CTC\_in}-1}$, where

$$z_i = \begin{cases} A_i, & i = 0,1,2,\cdots,N_{CTC\_in}-1, \\ B_{i-N_{CTC\_in}}, & i = N_{CTC\_in}, N_{CTC\_in}+1,\cdots,2N_{CTC\_in}-1, \\ Y1_{i-2N_{CTC\_in}}, & i = 2N_{CTC\_in}, 2N_{CTC\_in}+1,\cdots,3N_{CTC\_in}-1, \\ Y2_{i-3N_{CTC\_in}}, & i = 3N_{CTC\_in}, 3N_{CTC\_in}+1,\cdots,4N_{CTC\_in}-1, \\ W1_{i-4N_{CTC\_in}}, & i = 4N_{CTC\_in}, 4N_{CTC\_in}+1,\cdots,5N_{CTC\_in}-1, \\ W2_{i-5N_{CTC\_in}}, & i = 5N_{CTC\_in}, 5N_{CTC\_in}+1,\cdots,6N_{CTC\_in}-1. \end{cases} \quad (5.7)$$

5.2.2.2 Branch Encoder Generator Polynomials

For the feedback branch. i.e., $Y1$ and $W1$, the generator polynomial shall be $G_1(D) = [D^3 + D + 1]$. For the parity bits of $Y2$ and $W2$, the generator polynomial shall be $G_2(D) = [D^3 + D^2 + 1]$.

The initial value of each encoder registers shall be set to zero prior to start of each code block.

5.2.2.3 CTC Interleaver

The CTC interleaver consists of two permutation steps. The first step is a permutation on the level of each couple individually, and the second step is on the level of the sequence of all couples.

The first step is defined as switching the elements of alternate couples.

For the input sequence of all couples written as $[(A_0, B_0), (A_1, B_1), (A_2, B_2) \dots]$, the output of the first step shall be $[(A_0, B_0), (B_1, A_1), (A_2, B_2) \dots]$. That is, the two elements of couple (A_i, B_i) shall maintain their order when i is even and shall be swapped with each other when i is odd. This operation shall be repeated for the entire block.

The second step provides the interleaved address i of the couple j. Given the permutation parameters as P_0, P_1, P_2 and P_3, the second step of the interleaver shall be defined as

$$P(j) = \begin{cases} 0, & j \bmod 4 = 0, \\ N_{CTC\_in}/2 + P_1, & j \bmod 4 = 1, \\ P_2, & j \bmod 4 = 2, \\ N_{CTC\_in}/2 + P_3, & j \bmod 4 = 3. \end{cases} \quad (5.8)$$

$$i = [P_0 \times j + P(j) + 1] \bmod N_{CTC\_in} \quad (5.9)$$

where j is the index of an input couple to the interleaver, i is the index of the corresponding output couple after interleaving, and mod is the modulo operation taking the remainder after division. Parameters P_0, P_1, P_2 and P_3 depend on the length of the sequence to be encoded. The CTC interleaver parameters for the different modulations and coding rates allowed for the DRC shall be as listed in Table 5.1.

Table 5.1 CTC Parameters for Different Modulations and Coding Rates

Modulation Mode	Code Rate	L_O	N_{CTC_in}	P_0	P_1	P_2	P_3
BPSK	1/3	96	48	13	24	0	24
QPSK	1/2	96	48	13	24	0	24
		192	96	7	48	24	72
		288	144	17	74	72	2
	3/4	144	72	11	6	0	6
		288	144	17	74	72	2
		432	216	13	108	0	108
16 QAM	1/2	192	96	7	48	24	72
		384	192	11	96	48	144
		576	288	23	50	188	50
	3/4	288	144	17	74	72	2
		576	288	23	50	188	50
		864	432	13	0	4	8

5.2.2.4 CTC Puncturing

In order to generate different CTC coding rates, parity bits shall be punctured from the encoding output. The puncturing method shall be as shown in Table 5.2.

Table 5.2 CTC Puncturing Method.

Code Rate	Retained Bits
1/3	$z_0, z_1, z_2, z_3, \dots, z_{6N_{CTC\_in}-1}$
1/2	$z_0, z_1, z_2, z_3, \dots, z_{4N_{CTC\_in}-1}$
3/4	$z_0, z_1, z_2, z_3, \dots, z_{2N_{CTC_{in}}-1},$ $z_{2N_{CTC\_in}}, z_{2N_{CTC\_in}+3}, z_{2N_{CTC\_in}+6}, z_{2N_{CTC\_in}+9}, \dots, z_{4N_{CTC\_in}-6}, z_{4N_{CTC\_in}-3}$

In summary, CTC parameters for different modulation and coding rate combinations are listed in Table 5.3. L_{CTC_out} is the number of bits after encoding. L_{punt_out} is the number of bits after puncturing. L_{symb} is the number of modulation symbols. L_{tile} is the number of tiles occupied with the corresponding modulation-coding scheme.

Table 5.3 Data Stream in Transmitter

Modulation Mode	Code Rate	L_I	L_O	L_{CTC_out}	L_{punt_out}	L_{symb}	L_{tile}
BPSK	1/3	80	96	288	288	288	9
QPSK	1/2	80	96	288	192	96	3
		176	192	576	384	192	6
		272	288	864	576	288	9
	3/4	128	144	432	192	96	3
		272	288	864	384	192	6
		416	432	1296	576	288	9
16 QAM	1/2	176	192	576	384	96	3
		368	384	1152	768	192	6
		560	576	1728	1152	288	9
	3/4	272	288	864	384	96	3
		560	576	1728	768	192	6
		848	864	2592	1152	288	9

5.3 Modulation

There are three kinds of modulation schemes supported in DRC: BPSK, QPSK, and 16 QAM. Gray mapping is used for mapping binary bits to modulation symbols. The input to the modulation block shall be the output of the CTC encoder after puncturing. The input bit sequence $z_0, z_1, z_2, z_3, \dots, z_{L_{CTC\_out}-1}$ shall be modulated as described in Table 5.4, Table 5.5 and Table 5.6, respectively, based on the configured modulation scheme. The resulting output sequence consists of L_{symb} complex modulated symbols $(i) = I + jQ;\ i = 0,1, \dots, L_{symb} - 1$.

5.3.1 BPSK

For BPSK modulation, each input bit z_i shall be mapped to a complex modulated symbol according to Table 5.4.

Table 5.4 BPSK Modulation Mapping

z_i	I	Q
0	$1/\sqrt{2}$	$1/\sqrt{2}$
1	$-1/\sqrt{2}$	$-1/\sqrt{2}$

5.3.2 QPSK

For QPSK modulation, each pair of input bits z_{2i}, z_{2i+1} shall be mapped to a complex modulated symbol according to Table 5.5.

Table 5.5 QPSK Modulation Mapping

z_{2i}, z_{2i+1}	I	Q
0,0	$1/\sqrt{2}$	$1/\sqrt{2}$
0,1	$1/\sqrt{2}$	$-1/\sqrt{2}$
1,0	$-1/\sqrt{2}$	$1/\sqrt{2}$
1,1	$-1/\sqrt{2}$	$-1/\sqrt{2}$

5.3.3 16 QAM

For 16 QAM, each set of four input bits $z_{4i}, z_{4i+1}, z_{4i+2}, z_{4i+3}$ shall be mapped to a complex symbol according to Table 5.6.

Table 5.6 16 QAM Modulation Mapping

$z_{4i}, z_{4i+1}, z_{4i+2}, z_{4i+3}$	I	Q
0,0,0,0	$1/\sqrt{10}$	$1/\sqrt{10}$
0,0,0,1	$1/\sqrt{10}$	$3/\sqrt{10}$
0,0,1,0	$3/\sqrt{10}$	$1/\sqrt{10}$
0,0,1,1	$3/\sqrt{10}$	$3/\sqrt{10}$
0,1,0,0	$1/\sqrt{10}$	$-1/\sqrt{10}$
0,1,0,1	$1/\sqrt{10}$	$-3/\sqrt{10}$
0,1,1,0	$3/\sqrt{10}$	$-1/\sqrt{10}$
0,1,1,1	$3/\sqrt{10}$	$-3/\sqrt{10}$
1,0,0,0	$-1/\sqrt{10}$	$1/\sqrt{10}$
1,0,0,1	$-1/\sqrt{10}$	$3/\sqrt{10}$
1,0,1,0	$-3/\sqrt{10}$	$1/\sqrt{10}$
1,0,1,1	$-3/\sqrt{10}$	$3/\sqrt{10}$
1,1,0,0	$-1/\sqrt{10}$	$-1/\sqrt{10}$
1,1,0,1	$-1/\sqrt{10}$	$-3/\sqrt{10}$
1,1,1,0	$-3/\sqrt{10}$	$-1/\sqrt{10}$
1,1,1,1	$-3/\sqrt{10}$	$-3/\sqrt{10}$

5.4 DFT Transform

The block of modulated symbols $d(0), d(1), \ldots, d\left(L_{symb} - 1\right)$ is divided into $2\ x\ L_{tile}$ sets. DFT transform shall be applied according to equations below:

$$x_{DFT}(l \cdot N_{sc\_data} + k) = \frac{1}{\sqrt{N_{sc-data}}} \sum_{i=0}^{N_{sc\_data}-1} d(l \cdot N_{sc\_data} + i) e^{-j2\pi ik / N_{sc\_data}}$$
$$k = 0, \cdots, N_{sc\_data} - 1 \quad (5.10)$$
$$l = 0, \cdots, 2L_{tile} - 1$$

5.5 Physical Resource Mapping

The complex-valued symbols $x_{DFT}(i)$ shall be mapped in sequence starting with $x_{DFT}(0)$ to the data cells of the assigned resource tiles. Within the tiles assigned for transmission, the mapping of $x_{DFT}(i)$ to cell shall be in increasing order of subcarrier index, symbol index and tile index successively. That is, mapping shall begin with the first data subcarrier of the first SC-FDMA symbol of the first assigned tile. Mapping shall continue with the remaining data subcarriers, in increasing order, of the first SC-FDMA symbol of the first assigned tile, before moving on to the first data subcarrier of the second SC-FDMA symbol of the first assigned tile. After each assigned tile has been completely filled and if complex-valued symbols $x_{DFT}(i)$ still remain to be mapped, mapping shall move on to the first data subcarrier of the first SC-FDMA symbol of the next assigned tile.

Tiles to be used for transmission shall be assigned by BTS. The corresponding tile index is broadcast through BCI.

5.6 Pilot Mapping

Scattered pilots are used in the DRC uplink.

5.6.1 Locations of the Pilots

Cell (k, l) (where k is the subcarrier index and l is the SC-FDMA symbol index) of the tiles assigned to a BAT shall be a scattered pilot if one of the conditions given in equations below is satisfied:

$$\begin{aligned} &\text{mod}(k+4,5) = 0, 1 \le k \le 600, l \text{ is even.} \\ &\text{mod}(k+3,5) = 0, 1447 \le k \le 2046, l \text{ is even.} \\ &\text{mod}(k+2,5) = 0, 1 \le k \le 600, l \text{ is odd.} \\ &\text{mod}(k+1,5) = 0, 1447 \le k \le 2046, l \text{ is odd.} \end{aligned} \quad (5.11)$$

5.6.2 Generation of the Pilot Sequence

The pilot sequence is generated from a frequency-domain Zadoff-Chu sequence. The complex value for a pilot on subcarrier k shall be given by equations below.

$$x_p(k) = \begin{cases} \sqrt{2}\exp[-j\pi \frac{k(k+1)}{1201}], & 1 \le k \le 600 \\ \sqrt{2}\exp[-j\pi \frac{(k-846)(k-845)}{1201}], & 1447 \le k \le 2046 \end{cases} \quad (5.12)$$

5.7 SC-FDMA Baseband Signal Generation

Each SC-FDMA symbol is composed of two parts: a useful part with time duration $N_{IFFT}T_s$ and a cyclic prefix with time duration $N_{CP}T_s$ preceding the useful part. The cyclic prefix shall be formed by copying the last N_{CP} values from the symbol's useful part and prepending those values immediately before the symbol's useful part. Each frame shall contain a guard period with time duration $N_{GP}T_S$ at the end of the frame. The system parameters of a DRC uplink frame are summarized in Table 5.7.

