INTERNATIONAL STANDARD

ISO/IEC 4005-4

First edition
2023-03

Telecommunications and information exchange between systems — Unmanned aircraft area network (UAAN) —

Part 4: Physical and data link protocols for video communication

Télécommunications et échange d'information entre systèmes — Réseau de zone de drones (Unmanned aircraft area network - UAAN) —

Partie 4: Protocoles de liaison de données et physiques pour la communication vidéo

Reference number
ISO/IEC 4005-4:2023(E)

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see https://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*.

A list of all parts in the ISO/IEC 4005 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

Unmanned aircrafts (UAs) operating at low altitudes will provide a variety of commercial services in the near future. UAs that provide these services are distributed in the airspace. In level II, many people operate their own UAs without the assignment of communication channels from a central control centre.

This document describes video communication, which is a wireless distributed communication. Video communication allows UAs distributed over the airspace to transmit video without serious interference to the relevant video receiver which is usually a controller. The channels used for video communication have a multi-channel structure, which enables UA and video receiver pairs to independently use the occupied communication link. A wireless distributed communication described by this document is intended to be used in licensed frequency bands.

The ISO/IEC 4005 series consists of the following four parts:

ISO/IEC 4005-1: To support various services for UAs, it describes a wireless distributed communication model and the requirements that this model shall satisfy.

ISO/IEC 4005-2: It describes communication in which all units involved in UA operation can broadcast or exchange information by sharing communication resources with each other.

ISO/IEC 4005-3: It describes the control communication for the controller to control the UA.

ISO/IEC 4005-4 (this document): It describes video communication for UAs to send video to a controller.

The International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) draw attention to the fact that it is claimed that compliance with this document may involve the use of patents.

ISO and IEC take no position concerning the evidence, validity and scope of these patent rights.

The holders of these patent rights have assured ISO and IEC that they are willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statements of the holders of these patent rights are registered with ISO and IEC. Information may be obtained from the patent database available at www.iso.org/patents.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those in the patent database. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Telecommunications and information exchange between systems — Unmanned aircraft area network (UAAN) —

Part 4: Physical and data link protocols for video communication

1 Scope

This document specifies communication protocols for the physical and data link layer of video communication, which is a wireless distributed communication network for units related with unmanned aircrafts (UAs) in level II.

This document describes video communication, which is one-to-one communication that transmits video from a UA to a video receiver. For the specific use of video communication, video can be transmitted from a UA to multiple receivers.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 4005-1, *Telecommunications and information exchange between systems — Unmanned aircraft area network (UAAN) — Part 1: Communication model and requirements*

ISO/IEC 4005-2, *Telecommunications and information exchange between systems — Unmanned aircraft area network (UAAN) — Part 2: Physical and data link protocols for shared communication*

ISO/IEC 4005-3, *Telecommunications and information exchange between systems — Unmanned aircraft area network (UAAN) — Part 3: Physical and data link protocols for control communication*

ISO 21384-4, *Unmanned aircraft systems — Part 4: Vocabulary*

3 Terms and definitions

For the purposes of this document, the terms and definitions defined in ISO/IEC 4005-1, ISO/IEC 4005-2, ISO/IEC 4005-3, ISO 21384-4 and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

4 Abbreviated terms

CC	Control Communication
CB	Coding Block
CRC	Cyclic Redundancy Check

CSCH	Control Subchannel
DL	Data Link
DLL	Data Link Layer
DQPSK	Differential Quadrature Phase Shift Keying
DS	Dedicated Slot
FN	Frame Number
GF	Galois Field
PCCC	Parallel Concatenated Convolutional Code
PB	Parsing Block
PH	Parsing Header
PKH	Packet Header
PN	Pseudo Noise
SA	Source Address
SC	Shared Communication
SRRC	Square Root Raised Cosine
TSB	Tone Slot Block
UTC	Coordinated Universal Time
VC	Video Communication
VSCH	Video Subchannel

5 Physical layer

5.1 Channel and frame structure for data channel

5.1.1 The number of data channels and bandwidth

The number of data channels is L. L is greater than or equal to one. The bandwidth of one data channel is 5 MHz as shown in Figure 1. The L is determined in the upper layer.

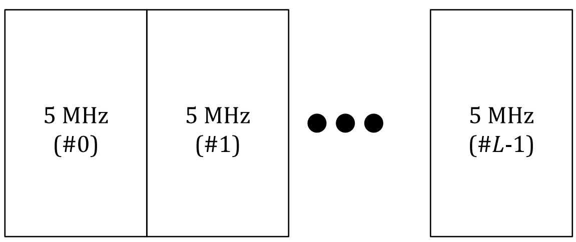
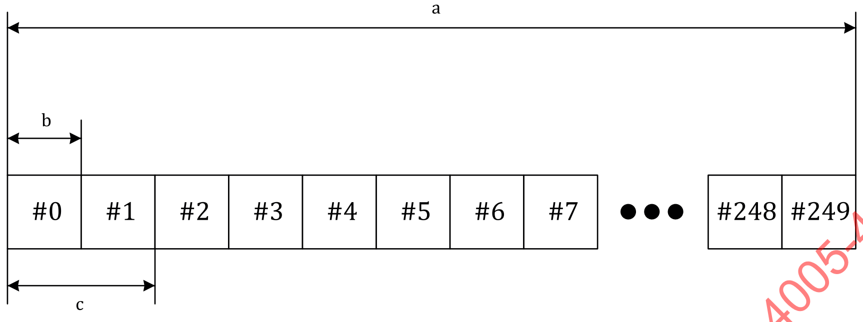


Figure 1 — Data channels in frequency region

5.1.2 Frame structure

The frame length of the data channel is 1 sec and consists of 250 slots. The one slot time T_s is 4 ms. A data slot block has 2 slots. Therefore, there are 125 data slot blocks in one frame, and the data slot block is 8 ms in length as shown in Figure 2. The frame number, *FN* changes from 0 to 59 in a 1 min interval, and has the same value as the second of the current time.



a 1 frame, T_f = 1 second = 250 T_s.

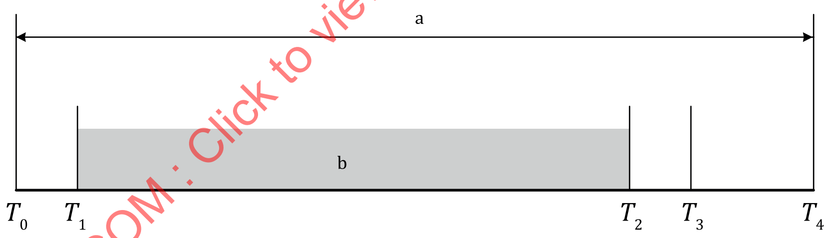
b 1 slot, T_s = 4 ms.

c 1 slot block, T_{sb} = 8 ms = 2 T_s.

Figure 2 — Data channel frame structure

5.1.3 Slot transmit time mask

The transmission time mask of a slot is as shown in Figure 3.



Key

T_0 0 µs

T_1, T_2, T_3, T_4 symbol offset from T_0

a 4 ms.

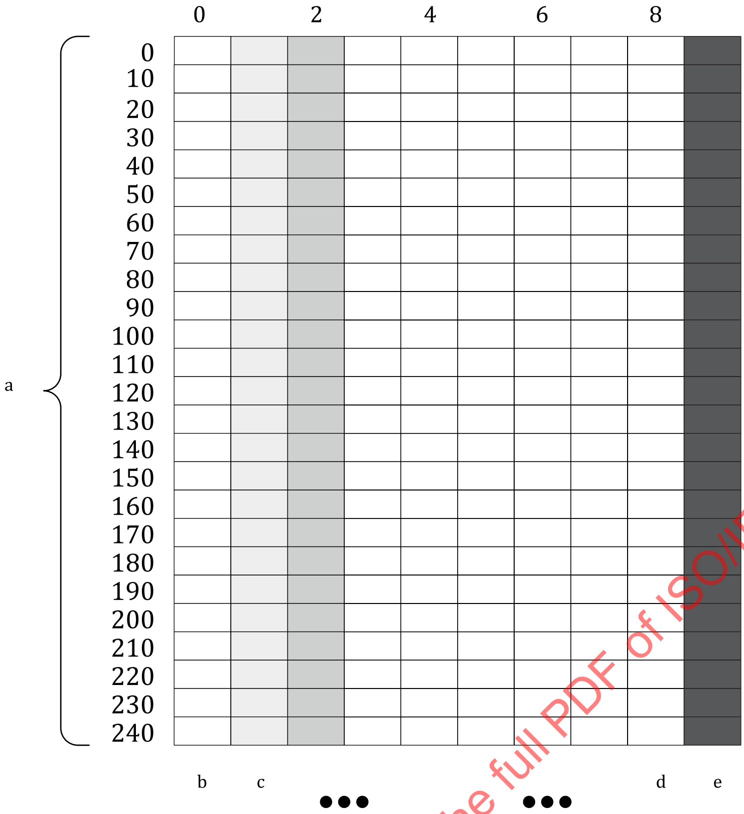
b Modulated signal.

Figure 3 — The transmission time mask of a slot

T_1, T_2, T_3, T_4 are symbol offsets from T_0 and symbol time is 1/2688000 sec. Each value is as follows: T_1 is 8, T_2 is 10380, T_3 is 10388, T_4 is 10752.

T_0 is 0 µs as the start time of the slot and the power amplifier is gated on and unmodulated fine signals begin to be transmitted. T_1 is an offset at which modulation signal transmission starts. T_2 is an offset at which the transmission of the modulated signal ends. T_3 is an offset at which the power amplifier is gated off, and transmission of unmodulated fine signals is stopped. The transmit power of T_0 to T_1, T_2 to T_3 shall be at least 50 dB less than the modulation signal transmit power.

5.1.4 Sub channels



a V_x
b $V_{x,0}$
c $V_{x,1}$
d $V_{x,8}$
e $V_{x,9}$

Figure 4 — Sub channel structure of video communication in even frame

One data channel consists of 10 subchannels as shown in Figure 4. Subchannel y of video channel x is composed of the following slot set.

$$V_{x,y} = S_{x,z}, S_{x,z+10}, S_{x,z+20}, \ldots, S_{x,z+240}$$

$$z = \begin{cases} y, & even\ frame \\ y+1-\lfloor (y \bmod 2)/2 \rfloor \times 2, & odd\ frame \end{cases} \quad (1)$$

where

y is subchannel number, y=0, 1, ..., 9;

$S_{x,z}$ is slot z of video channel x.

The subchannel consists of 25 slots, the *i*-th slot resource of the subchannel *y* of the channel *x* is indicated by $SR_{x,y,i}$, and the subchannel *y* of frequency channel *x* is indicated by $V_{x,y}$. Therefore, $V_{x,y}$ is as follows:

$$V_{x,y} = SR_{x,y,0}, SR_{x,y,1}, \ldots, SR_{x,y,24} \quad (2)$$

where $SR_{x,y,i}$ is *i*-th slot resource of subchannel *y* of channel *x*, *i*=0, ..., 24.

All slots of video channel are downlink.

5.1.5 Dedicated subchannels

The upper layer can predetermine one or several subchannels as dedicated subchannels. In this case, the tone subslot set mapped with the dedicated subchannel is not used as a competition tone and can be used for other purposes.

Dedicated subchannel information is received from an upper layer through UPtoDL. InfoDedicatedChannel.

5.2 Channel and frame structure for tone channel

5.2.1 General

The tone channel of video communication means a competitive tone channel. The tone channel used for video communication resource allocation and the tone channel used for control communication resource allocation are the same channel (see ISO/IEC 4005-3).

5.2.2 Slot transmit power

The maximum transmission power PmaxTCH of the tone slot mapped to the video subchannel (VSCH) is received as UPtoDL.InfoPowerParamVCH from the upper layer. The power of the tone subslot signal is determined by adding the PTX_VCHTCH_differ value to the transmission power of the mapped VSCH.

5.3 Encoding procedure

The encoding follows the following procedure. CRC encoding, turbo coding, rate matching, interleaving, modulation mapping, burst mapping, and pulse mapping are performed in this order as shown in Figure 5.

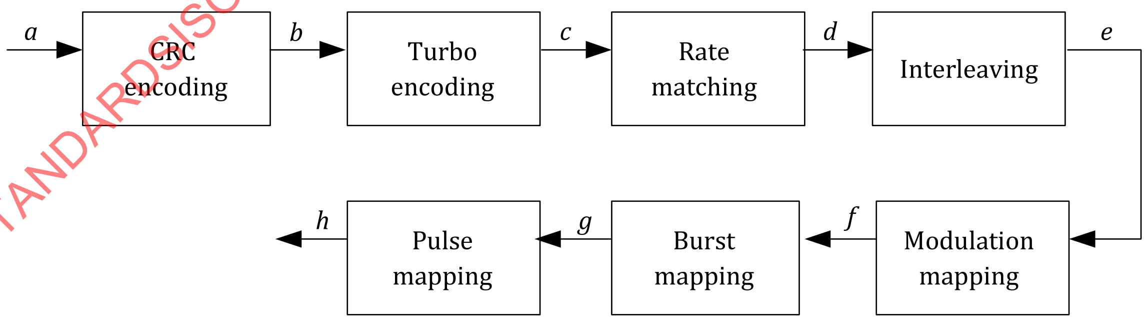


Figure 5 — Encoding procedure

The number of symbols according to each encoding stage is shown in Table 1, where the encoding input consists of two code blocks, CB0 and CB1 as shown in Figure 14. Each code block undergoes CRC encoding, turbo coding, rate matching, interleaving, and modulation mapping processes, respectively. The length of each code block in these processes is 4094, 4928, 9868, 9856, 9856, and 4928. The two code blocks are merged into one burst during burst mapping.

Table 1 — Number of symbols at each encoding stage

Stage	Number of symbols
a	4904 × 2 (binary)
b	4928 × 2 (binary)
c	9868 × 2 (binary)
d	9856 × 2 (binary)
e	9856 × 2 (binary)
f	4928 × 2 (complex)
g	10364 (complex)
h	10372 × *OS* (complex)

5.3.1 CRC encoding

The input bits are defined as $a_0, a_1, a_2, a_3, ..., a_{A\text{-}1}$ and parity bits as $p_0, p_1, p_2, p_3, ..., p_{23}$ where A represents the number of input sequences. Parity bits are generated through CRC generation polynomial as follows.

$$g_{\text{CRC}}(D) = D^{24} + D^{22} + D^6 + D^5 + D + 1 \tag{3}$$

The encoding performed through the cyclic generator polynomials has a systematic form as follows. The resulting polynomial has zero remainder when it is divided by $g_{\text{CRC}}(D)$ on GF(2).

$$a_0 D^{A+23} + a_1 D^{A+22} + ... + a_{A\text{-}1} D^{24} + p_0 D^{23} + p_1 D^{22} + ... + p_{22} D^1 + p_{23} \tag{4}$$

After CRC insertion, bits are represented by $b_0, b_1, b_2, b_3, ..., b_{B\text{-}1}$ (where $B = A + 24$), and the relationship between a_k and b_k is as follows.

$$b_k = \begin{cases} a_k, & for\, k = 0,1,2,\cdots,A-1 \\ P_{k-A}, & for\, k = A, A+1, A+2,\cdots,A+23 \end{cases} \tag{5}$$

5.3.2 Turbo encoding

The turbo encoder consists of Parallel Concatenated Convolutional Code (PCCC) with two 8-state constituent encoders and one turbo coded internal interleaver. The coding rate of the turbo encoder is 1/2. The structure of the turbo encoder is shown in Figure 6. The PCCC transfer function is as follows.

$$G(D) = [1, g_1(D)/g_0(D)] \tag{6}$$

where $g_0(D) = 1{+}D^2{+}D^3$, $g_1(D) = 1{+}D{+}D^3$.

