
Juphoon Protocol Framework

Talk Burst Control Protocol

Published: Nov 2007

For more information on Juphoon Protocol Framework, see <http://www.juphoon.com>

Juphoon TBCP Function Definition

Juphoon System Software Corporation.

<http://www.juphoon.com>

Tel: +86-574-87287820

Fax: +86-574-87304379

Text Part Number:

Copyright © 2007, Juphoon System Software Corporation.

All rights reserved.

Contents

1. INTRODUCTION.....	3
1.1 PURPOSE.....	3
1.2 AUDIENCE	3
1.3 SCOPE.....	3
1.4 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS	3
2. SYSTEM ENVIRONMENT.....	4
2.1 TBCP FRAMEWORK	4
2.2 BASIC DATA TYPES	4
3. SUPPORTED TBCP MESSAGES	5
4. USER INTERFACES	6
4.1 INTERFACE STRUCTURES	6
4.1.1 ZOS Structures.....	6
4.1.2 TBCP Structures	6
4.2 USEFUL INTERFACES	7
4.2.1 Tbcpl_SendTBReq.....	7
4.2.2 Tbcpl_SendTBGrtd	8
4.2.3 Tbcpl_SendTBDeny	9
4.2.4 Tbcpl_SendTBRls.....	9
4.2.5 Tbcpl_SendTBIidle.....	10
4.2.6 Tbcpl_SendTBTaken	10
4.2.7 Tbcpl_SendTBRvk.....	11
4.2.8 Tbcpl_SendTBAck.....	12
4.2.9 Tbcpl_SendTBRqsReq.....	13
4.2.10 Tbcpl_SendTBRqsRsp.....	13
4.2.11 Tbcpl_SendConn.....	14
4.2.12 Tbcpl_SendDisc.....	15
4.2.13 Tbcpl_DecodeMsg.....	15
4.3 UTILITY INTERFACES	16
4.3.1 Tbcpl_MsgDescFromType.....	16
4.4 VERSION INTERFACES.....	17
4.4.1 Tbcpl_GetVersion	17

List of Tables

Table 2-1 Basic Data Types	4
----------------------------------	---

List of Figures

Figure 2-1 TBCP Framework.....	4
--------------------------------	---

1. Introduction

A Talk Burst is the flow of media from a PoC Client while that has the permission to send media.

Talk Burst Control is a control mechanism that arbitrates requests from the PoC Clients, for the right to send media.

Talk Burst Control Protocol (TBCP) is a protocol for performing Talk Burst Control.

1.1 Purpose

This document is provided to developers who will develop their own products with the functions of TBCP furnished by Juphoon.

1.2 Audience

The readers of this document are assumed to have a working knowledge of Talk Burst Control and PoC.

1.3 Scope

This document provides several functions of TBCP and their usages but not the description about the design and the realization of it.

1.4 Definitions, Acronyms, and Abbreviations

The following definitions, acronyms, and abbreviations are used in this document:

Abbreviation	Description
CNAME	Canonical name
NAME	User Name SDES Item
PoC	Push To Talk over Cellular
RTCP	Real-Time Transport Control Protocol
RTP	Real-Time Transport Protocol
SDES	Source Description RTCP Packet
SSRC	Synchronization source
TBCP	Talk Burst Control Protocol
ZOS	Zero Operating System

2. System Environment

2.1 TBCP Framework

The picture below illustrates the environment in which Juphoon TBCP operates.