Table 5.7 System Parameters for SC-FDMA in DRC

Parameter	Value
IFFT point (N_{IFFT})	2048
DFT point (N_{DFT})	16
Number of active subcarriers	1200
Sampling rate	10 MHz
Time period between samples (T_s)	0.1 µsec
Cyclic prefix (N_{CP})	210
Guard period (N_{GP})	648
Subcarrier spacing	4.883 kHz
Actual occupied bandwidth	5.859 MHz

The time-continuous signal $s(t)$ in SC-FDMA symbol l shall be defined by

$$s_l(t) = \sum_{k=-1024}^{1023} x_{\hat{k},l} e^{j2\pi(k+1/2)\Delta f(t-N_{CP}T_s)} \qquad (5.13)$$

where $0 \le t < (N_{IFFT} + N_{CP})\times T_s$, $\hat{k} = mod(k + 2048, 2048), \Delta f = 4.883\ kHz$ and $x_{k,l}$ is the symbol value of cell (k, l).

The SC-FDMA symbols in a frame shall be transmitted in increasing order of l.

5.8 Random Access

BAT can achieve both time and frequency synchronization, and obtain resources for subsequent data transmission using a random access burst.

When a BAT wishes to transmit data on the DRC, the BAT shall first initiate a random access procedure in order to access the DRC. The structure of the random access burst shall be as shown in Figure 5.8.

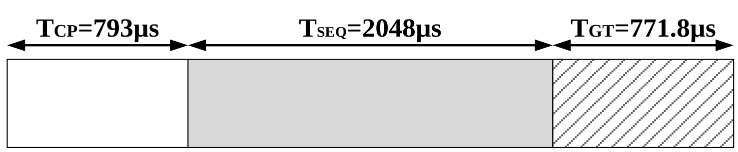

Figure 5.8 The structure of a PRACH preamble symbol

After a connection has been established, when a BAT needs to transmit more information on the DRC, then the BAT shall use bandwidth request to acquire further uplink resources. When a connection is terminated, the BAT loses its synchronization and needs to initiate a random access again, if the BAT wishes to transmit other data.

A random access burst is generated from a 1777-point Zadoff-Chu sequence. The sequence shall be defined as

$$Z(k) = \exp\left[-j\frac{\pi\mu k(k+1)}{N_{ZC}}\right], 0 \le k \le 1776 \tag{5.14}$$

where $N_{ZC} = 1777$, μ is a positive integer with $0 < \mu < N_{ZC}$. Different cyclically shifting points N_{CS} for the μ-th root ZC sequence will generate a new sequence. The cyclic shift point is set as $N_{CS} = 888$ in DRC system, which is explained in Annex A.4. Therefore, there are 3552 distinct preamble sequences in the DRC uplink.

All random access sequences are indexed by n_{zc}, the range of which is from 0 to 3551. For the preamble sequence with index n_{zc}, the sequence shall be defined as equations below:

$$Z_{n_{zc}}(k) = \begin{cases} \exp\left[-j\frac{\pi(n_{zc}+1)k(k+1)}{N_{ZC}}\right], & 0 \le n_{zc} < N_{ZC}-1 \\ \exp\left[-j\frac{\pi(n_{zc}-N_{ZC}+1)(k+N_{cs}-1)(k+N_{cs})}{N_{ZC}}\right], & N_{ZC}-1 \le n_{zc} \le 2N_{ZC}-3 \end{cases} \tag{5.15}$$

PRACH subcarriers have one tenth the subcarrier spacing of data subcarriers. The ZC sequence $Z_{n_{zc}}(k)$ shall be mapped to PRACH subcarriers in the frequency domain as shown below:

$$X_{n_{zc}}(k) = \begin{cases} Z_{n_{zc}}(k), & 0 \le k \le 888, \\ 0, & 889 \le k \le 1159, \\ Z_{n_{zc}}(k-271), & 1160 \le k \le 2047. \end{cases} \tag{5.16}$$

The random access sequence then passes through a 2048-point IFFT to generate the PRACH sequence in the time domain. The time-continuous random access signal $x_{n_{zc}}(t)$ shall be defined by follow equation.

$$x_{n_{zc}}(t) = \sum_{k=-1024}^{1023} X_{n_{zc}}(\hat{k}) e^{j2\pi(k+905)\Delta f^{RA}(t-N_{CP}^{RA}T_S^{RA})} \tag{5.17}$$

where $0 \le t < (N_{IFFT} + N_{CP}^{RA}) \times T_S^{RA}$, $\hat{k} = mod(k+2048, 2048)$ and $f^{RA} = 488.3\ Hz$

The Guard Time (GT) shall be padded with zeros at the end of the PRACH sequence in the time domain.

All of the parameters for random access are summarized in Table 5.8.

Table 5.8 PRACH Parameters

Parameter	Value
Sampling rate (f_S^{RA})	1 MHz
Time period between samples (T_S^{RA})	1 µsec
PRACH subcarrier spacing (Δf^{RA})	488.3 Hz
Preamble Type	ZC sequence
Preamble Length	1777 points
PRACH Cyclic prefix (N_{CP}^{RA})	793
PRACH Cyclic prefix duration	793 µsec
PRACH Sequence $\left(N_{SEQ}^{RA}\right)$	2048
PRACH Sequence duration	2048 µsec
Guard Time (GT) duration	771.8 µsec
Total duration of PRACH	3612.8 µsec
PRACH occupied bandwidth (BW^{RA})	878.9 kHz

5.9 Synchronization Procedure

The DRC system uses a synchronization procedure to ensure that a BTS synchronously receives all uplink frames transmitted by different BATs. This is achieved by compensating for propagation delays and aligning all uplink frames to the BTS time reference.

The DRC system works with uplink frame length equal to $T_F = 10msec$. A BTS should set up its time reference by Global Navigation Satellite System (GNSS) or Network Time Protocol (NTP) [4]. The time reference shall be time aligned with 1pps timing pulse and divided into 100 intervals (uplink frames) for each second. Every set of 1000 intervals (or uplink frames) shall begin at an integer multiple of ten seconds and the intervals (uplink frames) within each set shall be indexed from 0 to 999.

In order to minimize interference, the following process shall be followed in the DRC system:

1) For the i th downlink frame, a BTS shall include the **L1D_time_sec**, **L1D_time_msec, L1D_time_usec**, **L1D_time_nsec** fields in that frame's preamble. The start time of the bootstrap's first sample for the ith frame is $t_{bootstrap}(i)$. The start time of the closest 10ms interval that begins prior to or coincident with the bootstrap of the ith downlink frame is $t_{relativetime}(i)$. The elapsed time is defined as $T_{L1}(i) = t_{bootstrap}(i) - t_{relativetime}(i)$.
2) A BAT will detect the first sample of the bootstrap belonging to the ith downlink frame at time $t_{sync}(i) = t_{bootstrap}(i) + T_{delay}$, where T_{delay} is the propagation delay. The BAT shall decode L1-Detail to obtain **L1D_time_sec**, **L1D_time_msec**, **L1D_time_usec**, **L1D_time_nsec**. The BAT shall then calculate $T_{L1}(i)$ from those fields.
3) The BAT shall reconstruct a time reference with 10ms intervals using an internal 10 MHz clock. The start time of the reconstructed time reference shall be $t_{sync}(i) - T_{L1}(i)$.
4) Multiple 10 msec intervals will occur during the ith downlink frame. The BAT shall randomly select a 10 msec interval using a uniform distribution and transmit an uplink PRACH signal in that randomly selected 10 msec interval.

5) When the BTS detects an uplink PRACH signal, the BTS shall calculate the T_{delay} and transmit it to the BAT via PLP-R. The BAT shall then adjust the reconstructed time reference to compensate for T_{delay} during its next transmission.

The synchronization procedure is shown in Figure 5.9.

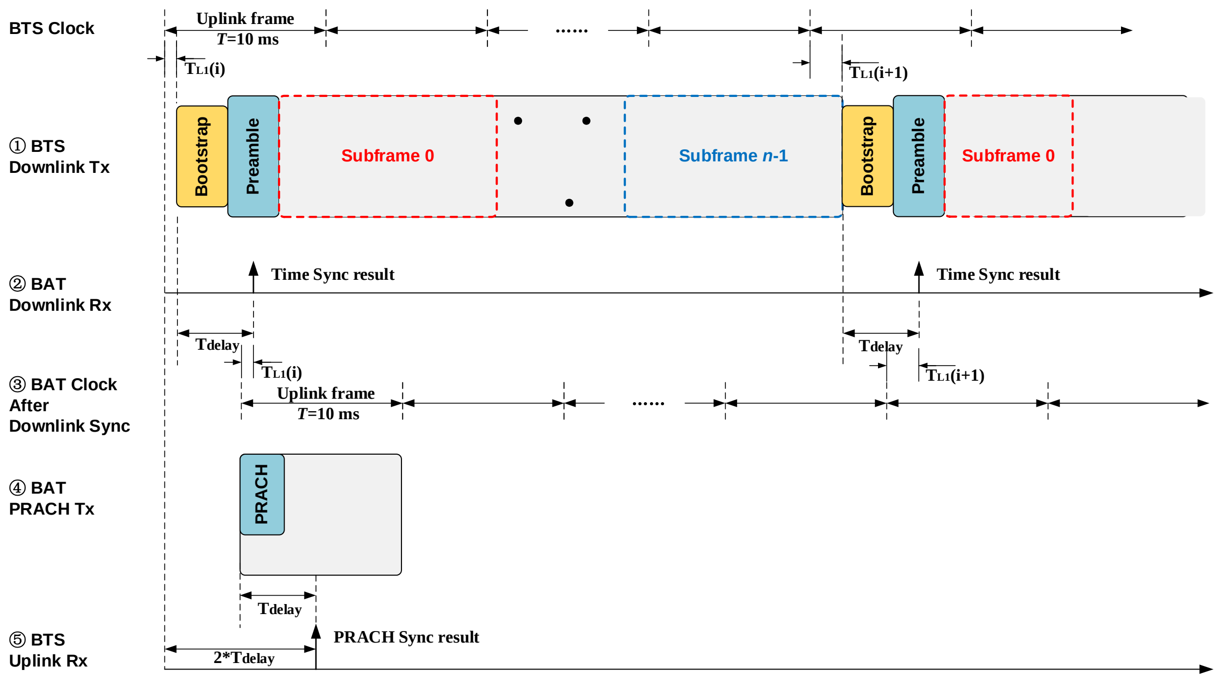

Figure 5.9 Synchronization procedure.

5.10 Mapping from DRC Uplink PDU to Transport Block (TB)

The CTC coding block is defined as a Transport Block (TB). The size of a TB shall be equal to the number of bits input to the CTC block defined in Section 5.2.

A DRC uplink MAC PDU is composed of different subheaders and their respective payloads (refer to Section 6.3). The size of an uplink MAC PDU may be larger than the size of a transport block, in which case the MAC PDU shall be fragmented into multiple TBs for transmission. When the size of a MAC PDU is smaller than the size of a TB, concatenation of multiple MAC PDUs into a TB shall not be allowed in the DRC uplink.

As shown in Figure 5.10, a TB is divided into three parts, TB Header, TB Payload and padding at the end of TB if the size of a MAC PDU is smaller than the size of the TB.

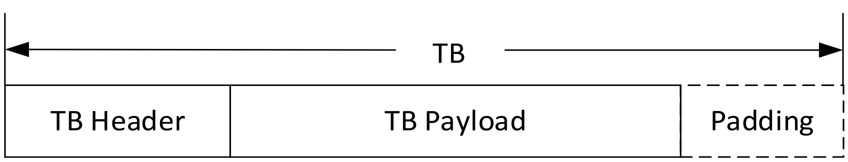

Figure 5.10 Transport Block (TB).

The structure of the TB header shall be as shown in Figure 5.11. FI, E, TBSN, C, R, PL are abbreviations of **fragment_indication**, **extension**, **tb_sequence_number**, **continuation**, **reserved** and

padding_length. The syntax of the transport_block() header shall be as specified in Table 5.9. The semantics of the fields in the transport_block() shall be as given immediately below the table.