When the input bits of the turbo encoder are encoded, the initial values of the shift registers of the 8-state constituent encoder shall all be zero.

For $k = 0, 1, 2, ..., B/2\text{-}1$, the output value of the turbo encoder is expressed as follows.

$$c_{4k} = x_{2k}$$

$$c_{4k+1} = z_{2k}$$

$$c_{4k+2} = x_{2k+1}$$

$$c_{4k+3} = z'_{2k+1} \tag{7}$$

Output bits of the first and second 8-state constituent encoders for turbo encoder input bits b_0, b_1, b_2, b_3, ..., b_{B-1} are z_0, z_1, z_2, z_3, ..., z_{B-1} and z'_0, z'_1, z'_2, z'_3, ..., z'_{B-1}, and the output bits through the turbo code internal interleaver that is described in Annex A are represented by b'_0, b'_1, b'_2, b'_3, ..., b'_{B-1}. These output bits are used as inputs for the second 8-state constituent encoder.

Trellis termination is performed by taking tail bits from shift register feedback after all information bits have been encoded. The generated tail bits are added after encoding of the information bits.

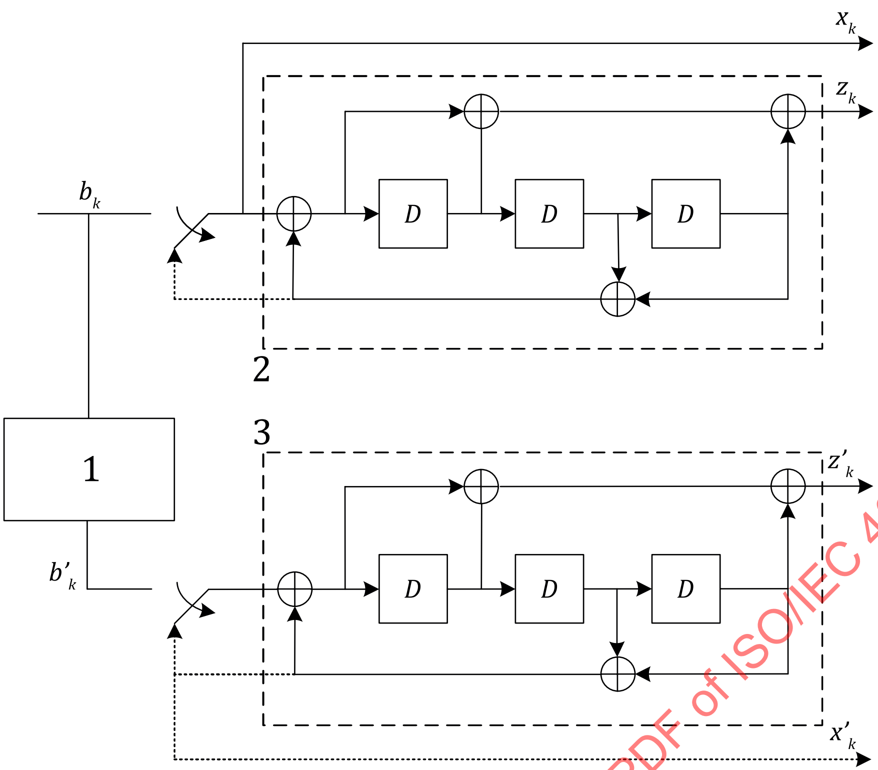
The first three tail bits are used for the first constituent encoder termination and not the second constituent encoder. The remaining three tail bits are used for the termination of the second constituent encoder and not the first constituent encoder.

The bits transmitted for trellis termination are determined as follows.

$$c_{2B} = x_B,\ c_{2B+3} = z_{B+1},\ c_{2B+6} = x'_B,\ c_{2B+9} = z'_{B+1}$$

$$c_{2B+1} = z_B,\ c_{2B+4} = x_{B+2},\ c_{2B+7} = z'_B,\ c_{2B+10} = x'_{B+2}$$

$$c_{2B+2} = x_{B+1},\ c_{2B+5} = z_{B+2},\ c_{2B+8} = x'_{B+1},\ c_{2B+11} = z'_{B+2} \tag{8}$$



Key

1 turbo code internal interleaver
2 first constituent encoder
3 second constituent encoder
D register
b_k a *k*-th bit of turbo encoder input
b'_k a *k*-th bit of turbo code internal interleaver output
x_k a *k*-th systematic bit of turbo encoder output
z_k a *k*-th bit of first constituent encoder output
x'_k a *k*-th bit of second constituent encoder output for trellis termination
z'_k a *k*-th bit of second constituent encoder output

Figure 6 — Turbo encoder structure

Input bit sequence of turbo code internal interleave, b_0, b_1, b_2, b_3, ..., $b_{B\text{-}1}$ and output bit sequence generated from turbo code internal interleaver, b'_0, b'_1, b'_2, b'_3, ..., $b'_{B\text{-}1}$ have the following relationship.

$$b'_i = b_j \quad (9)$$

where the mapping between the output bit index *i* and the input bit index *j* shall follow Table A.1 of Annex A where *j* and *i* are as follows, and row and column numbers start at zero.

$$j = (\text{number shown in table}) - 1$$

$$i = (\text{row number in table}) \times 16 + (\text{column number in table}) \quad (10)$$

5.3.3 Rate matching

Rate matching outputs d_0, d_1, d_2, d_3, ..., $d_{D\text{-}1}$ by puncturing the input bits c_0, c_1, c_2, c_3, ..., $c_{C\text{-}1}$. The puncturing bit numbers are as follows.

— 821, 1643, 2461, 3283, 4101, 4923, 5741, 6563, 7381, 8203, 9021, 9843

5.3.4 Interleaving

The interleaver uses block interleaving with 77 rows and 128 columns.

$$e_m = d_n$$

$$m = (n \text{ x } 77)\ \%9856 + \lfloor n/128 \rfloor \quad (11)$$

where $\lfloor x \rfloor$ means the largest integer among integers less than or equal to x and $0 \le n \le 9855$.

5.3.5 Modulation mapping

Modulation mapping generates a complex symbol f_n from the input bit e_m, $0 \le n \le 9855$, $0 \le m \le 4927$. Two input bits are mapped to one complex number as shown in Table 2.

Table 2 — Modulation mapping

$e_{2n}e_{2n+1}$	00	01	10	11
f_n	$exp(j/4\pi)$	$exp(j{\cdot}7/4\pi)$	$exp(j{\cdot}3/4\pi)$	$exp(j{\cdot}5/4\pi)$

5.3.6 Burst mapping

Output complex symbols $g_0, g_1, ..., g_{4927}$ are generated from $f_0, f_1, ..., f_{4927}$ of CB0 and $f_0, f_1, ..., f_{4927}$ of CB1.

$$g_n = \prod_{k=0}^{n} c(k) \quad (12)$$

where $c(n)$ is shown in Table 3.

Table 3 — $c(n)$

n	$c(n)$	Number of symbols
0, 1	$TSS(n)$	2
2, ..., 37	$PTS1(n\text{-}2)$	36
38, ..., 767	$f_{n\text{-}38}$ of CB0	730
768, ..., 803	$PTS1(n\text{-}768)$	36
804, ..., 1533	$f_{n\text{-}74}$ of CB0	730
1534, ..., 1569	$PTS1(n\text{-}1534)$	36
1570, ..., 2299	$f_{n\text{-}110}$ of CB0	730
2300, ..., 2335	$PTS1(n\text{-}2300)$	36
2336, ..., 3065	$f_{n\text{-}146}$ of CB0	730
3066, ..., 3101	$PTS1(n\text{-}3066)$	36
3102, ..., 3831	$f_{n\text{-}182}$ of CB0	730
3832, ..., 3867	$PTS1(n\text{-}3832)$	36
3868, ..., 4597	$f_{n\text{-}218}$ of CB0	730
4598, ..., 4633	$PTS1(n\text{-}4598)$	36

Table 3 *(continued)*

n	*c*(*n*)	Number of symbols
4634, ..., 5181	$f_{n\text{-}254}$ of CB0	548
5182, ... 5363	$f_{n\text{-}5182}$ of CB1	182
5364, ...,5399	*PTS*1(*n*-5364)	36
5400, ..., 6129	$f_{n\text{-}5218}$ of CB1	730
6130, ..., 6165	*PTS*1(*n*-6130)	36
6166, ..., 6895	$f_{n\text{-}5254}$ of CB1	730
6896, ..., 6931	*PTS*1(*n*-6896)	36
6932, ..., 7661	$f_{n\text{-}5290}$ of CB1	730
7662, ..., 7697	*PTS*1(*n*-7662)	36
7698, ..., 8427	$f_{n\text{-}5326}$ of CB1	730
8428, ..., 8463	*PTS*1(*n*-8428)	36
8464, ..., 9193	$f_{n\text{-}5362}$ of CB1	730
9194, ..., 9229	*PTS*1(*n*-9194)	36
9230, ..., 9959	$f_{n\text{-}5398}$ of CB1	730
9960, ..., 9995	*PTS*1(*n*-9960)	36
9996, ..., 10361	$f_{n\text{-}5434}$ of CB1	366
10362, 10363	*TSS*(*n*-10362)	2

where *TSS*(*n*) and *PTS*1(*n*) are shown in Table 4 and Table 5 respectively.

Table 4 — *TSS*(*n*)

***TSS*(0)**	***TSS*(1)**
exp(*j*·3/4π)	*exp*(*j*·7/4π)

Table 5 — *PTS*1(*n*)

n	*PTS*1(*n*)	*n*	*PTS*1(*n*)	*n*	*PTS*1(*n*)
0	*exp*(*j*·5/4π)	12	*exp*(*j*·5/4π)	24	*exp*(*j*·7/4π)
1	*exp*(*j*·7/4π)	13	*exp*(*j*/4π)	25	*exp*(*j*·5/4π)
2	*exp*(*j*·7/4π)	14	*exp*(*j*/4π)	26	*exp*(*j*·7/4π)
3	*exp*(*j*·5/4π)	15	*exp*(*j*·5/4π)	27	*exp*(*j*/4π)
4	*exp*(*j*/4π)	16	*exp*(*j*·7/4π)	28	*exp*(*j*·5/4π)
5	*exp*(*j*/4π)	17	*exp*(*j*/4π)	29	*exp*(*j*·3/4π)
6	*exp*(*j*·3/4π)	18	*exp*(*j*·5/4π)	30	*exp*(*j*·3/4π)
7	*exp*(*j*·5/4π)	19	*exp*(*j*·3/4π)	31	*exp*(*j*/4π)
8	*exp*(*j*·3/4π)	20	*exp*(*j*·7/4π)	32	*exp*(*j*/4π)
9	*exp*(*j*/4π)	21	*exp*(*j*/4π)	33	*exp*(*j*·5/4π)
10	*exp*(*j*·5/4π)	22	*exp*(*j*/4π)	34	*exp*(*j*·3/4π)
11	*exp*(*j*·5/4π)	23	*exp*(*j*·3/4π)	35	*exp*(*j*·7/4π)

5.3.7 Pulse mapping

The complex symbol g_m is converted into a complex signal h_n, where the oversampling ratio of the filter is OS times and depends on implementation. For $0 \le n < 10372 \times OS$, the complex signal is defined as follows.

$$h_n = w\left(\frac{nT_s}{OS}\right)\sum_{m=0}^{10363} p\left(\left(\frac{n}{OS} - m - 4\right)T_s\right)g_m \tag{13}$$

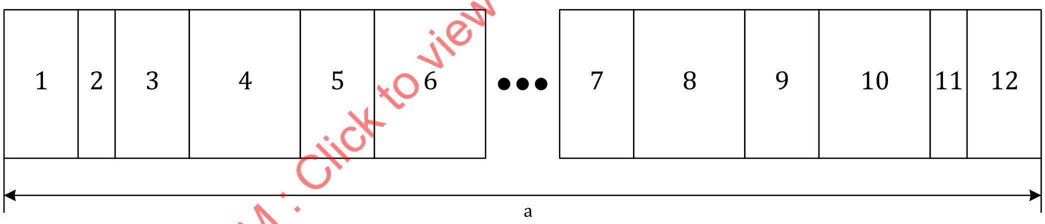
where symbol duration T_s is the 1/2688000 second and pulse shape $p(t)$ is defined as SRRC function of roll-off factor 0,35 as follows.

$$p(t) = \frac{1}{1+\frac{(1-\alpha)\pi}{4\alpha}} \cdot \frac{\cos\left(\frac{(1+\alpha)\pi t}{T_s}\right) + \frac{\sin((1-\alpha)\pi t/T_s)}{4\alpha t/T_s}}{1-(4\alpha t/T_s)^2} \tag{14}$$

The window function $w(t)$ is defined as follows.

$$w(t) = \begin{cases} (1/2)(1-\cos(\pi t/2T_s)), & 0 \le t < 2T_s \\ 1, & 2T_s \le t < 10370T_s \\ (1/2)\left(1-\cos\left(\frac{\pi}{2T_s}(t-10372T_s)\right)\right), & 10370T_s \le t < 10372T_s \\ 0, & otherwise \end{cases} \tag{15}$$

The modulated signal is shown in Figure 7. Timing of modulated signal transmission is as described in 5.1.3, i.e. the modulated signals are transmitted in the time intervals of T_1 to T_2 as shown in Figure 3.



Key

1 filter ripple (4 symbols)
2 TSS (2 symbols)
3 PTS1 (36 symbols)
4 data (730 symbols)
5 PTS1 (36 symbols)
6 data (730 symbols)
7 PTS1 (36 symbols)
8 data (730 symbols)
9 PTS1 (36 symbols)
10 data (336 symbols)
11 TSS (2 symbols)
12 filter ripple (4 symbols)
a Modulated signal (10372 DQPSK symbol).

Figure 7 — Modulated signal structure

5.4 Physical layer procedure

5.4.1 Synchronization

All messages shall be transmitted based on UTC time. All times are measured based on UTC.

The synchronization mode of the unit includes 'A sync', 'B sync' and 'C sync'.

— A sync is synchronization obtained from UTC.

— B sync is secondary synchronization acquired from the synchronization signal of the A sync unit.

— C sync is sync status within 20 seconds after sudden loss of sync in A or B sync mode.

A sync unit shall know the date, hour, minute, second, slot number.

The time error of A sync shall be within ±0,4 µs. The time error of B sync shall be within ±4 µs. The time error of C sync shall be within ±5 µs.

The frequency error of A sync shall be within ±0,1 ppm. The frequency error of the B sync shall be within ±0,2 ppm. The frequency error of the C sync shall be within ±0,3 ppm.

