

NoraVR (Nora Virtual Repeater)

Specification

1. 概要	_____	P03
2. イメージブロック図	_____	P04
3. 動作フローイメージ	_____	P05
4. 通信プロトコル仕様	_____	P06
① 基本コマンドフレーム構成	_____	P06
② 共通事項	_____	P07
③ 通信動作シーケンス	_____	P09
1. ログインシーケンス(LOGINUSR)	_____	P09
2. ログインシーケンス(LGINUSR2)	_____	P10
3. ログアウトシーケンス(LOGOUT)	_____	P10
4. 接続維持シーケンス(PING/PONG)	_____	P11
5. 設定更新シーケンス(CONFSET)	_____	P11
6. リフレクタリンク状態の要求シーケンス(RLINKGET)	_____	P12
7. リフレクタリンク状態の通知シーケンス(RLINK)	_____	P12
8. アップリンク音声シーケンス(VTPCM/VTOPUS/VTAMBE)	_____	P13
9. ダウンリンク音声シーケンス(VTPCM/VTOPUS/VTAMBE)	_____	P13
10. ルーティングサービス名称の取得シーケンス(RSRVGET)	_____	P14
11. ルーティングサービス名称の更新通知シーケンス(RSRV)	_____	P14
12. レピータ情報の取得シーケンス(RINFOGET)	_____	P15
13. アクセスログの取得シーケンス(ACLOGGET)	_____	P16
14. アクセスログの更新通知シーケンス(ACLOG)	_____	P16
15. ユーザーリストの取得シーケンス(USLSTGET)	_____	P17
16. ユーザーリストの更新通知シーケンス(USLST)	_____	P17

4. 通信プロトコル仕様(続き)

④ コマンドタイプ別フィールド構成

1. ACK____	_____	P19
2. NAK____	_____	P19
3. LOGOUT__	_____	P19
4. LOGINUSR	_____	P20
5. LOGIN_CC	_____	P20
6. LOGIN_HS	_____	P20
7. LGINUSR2	_____	P21
8. LOGINACK	_____	P22
9. CONFSET_	_____	P23
10. PING____	_____	P23
11. PONG____	_____	P23
12. VTPCM____	_____	P24
13. VTOPUS__	_____	P25
14. VTAMBE__	_____	P26
15. RLINKGET	_____	P27
16. RLINK____	_____	P27
17. RSRVGET	_____	P27
18. RSRV____	_____	P27
19. RINFOGET	_____	P28
20. RINFO____	_____	P28
21. ACLOGGET	_____	P28
22. ACLOG____	_____	P29
23. USLSTGET	_____	P30
24. USLST____	_____	P31

5. 付録	_____	P33
-------	-------	-----

6. 改版履歴	_____	P35
---------	-------	-----

7. 関連情報(引用元)	_____	P35
--------------	-------	-----

8. 連絡先	_____	P35
--------	-------	-----

NoraVR(Nora Virtual Repeater)仕様

1. 概要

一般的なDSTARレピータが空中線(RF)で構成されているのに対して、公衆回線網上に「仮想的な」レピータシステム(VirtualRepeater)を形成する機能を提供する。リフレクターとしても機能し、加えてコールサインルーティングにも対応する。この仕様書は、ユーザー側でクライアントサイドを開発・設計する為のNoraVirtualRepeater(以下NoraVR)システム(サーバー・クライアント)の詳細仕様を示すものである。

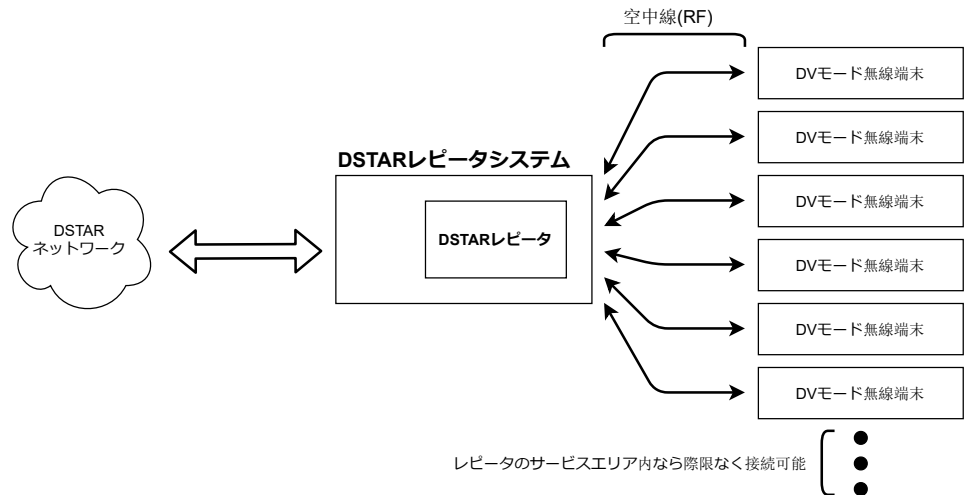
【機能・特徴】

- 仮想的なレピータシステム(=リフレクター同等)としての機能
- TO(コールサインルーティング)を使用したゲート越え機能
- ログイン時にパスワードを設定可能
- 低ビットレートに対応し低速回線でも音声破綻しにくい
- ユーザー側が音声コーデックを選択可能(帯域高 PCM<->Opus 64k<->Opus 24k<->Opus 8k 帯域低)
- AMBEパススルーに対応(サーバー側のAMBEエンコード・デコードをパススルー)
- パケット通信に対応(DSTAR仕様のファストデータ・スローデータ使用)
- ユーザーがクライアント(端末)側を自由に開発・設計可能+通信ライブラリを公開
- エコーバック機能(送信した音声そのままオウム返しされる)
- ログイン時にRF出力機能を有するかサーバーに宣言する事でRFが出力される端末を選別可能

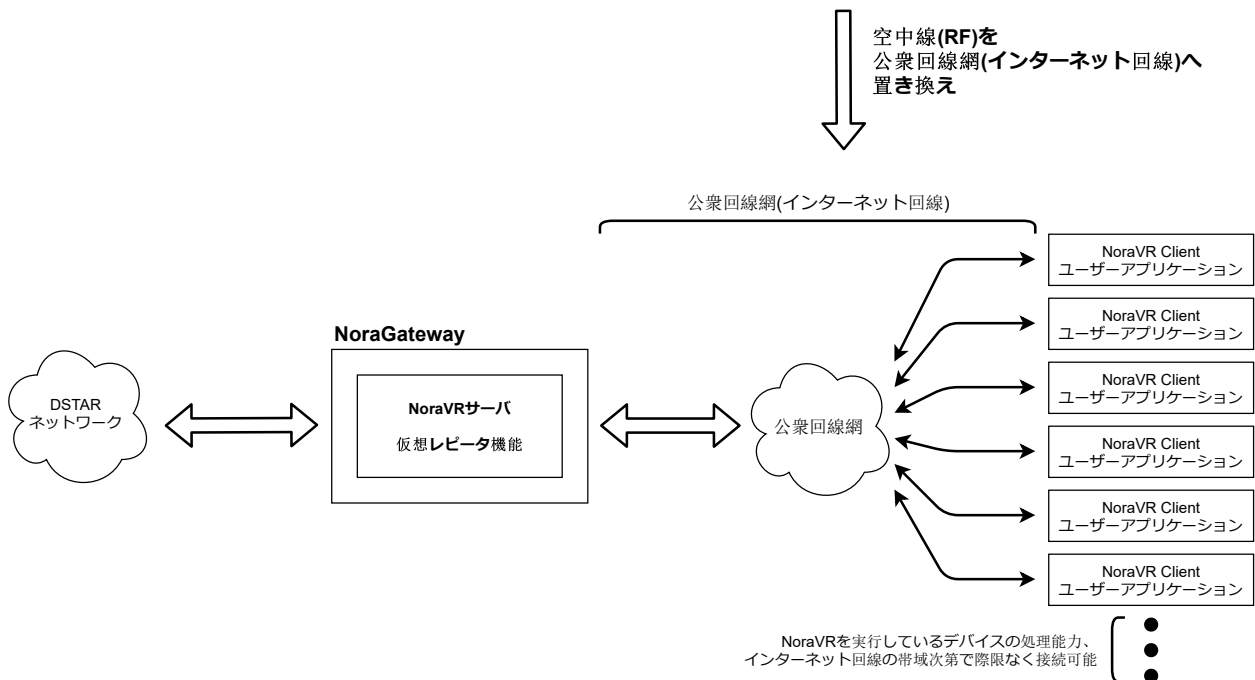
ユーザー側から見て一般的なレピータシステムと同等の機能を備えており、加えてNoraGatewayで提供されている以下の機能を含め全て使用可能

- 接続ユーザー同士の山掛け交信
- リフレクターへの接続(DExtra/DCS/DPlus/JARL MultiForward Link)
- ゲート越え交信(コールサイン指定・ゲート指定)
- 制御コマンドによるNoraGatewayの制御(_RLMACEなど)

サーバー側(NoraVR)はNoraGatewayに内蔵されており、DSTAR関連のアプリケーションを自作する際に障壁となるAMBEコーデックのエンコード・デコードはサーバー側で処理される。これにより、クライアント側でAMBEコーデックを扱う必要がない。また、DSTARネットワークへの接続はNoraGatewayが担当し、NoraVRクライアント側では、宛先を指定して音声(またはデータ)をサーバー側(NoraVR)へ送信するだけで指定の対象へ送信可能となり、比較的容易にユーザー側でアプリケーションを自由に開発・設計する事が可能となっている。



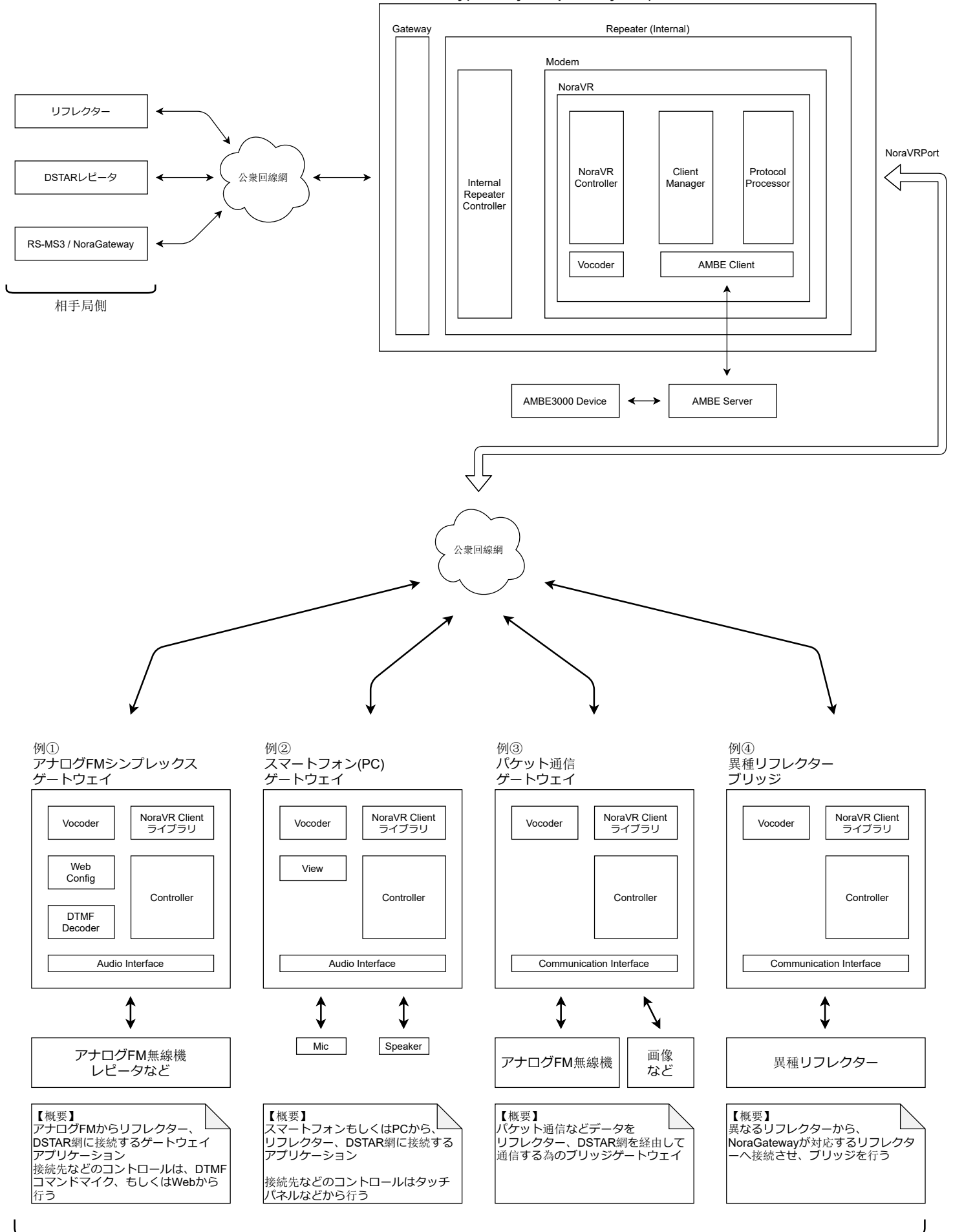
図① 既存のDSTARレピータシステム(抜粋)



図② NoraVR仮想レピータシステム

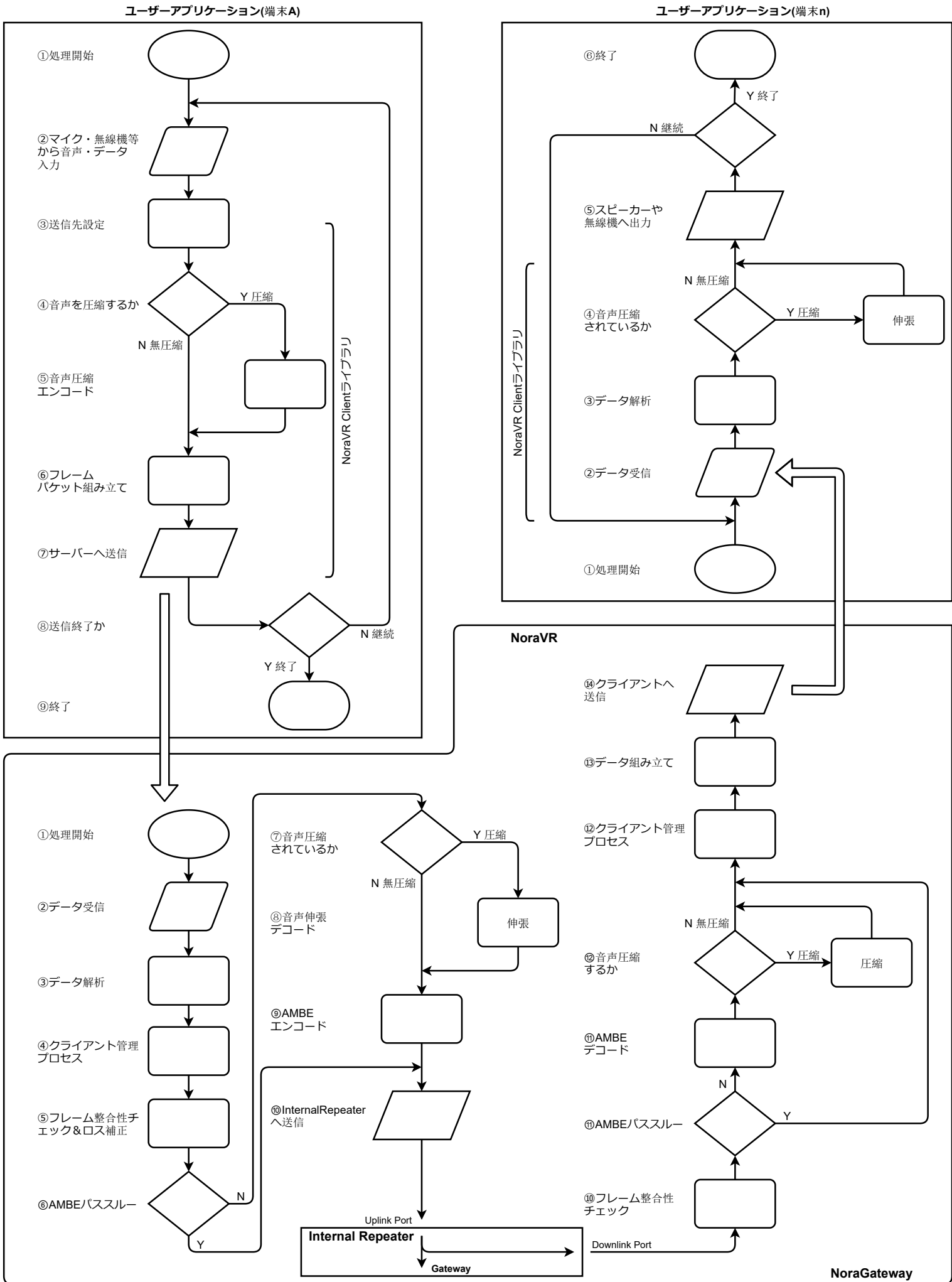
2. イメージブロック図

NoraGateway(Gateway & Repeater System)