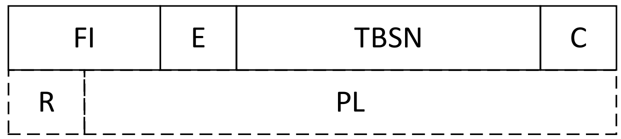

Figure 5.11 TB Header.

Table 5.9 Transport Block Header Syntax

Syntax	No. of Bits	Format
transport_block () {		
fragment_indication	2	uimsbf
extension	1	uimsbf
tb_sequence_number	4	uimsbf
continuation	1	uimsbf
reserved	1	'1'
padding_length	7	uimsbf
}		

fragment_indication – This field shall indicate the transport block payload fragment of a MAC PDU.

If **fragment_indication** is set to '0', the payload in this transport block is a whole MAC PDU.

If **fragment_indication** is set to '1', the payload in this transport block is the first fragment of a MAC PDU.

If **fragment_indication** is set to '2', the payload in this transport block is the last fragment of a MAC PDU.

If **fragment_indication** is set to '3', the payload in this transport block is one fragment of a MAC PDU, but neither the first nor the last fragment.

extension – This field shall indicate the transport block payload start point.

If **extension** is set to '0', the payload in this transport block starts from the second byte.

If **extension** is set to '1', the payload in this transport block starts from the third byte. The first bit of the second byte in this transport block is reserved. The last 7 bits of the second byte in this transport block indicate the padding length.

tb_sequence_number – This field shall indicate the transport block sequence number.

If a MAC PDU is not fragmented, **tb_sequence_number** shall always be set to '0'. If a MAC PDU is fragmented, **tb_sequence_number** of the first fragment shall be set to '1' and following fragments shall be incremented by '1'. After **tb_sequence_number** reaches '15', it shall be wrapped to '1'.

continuation – This field shall indicate the current transport block payload is a continuation from the last DRC uplink MAC PDU.

This field is only valid when **tb_sequence_number** is larger than '1'. When **tb_sequence_number** = '1' and the current transport block is the first fragment of the MAC PDU, **continuation** shall be set to '0'. When **tb_sequence_number** is not '0' and the payload in this transport block is a continuation of the previous DRC uplink MAC PDU, **continuation** shall be set to ‘1’. When **tb_sequence_number** = ‘0’, **continuation** shall be set to ‘0’.

padding_length – This field shall indicate the transport block payload length of padding bytes.

6. MAC SPECIFICATION

6.1 Introduction

This chapter defines the MAC (Media Access Control) protocol of the DRC, which specifies functions for resource sharing in the uplink channel and the control signaling required in the downlink.

6.1.1 Services

The MAC layer services are defined as:

1. Services provided to upper layers
   a) Packet Data;
   b) Reconfiguration of the uplink wireless resources.
2. Services required from the physical layer
   a) Received Data after error correction;
   b) Report of channel status information.

6.1.2 Functions

1. Multiplexing of data packets from different connections to MAC PDU;
2. De-multiplexing of MAC PDU packets;
3. Resource scheduling.

6.2 MAC Procedures

6.2.1 Random Access Procedure

The random access procedure describes how a Broadcast Access Terminal (BAT) initiates an access request through the random access channel, and establishes a wireless connection between the BAT and BTS (Broadcast Television Station).

BTS shall notify all BATs the network configuration through downlink Broadcast Control Information (BCI). When needed, a BAT initiates random access based upon information obtain from BCI. The process of ranging access includes the following 6 steps:

1) The BAT shall acquire downlink frequency and time synchronization from the downlink bootstrap as specified in Section 5.9.
2) The BAT shall get the system information from the downlink broadcast control information (BCI). The BAT shall randomly select, with a uniform probability distribution, a random access sequence from the defined sequence set, shall transmit the selected random access sequence in the uplink PRACH channel, and shall start the Ranging Response Timer (RRT). The duration of the Ranging Response Timer (RRT) shall be equal to $10N_{RR}$ msec, where N_{RR} is the number of uplink frames in waiting after sending a random access sequence and each N_{RR} frame duration is 10 msec. The value of N_{RR} is **frame_wait_count** broadcasted by the BTS in BCI.
3) When a DRC random access sequence is successfully received by the BTS, the BTS shall transmit a random access response in the downlink channel, including the root value of

the random access sequence, time adjustment, power adjustment, and scheduled resource for further registration request.

4) When a random access response is correctly received by the BAT, the BAT shall stop the RRT timer, shall adjust its time and transmission power according to the received random access response, shall send a registration request to the BTS using the resources allocated in step 3, and shall start the Registration Confirmation Response Time (RCRT). The duration of RCRT shall equal to $10N_{RR}$ msec.
5) After receiving the uplink registration request message, the BTS shall transmit a registration confirmation response message including the TUID and the first Connection ID (CID) to the BAT.
6) When a BAT does not receive the random access response before the timeout of RRT timer or does not receive the RCRT timer before the timeout of RCRT timer, the BAT shall declare a failure in this ranging access and shall initiate a backoff. When a backoff ends, the BAT shall schedule another ranging until the maximum number of ranging attempts is reached.

The flow chart of a successful ranging access procedure is illustrated in Figure 6.1.

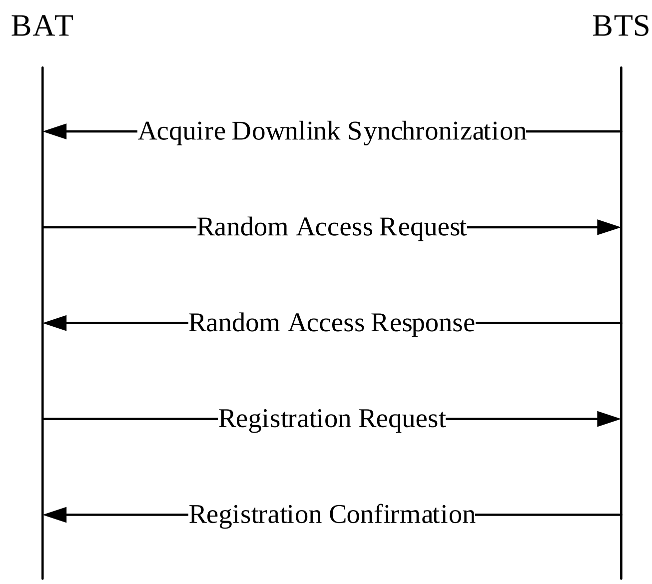

Figure 6.1 Ranging access procedure.

The Binary Exponential Backoff (BEB) mechanism shall be used in DRC. The user's initial backoff window size and the maximum backoff window size are indicated by the BTS in the downlink BCI. After a BAT retransmits a random access sequence, the BAT shall double its current backoff window size until the maximum backoff window size is reached or exceeded. When a BAT's current backoff window size exceeds the maximum backoff window size, the BAT shall set its current backoff window size to equal the maximum backoff window size.

When the current backoff window size for a BAT which needs to retransmit a random access sequence is W, the BAT shall randomly choose an integer value y from the interval $[0, W-1]$ with uniform distribution, shall reselect a random access sequence, and shall transmit that random access sequence in the uplink frame, which occurs y uplink frames after the current frame. When the maximum number of random access sequence transmissions is reached, the BAT shall stop the random access procedure and shall report the event to upper layers.

In the random access procedure, the transmission power shall be set as the following equation.

$$P_{tx} = P_{rx}^{\min} + L_{dl} + (N_{trans} - 1) \times P_{step} \tag{5.18}$$

where P_{tx} is the transmission power, L_{dl} is the estimated downlink path loss, N_{trans} is the number of transmissions (including the current transmission), P_{rx}^{min} is the minimum received power at the BTS for correct decoding of the data, and P_{step} is the power ramp step after a transmission failure. Both P_{rx}^{min} and P_{step} are broadcasted by the BTS in the downlink BCI.

The flow charts for the ranging access operation at the BAT and the BTS sides are shown in Figure 6.2 and Figure 6.3, respectively.

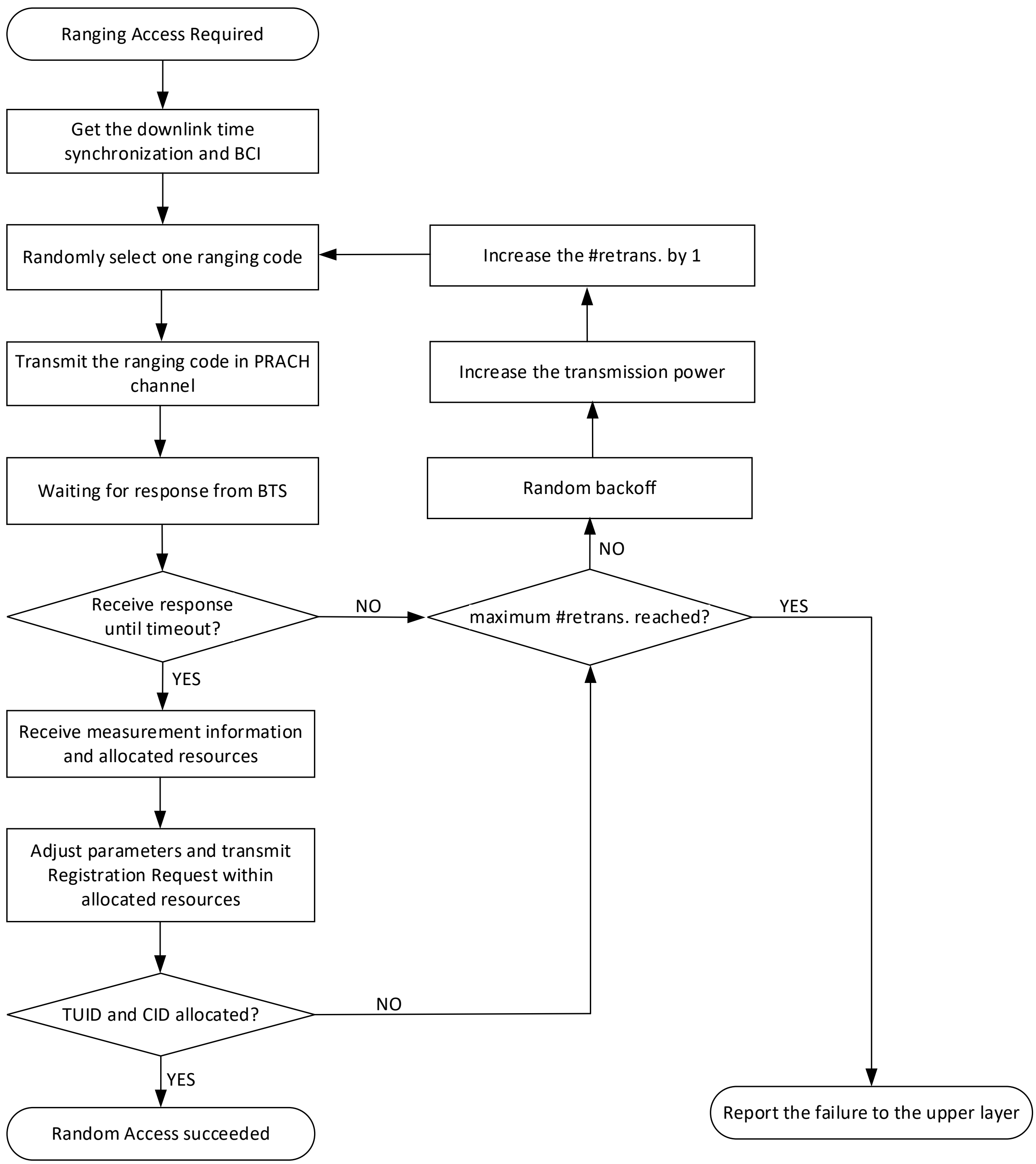

Figure 6.2 Ranging process at the BAT.