5.4.2 Subchannel power

The maximum power PmaxVCH of the VSCH is received as UPtoDL.InfoPowerParamVCH from an upper layer. The maximum transmission power and minimum transmission power of each VSCH are received as UPtoDL.InfoPowerParamVCHsub from the upper layer. The power control of each VSCH is described in the resource allocation procedure.

5.4.3 Measurements

The physical layer shall have the ability to measure the following parameters. The received signal power of a tone subslot, the received signal power of a data slot, and propagation delay time of the received data signal shall be measured. The receiving power determination point shall be the receiving antenna connector.

5.4.4 Coexistence operation

If the hardware of shared communication described in ISO/IEC 4005-2 and the hardware of control communication described in ISO/IEC 4005-3 and the hardware of video communication described in this document are completely physically isolated and do not affect each other at all, it is possible that coexistence operation is not performed, which is implementation dependent. In general, the three communications affect each other, and in this case, the following coexistence operation shall be performed.

The TX operation of a shared slot includes the TX of the corresponding shared slot and the TX operation in the mapped tone subslot set. The TX operation of a control communication includes TX of the mapped tone subslot set and TX in the subchannel for transmitting control data. The TX operation of video communication includes TX of a mapped tone subslot set and TX in a subchannel for transmitting video data.

When a UA periodically broadcasts its information to a shared slot of a shared channel, a shared slot and a set of tone subslots mapped to the shared slot generally require 1 slot and 4 slots, respectively, for TX operation. If the TX operation of the shared slot used for mandatory periodic broadcasting and the TX operation of the video channel overlap, the TX operation of the shared slot shall be performed.

A control subchannel (CSCH) and a VSCH shall be allocated so that they do not overlap in time.

The TX time of the tone subslot set mapped with mandatory periodically broadcasted shared slot, the TX time of the tone subslot set mapped with the CSCH, and the TX time of the tone subslot set mapped with the VSCH shall not overlap each other. If the video tone slot block type is TSBtype0, the video tone subslot set and the shared tone subslot set can be located in the same TSB. In this case, the two tone subslot set numbers shall be different. If the video TSB type is not TSBtype0, the video tone subslot set and the shared tone subslot set cannot be located in the same TSB.

The TX operation time of the tone subslot set mapped with a CSCH may overlap the TX time of a VSCH, and in this case, the corresponding video slot cannot be transmitted. The TX operation time of the tone subslot set mapped with a VSCH may overlap with the slot TX time of a CSCH, and in this case, the corresponding control slot cannot be transmitted.

6 Data link layer

6.1 General

The data link layer allocates subchannels consisting of 25 slots to controllers and UAs. The controller can use this subchannel to receive video from the UA. In this case, the controller shall include a video channel receiver. The representative application service of video communication is that the controller receives video from the UA, but it is possible to provide other services through the formed video link. This is determined at the upper layer.

The process of using the VSCH is almost the same as the process of using the CSCH and shown in Figure 8.

— negotiation of subchannel number to be allocated;

— competition for allocation and generated link confirmation;

— occupation and management of subchannels;

— subchannel return or reallocation.

Firstly, the UA and the controller each generate a map of the available subchannels. The controller selects one of the subchannels available together and the controller transfers the selected subchannel number to the UA where the subchannel negotiation is performed by the SC DLL or the CC DLL.

After that, the UA and the controller attempt to allocate subchannel at the same time. Subchannel can be allocated only when the UA and the controller succeed in allocation at the same time. The controller shall confirm whether a link is generated.

If the subchannel allocation is successful, the UA and the controller simultaneously perform slot clearing to occupy the subchannel. While occupying a subchannel, the UA and the controller constantly check for collisions of subchannel resources. They also calculate the amount of interference from neighboring channels.

If collision of subchannel resources or interference with neighboring channels exceeding the threshold is detected, the UA and the controller reallocate the subchannel. To do this, the UA and the controller decide which subchannels to reallocate and perform allocation competition on that subchannel.

The UA and the controller return the subchannel when they can no longer maintain or need to maintain them.

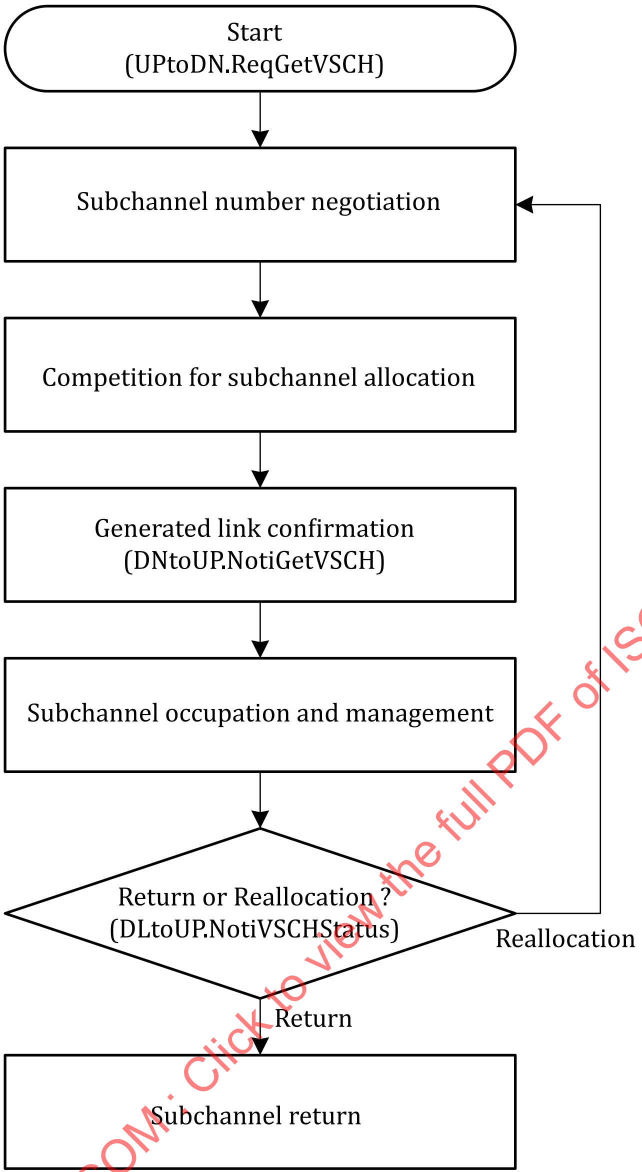


Figure 8 — Subchannel use procedure

6.2 Channel mapping and measurements

6.2.1 General

Allocating subchannel resources is performed by tone channel. One subslot set in a tone channel and one subchannel have a mapping relationship. When one tone subslot set is allocated, a subchannel mapped thereto is allocated.

In order for the UA and the controller to allocate a subchannel, the UA and the controller shall find subchannels that can be allocated at the same time. To this end, the UA and the controller determine the allocable subchannels by calculating the interference power for each subchannel.

6.2.2 Mapping of communication resources and subslot sets

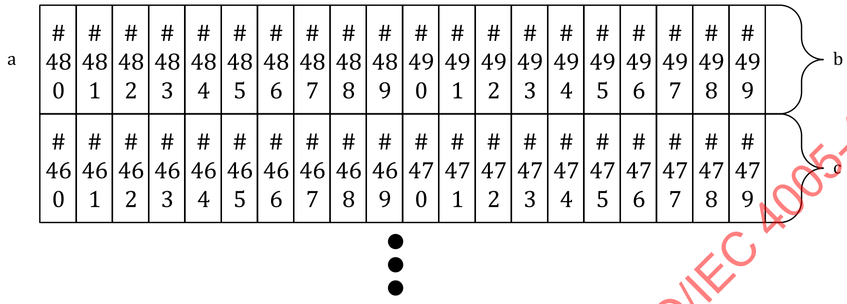
The competition for allocating a subchannel is performed in the subslot set mapped thereto. Subslot sets are mapped to subchannels.

The tone subslot set $\{S_{480+m-n}\}$ is mapped to the subchannel $V_{x,y}$, where m is $(((x \; mod \; 2)+ 2y+8) \; mod \; 20) - 20 \times \lfloor x/2 \rfloor$ and n is as follows.

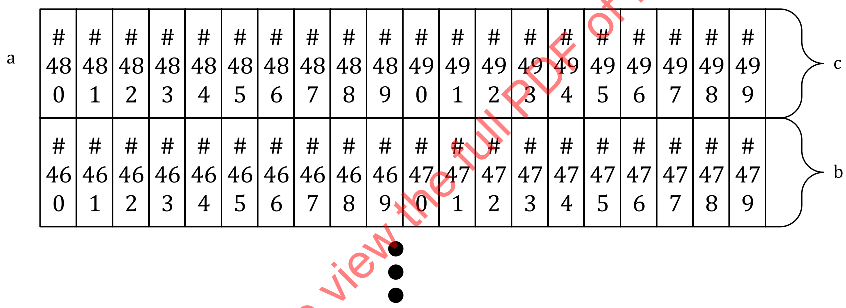
$$n = \begin{cases} 0, & even\,frame \\ 20-(\lfloor x/2 \rfloor \mathrm{mod}\, 2)\times 40, & odd\,frame \end{cases} \qquad (16)$$

Thereafter, $\{S_{480+m-n}\}$ mapped to $V_{x,y}$ is expressed as $\{S_{x,y}\}$.

This mapping shape is shown in Figure 9 key.



a) even frame



b) odd frame

a Tone subslot set.

b $V_{0,6}$, $V_{1,6}$, ..., $V_{1,4}$, $V_{0,5}$, $V_{1,5}$.

c $V_{2,6}$, $V_{3,6}$, ..., $V_{3,4}$, $V_{2,5}$, $V_{3,5}$.

Figure 9 — Mapping of video subchannel (VSCH) and tone subslot sets

6.2.3 Interference power calculation

In order to allocate the VSCH, the controller shall calculate the interference power in the allocable subchannel. Interference constants for calculating the interference power are received as UPtoDL. InfoICConstant in the upper layer. The unit is dB. The estimated interference power of the subchannel is expressed as $PImVCH_{x,y}$.

The interference power of the subchannel $V_{x,y}$ experienced by the controller is calculated as follows.

$$P\mathrm{Im}cVCH_{x,y} = \sum_{i=0,i\neq x}^{N-1}(PmdVCH_{i,y} - IC_{|x-i|}) \qquad (17)$$

where $PmdVCH_{i,y}$ is the reception power of the tone transmitted by the UA in the tone subslot set mapped with $V_{i,y}$ and the unit of this is dBm. The unit of $(PmdVCH_{i,y} - IC_{|x-i|})$ is also dBm.

6.2.4 Subchannel map

Each unit shall make a subchannel map indicating the availability of subchannels. The subchannel map is expressed by 2 bits per subchannel.

In the case of a controller, a subchannel map shall be made as follows in consideration of the subchannel interference.

If the subchannel interference $PImVCH_{x,y}$ is PTH_SMI0 or less, it is written as '11'. If it is greater than PTH_SMI0 and less than PTH_SMI1, it is written as '10', if it is greater than PTH_SMI1 and less than PTH_SMI2, it is written as '01', and if it is greater than PTH_SMI2, it is written as '00', where, PTH_SMI0, PTH_SMI1, and PTH_SMI2 are threshold values for subchannel map creation, and are received as UPtoDL.InfoPowerParamVCH from an upper layer. In addition, the upper layer can designate available subchannels as UPtoDL.InfoApprovedSubchMap, and subchannels that cannot be used in this interface are indicated as '00'.

A UA that has allocated a VSCH shall broadcast its location and transmission power through the SC using a PB 0x88 parsing header. If the UA has allocated both VSCH and CSCH, it uses PB 0x8A where ‘0x’ means hex notation

Each unit shall receive SC signals transmitted by other UAs. Each unit can know the currently occupied subchannels from the SC signal, and if the distance to the UA occupying the subchannels is greater than d_map1, subchannels whose transmission power is less than PTXmap0 are expressed as '10', and PTXmap0 subchannels above and below PTXmap1 are expressed as '01', and subchannels above PTXmap1 are expressed as '00'. If the distance to the UA occupying the subchannels is greater than d_map0 and less than or equal to d_map1, subchannels with transmission power less than PTXmap2 are expressed as '10', and subchannels greater than PTXmap2 and smaller than PTXmap3 are expressed as '01' Is expressed as, and subchannels greater than or equal to PTXmap3 are expressed as '00', where PTXmap0, PTXmap1, PTXmap2, PTXmap3, d_map0, and d_map1 are threshold values for subchannel map creation, and are received as UPtoDL.InfoPowerParamVCH from an upper layer.

As a result of receiving the tone signal of the tone subslot set, if the received power is greater than PTH_TONE, i.e. if the subchannel is occupied, the subchannel is denoted as '00'. In addition, if there is no information on the subchannel, it is expressed as '00'. The number of bits of the subchannel map is 20*L*. *L* is the number of VCH channels. The data link layer can transmit the subchannel map by dividing it into one packet or several packets.

When creating a subchannel map, a UA can make a subchannel map with '00' for subchannels that are overlapped with an own occupied shared slot and a tone subslot set mapped to this.

Among the 10 subchannels existing in one video channel, 4 subchannels correspond to this. To this end, the upper layer sets the SCmake00 value of the UPtoDL.InfoMapOption interface to '1' or '3'. In addition, the UA can make a subchannel map with '00' for subchannels in which transmission overlaps with the tone subslot set mapped to the CSCH currently occupied by the UA. To this end, the upper layer sets the SCmake00 value of the UPtoDL.InfoMapOption interface to '2' or '3'.

6.3 Subchannel negotiation for allocation

6.3.1 General

The upper layer can determine in advance a CSCH to be allocated and transfer it to the data link. In this case, UPtoDL.ReqAllocatingDedicatedVSCH or UPtoDL.ReqUsingDedicatedVSCH is used. When a subchannel to be allocated is not determined, the UA and the controller shall first negotiate which subchannel to allocate before allocating a subchannel as shown in Figure 10. In this case, UPtoDL.ReqGetVSCH is used.

a) For subchannel negotiation, the controller first requests the UA to transmit a subchannel map with a PB 0x12.

b) When the UA receives this request, it transmits its own subchannel map to the controller using PB 0x13.

c) After receiving the UA's subchannel map, the controller selects one of the allocable subchannels. Then, it requests the UA to allocate the selected subchannel using PB 0x14.

d) If the subchannel requested for allocation is allocable, the UA transmits an ACK using PB 0x15 and goes to the subchannel allocation stage. If it is not allocable, it transmits its own subchannel map again with NACK.

e) When the controller receives an ACK, the controller goes to the subchannel allocation stage. Or receiving the NACK, the controller performs c) again. If all ACK/NACK is not received, the controller performs a) again or report a failure to the upper layer.