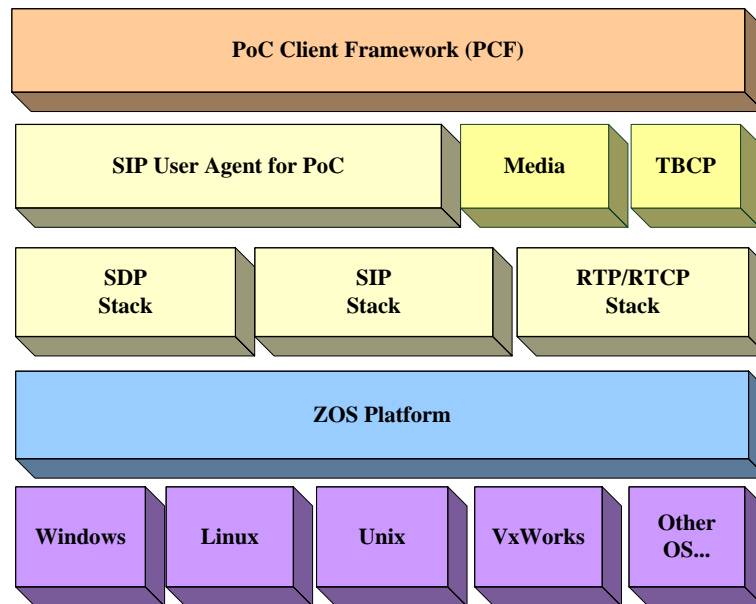


Figure 2-1 TBCP Framework

2.2 Basic Data Types

There are some basic data types provided by ZOS platform listed below.

Name	Type
ZDOUBLE	double
ZFLOAT	float
ZLONG	long
ZINT	int
ZSHORT	short
ZCHAR	char
ZULONG	unsigned long
ZUINT	unsigned int
ZSIZE_T	unsigned int
ZUSHORT	unsigned short
ZUCHAR	unsigned char
ZBOOL	int
ZVOID	void

Table 2-1 Basic Data Types

3. Supported TBCP Messages

Juphoon TBCP supports the following TBCP messages:

- TBCP Talk Burst Request. This is used by the PoC Client to request permission from the PoC Server to send a Talk Burst.
- TBCP Talk Burst Granted. This is used by the PoC Server to notify the PoC Client that it has been granted permission to send a Talk Burst.
- TBCP Talk Burst Deny. This is used by the PoC Server to notify a PoC Client that it has been denied permission to send a Talk Burst.
- TBCP Talk Burst Release. This is used by the PoC Client to notify the PoC Server that it has completed sending the Talk Burst.
- TBCP Talk Burst Idle. This is used by the PoC Server to notify all PoC Clients that no one has the permission to send a Talk Burst at the moment and that it may accept the TBCP Talk Burst Request message.
- TBCP Talk Burst Taken. This is used by the PoC Server to notify all PoC Clients, except the PoC Client that has been given permission to send a Talk Burst that another PoC Client has been given permission to send a Talk Burst.
- TBCP Talk Burst Revoke. This is used by the PoC Server to revoke the media resource from a PoC Client and can be used for preemption functionality, but is also be used by the system to prevent overly long use of the media resource.
- TBCP Talk Burst Acknowledgement. This is used by the PoC Client, when Acknowledgement is required in the received TBCP message.
- TBCP Disconnect. This is used by the PoC Server to close the PoC Session using a Pre-established Session while maintaining the Pre-established Session.
- TBCP Connect. This is used by the PoC Server to notify all PoC Clients using Pre-established Session, that PoC session is connected.
- TBCP Talk Burst Request Queue Status Request. This is used by the PoC Client to request the current queue position of a queued TBCP Talk Burst Request message.
- TBCP Talk Burst Request Queue Status Response. This is used by the PoC Server to notify the PoC Client that the TBCP Talk Burst Request has been queued and is used to respond to a Talk Burst Queue status Request message from the PoC Client.