ユーザーアプリケーション(自局側)
ユーザー側で自由に開発・設計可能

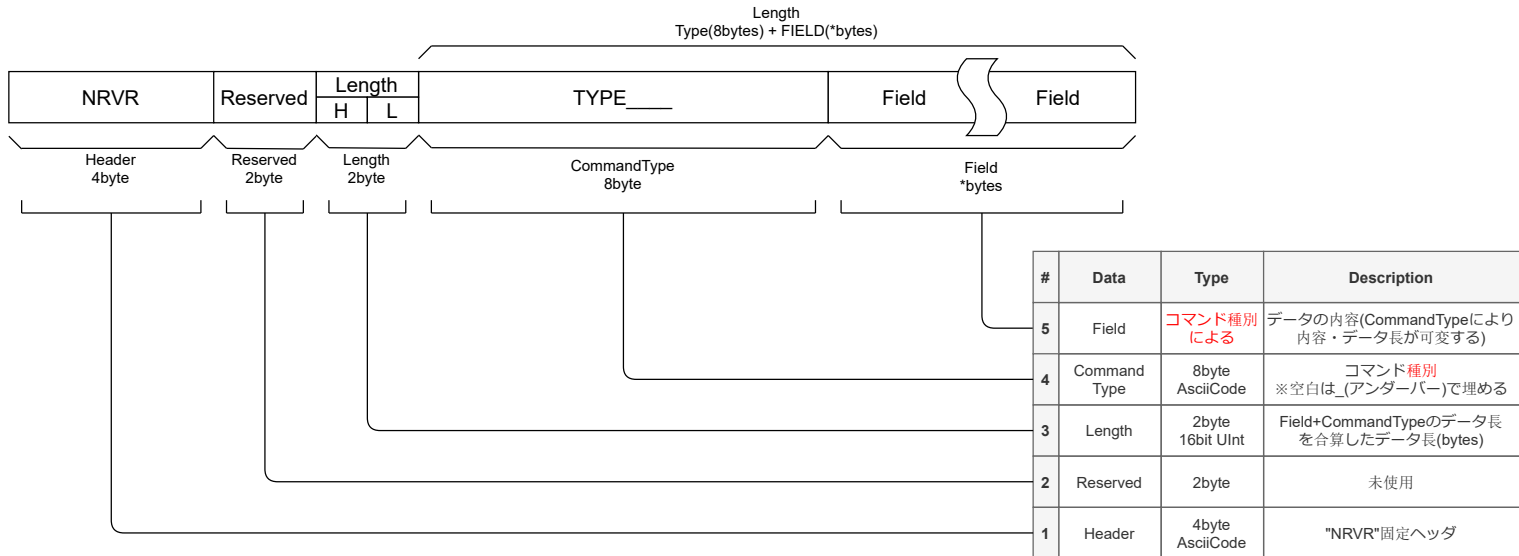
3. 動作フローイメージ



4. 通信プロトコル仕様

① 基本コマンドフレーム構成

※基本的にUDPパケットを使用し、示しているデータはパケット内のデータを示す
 ※フォーマット規格外のデータが送信された場合にはサーバーは無応答もしくは、NAKを返却する



② 共通事項

・通信プロトコル

通信プロトコルには、UDPを使用する。サーバー側は、単一のポート(NoraVRポート)を開放し、クライアント側はそのNoraVRポートに対して全てのコマンドの送受信を行う。

・クライアントID

ログイン時にLOGIN_ACKコマンド内で、クライアントIDが発行されるので、必ず保持しておくこと
 サーバー側は、各種コマンドを受信した際にクライアントIDの照合を行い、正規のクライアントであるか確認を行う

・生存確認(Ping/Pong)

サーバー側は基本的に、受信したPingに対してPongを返却する。クライアント側は、3~8秒程度の間隔でサーバー側に送信すること。
 ただし、サーバー側は、クライアント側からPingが一定時間送信されなかった場合にのみ、Pongを自動的に送信するので考慮しておくこと。
 クライアント側は、音声パケットの送受信中においても、Pingを送信すること。サーバー側は一定時間(1分程度)、クライアント側から、Pingが送出されなければ、コネクションが切れたものとして強制的にログアウトを行う。

・コマンド応答(ACK/NAK)

サーバー側は全ての要求に対して、NAK応答を返却する場合がある(コネクションが失われているなど)
 ACKに関しては各コマンド毎に扱いが異なり、ACKを返すコマンドもあれば、LOGIN_ACKなどの専用ACKコマンドが用意されているコマンドも存在する。

・プロトコルバージョンの互換性

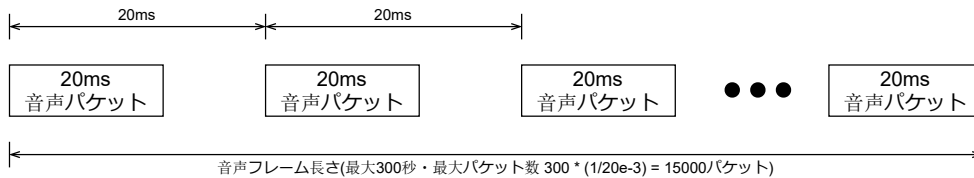
可能な限りの後方互換性を保つため、規定されている各コマンドはReserved(予約)としている領域以外は変更しないものとするが、実装した時点でReservedとなっても、後のバージョンでデータが入っている可能性があるため、無視するなど考慮すること。
 また、未対応のコマンドを受信した場合には、データをパケット丸ごと破棄すること。

・プロトコルバージョンの決定

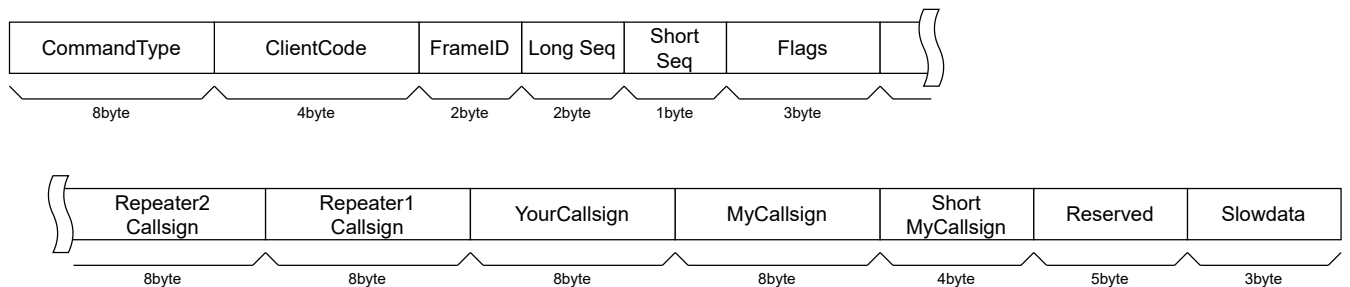
クライアント側が認識するプロトコルバージョンは、LOGINACKコマンドにて返答されるプロトコルバージョンにて最終決定を行うこと。
 尚、LGINUSR2にてサーバー側に希望するプロトコルバージョンは、サーバー側の対応バージョンが古い等の理由により、必ずしも希望するバージョンにならない事がある。

・音声フレーム

音声パケットの間隔(=長さ)、0.02秒(=20ms)分とする。また、音声アプリケーションにおける、ソース音声サンプリングフォーマットは、8ksps 16bit 符号付きを推奨する。
 音声パケットの送出間隔については、可能な限り20ms間隔を保つものとするが、特に規定はしない。
 音声フレームの最大長は300秒とし、サーバー側はこれを超えた場合、音声フレームを強制終了する。
 クライアント側は、180秒程度で制限を設ける事を推奨する。



・音声パケット中の共通ヘッダ情報各パラメータ

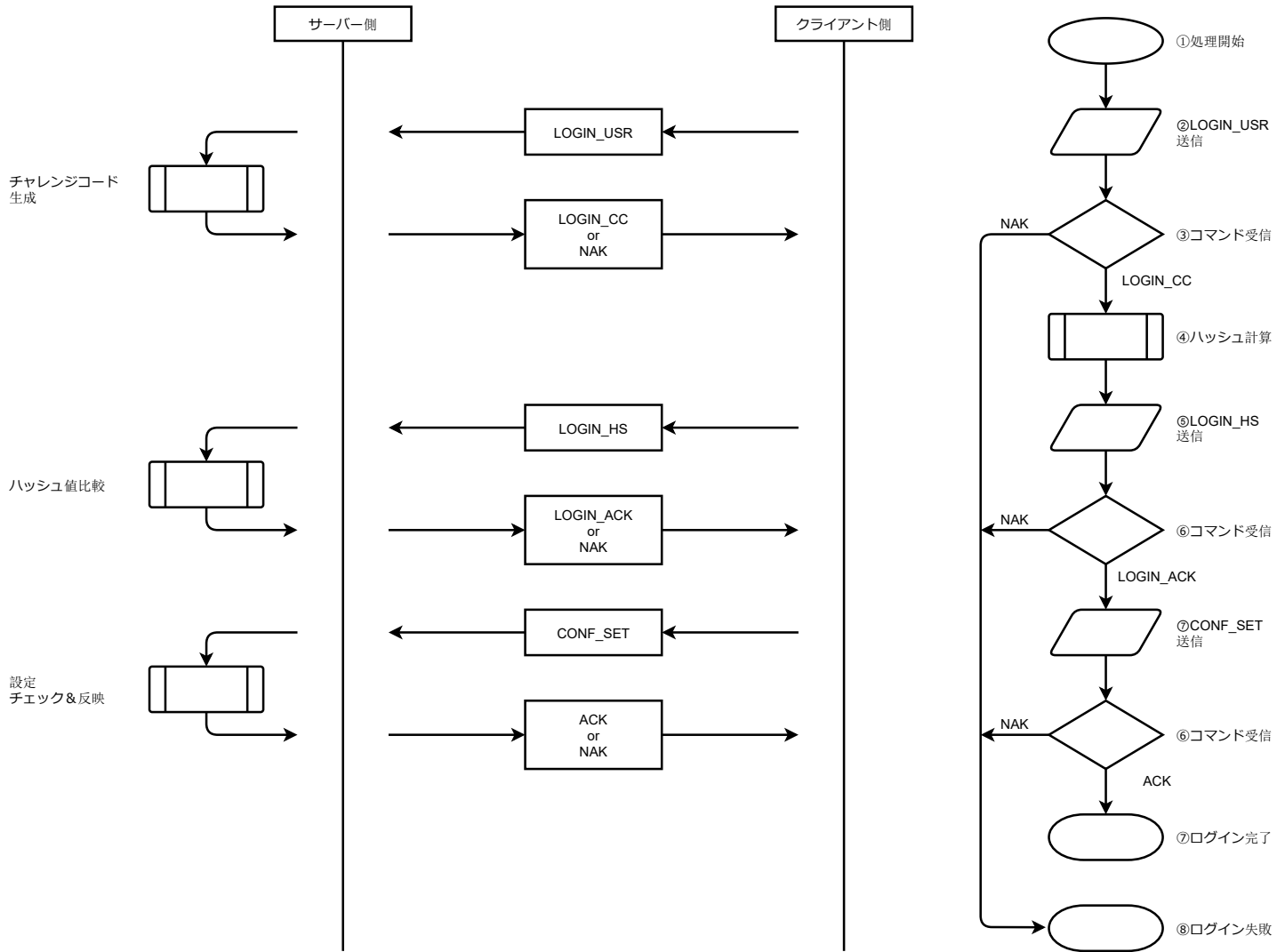


CommandType	VTAMBE_VTOPUS_VTPCM_のいずれか
FrameID	0x0001~0xFFFFの範囲でランダム値を入れる ※単一フレーム中は同一の値にすること
LongSequence	0x0000~0xFFFFの範囲でインクリメントして循環させる ※単一フレーム中の初回送信の際は0x0000から開始すること
ShortSequence	0x00~0x14の範囲でインクリメントして循環させる 7ビット目を立てることにより、フレーム終端を通知する (送信終了の際も、最後の音声パケットにおいては、7ビット目を立てること) ※単一音声フレーム中の初回送信の際は0x00から開始すること
Flags	DSTARヘッダのフラグ(3byte) ※送信する際には、1byte目のフラグの7bit目を立てる必要があるので注意すること(詳細はDSTAR仕様書を参照)
Repeater2 Callsign	LOGIN_ACKコマンドにて取得したコールサインをセットする 山かけの場合Repeater Callsignをセットし、ゲート超えや制御コマンドはGateway Callsignをセットする ※空白は0x20(スペース)で埋めること
Repeater1 Callsign	LOGIN_ACKコマンドにて取得したRepeater Callsignをセットする ※空白は0x20(スペース)で埋めること
Your Callsign	送信先コールサインをセットすること 例として、山かけ・リフレクターの場合CQCQCQ、コール指定の場合JJ0TPX F、エリア指定の場合/JQ1ZYC Fなど ※空白は0x20(スペース)で埋めること
My Callsign	自局コールサイン(8文字)をセットする ※空白は0x20(スペース)で埋めること
Short My Callsign	自局コールサイン(4文字)をセットする ※空白は0x20(スペース)で埋めること
Slowdata	DSTARスローデータをセットする(詳細はDSTAR仕様書を参照) 例として、Nullデータとして指定されている0x66 0x66 0x66を入れる場合には、0x70 0x4F 0x93で各々xorを取って からセットする Nullデータ 0x66 0x66 0x66 xor 0x70 0x4F 0x93 → 0x16 0x29 0xF5