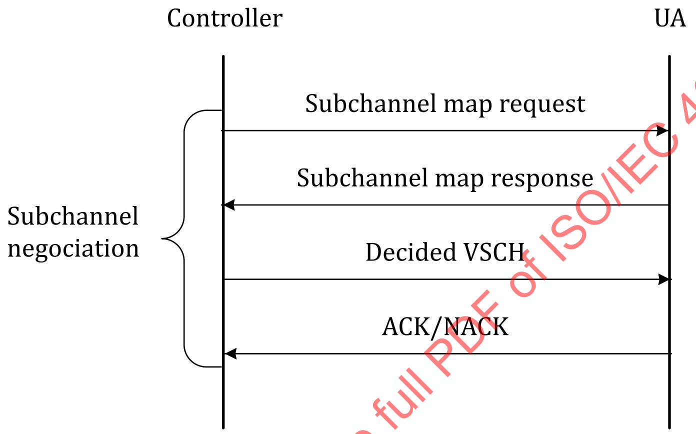


Figure 10 — General subchannel negotiation procedure

When the UA and the controller negotiate a subchannel number, they perform subchannel negotiation using shared communication or a dedicated slot or CSCH. Transmission power control related to subchannel negotiation is performed using PB 0x16 and PB 0x17. PB 0x16 represents its own transmission power and reception power of the counterpart's signal, and PB 0x17 represents its own transmission power and designates the power to be transmitted by the counterpart.

6.3.1.1 Generation of effective subchannel map

The controller receives the subchannel map from the UA and generate an effective subchannel map. The effective subchannel map is generated by selecting a lower value from the two subchannel maps.

6.3.1.2 Parsing block of shared channel related to subchannel allocation

The UA can broadcast its subchannel map using PB 0x89 in the slot of the shared channel. The parsing field of PB 0x89 is shown in Table 6.

Table 6 — Parsing field of PB 0x89

Bits	Description
[0:5]	Starting channel number
[6:10]	Map length that indicates [6:10]*20.
[11:[6:10]*20+10]	Variable subchannel map
[x:y] means from the x-th bit to the y-th bit of the related field.	

6.3.1.3 Parsing block of control channel related to subchannel allocation

The parsing blocks used in the control channel are 0x12, 0x13, 0x14, 0x15, 0x16, 0x17. PB 0x12 has no parsing field.

The parsing field of PB 0x13 is shown in Table 7.

Table 7 — Parsing field of PB 0x13

Bits	Description
[0:4]	Starting channel number
[5:9]	Map length is [5:9]*20.
[10:[5:9]*20+9]	Variable subchannel map

The parsing field of PB 0x14 is shown in Table 8.

Table 8 — Parsing field of 0x14

Bits	Description
[0:4]	Channel number
[5:9]	Subchannel number

The parsing field of PB 0x15 is shown in Table 9 and Table 10.

Table 9 — Parsing field of PB 0x15 with ACK

Bits	Description
[0]	This bit is '1' and means ACK.

Table 10 — Parsing field of PB 0x15 with NACK

Bits	Description
[0]	This bit is '0' and means NACK.
[1:5]	Starting channel number
[6:10]	Map length is [6:10]*20.
[11:[6:10]*20+10]	Variable subchannel map

The parsing field of PB 0x16 is shown in Table 11.

Table 11 — Parsing field of PB 0x16

Bits	Description
[0:6]	Own transmit power with the range of -40 dBm(0) ~ 40 dBm(80)
[7:13]	Received power of counterpart signal with the range of -115 dBm(1) ~ -10dBm(106), 0xFF means 'unknown' and 0x00 means smaller than -115dBm.

The parsing field of PB 0x17 is shown in Table 12.

Table 12 — Parsing field of PB 0x17

Bits	Description
[0:6]	Own transmit power with the range of -40 dBm(0) ~ 40 dBm(80)
[7:13]	Required transmit power of counterpart, the range is same with [0:6].

6.3.1.4 Subchannel selection for allocation

The data link of the controller generates an effective subchannel map using the subchannel map transmitted by the UA and its own subchannel map. In this case, when the UA transmits only a part of the subchannel map, all values that are not transmitted are regarded as a value of '00'. The effective subchannel map is achieved by selecting a lower value from the two subchannel maps, i.e. if one of the two values is '00', regardless of the other value, the value of the corresponding subchannel is '00'. The data link of the controller determines one subchannel for allocation from the effective subchannel map. If there is no valid subchannel, it is reported to the upper layer.

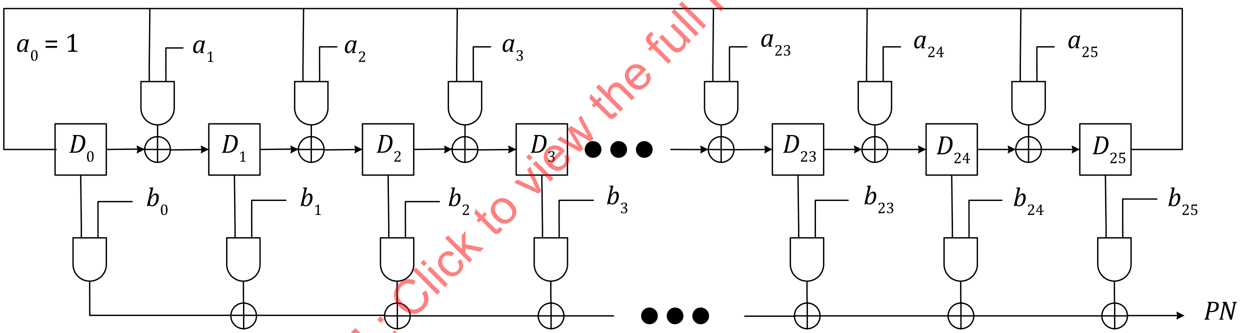
If there are subchannels having a value of '11', the controller shall select one among these subchannels.

If there are no subchannels having a value of '11', the controller can select one subchannel from among the subchannels having a value of '10' only when the value of SubChannelAbility10 of the UPtoDL.InfoMapOption interface is set to 1.

If there are no subchannels having a value of '11' or '10', the controller can select subchannel from among subchannels having a value of '01' only when the value of SubChannelAbility01 is set to 1.

If the number of selectable subchannels is $NUM_{\text{selectable\_subch}}$, the controller calculates the *Nordinal*-th usable subchannel from the effective subchannel map using a PN code generator. The order from the zeroth subchannel to the ($NUM_{\text{selectable\_subch}}$ -1)-th subchannel is determined by the following rules.

The smaller the *x* value in the subchannel $V_{x,y}$, the faster the order. And when the *x* values are the same, the smaller the *y* value, the faster the order.



Key

a_k a *k*-th bit for AND operation with D_{25}

b_k a *k*-th bit for AND operation with D_k

D_k a *k*-th register

PN PN code generator output

Figure 11 — PN code generator

The *Nordinal* is calculated with PN code generator shown in Figure 11 as follows where $[x_{i-1}, x_{i-2}, ..., x_0]$ is the *i*-bit binary representation of *x*.

$$[D_{25}(0), D_{24}(0), ..., D_0(0)] = [SA_{25}, SA_{24}, ..., SA_0]$$

$$[b_{25}, b_{24}, ..., b_{17}] = [SA_8, SA_7, ..., SA_0]$$

$$[b_{16}, b_{15}, ..., b_{11}] = [f_5, f_4, ..., f_0]$$

$$[b_{10}, b_9, ..., b_0] = [M_5, M_4, M_3, M_2, M_1, M_0, H_4, H_3, H_2, H_1, H_0]$$

$[a_{25}, a_{24}, ..., a_0] = [SA_0, SA_1, ..., SA_{24}, 1]$

$PN(clk) = (D_0(clk)\ \&\ b_0)\ \hat{}\ (D_1(clk)\ \&\ b_1)\ \hat{}\ ...\ \hat{}\ (D_{25}(clk)\ \&\ b_{25})$

$N_{PN} = [PN(15), PN(14), ..., PN(1), PN(0)]$

$$Nordinal = floor((N_{PN} \times NUM_{selectable\_subch})/2^{16}) \quad (18)$$

where

$D_x(clk)$	is a x-th bit of shift register D at clock clk;
SA_x	is a x-th bit of source address, the number of SA bits is greater than or equal 26;
f_x	is a x-th bit of frame number, the number of f bits is 6;
M_x	is a x-th bit of minute, the number of M bits is 6;
H_x	is a x-th bit of hour for 24-hour clock, the number of H bits is 5 and H has the value from 0 to 23;
$PN(clk)$	is a pseudo random bit of PN code generator at clock clk;
&	is AND bit operation that means a logical multiplication;
^	is exclusive OR bit operation that means exclusive logical sum operation;
$floor(x)$	means the largest integer among integers smaller than or equal to x.

6.3.2 Subchannel negotiation using shared channel

6.3.2.1 General

UAs and controllers can perform subchannel negotiation using shared channels. To do this, the controller shall allocate a talk slot of a shared channel to communicate with the UA. Allocating a talk slot in a shared channel shall follow the scheme described in ISO/IEC 4005-2. When transmitting and receiving data related to the video channel in the shared channel, PKH 0x06 is used. When PKH 0x06 is used, the PBs from 0x00 to 0x7F mean the PBs of the video channel. VC DLL receives UPtoDL.ReqGetVSCH from the upper layer. At this time, if the NegoMethod parameter is '0', VCtoSC.ReqNegoVSCH, which is a subchannel negotiation request, is transmitted to SC DLL. Upon receiving this, the SC DLL shall perform subchannel negotiation. When the subchannel negotiation is over, the SC DLL delivers SCtoVC.NotiNegoVSCH to the VC DLL. If the subchannel negotiation is successful, the VC DLL enters the subchannel negotiation stage, and if the subchannel negotiation fails, the VC DLL notifies the upper layer of the failure to DLtoUP.NotiGetVSCH.

The controller's SC DLL performs subchannel negotiation shown in Figure 12 as follows.

a) SC DLL receives the VCtoSC.ReqNegoVSCH requesting subchannel allocation from VC DLL. At this time, m and n are set. m and n can have negative values.

b) SC DLL requests the UA to transmit the UA subchannel map. For this, the controller shall transmit a request packet by allocating the talk slot of the SC. The PKH of the SC used here is 0x06, and the subchannel map transmission request PB is 0x12, where the controller SC DLL can use PB 0x81 to utilize the SC broadcast slot of UA for subchannel negotiation. If talk slot allocation and transmission fails, m is increased.

c) The SC DLL repeats b), increasing m until it receives PB 0x13. If PB 0x13 is not received and m is greater than 3, the SC DLL reports it to the VC DLL using SCtoVC.NotiNegoVSCH and performs

d) when the subchannel map is received, where the controller can receive the subchannel map through the response slot related to the allocated talk slot or the currently occupied broadcast slot of the UA.

d) After receiving the UA's subchannel map, the SC DLL generates an effective subchannel map and selects one subchannel from the effective subchannel map.

e) After allocating the talk slot, the SC DLL transmits the allocation request for the selected subchannel to the UA using PB 0x14. If talk slot allocation and transmission fails, *n* is increased.

f) When SC DLL receives ACK from PB 0x15, it notifies VC DLL to SCtoVC.NotiNegoVSCH. Upon receiving the NACK, d) is performed again. If PB 0x15 is not received, *n* is increased, and if *n* is greater than 3, it is reported to the VC DLL as SCtoVC.NotiNegoVSCH, and if it is less than or equal to 3, d) is performed again, where the controller receives ACK/NACK through a response slot related to the allocated talk slot or a currently occupied broadcast slot of the UA.

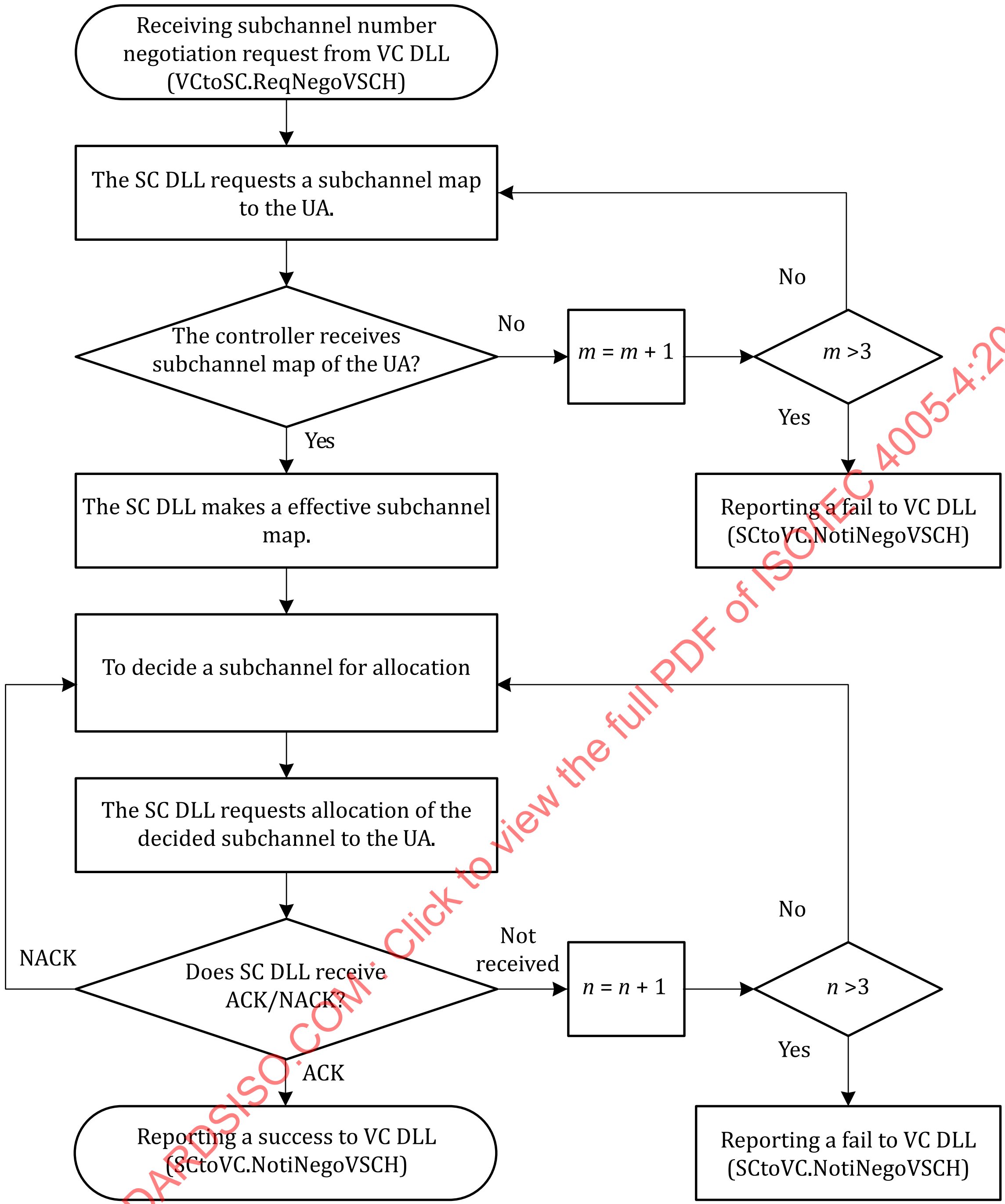


Figure 12 — Video subchannel (VSCH) negotiation procedure of controller using shared channel

The UA's SC DLL shall respond to the controller's request for subchannel negotiation. First, when a subchannel map request is received by PB 0x12, the UA's SC DLL shall transmit its subchannel map to the controller using PB 0x13. When a UA transmits a subchannel map, it shall transmit a subchannel map of a length that can be transmitted.