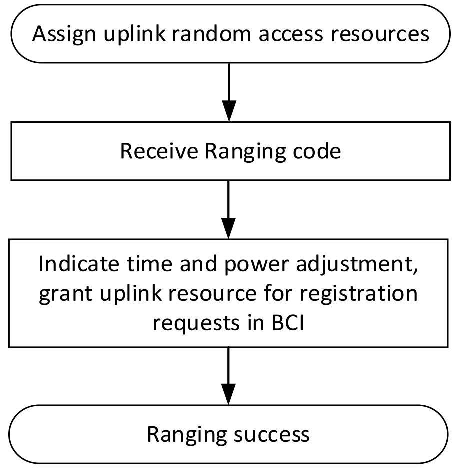

Figure 6.3 Ranging process at the BTS.

6.2.2 Registration Request

Registration request shall be used by a BAT to register to the network, after the ranging process succeed. In the registration process, the BAT shall send its MAC address to BTS, and the BTS shall send back the assigned temporary user ID (TUID) to the receiver. The length of MAC address is 48 bits, which can accommodate an extremely large number of users. However, the maximum number of users covered by a broadcast tower is much limited. Consequently, a shorter TUID is used to reduce the overhead of user ID. The length of TUID is defined as 18 bits. Therefore, the maximum number of users supported by a BTS is 262,144.

6.2.3 Uplink Resource Request

An uplink resource request shall be used by a BAT to request resources for uplink data transmission. When there is no uplink connection and a BAT wishes to perform uplink data transmission, the BAT shall request uplink resources with a random access ranging procedure. Conversely, when there is at least one active connection between a BTS and a BAT, the BAT can send buffer status report for resource request for an existing connection. The BAT can also send a connection establishment request for establishment of a new connection and then send bandwidth allocation request for additional uplink resources.

6.2.4 Paging Request

When the BTS needs to contact a specific BAT that is powered on and registered, but it has no active connection, the BTS has to use paging. Idle BATs that are completely turned off or just monitoring the bootstrap for Emergency Alert wake-ups cannot be contacted via paging. The paging information, which includes the MAC address of the paged BAT and the uplink random access sequence allocated to that BAT for ranging, is broadcasted in the downlink BCI. When a BAT receives a paging request intended for that BAT, the BAT shall start a connection establishment procedure.

The flow chart of paging at the BAT side is shown in Figure 6.4. Each powered on and registered BAT without active connections shall monitor the downlink MAC PDUs in the PLP-R for paging subheaders. When an active BAT receives a Paging Request, the BAT shall send the assigned random access sequence in the allocated resources, and shall wait for a response from the BTS. When a response is received, the BAT shall adjust its timing and power according to the measurement information, and shall transmit a registration request within the allocated resources. Paging succeeds when the BAT receives a confirmation message from the BTS, before the RRT timer expires.

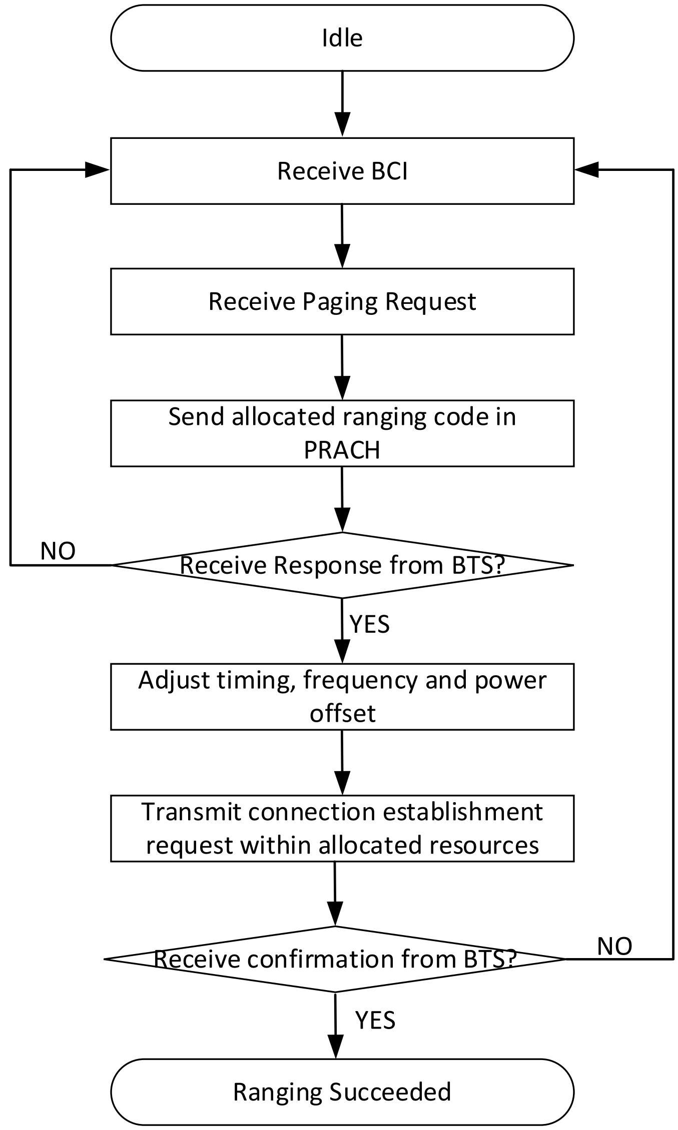

Figure 6.4 Flow chart of paging at the BAT side.

The flow chart of paging at the BTS side is shown in Figure 6.5. A BTS sends Broadcast and Control Message periodically and Paging Requests to BATs as requested by upper layers. When the BTS successfully decodes the random access sequence from a paged BAT, the BTS will send a response to the corresponding BAT containing a time advance and power adjustment, as well as resource allocation message. When no transmissions are received for the allocated random access sequence and the maximum number of transmission attempts is not reached, the BTS will send a Paging Request again. When the BTS receives a registration request from the BAT, the BTS will transmit a confirmation message and the paging is considered to be successful. When the maximum number of transmission attempts is reached and no connection establishment request is received from the paged BAT, the paging is considered to have failed. The maximum number of transmission attempts can be configured in the downlink BCI.

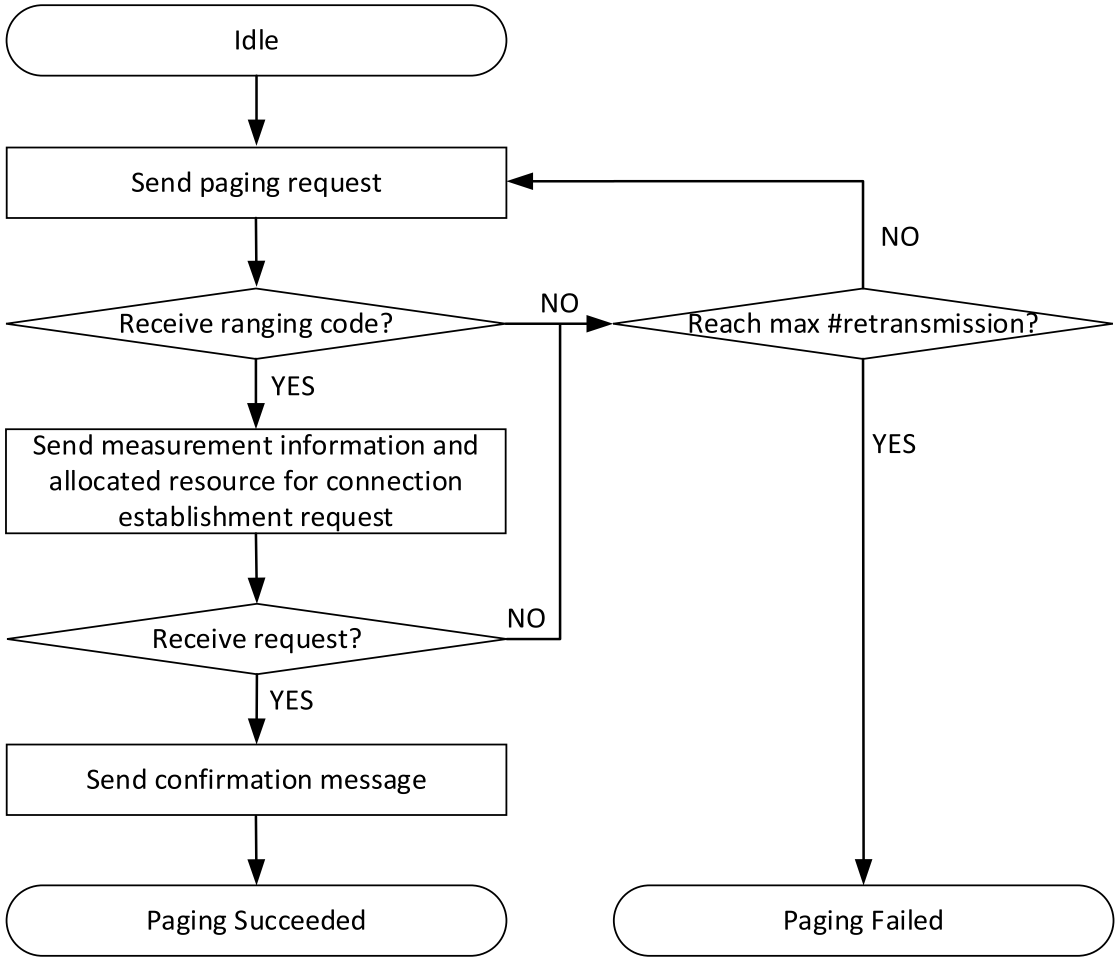

Figure 6.5 Flow chart of paging at the BTS side.

6.2.5 ARQ Procedures

Window-based selective ARQ procedures are performed for both the DRC uplink MAC PDUs and the DRC downlink MAC PDUs. The sequence number (SN) field defined in the header of each PDU is used to identify that specific PDU.

6.2.5.1 Parameters Definition

Three parameters, *TxW, RxW, and MRC,* are defined for the window-based selective ARQ.

a) TxW: Transmission Window. It defines the maximum allowed difference between the newest sequence number of the DRC uplink/downlink MAC PDU having been transmitted and the oldest sequence number of the MAC PDU that has not yet been acknowledged.
b) RxW: Receiving Window. It defines the maximum allowed difference between the newest sequence number of the DRC uplink/downlink MAC PDU having been received and the newest sequence number of the DRC uplink/downlink MAC PDU having not been received.
c) MRC: Maximum Retransmission Count for normal DRC uplink/downlink MAC PDU ;

6.2.5.2 Transmitter-side variables

Three transmitter-side variables, *V(S), V(AS), and V(MS)*, are defined for the window-based selective ARQ as follows:

a) $V(S)$: Sequence Number for Sending. It defines the sequence number of the next DRC uplink/downlink MAC PDU to be transmitted for the first time (i.e. excluding retransmitted DRC uplink/downlink MAC PDUs). The initial value of $V(S)$ shall be 0. The maximum value of $V(S)$ is 255. When $V(S)$ is larger than 255, it shall be re-set to $V(S)$ modulo 256.

b) $V(AS)$: Sequence Number waiting for Acknowledgement at the Sender side. It defines the sequence number of the DRC uplink/downlink MAC PDU following the last in-sequence acknowledged DRC uplink/downlink MAC PDU. That is, $V(AS)$ represents the sequence number of the oldest DRC uplink/downlink MAC PDU that has not yet been acknowledged. The initial value of $V(AS)$ shall be 0. The maximum value of $V(AS)$ is 255. When $V(AS)$ is larger than 255, it shall be re-set to $V(AS)$ modulo 256.

c) $V(MS)$: Maximum Sequence Number for Sending. It defines the upper limit on the sequence number of DRC uplink/downlink MAC PDUs that may be transmitted. A BAT shall not transmit a DRC uplink/downlink MAC PDU with a sequence number greater than or equal to $V(MS)$. $V(MS) = (V(AS) + TxW)$ modulo 256. $V(MS)$ shall be updated when $V(AS)$ or TxW are updated. The initial value of $V(MS)$ shall be TxW.