4. User Interfaces

4.1 Interface Structures

4.1.1 ZOS Structures

4.1.1.1 *ST_ZOS_SSTR*

```
typedef struct tagZOS_SSTR
{
    ZCHAR *pcStr;           /* string pointer */
    ZUSHORT wLen;          /* string length */
    ZUCHAR aucSpare[2];    /* for 32 bit alignment */
} ST_ZOS_SSTR;
```

4.1.2 TBCP Structures

4.1.2.1 *ST_TBCP_SDES_ITEM*

```
typedef struct tagTBCP_SDES_ITEM
{
    ZUCHAR ucType;         /* item type EN_TBCP_ITEM_TYPE */
    ZUCHAR aucSpare[3];    /* for 32 bit alignment */
    ST_ZOS_SSTR stStr;     /* item string */
} ST_TBCP_SDES_ITEM;
```

4.1.2.2 ST_TBCP_MSG

```

typedef struct tagTBCP_MSG
{
    ZUCHAR ucPres;                /* present flag */
    ZUCHAR ucType;                /* message type EN_TBCP_MSG_TYPE */
    ZUCHAR aucSpare[2];          /* for 32 bit alignment */
    union
    {
        ST_TBCP_TB_REQ stTbReq;   /* talk burst request */
        ST_TBCP_TB_GRTED stTbGrted; /* talk burst granted */
        ST_TBCP_TB_DENY stTbDeny; /* talk burst deny */
        ST_TBCP_TB_RLS stTbRls;   /* talk burst release */
        ST_TBCP_TB_TAKEN stTbTaken; /* talk burst taken */
        ST_TBCP_TB_RVK stTbRvk;   /* talk burst revoke */
        ST_TBCP_TB_ACK stTbAck;   /* talk burst acknowledgement */
        ST_TBCP_TB_RQS_RSP stTbRqsRsp; /* talk burst request queue status rsp */
        ST_TBCP_CONN stConn;      /* connect */
    } u;
    ST_ZOS_DBUF *pstMemBuf;       /* memory buffer */
} ST_TBCP_MSG;

```

4.2 Useful Interfaces

These interfaces are included in `tbcp_ui.h`

4.2.1 TbcP_SendTBReq

The PoC Client sends a TBCP Talk Burst Request to the PoC Server.

```

ZINT TbcP_SendTBReq(ZULONG dwRtpId, ZUSHORT wPriority,
                    ZUCHAR *pucTimeStamp);

```

[Parameters]

Input parameters:

ZULONG dwRtpId
The PoC session ID.

ZUSHORT wPriority
The priority of the TBCP Talk Burst Request.

ZUCHAR *pucTimeStamp
The time stamp.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.2 TbcP_SendTBGrtd

The PoC Server sends a TBCP Talk Burst Granted message to notify the PoC Client that it has been granted permission to send a Talk Burst.

```
ZINT TbcP_SendTBGrtd(ZULONG dwRtpId, ZUSHORT wT2Tmr, ZULONG dwPCount);
```

[Parameters]**Input parameters:**

ZULONG dwRtpId
The PoC session ID.

ZUSHORT wT2Tmr
The T2 timer - the stop talking timer.

ZULONG dwPCount
The amount of participants in this PoC Session.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.3 TbcP_SendTBDeny

The PoC server sends a TBCP Talk Burst Deny message to a PoC Client to notify it that it has been denied permission to send a Talk Burst.

```
ZINT TbcP_SendTBDeny(ZULONG dwRtpId, ZUCHAR ucRsnCode);
```

[Parameters]

Input parameters:

ZULONG dwRtpId

The PoC session ID.

ZUCHAR ucRsnCode

The reason for the denial. The reason can be EN_TBCP_RVK_RSN_ONLY_ONE, EN_TBCP_RVK_RSN_TOO_LONG, EN_TBCP_RVK_RSN_NO_PERM, or EN_TBCP_RVK_RSN_PRE_EMPTED.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.4 TbcP_SendTBRls

The PoC Client sends a TBCP Talk Burst Release message to notify the PoC Server that it has completed sending the Talk Burst.

```
ZINT TbcP_SendTBRls(ZULONG dwRtpId, ZUSHORT wLastSeq);
```

[Parameters]

Input parameters:

ZULONG dwRtpId

The PoC session ID.