余白

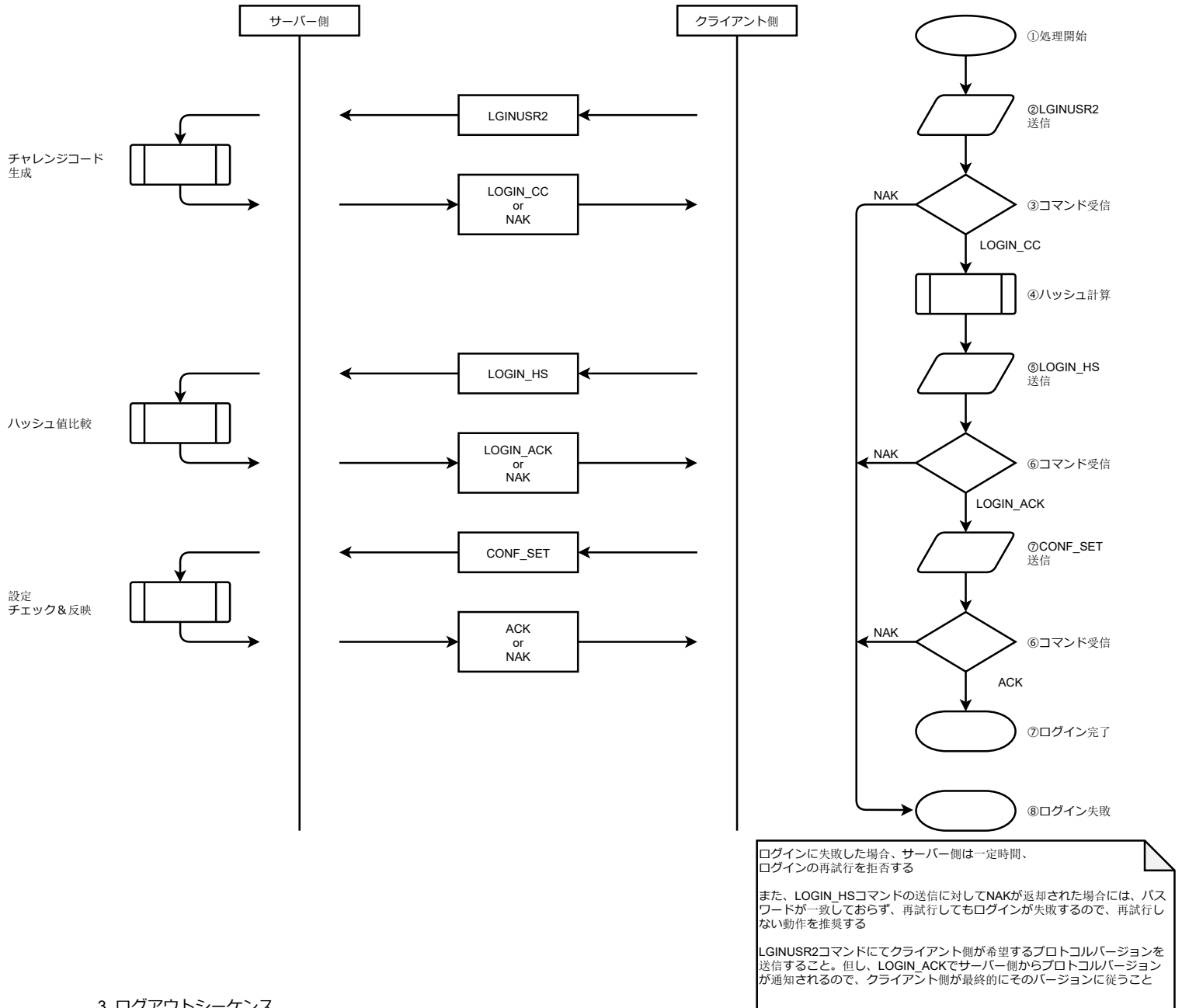
③ 通信動作シーケンス

1. ログインシーケンス

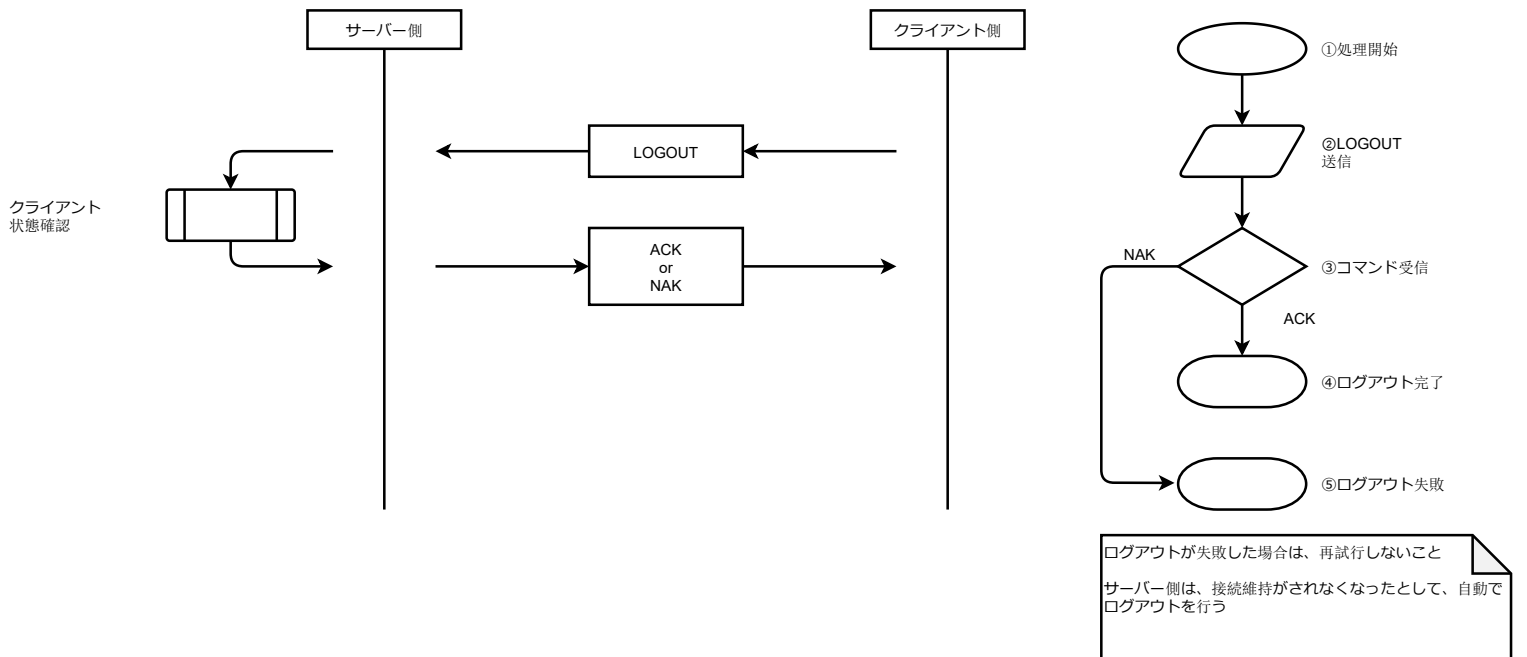


ログインに失敗した場合、サーバー側は一定時間、ログインの再試行を拒否する
 また、LOGIN_HSコマンドの送信に対してNAKが返却された場合には、パスワードが一致しておらず、再試行してもログインが失敗するので、再試行しない動作を推奨する

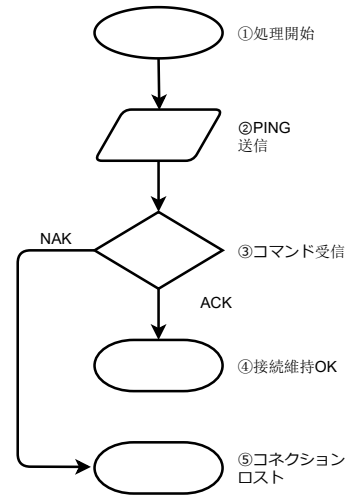
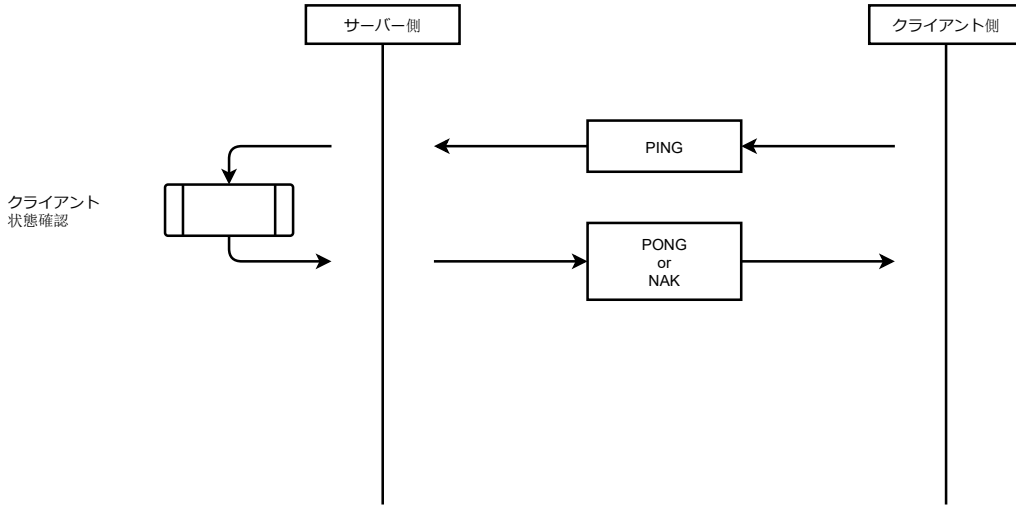
2. ログインシーケンス(プロトコルバージョンv2以降)



3. ログアウトシーケンス

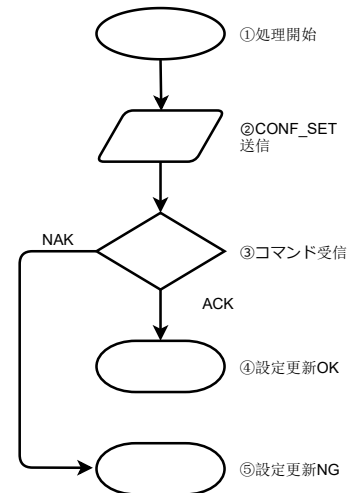
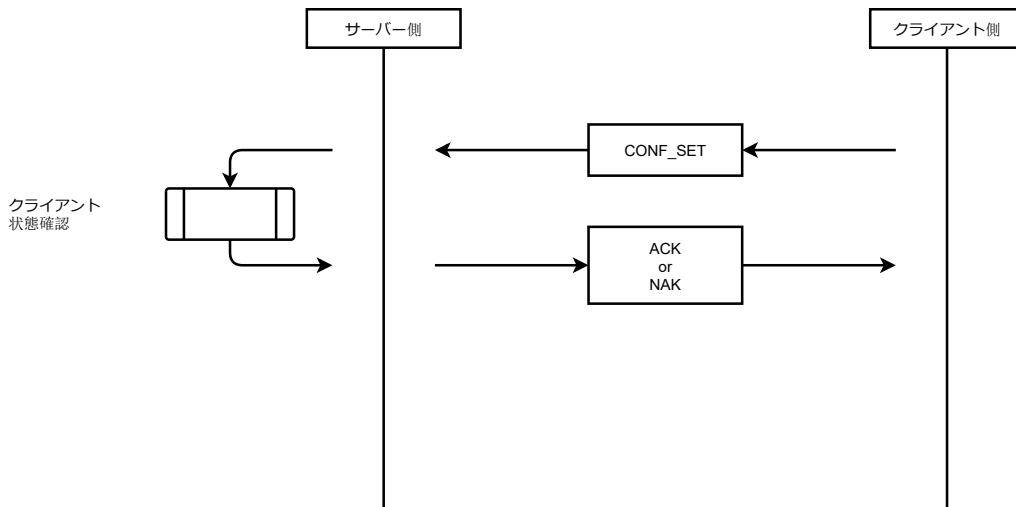


4. 接続維持シーケンス



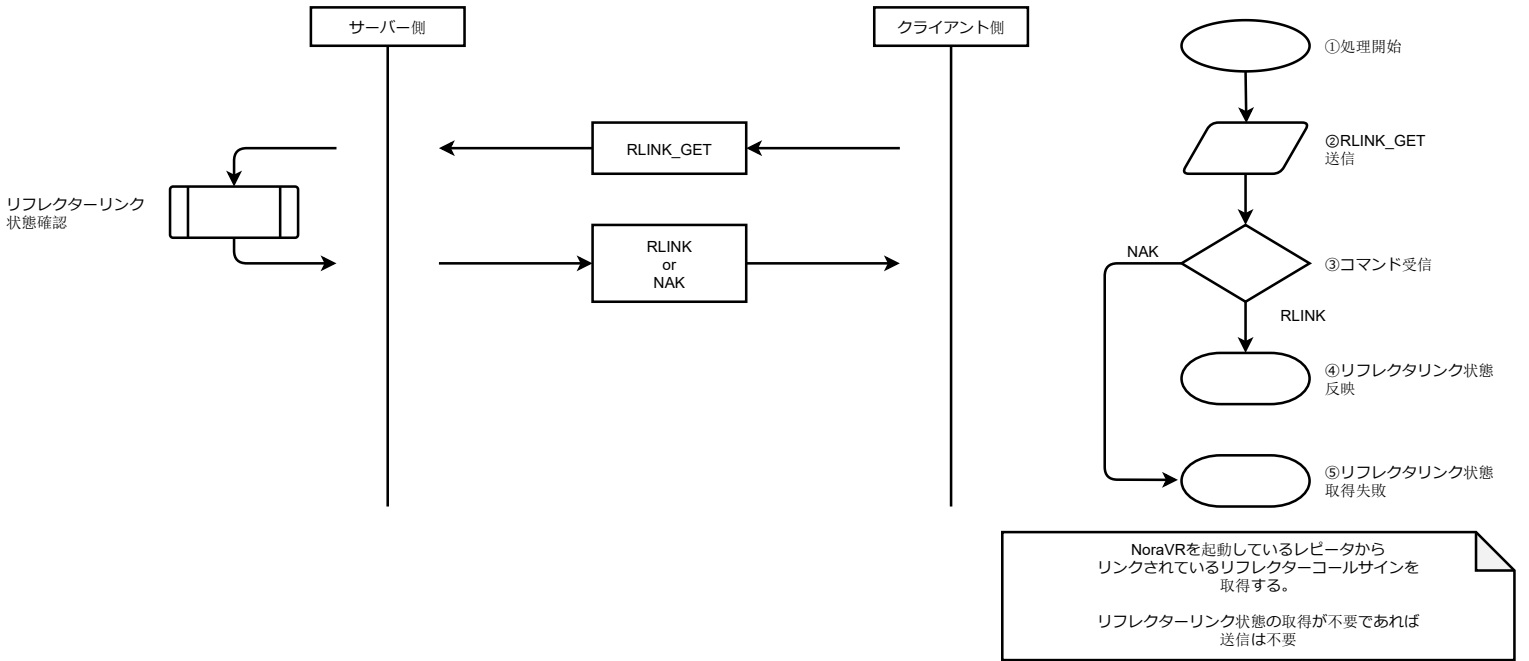
PINGは音声パケットなどの他のデータを送受信中においても、一定間隔で送信すること
よって、PINGを送信してからPONGを待たずに他の処理を行い、PONGを受信したら単純に最終PONG受信時刻を更新するような処理を推奨する