The UA's SC DLL receives a subchannel allocation request by PB 0x14, and if the subchannel is available for allocation, transmits an ACK to the controller using PB 0x15, and informs the VC DLL of the ACK transmission through SCtoVC.NotiNegoVSCH. If the subchannel cannot be allocated, it transmits NACK and its own subchannel map to the controller. After UA's VC DLL receives SCtoVC.NotiNegoVSCH, if the parameter IsSuccess is 1, it enters the resource allocation step.

If PB 0x81 is included in the packet sent by the controller and the parsing field RequestMethod has a value of '0', the UA responds with the SC broadcast slot it has already occupied. If the parsing field RequestMethod has a value of '1', the UA responds with the SC broadcast slot it has already occupied, and simultaneously performs response clearing and allocation competition, and responds with a talk slot allocated by the controller. If PB 0x81 is not included in the controller's packet, response clearing and allocation competition are performed to respond with a talk slot allocated by the controller.

When responding to the SC broadcast slot, the video channel parsing block is transmitted using PB 0x8C. The parsing field of the PB 0x8C carries the video parsing block. When responding with an SC talk slot, PKH 0x06 is used.

6.3.2.2 Transmission power in subchannel negotiation

When the controller and UA perform subchannel negotiation with the SC, they use the transmit power specified in the SC.

6.3.3 Subchannel negotiation using dedicated slot

6.3.3.1 General

Dedicated slots designated by upper layer in ISO/IEC 4005-3 can be used for VSCH negotiation.

The UA and the controller can perform VSCH negotiation using a predetermined dedicated slot. The procedure for the controller to perform subchannel negotiation is as follows.

a) The VC DLL receives the subchannel allocation command UPtoDL.PBReqGetVSCH from the upper layer. At this time, NegoMethod is '1', and *m* and *n* are set and VC DLL sends VCtoCC.ReqNegoVSCH to CC DLL.

b) The CC DLL requests the UA to transmit the UA subchannel map using PB 0x12. For this, the controller uses a dedicated slot.

c) The CC DLL repeats b) while increasing *m* until it receives the UA subchannel map through PB 0x13. If the subchannel map is not received and *m* is greater than 3, the CC DLL delivers the failure to the VC DLL with CCtoVC.NotiNegoVSCH, and the VC DLL reports this to the upper layer with DLtoUP.NotiGetVSCH, if the subchannel map is received then and d) is performed, where the CC DLL of the controller receives the subchannel map in the next dedicated slot of the dedicated slot used in b).

d) After receiving the UA's subchannel map, the CC DLL makes an effective subchannel map, and selects one subchannel from the effective subchannel map.

e) The CC DLL transmits the allocation request for the selected subchannel to the UA using PB 0x14.

f) When the CC DLL receives an ACK from the UA, it transmits it to the VC DLL using CCtoVC.NotiNegoVSCH. Upon receiving the ACK, the VC DLL reports it to the upper layer as DLtoUP.NotiGetVSCH and goes to the subchannel allocation stage. When the controller's CC DLL receives a NACK from the UA, it executes d) again. When neither ACK nor NACK is received, *n* is increased, and if *n* is greater than 3, CC DLL transmits a failure to the VC DLL using CCtoVC.NotiNegoVSCH. Upon receiving the failure, the VC DLL reports the failure to the upper layer with DLtoUP.NotiGetVSCH. If *n* is less than or equal to 3, the CC DLL performs d) again where the CC DLL receives ACK or NACK in the next dedicated slot of the dedicated slot used in e).

The UA continuously receives its dedicated slots according to the settings of the upper layer. When a UA receives a packet requiring a response from the controller, it shall transmit a response in a dedicated slot next to the dedicated slot in which the request was received.

When a subchannel map request PB 0x12 is received in a dedicated slot, the UA's CC DLL shall transmit its subchannel map to the controller using PB in a dedicated slot next to the dedicated slot in which the request was received.

After receiving the subchannel allocation request PB 0x14, the UA's CC DLL responds with PB 0x15. At this time, if the subchannel is available for allocation, ACK is transmitted to the controller, and ACK transmission is notified to the VC DLL using CCtoVC.NotiNegoVSCH. When the VC DLL receives the ACK transmission, it enters the subchannel allocation stage. If the subchannel cannot be allocated, the UA's CC DLL transmits NACK and its subchannel map to the controller.

6.3.3.2 Transmission power in subchannel negotiation

The power during subchannel negotiation is controlled by the CC DLL, as described in ISO/IEC 4005-3. However, instead of PB 0x06 and PB 0x07, PB 0x16 and PB 0x17 are used.

6.3.4 Subchannel negotiation using CSCH

6.3.4.1 General

When the CSCH is in use, the controller can use the CSCH for VSCH negotiation.

The procedure for the controller to perform subchannel negotiation is as follows.

a) The VC DLL receives the subchannel allocation command UPtoDL.PBReqGetVSCH from the upper layer. At this time, NegoMethod is '3', and *m* and *n* are set. VC DLL sends VCtoCC.ReqNegoVSCH to CC DLL.

b) The CC DLL requests the UA subchannel map transmission to the UA using PB 0x12. For this, the controller uses the uplink slot of the occupied CSCH.

c) The CC DLL repeats b) while increasing m until it receives the UA subchannel map through PB 0x13. If the subchannel map is not received and *m* is greater than 3, the CC DLL delivers the failure to the VC DLL with CCtoVC.NotiNegoVSCH and the VC DLL reports this to the upper layer with DLtoUP.NotiGetVSCH. When the subchannel map is received, d) is performed. Where, the CC DLL of the controller receives the subchannel map in the first downlink slot after the uplink slot used in b).

d) After receiving the UA's subchannel map, the CC DLL makes an effective subchannel map, and selects one subchannel from the effective subchannel map.

e) The CC DLL transmits the allocation request for the selected subchannel in the uplink slot to the UA using PB 0x14.

f) When the CC DLL receives an ACK from the UA, it transmits it to the VC DLL using CCtoVC.NotiNegoVSCH. Upon receiving the ACK, the VC DLL reports it to the upper layer as DLtoUP.NotiGetVSCH and goes to the subchannel allocation stage. When the CC DLL of the controller receives a NACK from the UA, it executes d) again. If both ACK and NACK are not received, *n* is increased, and if *n* is greater than 3, the failure is transmitted to the VC DLL using CCtoVC.NotiNegoVSCH. Upon receiving the failure, the VC DLL reports the failure to the upper layer with DLtoUP.NotiGetVSCH. If *n* is less than or equal to 3, the CC DLL performs d) again, where the CC DLL receives ACK or NACK in the first downlink slot after the uplink slot used in e).

In this procedure, the controller can transmit a response in a plurality of uplink slots for one downlink slot, i.e. the controller can transmit PB 0x12 or PB 0x14 in a plurality of slots. This is implementation dependent.

When the UA receives PB 0x12 or PB 0x14 from the controller, it shall transmit PB 0x13 or PB 0x15 in the first downlink slot after the uplink slot that received it.

When the UA receives PB 0x12, the UA's CC DLL shall transmit its subchannel map to the controller using PB 0x13.

The UA's CC DLL responds with PB 0x15 after receiving PB 0x14. At this time, if the subchannel is available for allocation, ACK is transmitted to the controller, and ACK transmission is notified to the VC DLL using CCtoVC.NotiNegoVSCH. When the UA's VC DLL receives an ACK transmission from the CC DLL, it enters the subchannel allocation stage. The UA's CC DLL transmits a NACK and its own subchannel map to the controller if the subchannel cannot be allocated.

6.3.4.2 Transmission power in subchannel negotiation

The transmission power of the CSCH is described in ISO/IEC 4005-3, and follows ISO/IEC 4005-3 when subchannel negotiation is performed using the CSCH occupied by the unit.

6.4 Subchannel allocation and generated link confirmation

6.4.1 General

When subchannel negotiation is completed, each unit shall first perform allocation competition in the tone subslot set mapped with the corresponding subchannel. At this time, the UA and the controller compete at the same time.

If the UA successfully allocates resources, it notifies the upper layer with DLtoUP.NotiGetVSCH. The UA immediately goes to the subchannel occupation stage without the generated link confirmation stage and starts transmitting video packets using the occupied VSCH.

If the controller fails to allocate resources, it reports the failure to the upper layer through DLtoUP.NotiGetVSCH. If the controller succeeds in allocating resources, it transmits PB 0x17 to the UA in the next frame and performs a generated link confirmation for 1 sec. The controller calculates the video packet error rate for 1 sec, and if this error rate is less than the threshold error rate LinkConfirmError, enters the resource occupation stage and reports the allocation success to the upper layer through DLtoUP.NotiGetVSCH. If the error rate for 1 sec is greater than the threshold error rate, the controller reports an allocation failure to the upper layer with DLtoUP.NotiGetVSCH. Figure 13 shows resource allocation competition and generated link confirmation of a controller.

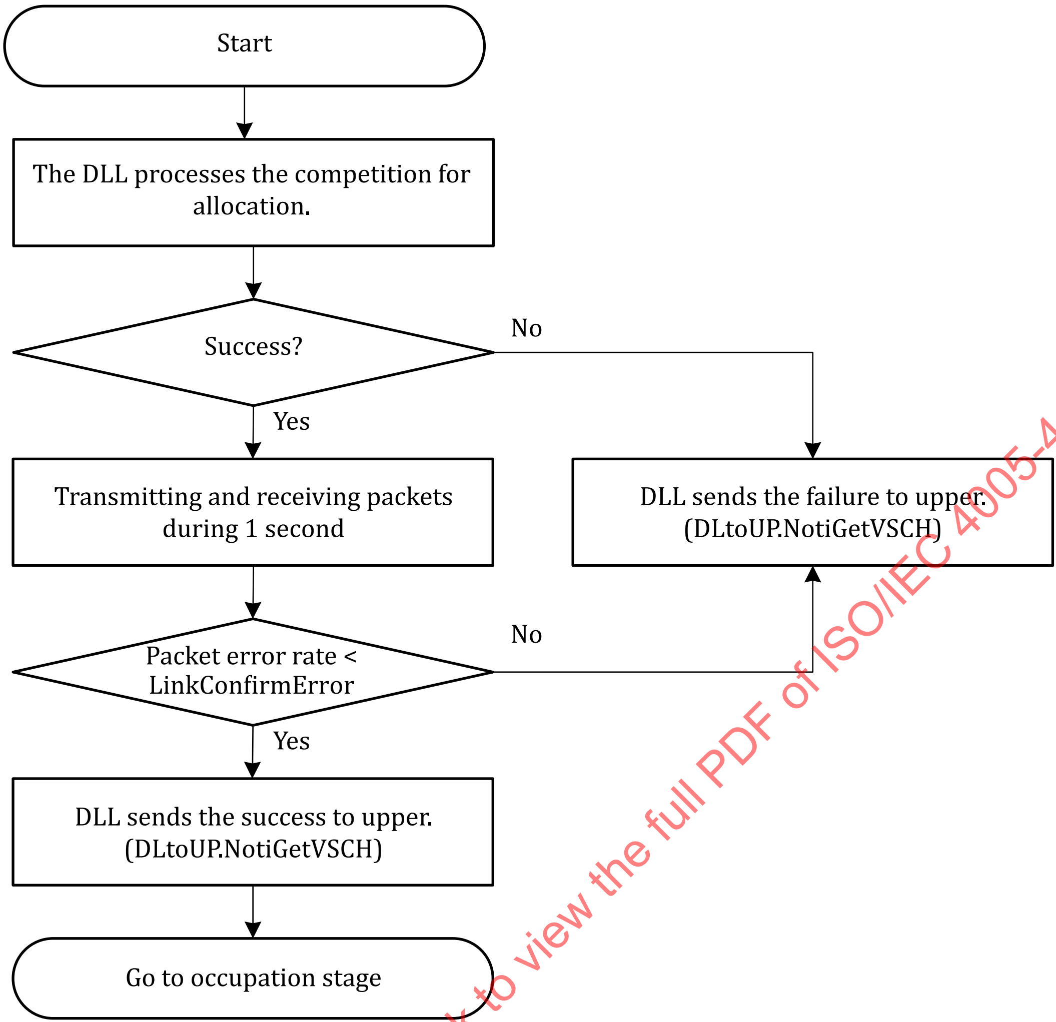


Figure 13 — Resource allocation competition and generated link confirmation of a controller

6.4.2 Subchannel resource allocation competition

6.4.2.1 General

The competition for allocating VSCH is same with that of CSCH except power control.

6.4.2.2 Tone subslot transmission power in subchannel allocation

When the controller and UA perform subchannel allocation, the transmit power of the tone subslot signal, *PTX_comp_Tone*, is calculated as follows:

First, when subchannel negotiation is performed with an SC signal, it is as follows:

$$PTX\_comp\_Tone = \mathrm{MIN}(\ (\mathrm{PtargetVCH} + PL\_SC + \mathrm{SNRrequiredVCH} + \mathrm{PmarginTCH}\ ),\ \mathrm{PmaxTCH}) \quad (19)$$

where

PtargetVCH	is the target reception power of the VSCH;
PL_SC	is the path loss measured from the received SC signal;

SNRrequiredVCH is the SNR required for the control channel to be received;

PmarginTCH is the power margin for the tone channel associated with the video channel.

The data link receives the main power parameter as UPtoDL.InfoPowerParamVCH from the upper layer.

Second, when negotiation is performed with a dedicated slot or a CSCH, the unit uses a value obtained by adding PTX_VCHTCH_differ to the last transmission power used in subchannel negotiation.

6.4.3 Generated link confirmation

6.4.3.1 General

Confirmation of link formation is performed by the controller.

If the controller's VC DLL succeeds in allocating resources, it performs a generated link confirmation for 1 sec from the next frame of the allocation success frame. It receives packets from the UA for 1 sec and calculates the received packet error rate. If the received packet error rate is less than the LinkConfirmError of UPtoDL.InfoVideoChannel, the VC DLL reports the allocation success to the upper layer as DLtoUP.NotiGetVSCH and enters the resource occupation state where the received packet error rate is a value obtained by dividing the number of packets received without an error for 1 sec by the number of slots received for 1 sec.

If the received packet error rate is greater than the LinkConfirmError of UPtoDL.InfoVideoChannel, the controller immediately returns the subchannel and transmits an allocation failure to the UA using PB 0x1C. Thereafter, when PB 0x1C is received from the UA, the VC DLL of the controller reports a failure to the upper layer with DLtoUP.NotiGetVSCH. If a PB 0x1C response is not received from the UA, PB 0x1C is retransmitted to the UA for a maximum of TimeOf1CTry seconds until PB 0x1C is received. Thereafter, even if the PB 0x1C response is not received, the controller's VC DLL reports the failure to the upper layer to DLtoUP.NotiGetVSCH.

When the UA in the subchannel occupation state receives PB 0x1C from the controller, it stops transmitting packets from the next frame and returns the occupied subchannel. At this time, the UA shall send a PB 0x1C to the controller to inform that it has returned the VSCH. In addition, the UA's VC DLL reports a failure to the upper layer with DLtoUP.NotiGetVSCH.

All VSCHs are made up of downlink slots. Therefore, the controller can transmit PB to the UA using SC, CSCH, and DS. The UA can transmit PB to the controller using SC, CSCH, DS, and VSCH. The specific transmission method is implementation dependent.