ZUSHORT wLastSeq

The sequence number of the last RTP Media packet.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.5 TbcP_SendTBIdle

The PoC Server sends a TBCP Talk Burst Idle message to notify all the PoC Clients that no one has the permission to send a Talk Burst at the moment and that it may accept the TBCP Talk Burst Request message.

```
ZINT TbcP_SendTBIdle(ZULONG dwRtpId);
```

[Parameters]

Input parameters:

ZULONG dwRtpId
The PoC session ID.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.6 TbcP_SendTBTaken

The PoC Server sends a TBCP Talk Burst Taken message to notify all PoC Clients, except the PoC Client that has been granted the permission to send a Talk Burst that another PoC Client has been given permission to send a Talk Burst.

```
ZINT TbcP_SendTBTaken(ZULONG dwRtpId, ZULONG dwSsrc,  
                      ST\_ZOS\_SSTR *pstCName, ST\_ZOS\_SSTR *pstName, ZULONG dwPCount,  
                      ZUCHAR ucNeedAck);
```

[Parameters]

Input parameters:

ZULONG dwRtpId

The PoC session ID.

ZULONG dwSsrc

The SSRC (Synchronization source).

[ST_ZOS_SSTR](#) *pstCName

The SDES CNAME (Canonical name).

[ST_ZOS_SSTR](#) *pstName

The SDES NAME (User Name SDES Item).

ZULONG dwPCount

The amount of participants in this PoC Session.

ZUCHAR ucNeedAck

Indicates whether or not the PoC Server needs the PoC Clients which will receive the TBCP Talk Burst Taken message to send a TBCP Talk Burst Acknowledgement message.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.7 TbcP_SendTBRvk

The PoC Server sends a TBCP Talk Burst Revoke message to revoke the media resource from a PoC Client. This interface can also be used by the system to send a TBCP Talk Burst Revoke to prevent overly long use of the media resource.

```
ZINT TbcP_SendTBRvk(ZULONG dwRtpId, ZUSHORT wRsnCode, ZUSHORT wAddInfo);
```

[Parameters]

Input parameters:

ZULONG dwRtpId
The PoC session ID.

ZUSHORT wRsnCode
The reason for the revoking. The reason can be EN_TBCP_RVK_RSN_ONLY_ONE (only one PoC User), EN_TBCP_RVK_RSN_TOO_LONG (Talk Burst too long), EN_TBCP_RVK_RSN_NO_PERM(no permission to send a Talk Burst), or EN_TBCP_RVK_RSN_PRE_EMPTED (Talk Burst preempted).

ZUSHORT wAddInfo
The additional information. It is needed when the revoking reason is EN_TBCP_RVK_RSN_TOO_LONG.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.8 TbcP_SendTBBack

The PoC Client sends a TBCP Talk Burst Acknowledgement message when acknowledgement is required in the received message.

```
ZINT TbcP_SendTBBack(ZULONG dwRtpId, ZUCHAR ucSubType, ZUSHORT wRsnCode);
```

[Parameters]**Input parameters:**

ZULONG dwRtpId
The PoC session ID.

ZUCHAR ucSubType
The subtype. The bit pattern in the subtype field shall be used for the TBCP Talk Burst Acknowledgement message is 00111.

ZUSHORT wRsnCode
The Talk Burst acknowledgement message reason type. The reason type can be EN_TBCP_ACK_RSN_ACCEPTED (accepted), EN_TBCP_ACK_RSN_BUSY (busy), EN_TBCP_ACK_RSN_NOT_ACCEPTED (not accepted).

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.9 TbcP_SendTBRqsReq

The PoC Client sends a TBCP Talk Burst Request Queue Status Request message to request the current queue position of a queued TBCP Talk Burst Request message.

```
ZINT TbcP_SendTBRqsReq(ZULONG dwRtpId);
```

[Parameters]

Input parameters:

ZULONG dwRtpId
The PoC session ID.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.10 TbcP_SendTBRqsRsp

The PoC Server sends a TBCP Talk Burst Request Queue Status Response message to notify the PoC Client that the TBCP Talk Burst Request has been queued and to respond to a Talk Burst Queue Status Request message from the PoC Client.

```
ZINT TbcP_SendTBRqsRsp(ZULONG dwRtpId, ZUCHAR ucPriority,  
                        ZUSHORT wQPos);
```

[Parameters]

Input parameters:

ZULONG dwRtpId
The PoC session ID.