5. 設定更新シーケンス

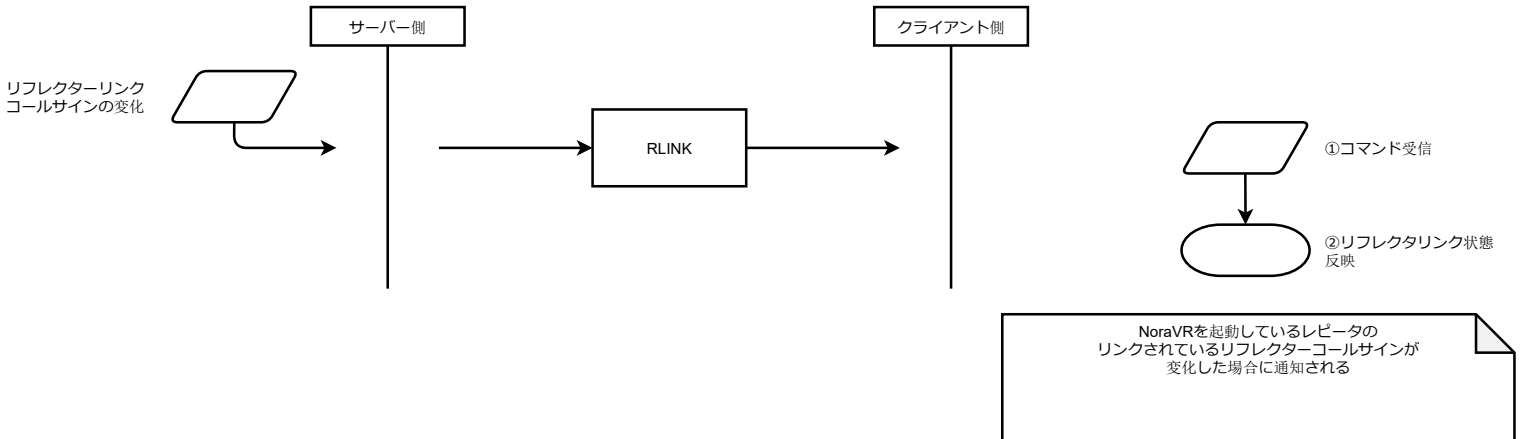


設定を更新する場合に送信すること
ログイン時においては、必ず送信すること
(送信されなければ音声パケットが流れない)
例として、エコーバックのON/OFF、
音声コーデックの変更など