6.2.5.3 Receiver-side variables

Three receiver-side variables, *V(AR), V(R), and V(MR)*, are defined for the window-based selective ARQ as follows:

a) $V(AR)$: Sequence Number without Acknowledgement for Receiving. It defines the sequence number of the DRC uplink/downlink MAC PDU following that of the last in-sequence DRC uplink/downlink MAC PDU received. $V(AR)$ shall be updated upon the receipt of the DRC uplink/downlink MAC PDU with sequence number equal to $V(AR)$. $V(AR)$ shall be updated to the sequence number of the oldest DRC uplink/downlink MAC PDU that has not yet been received. The initial value of $V(AR)$ shall be 0. The maximum value of $V(AR)$ is 255. When $V(AR)$ is larger than 255, it shall be re-set to $V(AR)$ modulo 256.

b) $V(R)$: Sequence Number for Receiving. It defines the sequence number of the DRC uplink/downlink MAC PDU following the newest sequence number of all received DRC uplink/downlink MAC PDUs. Let x denote the sequence number of a DRC uplink/downlink MAC PDU that has been received. If $V(R) \leq V(R) < V(MR) \leq 255$, $V(R)$ shall be set to $(x + 1)$ modulo 256 and the DRC uplink/downlink MAC PDU shall be processed if it has not been received before. When $V(MR) < V(R) \leq 255$, if $V(R) \leq x \leq 255$ or $x < V(MR)$, $V(R)$ shall be set to $(x + 1)$ modulo 256 and the DRC uplink/downlink MAC PDU shall be processed if it has not been received before. The initial value of $V(R)$ shall be 0. The maximum value of $V(R)$ is 255. When $V(R)$ is larger than 255, it shall be re-set to $V(R)$ modulo 256.

c) $V(MR)$: Maximum sequence number for Receiving. It defines the upper limit on the sequence numbers of the DRC uplink/downlink MAC PDUs that shall be processed when received. When a DRC uplink/downlink MAC PDU is received with sequence number x such that $x < V(R)$ or $x \geq V(MR)$, the DRC uplink/downlink MAC PDU shall be discarded. $V(MR) = V(AR) + RxW$. $V(MR)$ shall be updated when $V(AR)$ or RxW is updated. The initial value of $V(MR)$ shall be RxW. When $V(MR)$ is larger than 255, it shall be re-set to $V(MR)$ modulo 256.

6.2.5.4 Timer Definition

A timer is defined for the window-based selective ARQ:

a) Timer_ACK: This timer is used by the sender to trigger retransmission of a DRC uplink/downlink MAC PDU, when no ACK message is received for that DRC uplink/downlink MAC PDU. When a DRC uplink/downlink MAC PDU is transmitted, Timer_ACK for that DRC uplink/downlink MAC PDU shall be started. When a positive or negative acknowledgement is received for a DRC uplink/downlink MAC PDU, the corresponding Timer_ACK shall be stopped. When neither a positive nor negative acknowledgement is received for a DRC uplink/downlink MAC PDU before the corresponding Timer_ACK expires, the DRC uplink/downlink MAC PDU shall be retransmitted. The value of Timer_ACK is configured by BCI.

6.2.5.5 Sender operation

Upon transmission of a new DRC uplink/downlink MAC PDU or upon retransmission of a previously transmitted DRC uplink/downlink MAC PDU, the sender shall process as follows:

a) If the MAC PDU is sent for the first time, the SN field of the DRC uplink/downlink MAC PDU shall be set equal to $V(S)$.
b) The transmission count of the DRC uplink/downlink MAC PDU shall be incremented by 1.
c) When the transmission count of a DRC uplink/downlink MAC PDU is equal to the Maximum Retransmission Count (MRC), this event shall be reported to upper layers.
d) If the transmission count is smaller than MRC:
    i. If $V(S)$ satisfies either of the two conditions: 1) $V(AS) \leq V(S) < V(AS) + T_xW \leq 255$, 2) $V(S) < V(AS)$ and $V(S) < V(MS)$, the DRC uplink/downlink MAC PDU shall be sent to the physical layer for transmission and Timer_ACK for that DRC uplink/downlink MAC PDU shall be started.
    ii. If $V(S)$ satisfies either of the two conditions: 1) $V(AS) + T_xW < V(S) \leq 255$, 2) $V(S) < V(AS)$ and $V(S) < V(MS)$, the DRC uplink/downlink MAC PDU shall be constructed while not to be transmitted at the present time.

Upon receipt of an ACK message, the sender shall process as follows:

a) When an acknowledged sequence numbers in the ACK message is larger than $V(MS)$, a connection crash message shall be sent to upper layers. Otherwise, $V(AS)$ and $V(MS)$ shall be updated according to the received ACK message. $V(AS)$ shall be set to the sequence number in the ACK message plus 1 modulo 256. $V(MS)$ shall be set to $V(AS) + T_xW$ modulo 256.
b) When there exists one or more negatively acknowledged DRC uplink/downlink MAC PDUs according to the ACK message, these MAC PDUs shall be considered for retransmission.
c) If a DRC uplink/downlink MAC PDU is negatively acknowledged more than once before the DRC uplink/downlink MAC PDU is retransmitted, the MAC PDU shall be considered for retransmission only once at the current time.

Upon expiration of Timer_ACK for a DRC uplink/downlink MAC PDU, the sender shall consider the DRC uplink/downlink MAC PDU for retransmission.

6.2.5.6 Receiver operation

Upon receipt of a DRC uplink/downlink MAC PDU, the receiver shall process as follows:

a) If the SN of the DRC uplink/downlink MAC PDU is smaller than $V(AR)$ or larger than $V(MR)$:
   i. Discard the DRC uplink/downlink MAC PDU.
b) If a packet with the same SN of the DRC uplink/downlink MAC PDU has been received:
   i. Discard the DRC uplink/downlink MAC PDU.
c) Else
   i. Update $V(AR)$, $V(R)$ and $V(MR)$ according to the rule defined in Section 6.2.5.3.
   ii. If there exists an unsent ACK message, update the length of the bitmap and the Bitmap in the existing ACK message;
   iii. If no ACK message exists, build a new ACK message, write the ACK information into the message, and notify the MAC scheduler;
   iv. If an ACK message is scheduled for transmission, the ACK message shall be included in a DRC uplink/downlink MAC PDU and the local ACK message shall be deleted.

6.3 Uplink MAC PDU Format

The structure of the Uplink MAC PDU (Protocol Data Unit) is shown in Figure 6.6. It includes four parts, i.e., MAC Header, Subheader MAC Payloads, Data Payload and Cyclic Redundancy Check (CRC). The CRC for an uplink MAC PDU shall be calculated over the contents of that MAC PDU's header and payload using the CRC defined in Section 5.2.1.

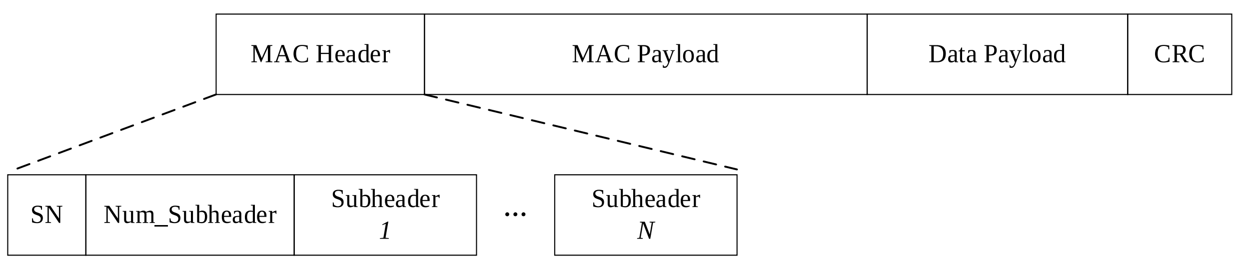

Figure 6.6 Structure of Uplink MAC PDU.

The syntax of an uplink_mac_header() shall conform to Table 6.1. The semantics of the fields in the uplink_mac_header() shall be as given immediately below the table. The sequence number (SN) in the uplink MAC Header shall start from 0x00, and shall increment by 1, whenever a new uplink MAC PDU is transferred. After reaching 0xFF, the SN shall be wrapped to 0x00. The uplink MAC PDU shall be byte aligned, i.e., its length shall be a multiple of 8 bits. The lengths of the MAC header and the payload are variable according to different situations.

Table 6.1 DRC Uplink MAC Header Syntax

Syntax	No. of Bits	Format
uplink_mac_header() {		
sequence_number_up	8	uimsbf
num_subheader_up	4	uimsbf
reserved	4	'1111'
for (i=0; i< **num_subheader_up**; i++) {		
subheader_type_up	4	Table 6.2
reserved	4	'1111'
}		
data_payloads_up	variable	uimsbf
}		

sequence_number_up – An 8-bit unsigned integer field that shall specify the sequence number of the current uplink MAC PDU.

num_subheader_up – A 4-bit unsigned integer field that shall specify the number of uplink subheaders included minus one.

data_payloads_up – Data payloads concatenated together in order according to the uplink data subheaders specified within the 'for' loop.

6.3.1 Definition of Uplink subheader Types

Uplink subheader types shall be as defined in Table 6.2. The priorities defined in Table 6.2 are used during the generation of a DRC uplink MAC PDU as specified in Section 6.3.2. A lower priority value shall correspond to a higher priority.

Table 6.2 DRC Uplink subheader Types

Value	Priority	Description
0	0	Registration Request
1	0	Uplink ACK Message
2	0	Connection Release Request
3	0	Power Down Request
4	1	Bandwidth Allocation Request
5	1	Status Report Message
6	2	Uplink Data
7~15		reserved

6.3.2 Assembly of DRC Uplink MAC PDU

The size of a DRC Uplink MAC PDU is determined by the resources allocated to the BAT by the BTS in UL-MAP packet defined in Section 6.5.2.

An uplink MAC PDU shall be constructed according to the subheader priorities defined in Table 6.2. Applicable subheaders with higher priority shall be added first to the uplink MAC PDU. Applicable subheaders with lower priority shall then be added to the uplink MAC PDU, if space is available. When the priorities of multiple applicable subheaders are the same, the subheader with the smallest subheader type value shall be included first.

6.4 Definitions of DRC Uplink subheader Payloads

6.4.1 Payload of Registration Request subheader

The syntax of the Registration Request subheader shall conform to Table 6.3. The semantics of the fields in the registration_request_subheader() shall be as given immediately below the table.

Table 6.3 Registration Request subheader Syntax

Syntax	No. of Bits	Format
registration_request_subheader() {		
user_id	48	uimsbf
qci	3	See Table 6.4
reserved	5	‘11111’
}		

user_id – A 48-bit unsigned integer field that shall specify the MAC address of the BAT.

Table 6.4 QoS Class Identifier

QCI	Service type	Priority	Packet Delay Budget	Packet Error Rate	Example Services
0	GBR	1	50 msec	10^{-3}	Real Time Gaming
1	GBR	2	100 msec	10^{-2}	Conversational voice
2	GBR	3	150 msec	10^{-3}	Conversational Video (Live Streaming)
3	GBR	4	300 msec	10^{-6}	Non-Conversational Video
4	Non-GBR	5	10 sec	10^{-6}	Voting, comments, program preference
5~7	reserved				

6.4.2 Payload of Uplink ACK Message subheader

The syntax of the Uplink ACK Message subheader shall conform to Table 6.5. The semantics of the fields in the ack_message_subheader() shall be as given immediately below the table.

Table 6.5 ACK Message subheader Syntax

Syntax	No. of Bits	Format
ack_message_subheader() {		
bitmap_length	8	uimsbf
first_sequence_number	8	uimsbf
bitmap	variable	uimsbf
padding	variable	uimsbf
}		

bitmap_length – An 8-bit unsigned integer field that shall specify the length of the bitmap minus 1.

first_sequence_number – A 8-bit unsigned integer field that shall specify the sequence number represented by the first bit in the bitmap.

bitmap – A variable-bit unsigned integer field that shall specify the meaning of the bits in the Bitmap. The meaning of the bit defined by bit_position ∈ [0, LB] shall be:

0x0: The uplink MAC PDU with SN = (**first_sequence_number** + bit_position) has not been correctly received.

0x1: The uplink MAC PDU with SN = (**first_sequence_number** + bit_position) has been correctly received.

padding – A variable-bit unsigned integer field that shall fill the **bitmap** for byte alignment.