ZUCHAR ucPriority
The priority of the Talk Burst Request.

ZUSHORT wQPos
The queue position of the TBCP Talk Burst Request.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.11 TbcP_SendConn

The PoC Server sends a TBCP Connect message to notify all PoC Clients using Pre-established session that PoC Session is connected.

```
ZINT TbcP_SendConn(ZULONG dwRtpId, ZUCHAR ucSesType, ZUCHAR ucAddInd,  
ST\_TBCP\_SDES\_ITEM *pstItem, ZULONG dwItemNum);
```

[Parameters]**Input parameters:**

ZULONG dwRtpId
The PoC session ID.

ZUCHAR ucSesType
The session type.

ZUCHAR ucAddInd
The additional indication.

[ST_TBCP_SDES_ITEM](#) *pstItem
The SDES item.

ZULONG dwItemNum
The maximum number of items that can be added into the message.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.12 TbcP_SendDisc

The PoC Server sends a TBCP Disconnect message to close the PoC Session using a Pre-established Session while maintaining the Pre-established Session.

```
ZINT TbcP_SendDisc(ZULONG dwRtpId);
```

[Parameters]

Input parameters:

ZULONG dwRtpId
The PoC session ID.

Output parameters:

None.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.2.13 TbcP_DecodeMsg

Decodes some data into a TBCP message.

```
ZINT TbcP_DecodeMsg(ZUCHAR ucSubtype, ZUCHAR *pucData, ZULONG dwLen,  
ST\_TBCP\_MSG **ppstMsg);
```

[Parameters]

Input parameters:

ZUCHAR ucSubtype

Type of the TBCP message. The type can be EN_TBCP_MSG_TB_REQ (Talk Burst Request), EN_TBCP_MSG_TB_GRTED (Talk Burst Granted), EN_TBCP_MSG_TB_TAKEN (Talk Burst Taken), EN_TBCP_MSG_TB_DENY (Talk Burst Deny), EN_TBCP_MSG_TB_RLS (Talk Burst Release), EN_TBCP_MSG_TB_IDLE (Talk Burst Idle), EN_TBCP_MSG_TB_RVK (Talk Burst Revoke), EN_TBCP_MSG_TB_ACK (Talk Burst Acknowledgement), EN_TBCP_MSG_TB_RQS_REQ (Talk Burst Request Queue Status Request), EN_TBCP_MSG_TB_RQS_RSP (Talk Burst Request Queue Status Response), EN_TBCP_MSG_DISC (TBCP Disconnect), or EN_TBCP_MSG_CONN (TBCP Connect).

ZUCHAR *pucData

The data to decode.

ZULONG dwLen

The maximum length of the message.

Output parameters:

[ST_TBCP_MSG](#) **ppstMsg

The decoded TBCP message.

[Return value]

Returns ZOK on success, or ZFAILED on failure.

4.3 Utility Interfaces

These interfaces are included in `tbcp_util.h`.

4.3.1 TbcP_MsgDescFromType

Uses the type of a message to get a short description of the message. For example if you input a message type like EN_TBCP_MSG_TB_REQ, a short description “TALK BURST REQUEST” will be returned.

```
ZUCHAR * TbcP_MsgDescFromType (ZUCHAR ucType);
```

[Parameters]

Input parameters:

ZUCHAR ucType

The message type.

Output parameters:

None.

[Return value]

Returns the description of the message type.

4.4 Version Interfaces

These interfaces are included in `tbcv_version.h`.

4.4.1 Tbcv_GetVersion

Gets the current the version of TBCP.

```
ZCHAR * Tbcv_GetVersion();
```

[Parameters]

Input parameters:

None.

Output parameters:

None.

[Return value]

Returns the version of the TBCP.