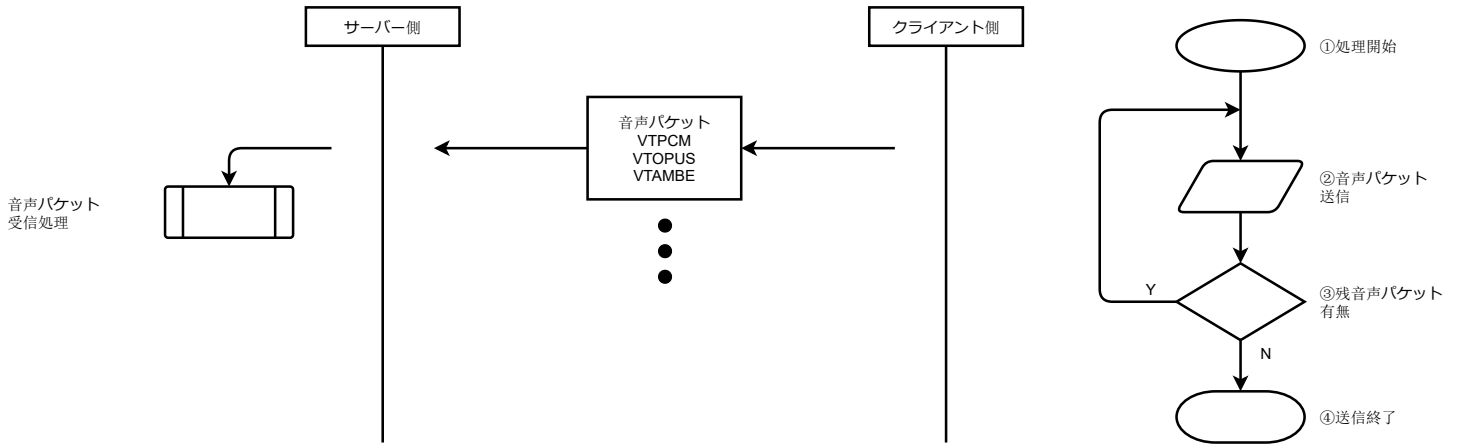
6. リフレクタリンク状態の要求シーケンス



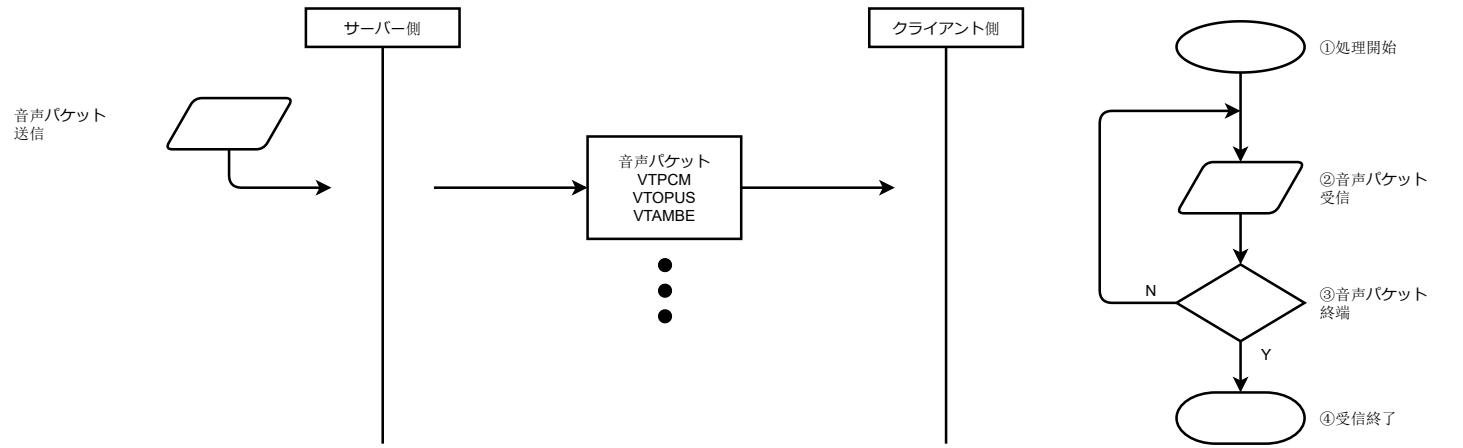
7. リフレクタリンク状態の通知シーケンス



8. アップリンク音声シーケンス

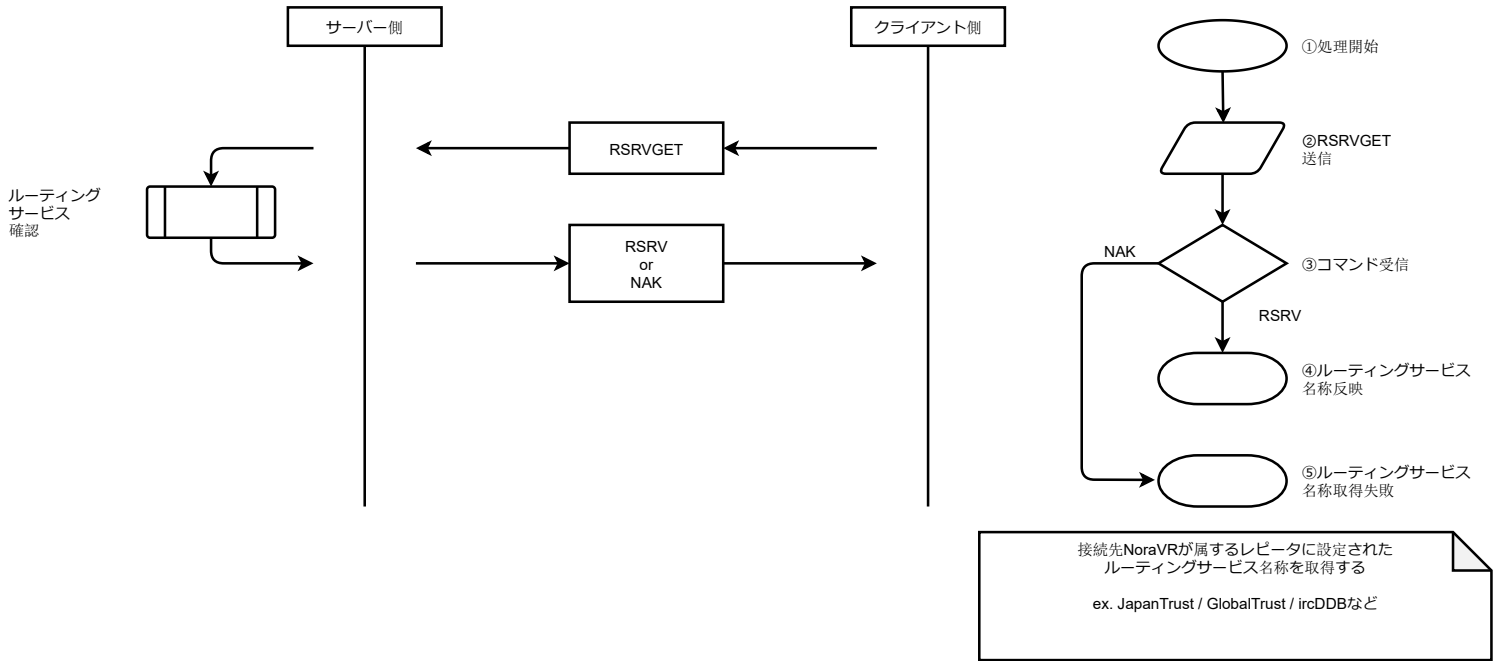


9. ダウンリンク音声シーケンス

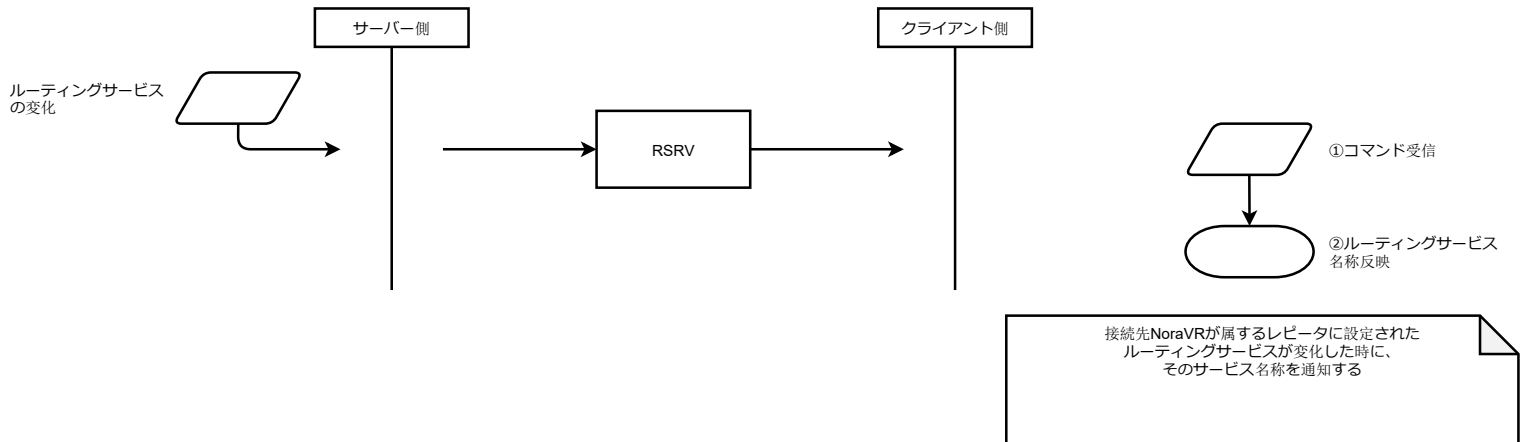


7. ルーティングサービス名称の取得シーケンス

対応プロトコルバージョン V2

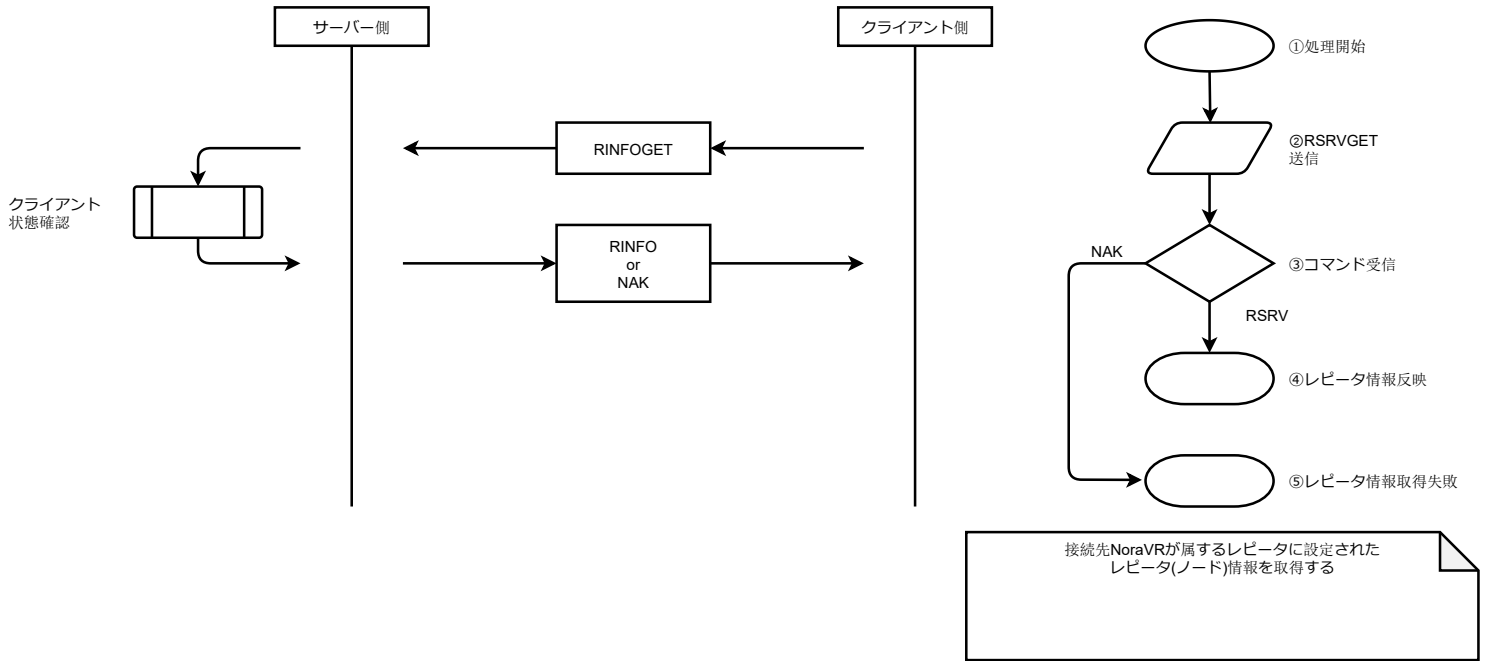


6. ルーティングサービス名称の通知シーケンス



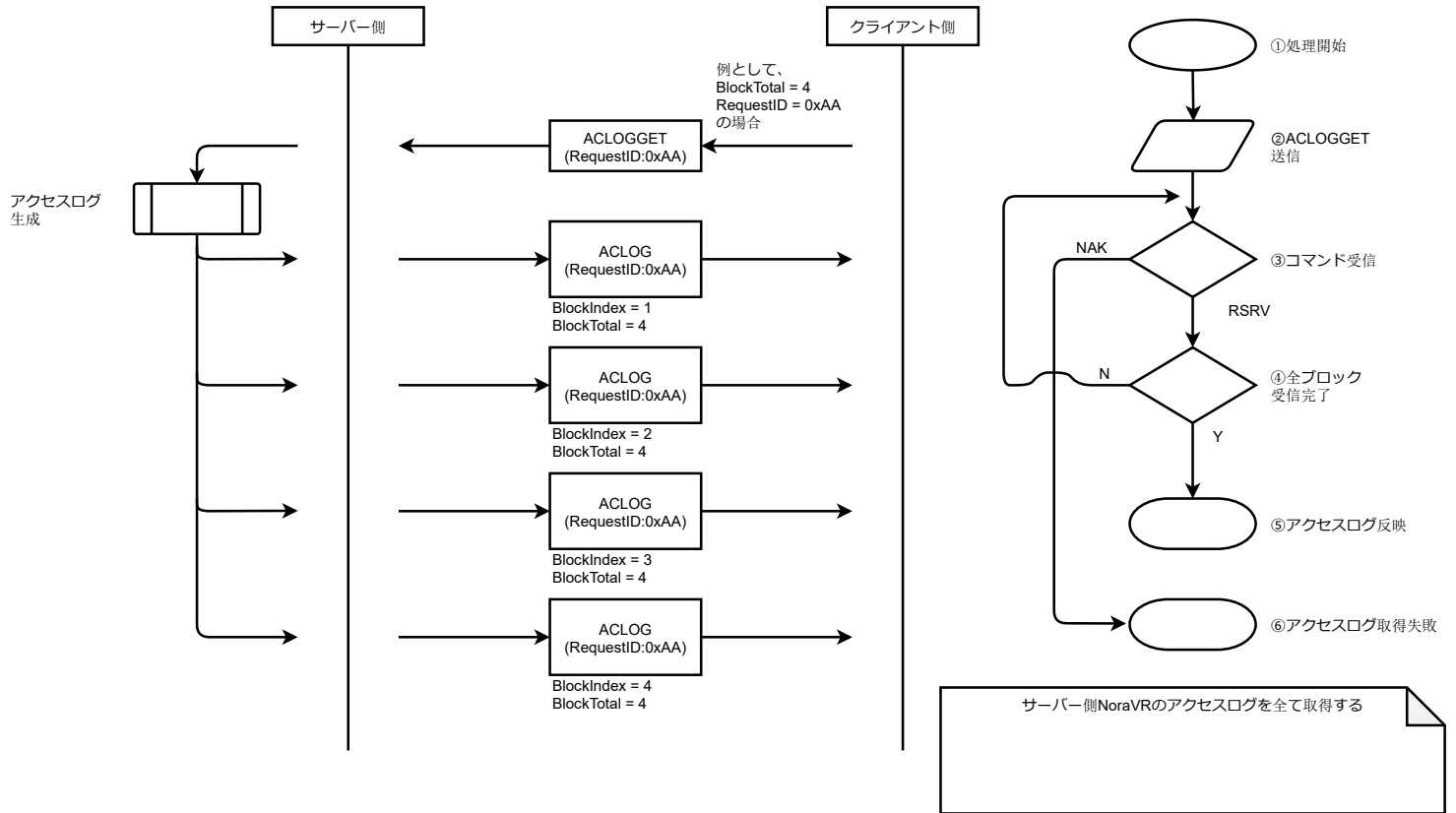
10. レピータ情報の取得シーケンス

対応プロトコルバージョン V2

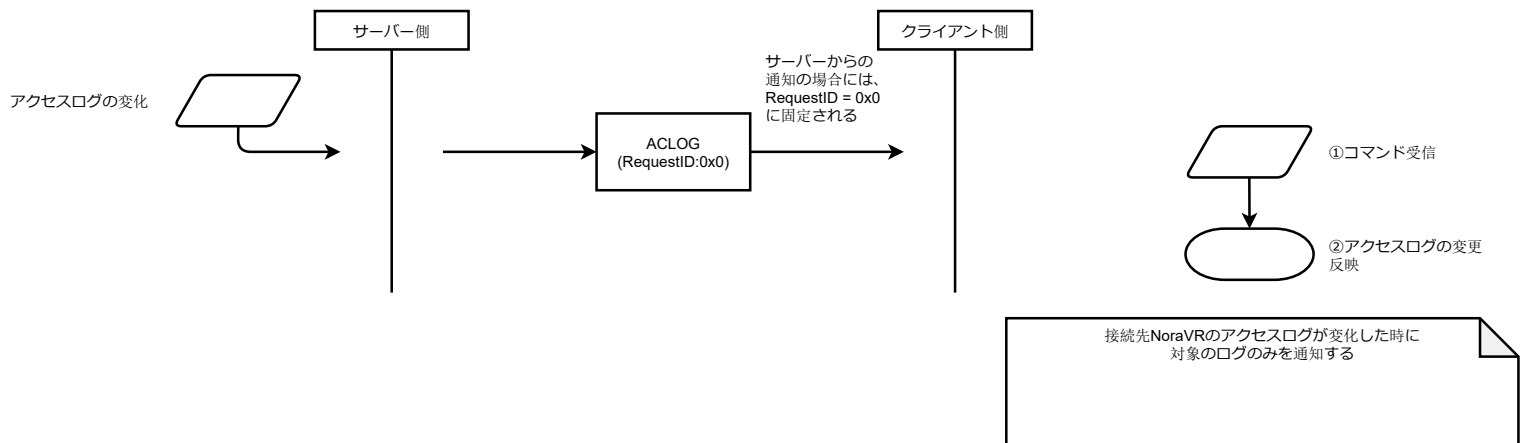


11. アクセスログの取得シーケンス

対応プロトコルバージョン V2

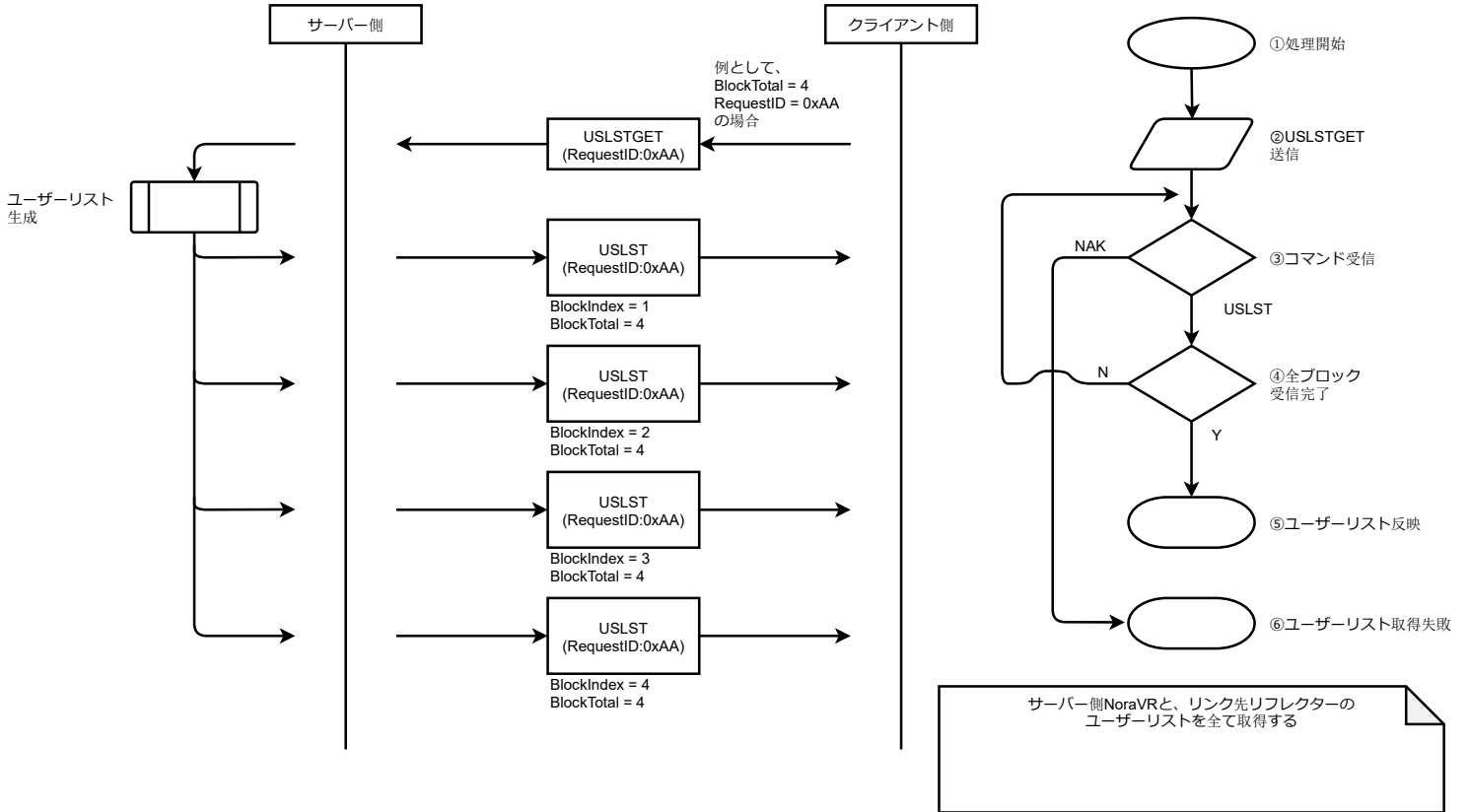


12. アクセスログの通知シーケンス

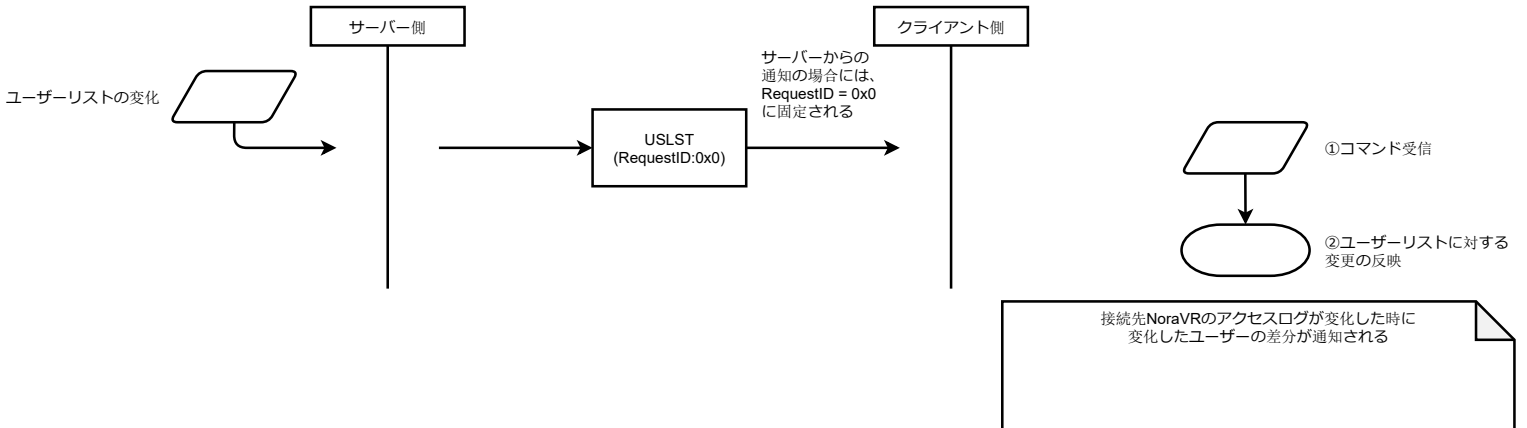


13. ユーザーリストの取得シーケンス

対応プロトコルバージョン V2



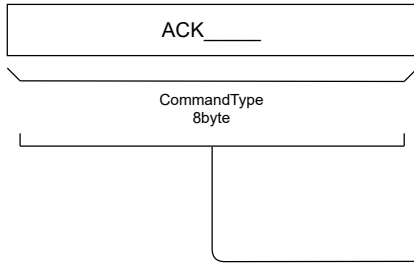
14. ユーザーリストの変更通知シーケンス



余白

④ コマンドタイプ別フィールド構成

1. ACK____

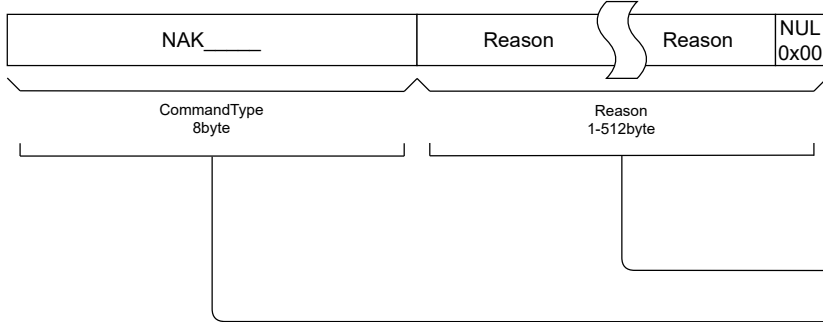


クライアント → サーバー
サーバー → クライアント
 対応プロトコルバージョン V1

サーバーが処理を受け付けた事を示す

#	Data	Length	Description
1	Login Callsign	8byte AsciiCode	"ACK____"固定ヘッダ ※_は5文字

2. NAK____

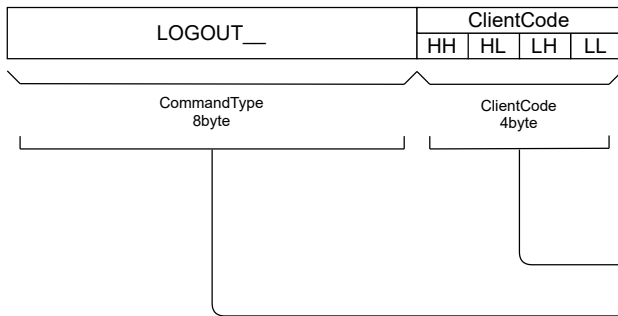


クライアント → サーバー
サーバー → クライアント
 対応プロトコルバージョン V1

サーバーが処理を受け付けられない事を示す
Reasonには処理を受け付けられない理由が入る

#	Data	Length	Description
2	Reason	1-512byte AsciiCode	NAKが返される理由 0x00にて終端されたAsciiCode
1	Login Callsign	8byte AsciiCode	"NAK____"固定

3. LOGOUT__



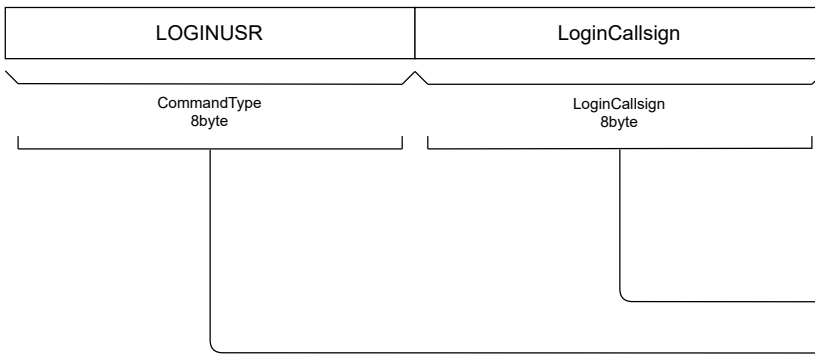
クライアント → サーバー
サーバー → クライアント
 対応プロトコルバージョン V1