In the generated link confirmation stage, the UA transmits video packets in slots of the allocated subchannel. When the UA transmits a packet, it can include a PB composed of ACKs for a packet received from the controller. These PBs can be specified in upper layers.

In the generated link confirmation stage, after receiving a video packet in a downlink slot of an allocated subchannel, the controller shall transmit PB 0x1B composed of ACKs for the received video packet to the UA at least once in the next frame.

6.4.3.2 Transmitting and receiving parsing block of controller

In the generated link confirmation stage, the controller can transmit the PB using SC, CSCH, and DS. Related interfaces are VCtoSC.PBTxReq and VCtoCC.PBTxReq. The method of transmission is implementation dependent. However, the controller cannot use the SC when the UA's SC slot signal is not received, cannot use the CSCH when the CSCH is not used, and cannot use the DS when the DS is not allocated.

The controller's VC DLL can receive PB through SC, CSCH, DS, and VSCH. Related interfaces are SCtoVC.PBRxFwd and CCtoVC.PBRxFwd.

6.4.3.3 UA transmission power control at generated link confirmation stage

The transmission power of the VSCH in one frame is the same. The UA determines the initial VSCH transmission power as follows.

a) In case of performing subchannel negotiation with SC signal:

$$PTX\_DR\_VCH\_ini = \mathrm{MIN}((\mathrm{PtargetVCH} + PL\_SCc + \mathrm{SNRrequiredVCH} + \mathrm{PmarginVCH}), \mathrm{PmaxVCH}) \quad (20)$$

where

PtargetVCH	is the target reception power of the VCH;
PL_SCc	is the path loss measured by the UA receiving the SC signal from the controller;
SNRrequiredVCH	is the SNR required to receive the video channel;
PmarginVCH	is the power margin for the video channel.

DLL receives main parameters from upper layer as UPtoDL.InfoPowerParamVCH.

b) When negotiation is performed with a dedicated slot or CSCH, the DLL calculates a value obtained by adding PTX_VCHCCH_differ to the last transmission power used during negotiation, *PTX_DR_DS_prev* or *PTX_DR_CC_prev*, as transmission power:

$$PTX\_DR\_VCH\_ini = (PTX\_DR\_DS\_prev \text{ or } PTX\_DR\_CC\_prev) + \mathrm{PTX\_VCHCCH\_differ} \quad (21)$$

The UA determines the VSCH transmission power plus PTX_VCHTCH_differ as the transmission power of the tone subslot.

The controller shall transmit PB 0x17 to the UA at least once in the generated link confirmation stage, and the UA's transmission power shall be set as above.

6.4.4 Broadcasting video subchannel (VSCH) information being allocated or occupied

The UA shall broadcast information on the VSCH that it is attempting to allocate or occupying through the occupying SC slot. The related PBs are PB 0x88 and PB 0x8A, and the related parsing fields are shown in Table 13 and Table 14.

Table 13 — Parsing field of PB 0x88 in SC

Bits	Description
[0:5]	Video channel number
[6:9]	VSCH number, the range is 0 ~ 9.
[10:33]	Own latitude, see ISO/IEC 4005-2:2023, B.1
[34:57]	Own longitude, see ISO/IEC 4005-2:2023, B.2
[58:69]	Own altitude, see ISO/IEC 4005-2:2023, B.3
[70:85]	Latitude difference between the controller and the UA, see ISO/IEC 4005-2:2023, B.7
[86:105]	Longitude difference between the controller and the UA, see ISO/IEC 4005-2:2023, B.8
[106:113]	Altitude difference between the controller and the UA, see ISO/IEC 4005-2:2023, B.9
[114:120]	VSCH TX power of the UA, the range of -40dBm(0) ~ 40dBm(80)

Table 14 — Parsing field of PB 0x8A in SC

Bits	Description
[0:4]	Control channel number

Table 14 *(continued)*

Bits	Description
[5:9]	CSCH number, the range is 0 ~ 19.
[10:34]	Own latitude, see ISO/IEC 4005-2:2023, B.1
[35:59]	Own longitude, see ISO/IEC 4005-2:2023, B.2
[60:71]	Own altitude, see ISO/IEC 4005-2:2023, B.3
[72:87]	Latitude difference between the controller and the UA, see ISO/IEC 4005-2:2023, B.7
[88:103]	Longitude difference between the controller and the UA, see ISO/IEC 4005-2:2023, B.8
[104:113]	Altitude difference between the controller and the UA, see ISO/IEC 4005-2:2023, B.9
[114:120]	CSCH TX power of the UA, the range of -40dBm(0) ~ 40dBm(80)
[121:126]	Video channel number
[127:130]	VSCH number, the range is 0 ~ 9.
[131:137]	VSCH TX power of the UA, the range of -40dBm(0) ~ 40dBm(80)

6.5 Subchannel occupation and collision management

6.5.1 General

In the occupation stage, the UA shall broadcast VSCH information in a SC slot using PB 0x88 or PB 0x8A.

The controller shall use PB 0x17 to control the power of the UA. The UA shall transmit the received power of the packet transmitted by the controller using PB 0x16.

At this stage, the UA and the controller occupy the subchannel. Occupation is achieved by slot clearing. The UA and the controller perform slot clearing tone and collision tone transmission in the tone subslot set mapped with the occupied subchannel. Resource collision monitoring is accomplished by detecting the collision tone of other units. Measurement of the amount of interference for neighbouring channels is also performed.

6.5.2 Power control in occupation stage

The controller uses PB 0x17 to inform its own transmission power and adjusts the UA's transmission power. The UA shall use PB 0x16 to inform the VSCH transmission power and the reception power of the packet including PB 0x17. The controller can calculate path loss using PB 0x16 received from the UA or PB 0x17 transmitted by itself. It is implementation dependent for the controller to perform the UA's transmit power control using the calculated path loss. However, the following rules shall be observed in transmit power control.

a) The transmit power within one frame is the same. The power change is applied when the frame changes.

b) If the transmission power is less than or equal to Pmax_dmap1, the maximum power increase per frame is 5 dB, and if the transmission power is greater than or equal to Pmax_dmap1 and less than Pmax_dmap2, the maximum power increase per frame is 3 dB, and if the transmission power is Pmax_dmap2 or more, the maximum increase power per frame is 1 dB. Where, Pmax_dmap1 and Pmax_dmap2 are received as UPtoDL.InfoPowerParamVCH from the upper layer.

c) The maximum power Pmax_occupy in the occupation state is generally equal to $\mathrm{PmaxVCH}_y^x$. However, when a unit using the same channel resource is found in the SC, power control is as follows. If the distance to the unit is greater than d_map1, Pmax_occupy is Pmax_dmap2, and if the distance to the unit is less than or equal to d_map1 and greater than d_map0, Pmax_occupy is Pmax_dmap1. If the distance to the unit is less than or equal to d_map0, the unit maintains the current power for the associated subchannel.

The UA and the controller determine the VSCH transmission power plus PTX_VCHTCH_differ as the transmission power of the tone subslot. The transmit power of the tone subslot cannot exceed PmaxTCH.

6.5.3 Subchannel occupation and return method

The occupation of the subchannel is achieved by the unit transmitting a tone signal in subslot 0 of the tone subslot set mapped with the subchannel, i.e. the UA and the controller each perform slot clearing. The return of the occupied subchannel is made by stopping slot clearing. The upper layer can request the return of the occupied VSCH by UPtoDL.ReqReturnVCH as necessary.

6.5.4 Collision tone transmission and collision management

Collision tone transmission and collision management of the VSCH are the same as those of the CSCH. However, the following are excluded.

a) PRXcollsiontoneThre0 and PRXcollsiontoneThre1 are received from the upper layer as UPtoDL. InfoPowerParamVCH.

b) When it is determined to return the subchannel resource, the unit shall inform the counterpart unit of it using PB 0x1D. The unit that receives the PB 0x1D indicating the decision to return the subchannel resource shall transmit the PB 0x1D whose parsing field value is 4. This PB 0x1D exchange shall take place within TimeOf1DTry seconds. If the PB 0x1D exchange fails, the unit immediately returns the channel.

6.5.5 Parsing block for video channel

6.5.5.1 ACK transmission

After receiving the VSCH slot, the controller can transmit an ACK for each slot. ACK transmission is performed when the value of UAAckReq is '1'.

The fields of PB 0x1B transmitted by the controller are shown in Table 15.

Table 15 — Parsing field of PB 0x1B

Bits	Description
[0:24]	25 ACK bits, 1 means ACK, 0 means NACK, ACKs for a downlink video slot during one frame

6.5.5.2 PB 0x18

PB 0x18 has a parsing block for the SC as a parsing field.

6.5.5.3 PB 0x19

PB 0x19 has a parsing block for the CC as a parsing field.

6.6 Reallocation

6.6.1 General

When the same resource collision occurs or when the neighbouring channel interference is greater than the threshold value, the unit that detects this shall reallocate the subchannel. First, the unit can detect the same resource collision as a collision tone or SC reception information. The neighbouring channel interference can be detected by receiving the mapping subslot sets of the tone channel or as SC reception information.

6.6.2 Reallocation decision

6.6.2.1 Reallocation decision due to same resource collision

The reallocation decision following collision tone detection is described in 6.5.3.

Units allocated the same subchannel can be detected by SC reception. In this case, subchannel reallocation is determined under the following conditions.

a) When another UA allocated the same subchannel is detected, and the distance between the UA and the controller connected to the UA is 500 m or more.

b) When another UA allocated the same subchannel is detected, and the UA's transmission power is Pmax_dmap2 or higher.

c) When another unit allocated the same subchannel is detected, and the distance between own unit and another unit is less than d_map0,

d) When the distance between own unit and another unit that has transmission power of Pmax_dmap1 or more is more than d_map0, and less than or equal to d_map0.

6.6.2.2 Reallocation decision due to neighboring channel interference

Since VSCH is a downlink, there is no channel interference between UAs or between controllers.

The measurement of the amount of interference due to neighboring channels using a competition tone channel is described in 6.2.2, and the unit decides subchannel reallocation under the following conditions.

a) When the calculated interference power is greater than or equal to (PImin – PImargin – 6dB) and less than (PImin – PImargin – 3dB), which lasts for 4 sec.

b) When the calculated interference power is greater than or equal to (PImin–PImargin–3dB) and less than (PImin–PImargin) , which lasts for 2 sec.

c) When the calculated interference power is greater than or equal to (PImin – PImargin).

PImin and PImargin are received as UPtoDL.InfoPowerParamVCH from the upper layer.

According to the SC reception information, the unit determines subchannel reallocation when interference with the neighboring channel is as follows.

The interference power is calculated from the transmission power and the current position of the UA using the interference channel, and if the interference power is greater than PIrealloc0, the controller determines reallocation, where the interference power is calculated as follows.

$$PI_{DjtoCi} = PTXvschDR_j - L_{fs}(d) - IC_{|i-j|} \tag{22}$$

where

i	is the channel number of the controller;
j	is the channel number used by the interference UA;
$PTXvschDR_j$	is the VSCH transmission power of the interference channel j;
d	is the distance between the controller and the interference UA in km.

The free space path loss $L_{fs}(d)$ is calculated as follows.

$$L_{fs}(d) = 32{,}45\ \text{dB} + 20\log_{10}(d_{km} \cdot f_{MHz}) \quad (23)$$

where f_{MHz} is the centre frequency of the interference channel used by the UA where PIrealloc0 is received as UPtoDL.InfoPowerParamVCH from the upper layer.

Subchannel reallocation is recommended when the neighbouring channel interference calculated from the SC reception information is the following conditions.

If the future interference power calculated from the transmission power and location and the future UA's way point of the UA using the neighbouring channel are expected to be greater than PIrealloc1, the controller can determine reallocation, where the interference power is calculated as follows.

$$PIf_{DjtoCi} = PTXvschDR_j - L_{fs}(d_f) - IC_{|i\text{-}j|} \quad (24)$$

where

i	is the channel number of the controller;
j	is the channel number used by the interference UA;
$PTXvschDR_j$	is the VSCH transmission power of the interference channel j;
d_f	is the minimum distance between the controller and the interfering channel UA that will be obtained within the next 10 sec.

It shall be calculated from the way points, which is implementation dependent. The free space path loss $L_{fs}(d_f)$ is calculated as follows.

$$L_{fs}(d_f) = 32{,}45\text{dB} + 20\log_{10}(d_{f\,km} \cdot f_{MHz}) \quad (25)$$

where f_{MHz} is the center frequency of the interference channel used by the UA where PIrealloc1 is received as UPtoDL.InfoPowerParamVCH from the upper layer.

6.6.2.3 Reallocation decision by increasing packet error rate

Subchannel reallocation by increasing packet error rate is implementation dependent.

6.6.3 Subchannel reallocation procedure

6.6.3.1 General

When the UA or controller decides to reallocate, the VC DLL shall notify the upper layer using DLtoUP. NotiVCHStatus and inform the counterpart unit of this using PB 0x1D. The unit that receives the PB 0x1D indicating the decision to return the subchannel resource shall transmit the PB 0x1D whose parsing field value is 4. This PB 0x1D exchange shall take place within TimeOf1DTry seconds. If the PB 0x1D exchange fails, the unit immediately returns the channel.

If the UA decides to reallocate, it can transmit its own subchannel map using PB 0x13 together with PB 0x1D. If the controller decides to reallocate, it can request the UA's subchannel map transmission using PB 0x12 with PB 0x1D.

When subchannel reallocation is performed, the existing allocated subchannel is occupied for maximum TVCHReturns. If reallocation is completed within TVCHReturn, the existing subchannel is immediately returned. TVCHReturn is received as UPtoDL.InfoTimeParam from the upper layer.

The subchannel reallocation procedure is the same as the subchannel allocation procedure except for subchannel negotiation.

Subchannel negotiation is first performed by a method indicated by the ReallocMethod parameter of UPtoDL.InfoVideoChannel. Regardless of the value of ReallocMethod, if the controller does not receive the UA's SC slot, the controller cannot perform subchannel negotiation with SC. Likewise, if the controller does not occupy the CC, subchannel negotiation cannot be performed with the CC. Likewise, if there is no DS allocated by the controller, subchannel negotiation cannot be performed with a DS.

The method of performing allocation subchannel negotiation between the UA and the controller is the same as in 6.3, but in the case of subchannel reallocation, additionally currently allocated subchannels can be used simultaneously, i.e. the UA can respond to subchannel negotiation using the currently allocated VSCH.

The operation of using the currently allocated VSCH together in the allocation subchannel negotiation is implementation dependent.

If subchannel reallocation is successful, each unit shall inform the upper layer of this as DLtoUP.NotiVCHStatus.

A UA or controller performing the subchannel reallocation procedure can continue to perform power control for the currently allocated subchannel.

6.6.3.2 Parsing block for reallocation

The configuration of the 0x1D parsing field indicating the reallocation decision is shown in Table 16.

