

YVCAPI
機能仕様書

Version 1.04.01

2022-03-28

ヤマハ株式会社

Copyright© 2017-2022 Yamaha Corporation

改訂履歴

作成年月日	仕様書バージョン	内容
2017/08/10	1.00.00	初版
2017/12/05	1.01.00	・Mac 対応 ・API 関数 YVC_DLLInfo を YVC_APIInfo に名称変更 ・型定義 YVC_DLL_INFO を YVC_API_INFO に名称変更 ・定数定義 YVC_LEN_DLL_VERSION を YVC_LEN_API_VERSION に名称変更
2018/07/24	1.02.00	・対応機器に YVC-200 を追加
2019/10/17	1.03.00	・対応機器に YVC-330 を追加
2021/06/22	1.04.00	・対応機器に YVC-331 を追加
2022/03/28	1.04.01	・ソフトウェア環境 の動作対象 OS を変更

目次

1. 概要 ……4
1.1 適用範囲 ……4
1.2 対象機器 ……4
1.3 用語定義 ……4
1.4 参照文書 ……5
2. 動作環境 ……6
2.1 ソフトウェア環境 ……6
2.2 対応バージョンについて ……6
3. API の構成 ……7
3.1 ファイル構成 ……8
4. API 仕様 ……9
4.1 API 関数一覧 ……9
4.2 API 関数詳細 ……10
4.3 制御コマンド ……15
4.4 制御コマンド一覧 ……16
4.5 制御コマンド詳細 ……21
4.6 イベント詳細 ……88
4.7 定数・型定義 ……90
4.8 エラー定義 ……101
4.9 制限事項 ……102
4.10 注意事項 ……102
5. API 使用例 ……103
6. 付録 ……107
6.1 コマンドの対応モードと対応ファームウェア一覧 ……107
6.2 マイクカプセル位置 ……110
6.3 オーディオ・ブロック図 ……112

1. 概要

本仕様書は、PC から YVC シリーズの機器を制御するための API（Application Programming Interface）仕様について、記述したものです。
PC 上の Web 会議アプリ、会議用ソフト等から API を利用することで、YVC シリーズの機器の制御、およびパラメータの設定や動作状態の取得が可能になります。

1.1 適用範囲

本仕様書は、HID デバイスとして PC に認識された YVC シリーズの機器を、HID 通信により独自のコマンドで制御し、パラメータの設定や取得をするための API の機能仕様について記述します。



1.2 対象機器

- Yamaha YVC-1000
- Yamaha YVC-300
- Yamaha YVC-330
- Yamaha YVC-331
- Yamaha YVC-200

1.3 用語定義

本仕様書やヘッダーファイルで使用される用語について、説明します。

用語	説明
HID	Human Interface Device の略称を示す
NORMAL	通常の起動モード（オーディオ機能あり）を示す
FWUP	ファームウェアを更新するためのモード（オーディオ機能なし）を示す
AGC	Automatic Gain Controller の略称を示す
AAT	Automatic Audio Tuning の略称を示す （自動音響調整）
BT	Bluetooth の略称を示す
SP	Speaker の略称を示す
MIC	Microphone の略称を示す
EQ	Equalizer の略称を示す
CH	Channel の略称を示す
FW	Firmware の略称を示す

INFO	Information の略称を示す
CMD	Command の略称を示す
ARG	Argument の略称を示す
PRM	Parameter の略称を示す

1.4 参照文書

各機器の機能の詳細については、以下の文書を参照してください。

- ユニファイドコミュニケーション マイクスピーカーシステム YVC-1000 取扱説明書
- YVC-300 ユニファイドコミュニケーション スピーカーフォン 取扱説明書
- YVC-330 ユニファイドコミュニケーション スピーカーフォン 取扱説明書
- YVC-331 ユニファイドコミュニケーション スピーカーフォン 取扱説明書
- YVC-200 ユニファイドコミュニケーション スピーカーフォン 取扱説明書

2. 動作環境

以下に本 API の動作環境について記述します。

2.1 ソフトウェア環境

本 API は以下の OS 上での動作を対象とします。

- **Windows**
  - Microsoft Windows 10 (32bit/64bit)
  - Microsoft Windows 11 (64bit)
- **Mac**
  - macOS 10.15
  - macOS 11
  - macOS 12