6.4.3 Payload of Connection Release Request subheader

When a connection between a BAT and the BTS is released by upper layers or a connection is idle for more than 50 seconds, the connection shall be released and the Connection Release subheader shall be transmitted by the BAT. The syntax of the Connection Release Request subheader shall conform to Table 6.6. The semantics of the fields in the connection_release_request_subheader() shall be as given immediately below the table.

Table 6.6 Connection Release Request subheader Syntax

Syntax	No. of Bits	Format
connection_release_request_subheader() {		
cid	5	uimsbf
reserved	3	'111'
}		

cid – A 5-bit unsigned integer field that shall specify the connection identification. A value of '0' shall be reserved for the request for a connection and not valid here. All other values are the connection identification.

6.4.4 Payload of Power Down Request subheader

When a BAT receives power down command or the power down is detected by a BAT, the power down request subheader shall be transmitted by the BAT. The payload of the Power Down Request subheader shall be empty.

6.4.5 Payload of Bandwidth Allocation Request subheader

In the case of DRC uplink data without scheduled transmission resource in the buffer of a BAT, the BAT shall transmit bandwidth allocation request subheader. The syntax of the Bandwidth Allocation Request subheader shall conform to Table 6.7. The semantics of the fields in the bandwidth_allocation_request_subheader() shall be as given immediately below the table.

Table 6.7 Bandwidth Allocation Request subheader Syntax

Syntax	No. of Bits	Format
bandwidth_allocation_request_subheader() {		
tuid	18	uimsbf
cid	5	uimsbf
qci	3	Table 6.4
reserved	6	'111111'
}		

tuid – An 18-bit unsigned integer field that shall specify the temporary user identification.

6.4.6 Payload of Status Report subheader

When a BAT sends the bandwidth allocation request or a BAT is requested by the BTS through status report request subheader, the BAT shall transmit the status report subheader. When sending bandwidth allocation request, the BAT shall send the status report subheader together with the bandwidth allocation request subheader. When responding the status report request, the status report subheader shall be transmitted in the next DRC uplink transmission, after receiving the status report request from the BTS.

The syntax of the Status Report subheader shall conform to Table 6.8. The semantics of the fields in the status_request_subheader() shall be as given immediately below the table.

Table 6.8 Status Report subheader Syntax

Syntax	No. of Bits	Format
status_request_subheader() {		
buffer_status	6	See Table 6.9
channel_estimation	6	uimsbf
transmission_power	6	uimsbf
num_connections	5	uimsbf
reserved	1	‘1’
}		

channel_estimation – A 6-bit unsigned integer field that shall specify the SINR of the bootstrap in dB. The actual value of SINR shall be equal to the value given here minus 16. For example, 0x00 and 0x3F indicate the SINR of bootstrap as -16 dB and 47 dB, respectively.

transmission_power – A 6-bit unsigned integer field that shall specify the transmission power used by BAT in dBm. The transmission power in dBm is equal to the value given here minus 33. For example, 0x00 and 0x3F indicate the actual transmission power of -33 dBm and 30 dBm, respectively.

num_connections – A 5-bit unsigned integer field that shall specify the number of active connections.

Table 6.9 Buffer Status Definition

Index	Buffer Depth (BD) [bytes]	Index	Buffer Depth value [bytes]
0	$BD = 0$	32	$1132 < BD \leq 1326$
1	$0 < BD \leq 10$	33	$1326 < BD \leq 1552$
2	$10 < BD \leq 12$	34	$1552 < BD \leq 1817$
3	$12 < BD \leq 14$	35	$1817 < BD \leq 2127$
4	$14 < BD \leq 17$	36	$2127 < BD \leq 2490$
5	$17 < BD \leq 19$	37	$2490 < BD \leq 2915$
6	$19 < BD \leq 22$	38	$2915 < BD \leq 3413$
7	$22 < BD \leq 26$	39	$3413 < BD \leq 3995$
8	$26 < BD \leq 31$	40	$3995 < BD \leq 4677$
9	$31 < BD \leq 36$	41	$4677 < BD \leq 5476$
10	$36 < BD \leq 42$	42	$5476 < BD \leq 6411$
11	$42 < BD \leq 49$	43	$6411 < BD \leq 7505$
12	$49 < BD \leq 57$	44	$7505 < BD \leq 8787$
13	$57 < BD \leq 67$	45	$8787 < BD \leq 10287$
14	$67 < BD \leq 78$	46	$10287 < BD \leq 12043$
15	$78 < BD \leq 91$	47	$12043 < BD \leq 14099$
16	$91 < BD \leq 107$	48	$14099 < BD \leq 16507$
17	$107 < BD \leq 125$	49	$16507 < BD \leq 19325$
18	$125 < BD \leq 146$	50	$19325 < BD \leq 22624$
19	$146 < BD \leq 171$	51	$22624 < BD \leq 26487$
20	$171 < BD \leq 200$	52	$26487 < BD \leq 31009$
21	$200 < BD \leq 234$	53	$31009 < BD \leq 36304$
22	$234 < BD \leq 274$	54	$36304 < BD \leq 42502$
23	$274 < BD \leq 321$	55	$42502 < BD \leq 49759$
24	$321 < BD \leq 376$	56	$49759 < BD \leq 58255$
25	$376 < BD \leq 440$	57	$58255 < BD \leq 68201$
26	$440 < BD \leq 515$	58	$68201 < BD \leq 79846$
27	$515 < BD \leq 603$	59	$79846 < BD \leq 93479$
28	$603 < BD \leq 706$	60	$93479 < BD \leq 109439$
29	$706 < BD \leq 826$	61	$109439 < BD \leq 128125$
30	$826 < BD \leq 967$	62	$128125 < BD \leq 150000$
31	$967 < BD \leq 1132$	63	$150000 < BD$

6.4.7 Payload of Uplink Data subheader

The syntax of the Uplink Data subheader shall conform to Table 6.10. The semantics of the fields in the uplink_data_subheader() shall be as given immediately below the table.

Table 6.10 Uplink Data subheader Syntax

Syntax	No. of Bits	Format
uplink_data_subheader() {		
cid	5	uimsbf
payload_length	11	uimsbf
}		

Payload_length – An 11-bit unsigned integer field that shall specify the number of payload bytes minus one.

6.5 Downlink Broadcast Control Data Format

The downlink control of DRC includes two parts: Downlink Broadcast Control Information (BCI) and Uplink MAP (UL-MAP). In Downlink BCI, system information is broadcasted to all BATs. UL-MAP specifies the uplink resource allocation for DRC. Each of the two control information is encapsulated into a control packet for ATSC Link-layer Protocol (ALP) [7], and transferred from the DRC downlink gateway to the ATSC 3.0 gateway through UDP/IP. At the beginning of a downlink packet, packet type (PT) is indicated by the first 3 bits and shall be as defined in Table 6.11.

Table 6.11 Downlink Packet Type (PT)

Value	Description
0x0	Downlink Broadcast Control Information (BCI).
0x1	Uplink Resource MAP (UL-MAP).
0x2	Downlink MAC PDU.
0x3 ~ 0x7	reserved

6.5.1 Downlink Broadcast Control Information (BCI)

The syntax for Downlink Broadcast Control Information packet shall conform to Table 6.12. The semantics of the fields in the downlink_broadcast_control_info() packet shall be as given immediately below the table.

Table 6.12 Downlink Broadcast Control Information Packet Syntax

Syntax	No. of Bits	Format
downlink_broadcast_control_info() {		
packet_type	3	Table 6.11
num_uplink_channels	4	uimsbf
reserved	1	'1'
for (i=0;i< **num_uplink_channels**;i++) {		
frequency	11	uimsbf
fft_size	1	uimbsf
transmission_power	6	uimbsf
tile_configuration	1	uimbsf
random_access_sequence	12	uimbsf
frame_wait_count	9	uimsbf
ranging_min_power	6	uimsbf
ranging_max_power	6	uimsbf
ranging_power_ramp_step	2	uimsbf
ranging_backoff_init_window	2	uimsbf
ranging_backoff_max_window	2	uimsbf
artt_timer	4	uimsbf
transmission_window_arq	8	uimsbf
receiving_window_arq	8	uimsbf
retrans_count_max	2	uimsbf
timer_ack	13	uimsbf
reserved	2	'11'
}		
}		

num_uplink_channels – A 4-bit unsigned integer field that shall specify the number of supported DRC uplink channels minus one.

frequency – An 11-bit unsigned integer field that shall specify the central frequency of the current DRC uplink channel in kHz with values between 0x000 to 0x3E8. Frequencies up to 1 GHz are supported. Values between 0x3E9 to 0x7FF are reserved.

fft_size – A 2-bit unsigned integer field that shall specify the number of points in an FFT. A value of '0' shall indicate 2048 point FFT. Values of '1' to '3' are reserved.

transmission_power – A 6-bit unsigned integer field that shall specify the downlink transmission power as measured in dBm minus one.

tile_configuration – A 1-bit unsigned integer field that shall specify the tile configuration for allocation. A value of '0' shall indicate tiles are allocated in time-frequency dimension. A value of '1' shall indicate tiles are allocated in frequency-time dimension. Definition of tile allocation schemes are defined in Section 5.1.

random_access_sequence – A 12-bit unsigned integer field that shall specify the available sequence for random access or paging. Let x represents the indicated value. The random access sequences with indices from 0 to x-1 shall be considered to be available for random access, and the random access sequences with indices from x to 0xDDF shall be considered to be available for paging. Values between 0xDE0~0xFFF are reserved.

frame_wait_count – A 9-bit unsigned integer field that shall specify the number of uplink frames for waiting after ranging (N_{RR}) minus one.

ranging_min_power – A 6-bit unsigned integer field that shall specify the minimum transmission power for ranging. Indicated values shall have the range:

0x00~0x32: The minimum ranging power equals to the value given here minus 50 in dBm. For example, 0x00 and 0x32 indicate -50dBm and 0dBm, respectively.

0x33~0x3F: reserved

ranging_max_power – A 6-bit unsigned integer field that shall specify the maximum transmission power for ranging. Indicated values shall have the range

0x00~0x32: The maximum ranging power equals to the value given here minus 20 in dBm. For example, 0x00 and 0x32 indicate -20dBm and 30dBm, respectively.

0x33~0x3F: reserved

ranging_power_ramp_step – A 2-bit unsigned integer field that shall specify the power ramp step in ranging. Indicated values shall have the range

0x0: 3dB

0x1: 6dB

0x2: reserved

0x3: reserved

ranging_backoff_init_window – A 2-bit unsigned integer field that shall specify the initial window size for backoff in ranging. Indicated values shall have the range

0x0: 16

0x1: 32

0x2: 64

0x3: reserved

ranging_backoff_max_window – A 2-bit unsigned integer field that shall specify the maximum window size for backoff in ranging. Indicated values shall have the range

0x0: 1024

0x1: 2048

0x2: 4096

0x3: reserved

arrt_timer – A 4-bit unsigned integer field that shall specify the ARTT timer value in msec. The ARTT timer shall be set equal to the value given here plus 1 and multiplied by 100, i.e., ARTT timer = (value given here +1)*100ms.

transmission_window_arq – An 8-bit unsigned integer field that shall specify the transmission window for ARQ minus one.

receiving_window_arq – An 8-bit unsigned integer field that shall specify the receiving window for ARQ minus one.

retrans_count_max – A 2-bit unsigned integer field that shall specify the maximum retransmission count for ARQ. Indicated values shall have the range

0x0: 4

0x1: 8

0x2: 16

0x3: reserved

timer_ack – A 13-bit unsigned integer field that shall specify the value of Timer_ACK in millisecond minus one. The maximum Timer_ACK is 8192ms maximum retransmission count for ARQ.

6.5.2 Uplink Resource Map Information

The syntax for Uplink Resource MAP packets shall conform to Table 6.13. The semantics of the fields in the uplink_resource_map() packet shall be as given immediately below the table.