Table 16 — Parsing field of PB 0x1D

Bits	Description
[0:2]	0 – Collision tone detection 1 – Collision detection of the same resource through SC reception 2 – Threshold excess of neighbouring channel interference 3 – Prediction of threshold excess of neighbouring channel interference 4 – Response to counterpart unit's decision to reallocate

6.7 Data exchange

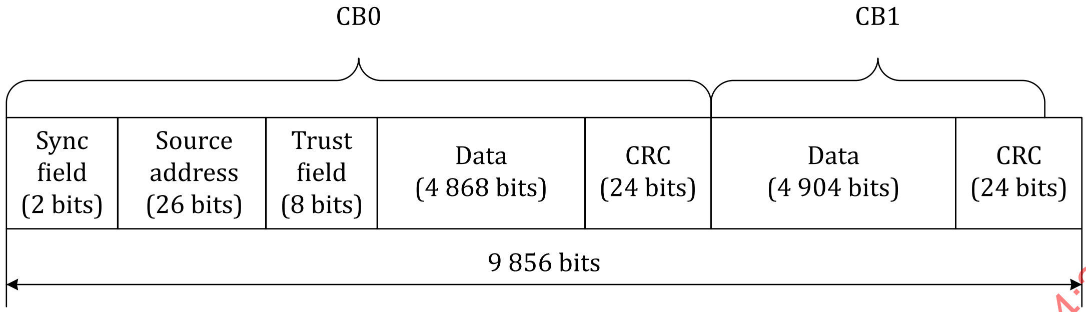
6.7.1 General

Parsing blocks that are equal to or higher than 0x80 share the same parsing blocks with SC. PBs that can be used in the video channel are PBs from 0x00 to 0x7F, which is shared with the control channel.

The upper layer can request packet transmission of the DLL using UPtoDL.ReqTxVCH. The DLL shall deliver the received packet to the upper layer through DLtoUP.RsvVCHData. At this time, if all the parsing blocks included in the received packet are in the header list of Table 18, it is possible that they are not delivered to the upper layer.

6.7.2 Data packet format

6.7.2.1 General



Key

CB0 code block 0

CB1 code block 1

Figure 14 — Video channel packet format

The packet format of the data link layer is shown in Figure 14. The source address is generally 26 bits, but the upper layer can allocate a larger address with UPtoDL.InfoPacketParam. The total length of the packet is 9856 bits, and the maximum length of data varies according to the length of the source address. The upper layer can deliver data of length less than or equal to the maximum data length to DLL. The CRC is added at the physical layer, and the part from sync mode to the first CRC is CB0, and the latter part is CB1.

6.7.2.2 Sync field

The sync field is 2 bits. The meaning of the sync field is shown in Table 17.

Table 17 — Sync field meaning

Value	Description
0x0	A sync
0x1	B sync
0x2	C sync
0x3	reserved

6.7.2.3 Source address

It is recommended to set the source address to 26 bits. However, the upper layer can set the unit address bit number larger. In this case, the upper layer shall shorten the maximum data field length by the increased number of bits.

6.7.2.4 Trust field

The trust field generation method of VC is same with that of CC.

6.7.2.5 Data field

6.7.2.5.1 General

The data field consists of parsing blocks. The parsing block consists of a parsing header and a parsing field. The parsing header is 8 bits, and the length of the parsing field depends on the parsing header.

Each parsing block can be inserted anywhere in the data field. However, a parsing block and another parsing block shall be attached without empty space, and the first parsing block shall be located immediately after the trust field.

The upper layer shall transmit data of a length less than or equal to the length of the data field to the data link layer. If the length of information provided from the upper layer is smaller than the length of the data field, the remaining bits are processed as follows.

If the length of unused extra data bits is less than 8, the extra data bits are filled with '0'. If the length of unused extra data bits is greater than or equal to 8, the extra data bits are filled with a parsing header '0x80' and a padding bit string of length 97 indicating padding. The padding bits are constructed by using a string of bits of length 97 repeatedly. A 97-bit sequence consists of the next 96-bit sequence followed by a bit of '1'.

- ‘0x71E5477D_A5B32BF7_E5469C8E’

where '_' is used to distinguish 32 bits without meaning.

6.7.2.5.2 Parsing header and parsing field

The parsing header consists of 8 bits. If the most significant bit of the parsing header is '1', it has the same meaning in common in SC, CC, and VC. When the most significant bit of the parsing header is '0' in VSCH, PH in Table 18 has meanings related to VC and CC. The upper layer can define a parsing header different from Table 18.

Table 18 — Parsing header list

Value	Description
0x12	Request to transmit a subchannel map, it does not have a parsing field.
0x13	Subchannel map
0x14	Subchannel allocation request
0x15	ACK on subchannel allocation request
0x16	Own transmission power and reception power of the counterpart's signal
0x17	Own transmission power and designation of the counterpart’s transmission power
0x1B	ACKs for downlink packets
0x1C	Link confirmation failure, this has no parsing field.
0x1D	Notification of reallocation decision

6.8 Synchronization

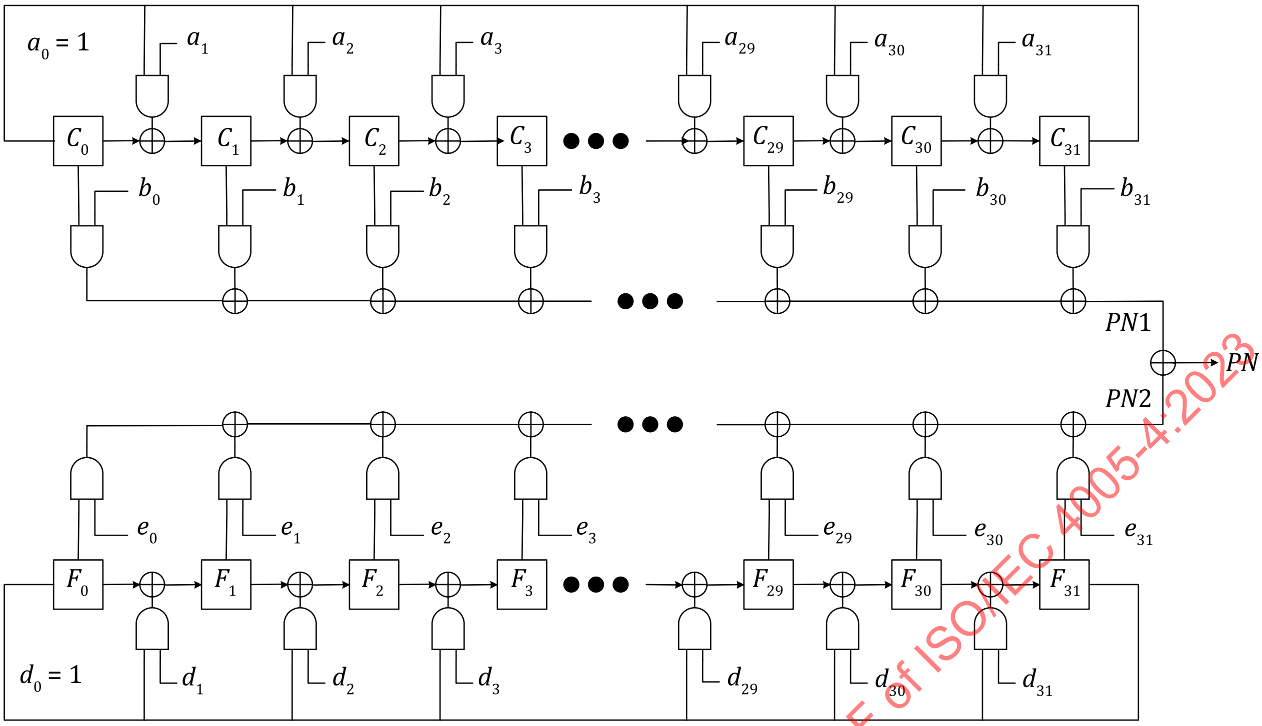
A sync, B sync and C sync units shall know the date, hour, minute, second, slot number in order to transmit a packet.

6.9 Data link layer security

In video communication, units generally perform one-to-one communication. Security can be applied depending on the service of one-on-one communication.

If the upper layer requests the use of data link security, then the following security shall be used.

The data link layer receives the security key U from the upper layer. The data field is scrambled by the PN code generated by the PN generator as shown in Figure 15.



Key

a_k	a k-th bit for AND operation with C_{31}
b_k	a k-th bit for AND operation with C_k
C_k	a k-th register of upper LFSR
F_k	a k-th register of lower LFSR
d_k	a k-th bit for AND operation with F_{31}
e_k	a k-th bit for AND operation with F_k
$PN1$	output of upper LFSR
$PN2$	output of lower LFSR
PN	output of PN code generator

Figure 15 — PN code generator used for data scrambling

$[a_{31}, a_{30}, ..., a_0] = [\mathrm{U}_{26}, \mathrm{U}_{25}, s_8, \mathrm{U}_{24}, ..., \mathrm{U}_{16}, s_6, \mathrm{U}_{15}, ..., \mathrm{U}_8, s_3, \mathrm{U}_7, ..., \mathrm{U}_0, s_0, 1]$

$[b_{31}, b_{30}, ..., b_0] = [\mathrm{U}_{55}, ..., \mathrm{U}_{47}, f_5, \mathrm{U}_{46}, ..., \mathrm{U}_{37}, f_3, \mathrm{U}_{36}, ..., \mathrm{U}_{28}, f_0, \mathrm{U}_{27}]$

$[C_{31}(0), C_{30}(0), ..., C_0(0)] = [\mathrm{U}_{77}, ..., \mathrm{U}_{73}, M_5, M_4, M_2, H_4, H_3, H_2, \mathrm{U}_{72}, ..., \mathrm{U}_{61}, Y_3, Y_1, W_3, W_1, \mathrm{U}_{60}, ..., \mathrm{U}_{56}]$

$[d_{31}, d_{30}, ..., d_0] = [\mathrm{U}_{103}, s_7, s_5, \mathrm{U}_{102}, ..., \mathrm{U}_{93}, s_4, \mathrm{U}_{92}, ..., \mathrm{U}_{85}, s_2, \mathrm{U}_{84}, ..., \mathrm{U}_{78}, s_1, 1]$

$[e_{31}, e_{30}, ..., e_0] = [\mathrm{U}_{132}, ..., \mathrm{U}_{126}, f_4, \mathrm{U}_{125}, ..., \mathrm{U}_{114}, f_2, \mathrm{U}_{113}, ..., \mathrm{U}_{105}, f_1, \mathrm{U}_{104}]$

$[F_{31}(0), F_{30}(0), ..., F_0(0)] = [\mathrm{U}_{154}, ..., \mathrm{U}_{150}, M_3, M_1, M_0, H_1, H_0, \mathrm{U}_{149}, ..., \mathrm{U}_{138}, Y_4, Y_2, Y_0, W_2, W_0, \mathrm{U}_{137}, ..., \mathrm{U}_{133}]$

$PN1(clk) = (C_0(clk)\ \&\ b_0)\ \hat{}\ (C_1(clk)\ \&\ b_1)\ \hat{}\ ...\ \hat{}\ (C_{31}(clk)\ \&\ b_{31})$

$$PN2(clk) = (F_0(clk) \& e_0) \wedge (F_1(clk) \& e_1) \wedge \ldots \wedge (F_{31}(clk) \& e_{31})$$

$$PN(clk) = PN1(clk) \wedge PN2(clk) \quad (26)$$

where

$C_x(clk)$	is a *x*-th bit of upper shift register *C* at *clk*;
$F_x(clk)$	is a *x*-th bit of lower shift register *F* at *clk*;
U_x	is a *x*-th bit of a security key provided by the upper layer;
f_x	is a *x*-th bit of frame number, the number of *f* bits is 6;
s_x	is a *x*-th bit of slot number, the number of *s* bits is 9;
M_x	is a *x*-th bit of minute, the number of *M* bits is 6;
H_x	is a *x*-th bit of hour for 24-hour clock, the number of *H* bits is 5 and *H* has the value from 0 to 23;
Y_x	is a *x*-th bit of date, with a value from 1 to 31, the number of *Y* bits is 5;
W_x	is a *x*-th bit of month, with a value from 1 to 12, the number of *W* bits is 4;
$PN(clk)$	is a pseudo random bit of PN code generator at *clk*, and *PN*(0) is the *PN* value when the initial value is loaded into the two PN code generation registers.

The first bit of the data field is scrambled with $PN(O_{offset})$. Subsequent data bits are also scrambled sequentially where O_{offset} is $[U_{159}, U_{158}, \ldots, U_{155}]$.

The padding block in data field is not scrambled.

6.10 Interface with upper layer

6.10.1 General

The interface with the upper layer consists of '10' and the interface header and the following parameters as shown in Figure 16. The number of bits of the header is 8 bits, and the number of parameters and the number of bits of each parameter are different for each interface header.

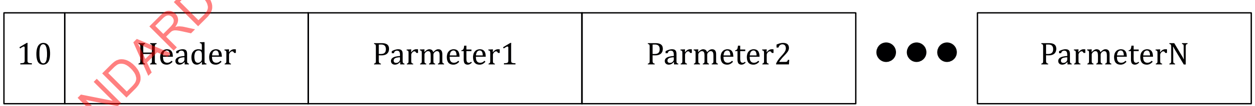


Figure 16 — Interface packet structure with upper layer

The upper layer can transmit the interface packet to the DLL, and the DLL can transmit the interface packet to the upper layer. In this case, sending ACK/NACK in response to the interface is optional and implementation dependent.

6.10.2 Initialization interface

6.10.2.1 General

The initialization interface is used when the unit is initialized. The upper layer can use this interface after initialization if necessary. Table 19 shows an initialization interface list.

Table 19 — Initialization interface list

Interface name	Header value	Direction
UPtoDL.InfoPowerParamVCH	0x01	to DL
UPtoDL.InfoPowerParamVCHsub	0x02	to DL
UPtoDL.InfoMapOption	0x06	to DL
UPtoDL.InfoApprovedSubchMap	0x07	to DL
UPtoDL.InfoICConstant	0x08	to DL
UPtoDL.InfoTimeParam	0x09	to DL
UPtoDL.InfoPacketParam	0x0A	to DL
UPtoDL.InfoVideoChannel	0x0D	to DL
UPtoDL.InfoSecurity	0x0E	to DL

6.10.2.2 UPtoDL.InfoPowerParamVCH

UPtoDL.InfoPowerParamVCH has two parameters. The meaning of each parameter is shown in Table 20.