クライアントからサーバーへログアウトを通知する

#	Data	Length	Description
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"LOGOUT__"固定

4. LOGINUSR

- クライアント → サーバー
 - サーバー → クライアント
- 対応プロトコルバージョン V1



クライアント側からのログイン時におけるログインコマンド

ログインユーザーとは、NoraVRにログインするコールサインであって、音声送信時に送信するMYコールサインに相当するものではない

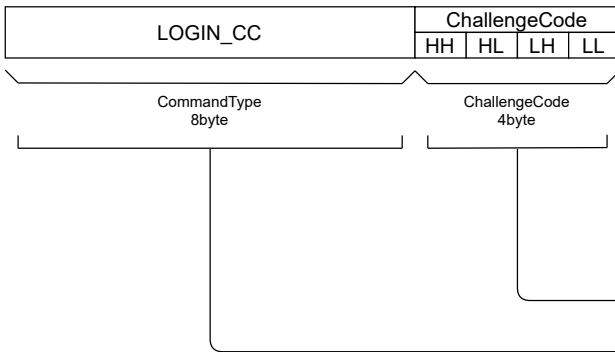
サーバー側で受け付けられると、LOGIN_CCもしくはNAKが返される

このコマンドは実装非推奨(LGINUSR2を使用のこと)

#	Data	Length	Description
2	Login Callsign	8byte AsciiCode	ログインコールサイン ※空白はスペースで埋める
1	Command Type	8byte AsciiCode	"LOGINUSR"固定

5. LOGIN_CC

- クライアント → サーバー
 - サーバー → クライアント
- 対応プロトコルバージョン V1



サーバーからクライアントへチャレンジコードを送信する

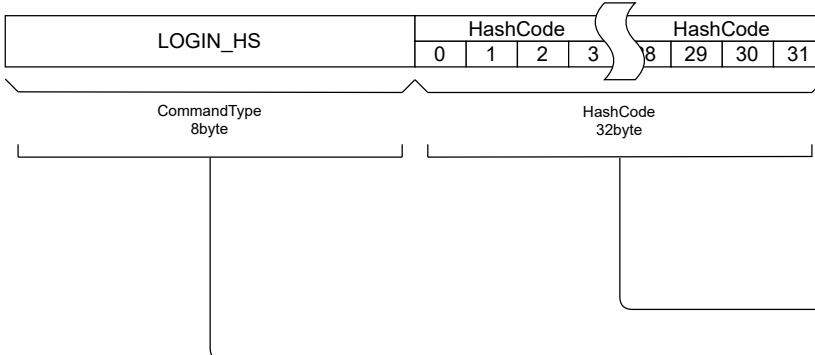
クライアントではこのチャレンジコードとログインパスワードを結合し、ハッシュ値(SHA256)を生成しておく

【ハッシュシードの構成】
チャレンジコード(4byte BEBinary) + ログインパスワード(Ascii)

#	Data	Length	Description
2	Challenge Code	4byte BE Binary	チャレンジコード 4byte
1	Command Type	8byte AsciiCode	"LOGIN_CC"固定

6. LOGIN_HS

- クライアント → サーバー
 - サーバー → クライアント
- 対応プロトコルバージョン V1



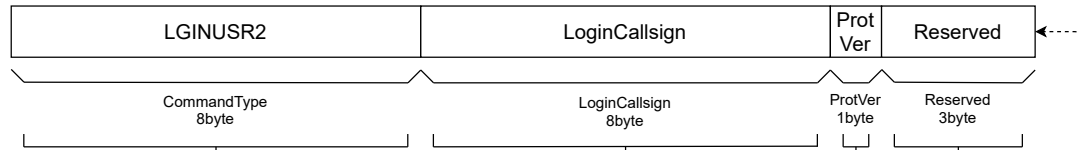
クライアントからサーバーへハッシュコードを送信する

サーバー側では受信したハッシュコードが生成されたハッシュコードと合致しているか確認し、合致していればLOGINACK、合致していなければNAKを返す

#	Data	Length	Description
2	HashCode	32byte LE Binary	ハッシュコード SHA256
1	Command Type	8byte AsciiCode	"LOGIN_HS"固定

7. LOGINUSR v2

■クライアント → サーバー
 □サーバー → クライアント
 対応プロトコルバージョン V2



クライアント側からのログイン時におけるログインコマンド

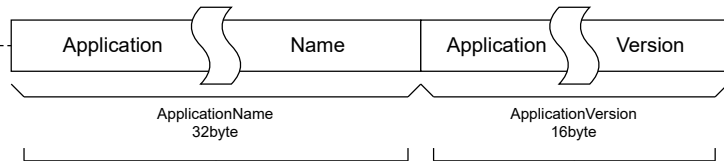
ログインユーザーとは、NoraVRにログインするコールサインであって、音声送信時に送信するMYコールサインに相当するものではない

サーバー側で受け付けられると、LOGIN_CCもしくはNAKが返される

ProtocolVersionには希望するプロトコルバージョンを送信する。最終的なプロトコルバージョンは、LOGIN_ACK内で返される ProtocolVersionをもって決定すること

ApplicationName、ApplicationVersionはサーバー側の管理用であり、クライアント側が自由に設定可能

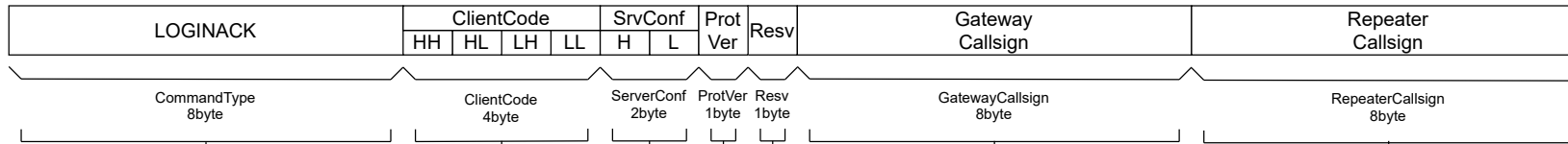
#	Data	Length	Description
4	Reserved	3byte Binary	予約領域
3	Protocol Verstion	1byte Binary	希望プロトコルバージョン
2	Login Callsign	8byte AsciiCode	ログインコールサイン ※空白はスペースで埋める
1	Command Type	8byte AsciiCode	"LOGINUSR2"固定



#	Data	Length	Description
6	Application Name	32byte AsciiCode	クライアントアプリケーション名 ※余白はNULL(0x0)で埋めること
5	Applicaition Version	16byte AsciiCode	クライアントバージョン ※余白はNULL(0x0)で埋めること

8. LOGINACK

■クライアント → サーバー
 □サーバー → クライアント
 対応プロトコルバージョン V1



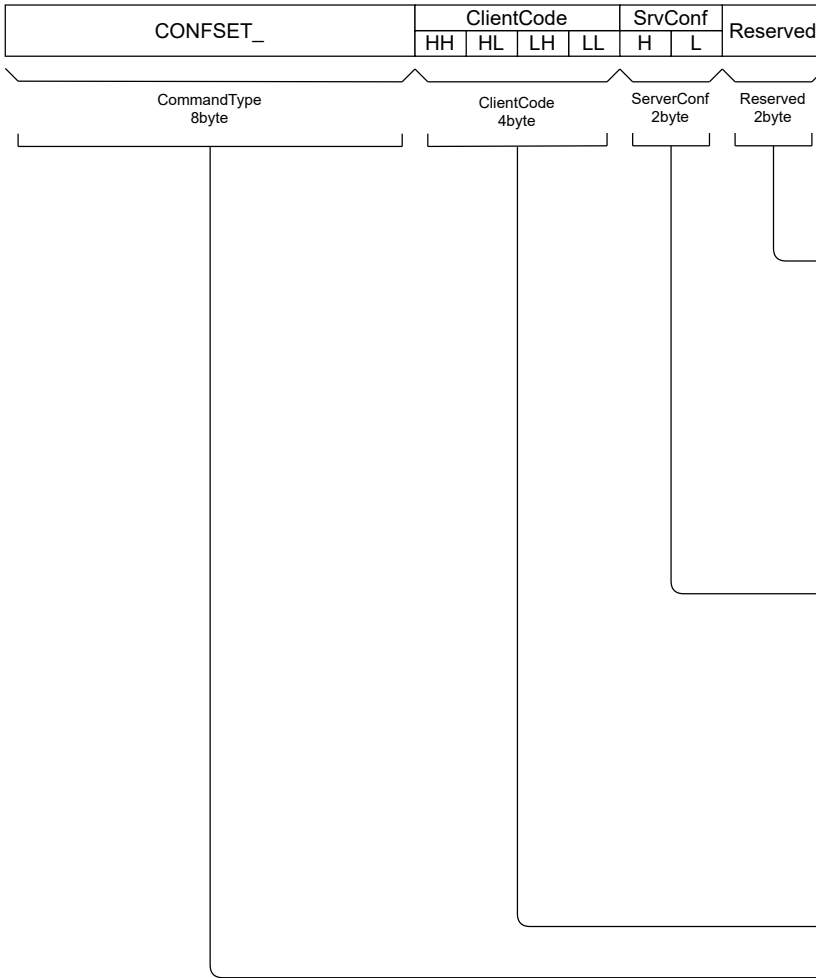
サーバーからクライアントへログインが完了した事を通知
 ログイン完了の通知と、クライアントコード、サーバーの設定を送信する。クライアントコードは、クライアント側で保持しておくこと。また、サーバー設定も同様に保持しておくこと。

#	Data	Length	Description		
5	Reserved	1byte Binary	Reserved		
4	Protocol Verstion	1byte Binary	動作プロトコルバージョン ※クライアント側は、このデータからプロトコルバージョンを決定すること		
3	Server Configuration	2byte BE binary	15	Reserved	
			14	Reserved	
			13	Reserved	
			12	Reserved	
			11	Reserved	
			10	Reserved	
			9	Reserved	
			8	Reserved	
			7	AMBE 1:enable/0:disable	
			6	Reserved	
			5	Reserved	
			4	Reserved	
			3	Supported Codec	Opus VBR 8k 1:enable/0:disable
			2		Opus VBR 24k 1:enable/0:disable
			1		Opus VBR 64k 1:enable/0:disable
			0		PCM 1:enable/0:disable
2	ClientCode	2byte BE UInt	クライアントコード クライアント側で保持しておく		
1	Command Type	8byte AsciiCode	"LOGIN_ACK"固定		

#	Data	Length	Description
7	Repeater Callsign	8byte AsciiCode	レピータコールサイン
6	Gateway Callsign	8byte AsciiCode	ゲートウェイコールサイン

9. CONFSET_

■クライアント → サーバー
 □サーバー → クライアント
 対応プロトコルバージョン V1

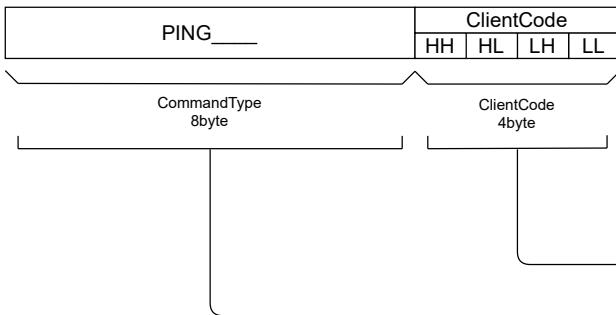


クライアントからサーバーへ設定変更を要求する
 クライアントから設定を送り、サーバー内の設定を書き換える
 SupportedCodecに関して複数のコーデックをenableにした場合、サーバー側においてenableにしたコーデックの中から自動で選択される

#	Data	Length	Description
4	Reserved	2byte BE binary	Reserved
3	Server Configuration	2byte BE binary	15 RFNode RF OutputNode 1:Yes/0:No
			14 Echoback Echoback 1:enable/0:disable
			13 Reserved
			12 Reserved
			11 Reserved
			10 LocalUserNotify 1:enable/0:disable
			9 RemoteUserNotify 1:enable/0:disable
			8 AccessLogNotify 1:enable/0:disable
			7 AMBE 1:enable/0:disable
			6 Reserved
			5 Reserved
			4 Supported Codec
			3 Opus VBR 8k 1:enable/0:disable
			2 Opus VBR 24k 1:enable/0:disable
			1 Opus VBR 64k 1:enable/0:disable
0 PCM 1:enable/0:disable			
2	ClientCode	4byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"CONFSET_"固定

10. PING_

■クライアント → サーバー
 □サーバー → クライアント
 対応プロトコルバージョン V1

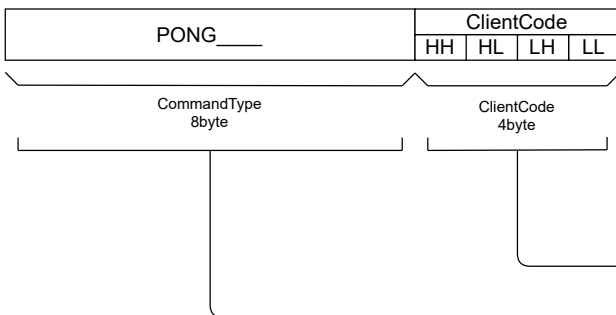


クライアントからサーバーへの生存確認

#	Data	Length	Description
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"PING_"固定

11. PONG_

□クライアント → サーバー
 ■サーバー → クライアント
 対応プロトコルバージョン V1

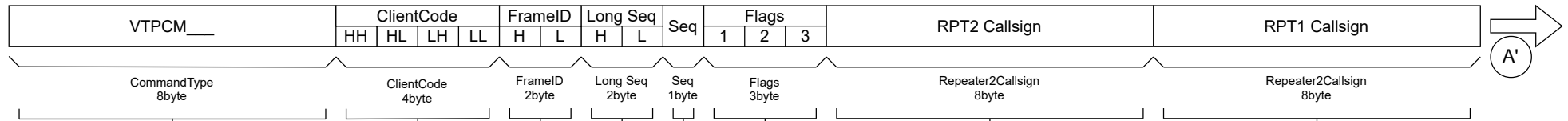


サーバーからクライアントへの生存確認

#	Data	Length	Description
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"PONG_"固定

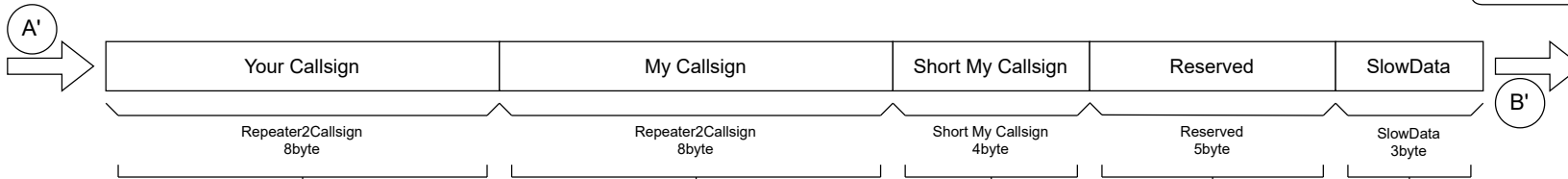
12. VTPCM__

■クライアント → サーバー
 ■サーバー → クライアント
 対応プロトコルバージョン V1

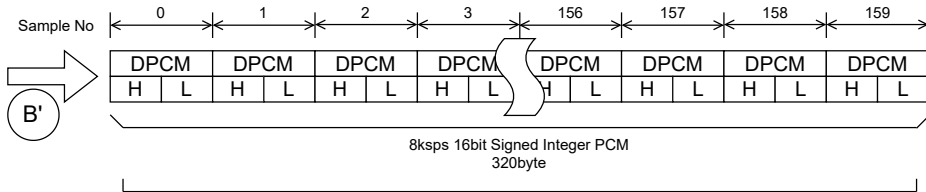


#	Data	Length	Description
2	ClientCode	4byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	コマンド種別 "VTPCM__"固定

#	Data	Length	Description
8	Repeater1 Callsign	8byte AsciiCode	レピータ1コールサイン
7	Repeater2 Callsign	8byte AsciiCode	レピータ2コールサイン
6	Flags	3byte binary	DSTARヘッダフラグ 送信の際は、0x40 00 00とすること
5	Short Sequence	1byte binary	0x00~0x14の範囲でパケット毎にインクリメントして循環させ、最終パケットで0x40とORする
4	Long Sequence	2byte BE binary	0x0000~0xFFFFの範囲でパケット毎にインクリメントして循環させる
3	FrameID	2byte BE binary	フレームID フレーム毎にランダム生成