Table 6.13 Uplink Resource Map Packet Syntax

Syntax	No. of Bits	Format
uplink_resource_map() {		
packet_type	3	Table 6.11
num_uplink_frames	9	uimsbf
reserved	4	'1111'
for (i=0;i< **num_uplink_frames**;i++) {		
num_users	9	uimsbf
reserved	7	'1111111'
for (i=0;i< **num_users**;i++) {		
tuid	18	uimbsf
num_connections	5	uimbsf
reserved	1	'1'
for (i=0;i< **num_connections**;i++) {		
cid	5	uimsbf
uplink_start_tile	11	uimsbf
resource_size	11	uimsbf
amc_type	3	uimsbf
reserved	2	'11'
}		
}		
}		
}		

num_uplink_frames – A 9-bit unsigned integer field that shall specify the number of uplink frames in the uplink resource map.

num_users – A 9-bit unsigned integer field that shall specify the number of users in the current uplink frame. The minimum number of users is zero, and the maximum number of users is 416. Values between 0x1A1~0x1FF are reserved.

tuid – An 18-bit unsigned integer field that shall specify the temporary user identification.

num_connections – A 5-bit unsigned integer field that shall specify the number of active connections.

uplink_start_tile – An 11-bit unsigned integer field that shall specify the index of the start tile allocated to the connection with identity **cid** of the BAT with identity **tuid** minus one.

resource_size – An 11-bit unsigned integer field that shall specify the number of tiles allocated to the connection with identity **cid** of the BAT with identity **tuid** minus one.

amc_type – A 3-bit unsigned integer field that shall specify the modulation and code rate. Indicated values shall have the range

'0': BPSK, code rate = 1/3

'1': QPSK, code rate = 1/2
'2': QPSK, code rate =3/4
'3': 16 QAM, code rate = 1/2
'4': 16 QAM, code rate = ¾
'5'~'7': reserved

6.6 Downlink MAC PDU Format

The structure of the Downlink MAC PDU is shown in Figure 6.7. It includes three parts, i.e., MAC Header, MAC Payload and Cyclic Redundancy Check (CRC). The CRC for a downlink MAC PDU shall be calculated over the contents of that MAC PDU's header and payload using the CRC defined in Section 5.2.1.

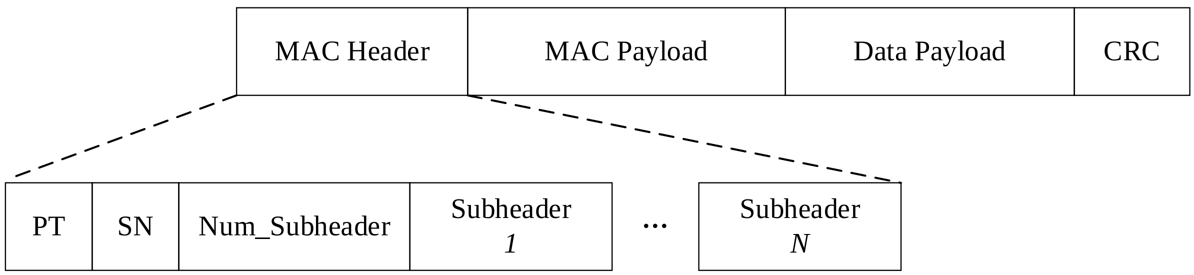

Figure 6.7 Structure of downlink MAC PDU.

The syntax of a downlink MAC header shall conform to Table 6.14. The semantics of the fields in the downlink_pdu_subheader() packet shall be as given immediately below the table. The sequence number (SN) in the downlink MAC Header shall start from 0x00, and shall increment by 1 whenever a new downlink MAC PDU is transferred. After reaching 0xFF, the SN shall be wrapped to 0x00. The downlink MAC PDU shall be byte aligned, i.e., its length shall be a multiple of 8 bits. The lengths of the MAC header and the payload are variable according to different situations.

Table 6.14 Downlink PDU subheader Syntax

Syntax	No. of Bits	Format
downlink_pdu_subheader() {		
packet_type	3	Table 6.11
sequence_number_down	8	uimsbf
num_subheader_down	15	uimsbf
for (i=0; i< **num_subheader_down**; i++) {		
subheader_type_down	4	Table 6.15
reserved	4	'1111'
}		
data_payloads_down	variable	uimsbf
}		

sequence_number_down – An 8-bit unsigned integer field that shall specify the sequence number of the current downlink MAC PDU.

num_subheader_down – A 15-bit unsigned integer field that shall specify the number of downlink subheaders included minus one.

data_payloads_down – Data payloads concatenated together in order according to the downlink data subheaders specified within the 'for' loop.

6.6.1 Definition of Downlink subheader Types

Downlink MAC PDU subheader types shall be as defined in Table 6.15.

Table 6.15 Types of Downlink MAC PDU subheaders

Value	Priority	Description
0	0	Initial Ranging Adjustment
1	0	Bandwidth Allocation Response
2	0	Status Report Request
3	0	Downlink ACK Message
4	0	Online Adjustment Message
5	1	Paging Request
6	1	Registration Confirmation
7	1	Connection Release Confirmation
8	1	Power Down Confirmation
9	2	Downlink Data
0xA~0xF		reserved

6.6.2 Assembly of the DRC Downlink MAC PDU

The size of a DRC Downlink MAC PDU is determined by the resources allocated in PLP-R.

A Downlink MAC PDU shall be constructed according to the subheader priorities defined in Table 6.15. Applicable subheaders with higher priority shall be added first to the downlink MAC PDU. When the priorities of multiple applicable subheaders are the same, the subheader with the smallest subheader type value shall be included first.

6.7 Definition of Downlink MAC subheaders

6.7.1 Payload of Initial Ranging Adjustment subheader

When the BTS correctly receives a ranging request identified by a specific random access sequence in a frame, it shall respond the initial ranging adjustment subheader to the BAT having sent the random access sequence in the frame. The syntax of the Initial Ranging Adjustment subheader shall conform to Table 6.16. The semantics of the fields in the downlink_range_init_subheader() packet shall be as given immediately below the table.

Table 6.16 Downlink Initial Ranging Adjustment subheader Syntax

Syntax	No. of Bits	Format
downlink_range_init_subheader() {		
frame_index	10	uimsbf
rand_access_seq_index	12	uimsbf
time_advance	9	uimsbf
power_offset	7	uimsbf
reserved	2	'11'
}		

frame_index – A 10-bit unsigned integer field that shall specify the index of the DRC uplink frame in which the random access sequence has been sent. Values between 0x3E8~0x7FF shall be reserved.

rand_access_seq_index – A 12-bit unsigned integer field that shall specify the index of received random access sequence. Values between 0xDE0~0xFFF shall be reserved.

time_advance – A 9-bit unsigned integer field that shall specify the amount of needed time advancement. It is denoted as N_{TA}. The time advance is equal to $T_A = 16 \times T_S \times N_{TA}$, where T_S is defined in Table 5.7. The positive value indicates that the time at the BAT is later than the time at BTS. The BAT needs to advance the time for compensation of the propagation delay. The maximum time advance is up to 817.6 μsec.

power_offset – A 7-bit unsigned integer field that shall specify the BAT offset power. The step shall be 1 dB. The exact power offset to be applied by the BAT shall be equal to the indicated power offset minus 64. For example, 0x00 and 0x7F mean offsets of -64dB and +63dB, respectively. A positive exact power offset shall indicate that the BAT shall increase its transmission power by the specified amount, and a negative exact power offset shall indicate that the BAT shall decrease its transmission power by the specified amount.

6.7.2 Payload of Bandwidth Allocation Response subheader

When the BTS receives a bandwidth allocation request from a BAT, it shall send the bandwidth allocation response to the BAT. The syntax of the Bandwidth Allocation Response subheader shall conform to Table 6.17. The semantics of the fields in the downlink_bw_allocation_subheader() packet shall be as given immediately below the table.

Table 6.17 Downlink Bandwidth Allocation Response subheader Syntax

Syntax	No. of Bits	Format
downlink_bw_allocation_subheader() {		
tuid	18	uimsbf
cid	5	uimsbf
bw_allocation_result	2	uimsbf
reserved	7	'1111111'
}		

bw_allocation_result – A 2-bit unsigned integer field that shall specify the allocated resource indicated by the next uplink map packet with a value of '1'. A value of '0' shall indicate rejected result. Values '2' and '3' shall be reserved.

6.7.3 Payload of Status Report Request subheader

When the BTS requires the status from a BAT for resource allocation or scheduling, the BTS shall send the status report request subheader to the BAT. The syntax of the Status Report Request subheader shall conform to Table 6.18.

Table 6.18 Downlink Status Report subheader Syntax

Syntax	No. of Bits	Format
downlink_status_report_subheader() {		
tuid	18	uimsbf
reserved	6	'111111'
}		

6.7.4 Payload of Downlink ACK Message subheader

The syntax of the Downlink ACK Message subheader shall conform to Table 6.19. The semantics of the fields in the downlink_ack_message_subheader() packet shall be as given immediately below the table.

Table 6.19 Downlink ACK Message subheader Syntax

Syntax	No. of Bits	Format
downlink_ack_message_subheader() {		
tuid	18	uimsbf
bitmap_length	8	uimsbf
first_sequence_number	8	uimsbf
bitmap	variable	uimsbf
padding	variable	uimsbf
}		

6.7.5 Payload of Online Adjustment subheader

The syntax of the Online Adjustment Message subheader shall conform to Table 6.20. The semantics of the fields in the online_adjust_subheader() packet shall be as given immediately below the table. For an online adjustment subheader received on DRC uplink frame n, the corresponding adjustment of the timing and power shall apply from the beginning of frame $(n+6)$ modulo 1000.

Table 6.20 Downlink Online Adjustment subheader Syntax

Syntax	No. of Bits	Format
online_adjust_subheader() {		
tuid	18	uimsbf
online_time_offset	6	uimsbf
online_power_offset	8	uimsbf
reserved	2	'11'
}		

online_time_offset – A 6-bit unsigned integer field that shall specify the BAT time offset. It is denoted as N_{TA}. The time offset to be applied shall be equal to $T_A = 16 \times T_S \times (N_{TA} - 32)$. A positive T_A shall indicate that the time at the BAT is later than the time at BTS and that the BAT shall advance its time by the magnitude of T_A. A negative T_A shall indicate that the time at the BAT is earlier than the time at BTS and that the BAT shall delay its time by the magnitude of T_A.

online_power_offset – A 6-bit unsigned integer field that shall specify the BAT power offset. The step size is 1dB. The exact power offset to be applied, in units of dB, shall be equal to the indicated power offset minus 32. For example, 0x00 and 0x3F mean offsets of -32dB and +31dB, respectively. When a positive power offset to be applied is calculated, the BAT shall increase its transmission power by the magnitude of the calculated power offset. When a negative power offset to be applied is calculated, the BAT shall decrease its transmission power by the magnitude of the calculated power offset.

6.7.6 Payload of Paging Request subheader

When the BTS is to find a BAT that is powered on and registered but has no active connection currently, the BTS shall send the paging request subheader to the BAT. The syntax of the Paging Request subheader shall conform to Table 6.21. The semantics of the fields in the downlink_paging_request_subheader() packet shall be as given immediately below the table.

Table 6.21 Downlink Paging Request subheader Syntax

Syntax	No. of Bits	Format
downlink_paging_request_subheader() {		
tuid	18	uimsbf
rand_access_seq_paging	12	uimsbf
reserved	2	‘11’
}		

rand_access_seq_paging – A 12-bit unsigned integer field that shall specify the index of received random access sequence to be used by the BAT in response to a Paging Request. Valid values for this index range from the indicated Partition of random access sequences as signaled in the Downlink Broadcast Control Information (see Table 6.12) to 0xDDF, inclusive.

6.7.7 Payload of Registration Confirmation subheader

When the BTS receives a Registration Request from a BAT, the BTS shall send the Registration Confirmation subheader in $5\times N_{RR}$ msec. The syntax of the Registration Confirmation subheader shall conform to Table 6.22.

Table 6.22 Downlink Registration Confirmation subheader Syntax

Syntax	No. of Bits	Format
downlink_reg_confirm_subheader() {		
user_id	48	uimsbf
tuid	18	uimsbf
cid	5	uimsbf
bw_allocation_result	2	uimsbf
reserved	7	‘1111111’
}		

6.7.8 Payload of Connection Release Confirmation subheader

When the BTS receives a Connection Release Request subheader from a BAT, the BTS shall send the Connection Release Confirmation subheader to the BAT. The syntax of the Connection Release Confirmation subheader shall conform to Table 6.23.