Table 20 — Parameters of UPtoDL.InfoPowerParamVCH

Parameter name	bits	Description
PmaxVCH	7	Max VCH tx power with the range of -40dBm(0) ~ 40dBm(80)
PminVCH	7	Min VCH tx power with the range of -40dBm(0) ~ 40dBm(80)
PtargetVCH	5	PtargetVCH is the target reception power of the VCH. -110dBm(0) ~ 79dBm(31)
PmarginVCH	4	Video channel power margin, 0 dB(0)~15 dB(15)
PmaxTCH	7	Max TCH transmission power, -43dBm(0) ~ 37dBm(80)
PmarginTCH	4	Power margin of tone channel mapped with the video channel, 0 dB(0)~15 dB(15)
PTX_VCHTCH_differ	4	Difference between subchannel transmission power and tone channel transmission power, -15 dB(0)~0 dB(15)
PTX_VCHCCH_differ	4	Difference between VSCH transmission power and CSCH transmission power, 0 dB(0)~15 dB(15)
SNRrequiredVCH	4	Required SNR of video channel, 0 dB(0)~15 dB(15)
PRXtoneCompeteThre	5	The detection threshold of the tone subslot performing competition, -115dBm(0) ~ -84dBm(31)
PRXcollsiontoneThre0	5	The reception power threshold of the collision tone for determining the subchannel return, -115dBm(0) ~ -84dBm(31)
PRXcollsiontoneThre1	5	The reception power threshold of the collision tone for determining the subchannel return, -115dBm(0) ~ -84dBm(31)
Pmax_dmap0	6	Power parameter to determine the maximum power in the occupation state, -20dBm(0) ~ 40dBm(60)

Table 20 *(continued)*

Parameter name	bits	Description
Pmax_dmap1	6	Power parameter to determine the maximum power in the occupation state, -20dBm(0) ~ 40dBm(60)
Pmax_dmap2	6	Power parameter to determine the maximum power in the occupation state, -20dBm(0) ~ 40dBm(60)
d_map0	6	Distance parameter used for subchannel management, d_map0 × 100m
d_map1	6	Distance parameter used for subchannel management, d_map1 × 100m
PImin	5	Maximum allowable value of the amount of interference due to neighboring channels for decision to return subchannel resources, -140dBm(0) ~ -109dBm(31)
PImargin	4	Interference margin value due to neighboring channels for decision to return subchannel resources, 0dB(0) ~ 15 dB(15)
PIrealloc0	5	Interference power threshold used when deciding to return subchannel resources based on SC reception information, -140dBm(0) ~ -109dBm(31)
PIrealloc1	5	Interference power threshold used when deciding to return subchannel resources based on SC reception information and future prediction, -140dBm(0) ~ -109dBm(31)
PTH_TONE	5	Receiving power threshold that determines whether the subchannel is occupied, -120dBm(0) ~ -89dBm(31)
PTH_SMI0	5	Interference power threshold used when units generate a subchannel map, -140dBm(0) ~ -109dBm(31)
PTH_SMI1	5	Interference power threshold used when units generate a subchannel map, -140dBm(0) ~ -109dBm(31)
PTH_SMI2	5	Interference power threshold used when units generate a subchannel map, -140dBm(0) ~ -109dBm(31)
PTXmap0	6	Power parameter used when generating a subchannel map from occupied subchannel information received by the SC, -40dBm(0) ~ 23dBm(63)
PTXmap1	5	Power parameter used when generating a subchannel map from occupied subchannel information received by the SC, -40dBm(0) ~ 23dBm(63)
PTXmap2	5	Power parameter used when generating a subchannel map from occupied subchannel information received by the SC, -40dBm(0) ~ 23dBm(63)
PTXmap3	5	Power parameter used when generating a subchannel map from occupied subchannel information received by the SC, -40dBm(0) ~ 23dBm(63)

6.10.2.3 UPtoDL.InfoPowerParamVCHsub

This interface is used to impose special power requirements on specified subchannels. UPtoDL.InfoPowerParamVCHsub has 4 parameters. The meaning of each parameter is shown in Table 21.

Table 21 — Parameters of UPtoDL.InfoPowerParamVCHsub

Parameter name	bits	Description
ChannelNum	6	Frequency channel number of a video channel
SubchannelNum	4	Subchannel number of the ChannelNum
MaxPwr	7	Max transmit power with the range of -40dBm(0) ~ 40dBm(80)
MinPwr	7	Min transmit power with the range of -40dBm(0) ~ 40dBm(80)

6.10.2.4 UPtoDL.InfoMapOption

UPtoDL.InfoMapOption has 3 parameters. The meaning of each parameter is shown in Table 22.

Table 22 — Parameters of UPtoDL.InfoMapOption

Parameter name	bits	Description
SubChannelAbility10	1	If this value is '1', units can allocate a subchannel having a value of '10' in the subchannel map.
SubChannelAbility01	1	If this value is '1', units can allocate a subchannel having a value of '01' in the subchannel map.
SCmake00	2	'0' – Subchannels related with SC, CC can be usable. '1' – Subchannels related with SC cannot be usable. '2' – Subchannels related with CC cannot be usable. '3' – Subchannels related with SC or CC cannot be usable.

6.10.2.5 UPtoDL.InfoApprovedSubchMap

UPtoDL.InfoApprovedSubchMap has 2 or more parameters. The meaning of each parameter is shown in Table 23.

Table 23 — Parameters of UPtoDL.InfoApprovedSubchMap

Parameter name	bits	Description
N	9	Subchannel map length, this is (TheNumOfVCH of UPtoDL.InfoVideoChannel) × 25.
EN_x	1 × N	Units shall allocate subchannel among enabled subchannels. If this value is '1', x-th subchannel is usable. x = 25 × channel number + subchannel number,

6.10.2.6 UPtoDL.InfoICConstant

UPtoDL.InfoICConstant has (N+1) parameters. The meaning of each parameter is shown in Table 24.

Table 24 — Parameters of UPtoDL.InfoICConstant

Parameter name	bits	Description
N	5	Number of IC parameters
IC_x	7 × N	x-th IC, 1≤x≤N, 30 dB(0)~157 dB(127)

6.10.2.7 UPtoDL.InfoTimeParam

UPtoDL.InfoTimeParam has 1 parameter. The meaning of each parameter is shown in Table 25.

Table 25 — Parameters of UPtoDL.InfoTimeParam

Parameter name	bits	Description
TVCHReturn	4	When subchannel reallocation is performed, the duration of the previously allocated subchannel, 0 second(0)~15 seconds(15)
TimeOf1DTry	4	When subchannel reallocation is decided, the maximum duration of the exchange of PB0x1D, 0 second(0)~15 seconds(15)
TimeOf1CTry	4	Duration for the controller to retransmit PB 0x1C to the UA until a response to PB 0x1C is received. 0 second(0)~15 seconds(15)

6.10.2.8 UPtoDL.InfoPacketParam

UPtoDL.InfoPacketParam has 2 parameters. The meaning of each parameter is shown in Table 26.

Table 26 — Parameters of UPtoDL.InfoPacketParam

Parameter name	bits	Description
MaxDataLen	16	Maximum length of data field
SrcAddrLen	6	The length of source address
SrcAddr	40	The unit's own address, Only the lower bits corresponding to the length of SrcAddrLen are meaningful.
DstAddr	40	The counterpart address, Only the lower bits corresponding to the length of SrcAddrLen are meaningful. In the case of a UA, this is the address of the controller, and in the case of a controller, this is the address of the UA. '0' means no counterpart.
UAAckReq	1	'0' – UA does not transmit ACKs of occupied VSCH slots. '1' – UA transmit ACKs of occupied VSCH slots.

6.10.2.9 UPtoDL.InfoVideoChannel

UPtoDL.InfoVideoChannel has 7 parameters. The meaning of each parameter is shown in Table 27.

Table 27 — Parameters of UPtoDL.InfoVideoChannel

Parameter name	bits	Description
TheNumOfVCH	6	The number of VCHs
TCHFreq	24	The centre frequency of TCH, TCHFreqkHz
FirstCenterFreq	24	The centre frequency of VCH^0, FirstCenterFreq kHz
IntervalOfVCH	12	The frequency interval between two VCHs, IntervalOfVCH kHz
LinkConfirmError	6	Packet error rate for link confirm during 1 frame error rate = LinkConfirmErrorx0.01

Table 27 *(continued)*

Parameter name	bits	Description
ReallocMethod	2	‘0’ – SC ‘1’ – Dedicated slot of control channel ‘2’ – CC
InterfaceAckResponse	2	‘0’ – no response ‘1’ – DLL responses to Upper layer. ‘2’ – Upper layer responses to DLL. ‘3’ – DLL & Upper layer response to each other.

6.10.2.10 UPtoDL.InfoSecurity

UPtoDL.InfoSecurity has 3 parameters. The meaning of each parameter is shown in Table 28.

Table 28 — UPtoDL.InfoSecurity parameter

Parameter name	bits	Description
TrustOffset	5	Input offset used for calculating trust field
K	102	Secret key between the trust check system and the unit
U	160	Secret key for data field scrambling

6.10.3 Dynamic interface

6.10.3.1 General

The dynamic interface is used after unit initialization. The unit shall receive all slots except the slot for performing the transmission. When data link transfers received packets to upper layer, DLtoUP.RsvVCHData is used. Other interface packets are used when a corresponding event occurs. Table 29 shows a dynamic interface list.

Table 29 — Dynamic interface list

Interface name	Header value	Direction
UPtoDL.ReqAllocatingDedicatedVCH	0x20	to DL
UPtoDL.ReqUsingDedicatedVCH	0x21	to DL
UPtoDL.PBReqGetVSCH	0x22	to DL
DLtoUP.NotiGetVSCH	0x23	to UP
DLtoUP.NotiVCHStatus	0x24	to UP
UPtoDL.ReqTxVCH	0x25	to DL
DLtoUP.RsvVCHData	0x26	to UP
UPtoDL.ReqReturnVCH	0x27	to DL
UPtoDL.ResponseACK	0x7E	to DL
DLtoUP.ResponseACK	0x7F	to UP

6.10.3.2 UPtoDL.ReqAllocatingDedicatedVCH

When receiving this interface, the DLL competes in the tone subslot set mapped with the corresponding subchannel, and then uses the corresponding channel when the competition wins. The DLL shall perform the functions of slot clearing, collision tone transmission, and collision tone detection in the occupation stage of the corresponding channel. UPtoDL.ReqAllocatingDedicatedVCH has 2 parameters. The meaning of each parameter is shown in Table 30.

Table 30 — Parameters of UPtoDL.ReqAllocatingDedicatedVCH

Parameter name	bits	Description
ChannelNum	6	Frequency channel number of a video channel
SubchannelNum	4	Subchannel number of the ChannelNum

6.10.3.3 UPtoDL.ReqUsingDedicatedVCH

A DLL that has received this interface can use the subchannel directly. DLL does not perform slot clearing, collision tone transmission, and collision tone detection in the subchannel occupation stage. UPtoDL.ReqUsingDedicatedVCH has 2 parameters. The meaning of each parameter is shown in Table 31.

Table 31 — Parameters of UPtoDL.ReqUsingDedicatedVCH

Parameter name	bits	Description
ChannelNum	6	Frequency channel number of a video channel
SubchannelNum	4	Subchannel number of the ChannelNum

6.10.3.4 UPtoDL.PBReqGetVSCH

UPtoDL.PBReqGetVSCH has 5 parameters. The meaning of each parameter is shown in Table 32.

Table 32 — Parameters of UPtoDL.PBReqGetVSCH

Parameter name	bits	Description
NegoMethod	2	’0‘ – SC ’1‘ – DS ‘3’ – CC
UsePB0x81	1	This value is used only when NegoMethod is ‘0’. ‘1’ means PB 0x81 use, ‘0’ means PB 0x81 not use.
RequestMethod	1	This value is used only when UsePB0x81 is ‘1’. ‘1’ – SC Bslot & Tslot use. ‘0’ – SC Bslot use.
m	3	-4(0) ~ 3(7)
n	3	-4(0) ~ 3(7)

6.10.3.5 DLtoUP.NotiGetVSCH

DLtoUP.NotiGetVSCH has 6 parameters. The meaning of each parameter is shown in Table 33.

Table 33 — Parameters of DLtoUP.NotiGetVSCH

Parameter name	bits	Description
Status	2	‘0’ – Subchannel allocation failure ‘1’ – Resource allocation failure ‘2’ – Link confirmation failure ‘3’ – Link confirmation success
NegoMethod	2	This value is meaningful when Status is ’0’. ‘0‘ – SC ’1‘ – DS ‘3’ – CC

Table 33 *(continued)*

Parameter name	bits	Description
m	3	The value of m when negotiation is finished. This is meaningful when Status is '0'. -4(0) ~ 3(7)
n	3	The value of n when negotiation is finished. This is meaningful when Status is '0'. -4(0) ~ 3(7)
ChannelNum	6	This value is meaningful when Status is '3'. Frequency channel number of a successful video channel
SubchannelNum	4	This value is meaningful when Status is '3'. Subchannel number of the successful ChannelNum

6.10.3.6 DLtoUP.NotiVCHStatus

DLtoUP.NotiVCHStatus has 3 parameters. The meaning of each parameter is shown in Table 34.

Table 34 — Parameters of DLtoUP.NotiVCHStatus

Parameter name	bits	Description
ChannelNum	6	Frequency channel number of an using video channel
SubchannelNum	4	Subchannel number of an using ChannelNum
Status	2	'0' – Reallocation due to the occurrence of the same resource collision '1' – Reallocation due to exceeding threshold of interfering channel power '2' – Reallocation success '3' – Subchannel return
ChannelNumNew	6	This value is meaningful when Status is '2'. Frequency channel number of a reallocated video channel
SubchannelNumNew	4	This value is meaningful when Status is '2'. Subchannel number of the reallocated ChannelNumNew

6.10.3.7 UPtoDL.ReqTxVCH

UPtoDL.ReqTxVCH has 5 parameters. The meaning of each parameter is shown in Table 35.

Table 35 — Parameters of UPtoDL.ReqTxVCH

Parameter name	bits	Description
ChannelNum	6	Frequency channel number of a video channel
SubchannelNum	4	Subchannel number of the ChannelNum
SubChSlotNum	5	Slot number of the subchannel, '0'~'24' is slot number of the subchannel. '31' mean 'as fast as possible'.
DataSecurity	2	'00' – No security '01' – Data scrambling with PN code '10', '11' - reserved
DataLen	16	The length of Data

Table 35 *(continued)*

Parameter name	bits	Description
Data	DataLen	Transmission Data

6.10.3.8 DLtoUP.RsvVCHData

When the VC DLL receives a packet from the other unit, it forwards it to the upper layer. DLtoUP.RsvVCHData has 3 parameters. The meaning of each parameter is shown in Table 36.

Table 36 — Parameters of DLtoUP.RsvVCHData

Parameter name	bits	Description
SrcAddr	26 (variable)	Source address of a received packet, this can be extended by SrcAddrLen of UPtoDL.InfoPacketParam
SlotNum	9	The slot number of the received packet
DataLen	16	The length of Data
Data	DataLen	Received data

6.10.3.9 UPtoDL.ReqReturnVCH

UPtoDL.ReqReturnVCH has 2 parameters. The meaning of each parameter is shown in Table 37.

Table 37 — Parameters of UPtoDL.ReqReturnVCH

Parameter name	bits	Description
ChannelNum	6	Frequency channel number of a video channel
SubchannelNum	4	Subchannel number of the ChannelNum

6.10.3.10 UPtoDL.ResponseACK

When the DLL sends an interface packet to the upper layer, the upper layer can respond using UPtoDL.ResponseACK. UPtoDL.ResponseACK has 2 parameters. The meaning of each parameter is shown in Table 38.

Table 38 — Parameters of UPtoDL.ResponseACK

Parameter name	bits	Description
Ack	1	'1' is ACK, '0' is NACK.
InterfaceHeader	8	Interface header received from DLL

6.10.3.11 DLtoUP.ResponseACK

When the upper layer sends an interface packet to the DLL, the DLL can respond with DLtoUP.ResponseACK. DLtoUP.ResponseACK has 2 parameters. The meaning of each parameter is shown in Table 39.

Table 39 — Parameters of DLtoUP.ResponseACK

Parameter name	bits	Description
Ack	1	'1' is ACK, '0' is NACK.
InterfaceHeader	8	Interface header received from Upper layer