#	Data	Length	Description
13	SlowData	3byte binary	低速データセグメント
12	Reserved	3byte binary	予約領域
11	Short MyCallsign	4byte AsciiCode	MY CALLSIGN(SHORT)
10	MyCallsign	8byte AsciiCode	MY CALLSIGN
9	Your Callsign	8byte AsciiCode	YOUR CALLSIGN

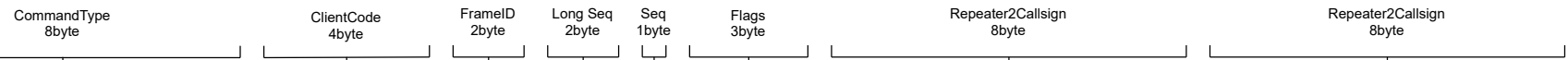
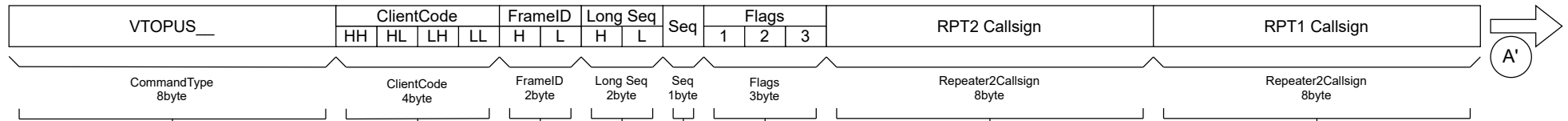


#	Data	Length	Description
14	PCM	2byte UInt *160sample	8ksps 16bit Signed Integer PCM

PCM(非圧縮)形式の音声パケットフォーマット
 ただし、LOGINACKにてPCMサポートビットが立っていない状態では送信不可

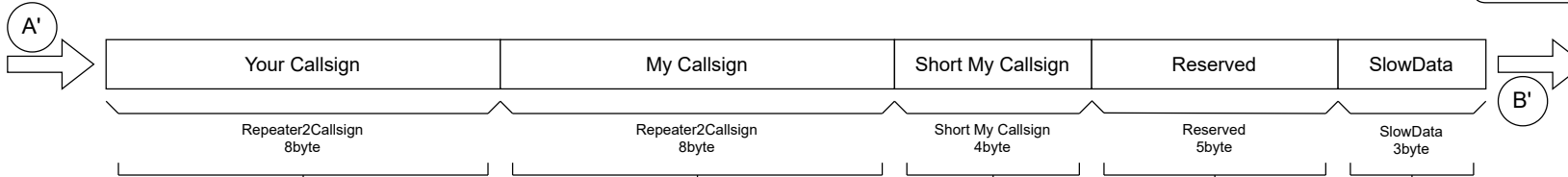
13. VTOPUS__

■クライアント → サーバー
 ■サーバー → クライアント
 対応プロトコルバージョン V1

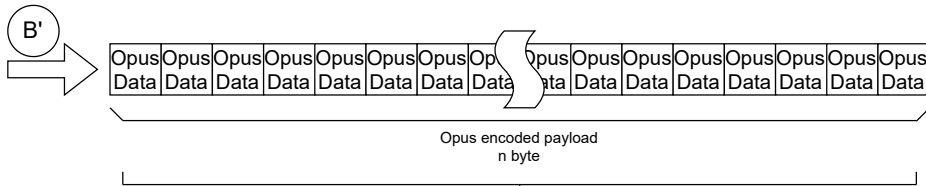


#	Data	Length	Description
2	ClientCode	4byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	コマンド種別 "VTOPUS_"固定

#	Data	Length	Description
8	Repeater1 Callsign	8byte AsciiCode	レピータ1コールサイン
7	Repeater2 Callsign	8byte AsciiCode	レピータ2コールサイン
6	Flags	3byte binary	DSTARヘッダフラグ 送信の際は、0x40 00 00とすること
5	Short Sequence	1byte binary	0x00~0x14の範囲でパケット毎にインクリメントして循環させ、最終パケットで0x40とORする
4	Long Sequence	2byte BE binary	0x0000~0xFFFFの範囲でパケット毎にインクリメントして循環させる
3	FrameID	2byte BE binary	フレームID フレーム毎にランダム生成



#	Data	Length	Description
13	SlowData	3byte binary	低速データセグメント
12	Reserved	3byte binary	予約領域
11	Short MyCallsign	4byte AsciiCode	MY CALLSIGN(SHORT)
10	MyCallsign	8byte AsciiCode	MY CALLSIGN
9	Your Callsign	8byte AsciiCode	YOUR CALLSIGN



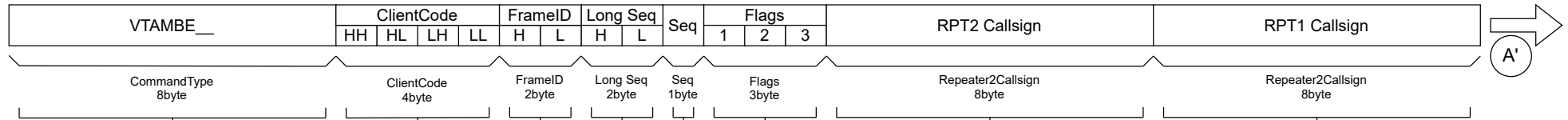
#	Data	Length	Description
14	Opus Voice	nbyte binary	8k/24k/64kbps VBR FEC Opus Encoded Voice

Opus形式で圧縮された音声パケットフォーマット
 8kbps/24kbps/64kbps VBR FEC付でエンコードを行うこと
 OpusCodecについての詳細は下記を参照のこと
<http://opus-codec.org/>
 ただし、LOGINACKにてOpus64k/24k/8kいずれかのサポートビットが立っていない状態では送信不可

25

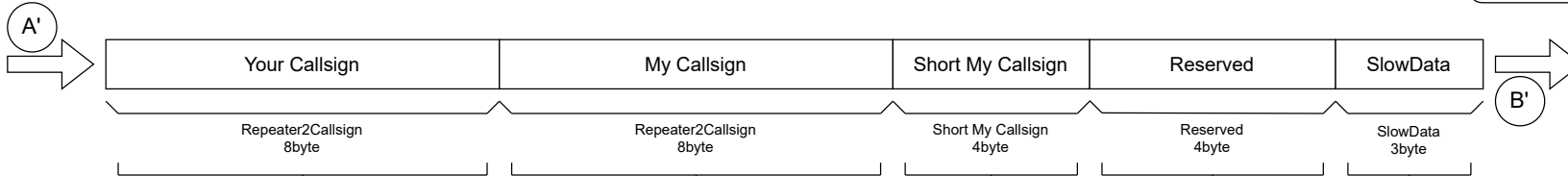
14. VTAMBE__

■クライアント → サーバー
 ■サーバー → クライアント
 対応プロトコルバージョン V1

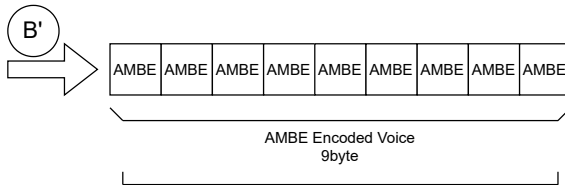


#	Data	Length	Description
2	ClientCode	4byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	コマンド種別 "VTAMBE__"固定

#	Data	Length	Description
8	Repeater1 Callsign	8byte AsciiCode	レピータ1コールサイン
7	Repeater2 Callsign	8byte AsciiCode	レピータ2コールサイン
6	Flags	3byte binary	DSTARヘッダフラグ 送信の際は、0x40 00 00とすること
5	Short Sequence	1byte binary	0x00~0x14の範囲でパケット毎にインクリメントして循環させ、最終パケットで0x40とORする
4	Long Sequence	2byte BE binary	0x0000~0xFFFFの範囲でパケット毎にインクリメントして循環させる
3	FrameID	2byte BE binary	フレームID フレーム毎にランダム生成



#	Data	Length	Description
13	SlowData	3byte binary	低速データセグメント
12	Reserved	3byte binary	予約領域
11	Short MyCallsign	4byte AsciiCode	MY CALLSIGN(SHORT)
10	MyCallsign	8byte AsciiCode	MY CALLSIGN
9	Your Callsign	8byte AsciiCode	YOUR CALLSIGN



#	Data	Length	Description
14	AMBE Voice	9byte binary	AMBE Encoded Voice 3.6kbps(FEC1.2kbps)

AMBE形式で圧縮された音声(データ)パケットフォーマット

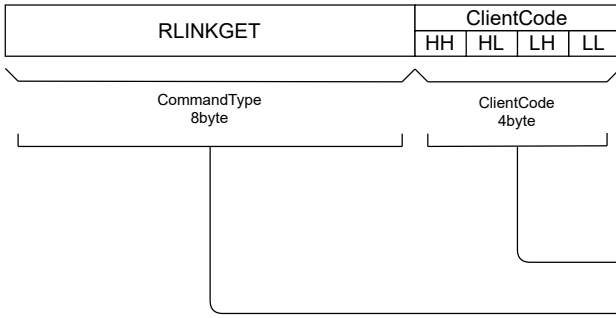
音声の場合には、3.6kbps(FEC1.2kbps)でエンコードされたデータをセットすること

ファストデータをセットする場合には、DSTAR仕様書を参照のこと

ただし、LOGINACKにてAMBEサポートビットが立っていない状態では送信不可

15. RLINKGET

■クライアント → サーバー
 □サーバー → クライアント
 対応プロトコルバージョン V1

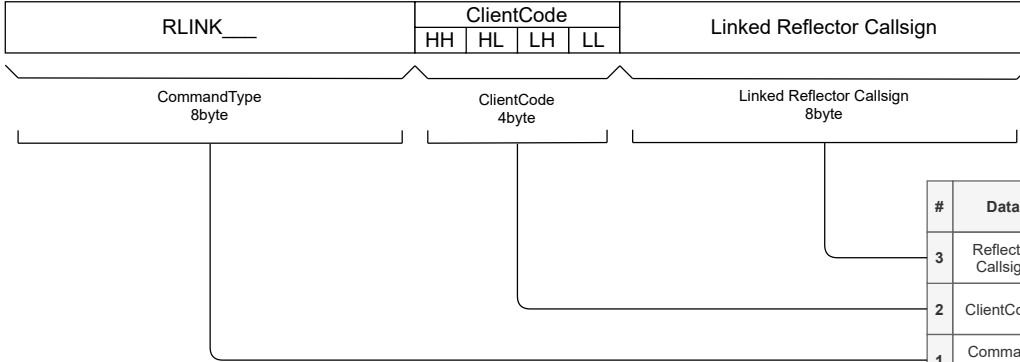


クライアントからサーバーへリフレクタリンク状態を要求

#	Data	Length	Description
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"RLINKGET"固定

16. RLINK__

□クライアント → サーバー
 ■サーバー → クライアント
 対応プロトコルバージョン V1

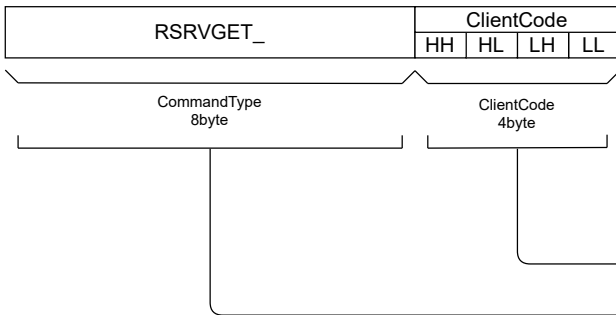


リフレクターのリンク状態を通知

#	Data	Length	Description
3	Reflector Callsign	8byte AsciiCode	リンクしている リフレクターコールサイン
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"RLINK__"固定

17. RSRVGET

■クライアント → サーバー
 □サーバー → クライアント
 対応プロトコルバージョン V2

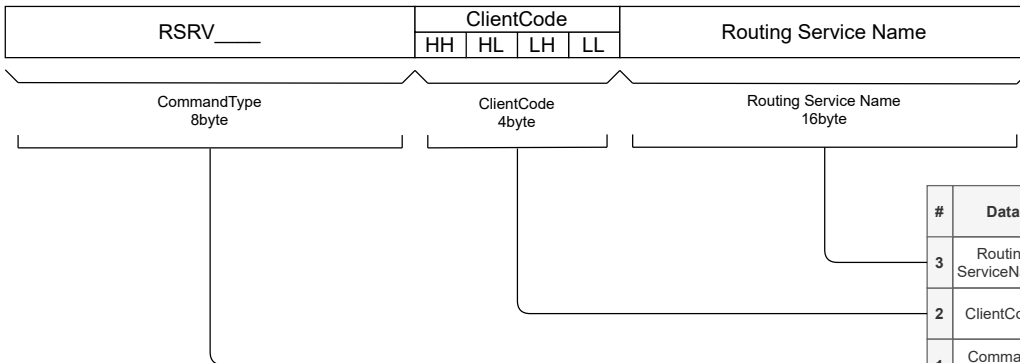


クライアントからサーバーへ現在のルーティングサービス名を要求

#	Data	Length	Description
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"RSRVGET_"固定

18. RSRV__

□クライアント → サーバー
 ■サーバー → クライアント
 対応プロトコルバージョン V2

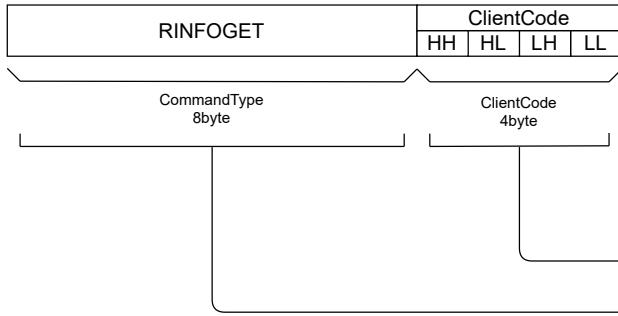


サーバーからクライアントへ現在のルーティングサービス名の変更または返答を通知

#	Data	Length	Description
3	Routing ServiceName	16byte AsciiCode	現在のルーティングサービス名 ※余りスペースはNUL(0x00)埋め
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"RSRV__"固定