Table 6.23 Downlink Connection Release subheader Syntax

Syntax	No. of Bits	Format
downlink_connection_release_subheader() {		
Tuid	18	uimsbf
cid	5	uimsbf
Reserved	1	‘1’
}		

6.7.9 Payload of Power Down Confirmation subheader

When the BTS receives a Power Down Request subheader from a BAT, the BTS shall send the Power Down Confirmation subheader to the BAT. The syntax of the Power Down Confirmation subheader shall conform to Table 6.24.

Table 6.24 Downlink Power Down subheader Syntax

Syntax	No. of Bits	Format
downlink_power_down_subheader() {		
Tuid	18	uimsbf
Reserved	6	'111111'
}		

6.7.10 Payload of Downlink Data subheader

The syntax of the Downlink Data subheader shall conform to Table 6.25. The semantics of the fields in the downlink_data_subheader() packet shall be as given immediately below the table.

Table 6.25 Downlink Data subheader Syntax

Syntax	No. of Bits	Format
downlink_data_subheader() {		
Tuid	18	uimsbf
cid	5	uimsbf
data_length	11	uimsbf
reserved	6	'111111'
}		

data_length – An 11-bit unsigned integer field that shall specify the downlink data in bytes minus one. The minimum actual data length shall be 1 byte, and the maximum actual data length shall be 1500 bytes, which is the same as the Maximum Transmission Unit (MTU) of Ethernet. Values between 0x5DC ~0x7FF shall be reserved.

7. DEFINITION OF DRCT

This specification defines a new Low Level Signaling (LLS) table (see A/331 [5]), the Dedicated Return Channel Table (DRCT). Its presence indicates that the payload carried in the PLP-R is for the Dedicated Return Channel (DRC). The DRCT is allocated the LLS_table_id 0x[TBD] (see Table 6.1 in A/331 [5]).

Each DRCT shall be repeated in the LLS in which it is transported at least once per ATSC 3.0 downlink physical layer frame.

7.1 XML Schema and Namespace

The DRCT shall be represented as an XML document containing a **DRCT** root element that conforms to the definitions in the XML schema that has namespace:

```
tag:atsc.org,2016:XMLSchemas/ATSC3/DRC/DRCT/1.0/
```

The definition of this schema is in an XML schema file, DRCT-1.0-20170809.xsd. The XML schema xmlns short name should be "**drct**".

The sub-string part of namespaces between the right-most two ‘/’ delimiters indicate major and minor version of the schemas. The schemas defined in this present document shall have version ‘1.0’, which indicates major version is 1 and minor version is 0.

The namespace designator, “xs:”, and many terms in the “Data Type” column of tables is a shorthand for datatypes defined in W3C XML Schema [9] and shall be as defined there.

In order to provide flexibility for future changes in the schema, decoders of XML documents with the namespaces defined in the present document should ignore any elements or attributes they do not recognize, instead of treating them as errors.

All element groups and attribute groups are explicitly extensible with elements and attributes respectively. Elements can only be extended from namespaces other than the target namespace. Attributes can be extended from both the target namespace and other namespaces. If the XML schema does not permit this for some element, that is an error in the schema.

XML schemas shall use processContents="strict" in order to reduce inadvertant typos in instance documents. Further, users are encouraged to modify the IETF FDT schema found in RFC 6726 [10] to change processContents to "strict". Similarly for the MBMS [11] 3GPP schemas.

In the event of any discrepancy between the XML schema definitions implied by the tables that appear in this document and those that appear in the XML schema definition files, those in the XML schema definition files are authoritative and take precedence.

The XML schema document for the schemas defined in this document can be found at the ATSC website.

7.2 DRCT Syntax

While the indicated XML schema specifies the normative syntax of the DRCT element, informative Table 7.1 below describes the structure of the DRCT element in a more illustrative way. The specifications following the table give the semantics of the elements and attributes.

Table 7.1 DRCT XML Format

Element or Attribute Name		Use	Data Type	Description
DRCT				Root element of the DRCT.
	`@bsid`	1	unsignedShort	Identifies the one or more Broadcast Streams comprising the Services.
	`@DestinationIpAddress`	1	IPv4address	A string containing the dotted-IPv4 destination address of the packets carrying data for the return channel.
	`@DestinationUdpPort`	1	unsignedShort	Port number of the packets carrying data for the return channel.
	`@SourceIpAddress`	1	IPv4address	A string containing the dotted-IPv4 source address of the packets carrying data for the return channel.

7.3 DRCT Semantics

The following text specifies the semantics of the elements and attributes in the DRCT.

DRCT – Root element of the DRCT.

`@bsid` – This list of one or more 16-bit unsigned integers shall identify the Broadcast Stream ID(s) of the original emission signal(s). The value of each `@bsid` shall be the same as the value signaled in L1D_bsid in L1-Detail Signaling in the physical layer (see A/322 [3]). In the case that the Service is delivered via channel bonding at the physical layer, the list shall include the BSID value of each RF emission involved in the bonding.

`@slsDestinationIpAddress` – A string containing the dotted-IPv4 destination address of the packets carrying data for the dedicated return channel. The syntax shall be as defined in RFC 3986 [8] Section 3.2.2.

`@slsDestinationUdpPort` – Port number of the packets carrying data for the dedicated return channel.

`@slsSourceIpAddress` – A string containing the dotted-IPv4 source address of the packets carrying data for the return channel. The syntax shall be as defined in RFC 3986 [8] Section 3.2.2.

Annex A: PRACH Design (Informative)

A.1 PRACH PREAMBLE STRUCTURE

As illustrated in Figure A.1.1, a PRACH preamble symbol is similar to that of an OFDM symbol. The total duration of a PRACH preamble symbol is 3.6128ms, including the Guard Time (GT) where nothing is transmitted. The Guard Time is reserved to compensate for propagation delay.

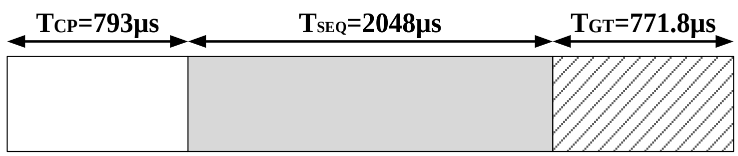

Figure A.1.1 The structure of a PRACH preamble symbol.

The duration of the PRACH preamble sequence was set based on the maximum round trip delay and the coverage performance. The details are given in the following sections.

A.2 MAXIMUM ROUND TRIP DELAY

In order to cover all users in a BTS's coverage range, the length of a random access sequence should be greater than the maximum round trip delay. In a cell with maximum target radius 100km, the length of a random access sequence T_{SEQ} should satisfy

$$T_{SEQ} \geq \frac{100km \times 2}{c} = 666.67\mu s \quad \text{(A.1)}$$

where c represents the speed of light.

In order to minimize the loss of orthogonality between PRACH subcarriers and PUSCH (Physical Uplink Shared Channel) subcarriers, the subcarrier spacing Δf in PUSCH should be an integer multiple of the PRACH subcarrier spacing Δf_{RA} , i.e.,

$$\Delta f_{RA} = \frac{1}{k} \Delta f \quad \text{(A.2)}$$

$$T_{SEQ} = kT_{SYM} \quad \text{(A.3)}$$

where T_{SEQ} stands for the time duration of PRACH, T_{SYM} is the duration of one PUSCH symbol used for data transmission and k is the integer up-sampling rate in PRACH generator. The value of k is set to 10 in this specification.

According to the uplink frame structure, the total duration of the PRACH is 3.6128ms, which is determined by the length of PRACH in the time domain. The duration of the preamble sequence can therefore be obtained as

$$\Delta f_{RA} = \frac{\Delta f}{10} \approx 488.3Hz, T_{SEQ} = \frac{1}{488.3} s = 2048\mu s \quad \text{(A.4)}$$

A.3 CP AND GT DURATIONS

According to the preamble sequence duration, the length of CP (Cyclic Prefix) can be calculated as

$$T_{cp} = \frac{T_{PRA} - T_{SEQ}}{2} + \frac{d}{2} = \left(\frac{3612.8 - 2048}{2} + \frac{21}{2}\right)\mu s = 792.9 \approx 793\mu s \quad \text{(A.5)}$$

where $d = 21\ \mu sec$ represents the maximum CP in an SC-FDMA symbol. The duration of the GT (Guard Time) is

$$T_{GT} = T_{PRA} - T_{SEQ} - T_{cp} = 771.8\mu s \quad \text{(A.6)}$$

A.4 THE CYCLIC SHIFT SIZE

Since ZC sequences with different root values are not ideally orthogonal, a single sequence with cyclic shifts should be used. The cyclic shift method can generate sufficient PRACH preamble sequences.

The upper bound of N_{CS} is $\lfloor 1777/2 \rfloor$=888 due to the cyclic shift operation.

The lower bound of the cyclic-shift size, N_{CS}, should guarantee that the duration of the sample sequence is larger than the time delay spread and time uncertainty between asynchronous users.

Therefore, the lower bound of N_{CS} is

$$N_{cs} \geq \left\lceil \left(\frac{20}{3} r - \tau_{ds}\right) \frac{N_{zc}}{T_{SEQ}} \right\rceil = 563 \quad \text{(A.7)}$$

where $r = 1000km$ represents the radius of the cell, $\tau_{ds} = 16.67\ usec$ represents the maximum delay spread, and $N_{zc} = 1777$ is the ZC sequence length.

In this specification, $N_{CS} = 888$ is chosen as the size of cyclic-shift. When generating random access sequences, for each index of root value, we can have 2 random access sequences by cyclic shift.

Annex B: Downlink Synchronization (Informative)

B.1 SYNCHRONIZATION ERROR ANALYSIS

The downlink broadcast operates frame by frame. BATs can acquire synchronization only at the beginning of a downlink frame. The synchronization error accumulates in the remaining time of the downlink frame. The accumulated error is determined by the length of the downlink frame and the clock precision at BATs.

According to A/322 [3], the maximum length of a broadcast frame is 5 seconds. The typical precision of a crystal clock is 10ppm. Thus the accumulated error can be calculated as

$$5 \times 10 \times 10^{-6} s = 5 \mu s \tag{B.1}$$

Similarly, if the precision of a clock at BATs is 50ppm in worst case, the accumulated error will be 25 μsec. This is tolerable by the DRC system.

Annex C: Signaling Overhead Analysis (Informative)

C.1 SIGNALING OVERHEAD ANALYSIS

The numbers of signaling bits of all MAC PDU subheaders are shown in Table C.1.1. Assuming that there are N_users users in the cell, without considering the overhead of BCI and UL-MAP, the overall signaling overhead in the downlink system can be calculated as $C\_ov \cdot N\_users$, where C_ov is the average cost in bits.

According to the resource allocation scheme given above, the maximum number of users that can be served within a DRC uplink frame is

$$N_{\max} = \frac{1320tiles - 72tiles}{3times} = 416 \qquad \text{(C.1)}$$

In extreme conditions, the average cost is $C_{ov} = 88$ and the number of users is N_max=416. Thus, the theoretical maximum signaling overhead could reach 3.66 Mbps.

Assuming the ten subheaders listed in Table C.1.1 appear with equal probability, the average subheader length is 46.4 bits. Thus, the average signaling overhead is 1.93 Mbps, when the maximum number of users (416) appear in each uplink frame. The average signaling overhead is 193Kbps when 41.6 users in average appear in each uplink frame.

Table C.1.1 Types and Lengths of subheaders in DRC Downlink

Subheader Index	Subheader	Total bits
0	Initial Ranging Adjustment	40
1	Bandwidth Allocation Response	40
2	Status Report Request	24
3	ACK Message	34 + variable
4	Online Adjustment Message	32
5	Paging Request	32
6	Registration Confirmation	88
7	Connection Release Confirmation	24
8	Power Down Confirmation	24
9	Downlink Data	40

End of Document