19. RINFOGET

■クライアント → サーバー
 □サーバー → クライアント
 対応プロトコルバージョン V2

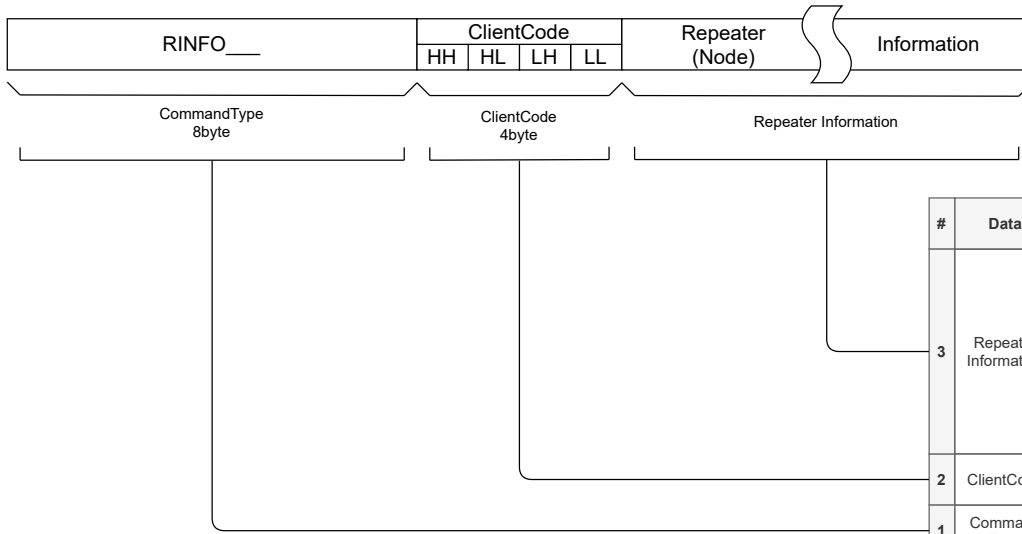


クライアントからサーバーへ
レピータデータを要求

#	Data	Length	Description
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"RINFOGET"固定

20. RINFO__

□クライアント → サーバー
 ■サーバー → クライアント
 対応プロトコルバージョン V2

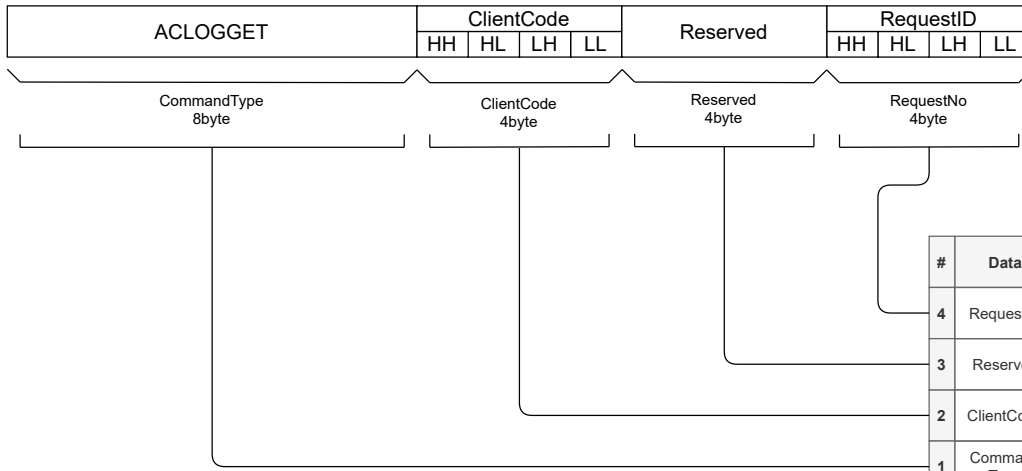


クライアントからサーバーへ
レピータデータを要求

#	Data	Length	Description
3	Repeater Information	JSON UTF-8	callsign コールサイン name 名称 location 設置場所 frequency 周波数(MHz) frequency_offset オフセット周波数(MHz) service_range サービスエリア(km) agl アンテナ地上高(m) url URL description1 備考1 description2 備考2
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"RINFO__"固定

21. ACLOGGET

■クライアント → サーバー
 □サーバー → クライアント
 対応プロトコルバージョン V2



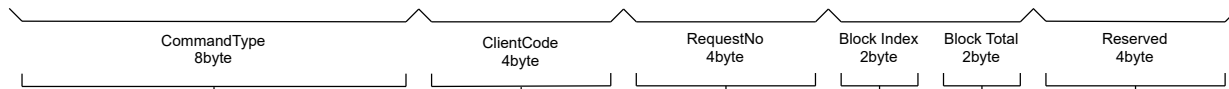
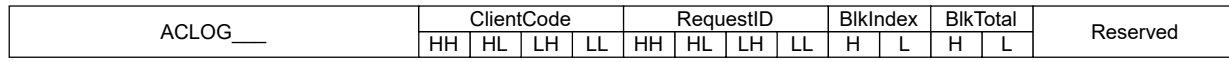
クライアントからサーバーへ
アクセスログデータを要求

#	Data	Length	Description
4	RequestID	4byte BE UInt	リクエストID(自由に設定可) ACLOG返答時に同じコードが使われる
3	Reserved	3byte	予約領域
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"ACLOGGET"固定

22. ACLOG

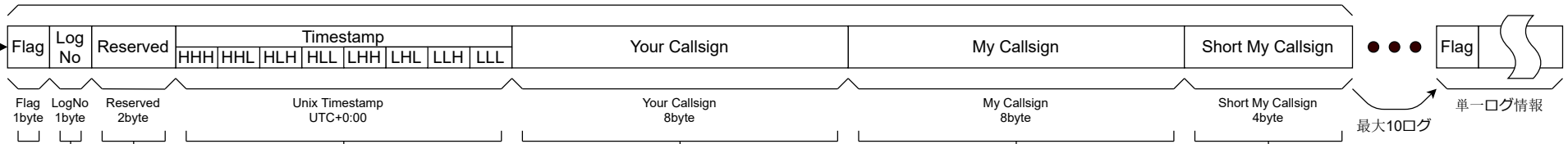
□クライアント → サーバー
 ■サーバー → クライアント
 対応プロトコルバージョン V2

サーバーからクライアントへ
 アクセスログを通知



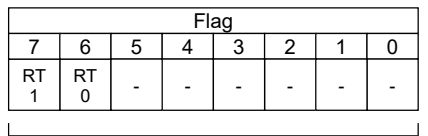
#	Data	Length	Description
6	Reserved	4byte	予約領域
5	Block Total	2byte BE UInt	総ブロック数
4	Block Index	8byte AsciiCode	ブロック番号
3	Request ID	2byte BE UInt	リクエストID (サーバーからの 一方的通知の場合には、0x0)
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"ACLOG_"固定

単一ログ情報



単一ログ情報

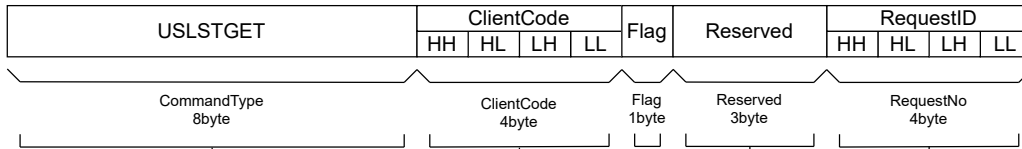
最大10ログ



#	Data	Length	Description						
13	Short MyCallsign	4byte AsciiCode	MY CALLSIGN(SHORT)						
12	MyCallsign	8byte AsciiCode	MY CALLSIGN						
11	Your Callsign	8byte AsciiCode	YOUR CALLSIGN						
10	Timestamp	8byte BE UInt	Unix Timestamp UTC +0:00						
9	Reserved	2byte	予約領域						
8	Log No	1byte UInt	ログ番号 1:最新ログ~n:最も古いログ						
7	Flag	1byte Binary	7	RT1 ROUTE1	00	Local	Local		
			6	RT0 ROUTE0	01	Local	GW		
			5	Reserved	10	GW	Local		
			4		Reserved	11	-	-	
			3			Reserved			
			2				Reserved		
			1					Reserved	
			0		Reserved				

23. USLSTGET

■クライアント → サーバー
 □サーバー → クライアント
 対応プロトコルバージョン V2



クライアントからサーバーへ
 ユーザーリストデータを要求

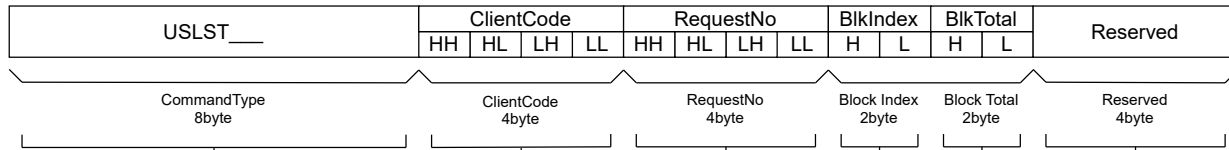
Flag							
7	6	5	4	3	2	1	0
Local	GW	-	-	-	-	-	-

#	Data	Length	Description		
5	RequestID	4byte BE UInt	リクエストID(自由に設定可) ACLOG返答時に同じコードが使われる		
4	Reserved	3byte	予約領域		
3	Flag	1byte Binary	7	Local	Include Local User 1:Yes/0:No
			6	Remote	Include Remote User 1:Yes/0:No
			5	Reserved	Reserved
			4		Reserved
			3		Reserved
			2		Reserved
			1		Reserved
			0		Reserved
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの		
1	Command Type	8byte AsciiCode	"USLSTGET"固定		

24. USLST__

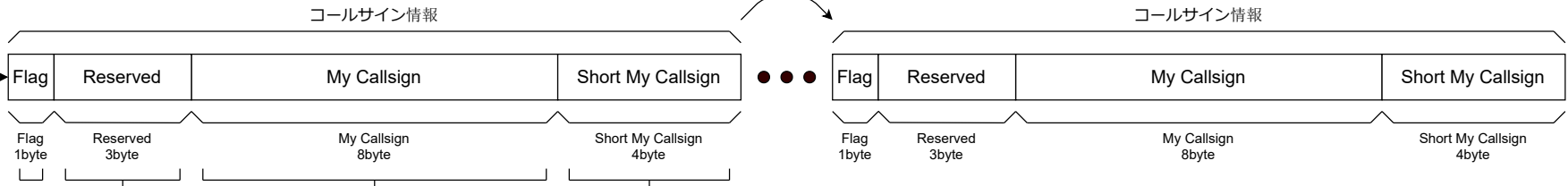
□クライアント → サーバー
 ■サーバー → クライアント
 対応プロトコルバージョン V2

サーバーからクライアントへ
 ユーザーリストを通知



#	Data	Length	Description
6	Reserved	4byte	予約領域
5	Block Total	2byte BE UInt	総ブロック数
4	Block Index	8byte AsciiCode	ブロック番号
3	Request No	2byte BE UInt	リクエスト番号 (サーバーからの 一方的通知の場合には、0x0)
2	ClientCode	2byte BE UInt	クライアントコード LOGINACKにて取得したもの
1	Command Type	8byte AsciiCode	"USLST__"固定

最大10コールサイン



#	Data	Length	Description
10	Short MyCallsign	4byte AsciiCode	MY CALLSIGN(SHORT)
9	MyCallsign	8byte AsciiCode	MY CALLSIGN
8	Reserved	3byte	予約領域
7	Flag	1byte Binary	7 Local Local User 1:Yes/0:No
			6 GW Remote User 1:Yes/0:No
			5 Reserved
			4 Reserved
			3 Reserved
			2 Reserved
			1 Reserved
0	LOGIN/OUT	1:LOGIN 0:LOGOUT	

Flag							
7	6	5	4	3	2	1	0
Local	Rmt	-	-	-	-	-	LGIN OUT

31

余白

5. 付録

①パケット通信例

1. ログイン

No	送信方向	パケットデータ
	送信情報	
1	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">←</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 44 4C 47 49 4E 55 53 52 32 NRVR DLGINUSR2 4A 49 31 52 4F 4A 20 43 02 00 00 00 4E 6F 72 61 JI1ROJ C Nora 56 52 52 65 6D 6F 74 65 43 6C 69 65 6E 74 00 00 VRRemoteClient 00 00 00 00 00 00 00 00 00 00 00 00 30 2E 30 2E 0.0. 32 61 00 00 00 00 00 00 00 00 00 00 00 2a </pre>
	ログインコールサイン JI1ROJ C 希望プロトコルバージョン 2 アプリ情報 NoraVRRemoteClient 0.0.2a	
2	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">→</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 0C 4C 4F 47 49 4E 5F 43 43 NRVR LOGIN_CC D3 A2 96 9B </pre>
	ログインチャレンジコード 0xD3A2969B	
3	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">←</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 28 4C 4F 47 49 4E 5F 48 53 NRVR (LOGIN_HS 4A 8B 08 01 15 A5 3A A9 EA 4E CF 1A C2 8F AB 8F J : N 6D C4 58 FB 0D 47 2F 8C 0A 68 AC 4D A2 B6 F2 7A m X G/ h M z </pre>
	ハッシュコード 7ad2b6a24d...1501088b4a	
4	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">→</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 20 4C 4F 47 49 4E 41 43 4B NRVR LOGINACK 02 C7 23 45 00 80 02 00 4A 4A 30 54 50 58 20 47 #E JJ0TPX G 4A 4A 30 54 50 58 20 42 JJ0TPX B </pre>
	クライアントコード 0x02c72345 サーバー設定 0x0080 プロトコルバージョン 2 ゲートウェイコールサイン JJ0TPX G レピータコールサイン JJ0TPX B	
5	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">←</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 10 43 4F 4E 46 53 45 54 5F NRVR CONFSET_ 02 C7 23 45 07 80 00 00 #E </pre>
	クライアントコード 0x02c72345 サーバー設定 0x0780	
6	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">→</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 08 41 43 4B 5F 5F 5F 5F 5F NRVR ACK_____ </pre>

2. ログアウト

1	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">←</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 0C 4C 4F 47 4F 55 54 5F 5F NRVR LOGOUT__ 02 C7 23 45 #E </pre>
	クライアントコード 0x02c72345	
2	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">→</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 08 41 43 4B 5F 5F 5F 5F 5F NRVR ACK_____ </pre>

3. 接続維持

1	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">←</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 0C 50 49 4E 47 5F 5F 5F 5F NRVR PING_____ 02 C7 23 45 #E </pre>
2	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">サーバー</div> <div style="font-size: 2em; margin: 0 10px;">→</div> <div style="border: 1px solid black; padding: 2px; margin-left: 5px;">クライアント</div> </div>	<pre> 4E 52 56 52 00 00 00 0C 50 4F 4E 47 5F 5F 5F 5F NRVR PONG_____ 02 C7 23 45 #E </pre>

余白

6. 改版履歴

2019.03.12	初版	初版発行
2020.02.02	Rev1	プロトコルバージョン2へ更新 以下のコマンドを追加 ○ログインコマンド(LGINUSR2) ○ルーティングサービス名称の取得・通知(RSRVGET/RSRV) ○レピータ(ノード)情報の取得(RINFOGET/RINFO) ○アクセスログの取得・通知(ACLOGGET/ACLOG) ○ユーザーリストの取得・通知(USLSTGET/USLST)

7. 関連情報(引用元)

1. JARL DSTAR仕様
アマチュア無線のデジタル化技術の標準方式
<http://www.jarl.com/d-star/shogen.htm>
2. OpusCodec
<http://opus-codec.org/>
3. The Format of D-Star Slow Data
By Jonathan Naylor, G4KLX
4. ircDDBGateway
<https://github.com/g4klx/ircDDBGateway>

8. 連絡先

JQ1ZYC 圏央道友会
kenoh_doyu@txb.sakura.ne.jp