※OS 上で、HID デバイスとして正常に認識・動作することを前提とします。

2.2 対応バージョンについて

本 API のバージョンに対応する YVC シリーズの機器の FW バージョンを以下に記述します。

- **Windows**

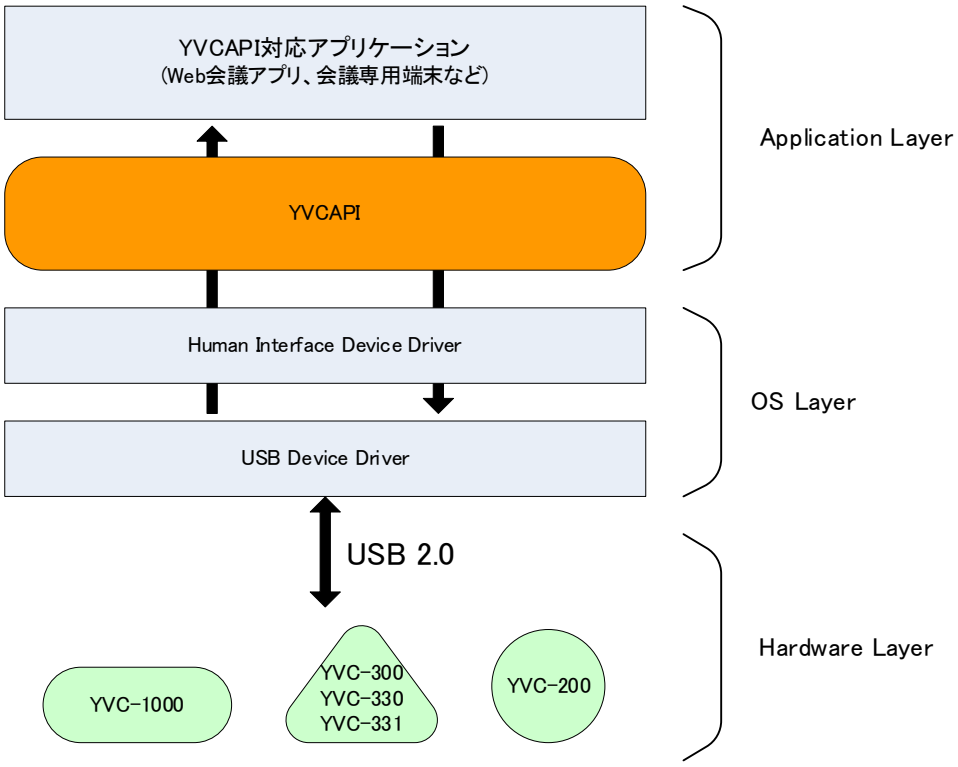
API のバージョン	YVC-1000 のバージョン	YVC-300 のバージョン	YVC-330 のバージョン	YVC-331 のバージョン	YVC-200 のバージョン
Ver. 1.00	Ver. 2.00 以降	Ver. 1.04 以降	-	-	-
Ver. 1.01	↑	↑	-	-	-
Ver. 1.02	↑	↑	-	-	Ver. 1.01 以降
Ver. 1.03	↑	↑	Ver. 1.02 以降	-	↑
Ver. 1.04	↑	↑	↑	Ver. 1.00 以降	↑

- **Mac**

API のバージョン	YVC-1000 のバージョン	YVC-300 のバージョン	YVC-330 のバージョン	YVC-331 のバージョン	YVC-200 のバージョン
Ver. 1.01	Ver. 2.00 以降	Ver. 1.04 以降	-	-	-
Ver. 1.02	↑	↑	-	-	Ver. 1.01 以降
Ver. 1.03	↑	↑	Ver. 1.02 以降	-	↑
Ver. 1.04	↑	↑	↑	Ver. 1.00 以降	↑

3. API の構成

本 API の構成について記述します。



3.1 ファイル構成

YVC シリーズを制御するための API の本体となるバイナリファイル、およびそのインターフェースを定義したヘッダーファイルで構成されます。

- **Windows 版**
  - YVCAPI.dll : DLL 本体
  - YVCAPI.h : ヘッダーファイル

- **Mac 版**
  - YVCAPI.framework : フレームワーク本体
  - YVCAPI.h : ヘッダーファイル

4. API 仕様

本 API の仕様について記述します。

4.1 API 関数一覧

API 関数の一覧を示します。

関数	内容
YVC_DeviceInfo	YVC シリーズの機器の接続情報を取得します。
YVC_DeviceAttach	YVC シリーズの機器と HID 経由で通信するためのハンドルを取得します。
YVC_DeviceControl	YVC シリーズの機器と HID 経由で通信し、機器の制御、およびパラメータの設定・取得します。
YVC_DeviceDetach	YVC シリーズの機器と HID 経由で通信するためのハンドルを解放します。
YVC_APIInfo	本 API のバージョン情報を取得します。

4.2 API 関数詳細

API 関数の詳細について記述します。

4.2-1 YVC_DeviceInfo

【書式】

```
int __stdcall YVC_DeviceInfo(
        YVC_DEV_TYPE        type,
        YVC_DEV_INFO        *info
);
```

【動作概要】

ホスト PC に接続されている YVC シリーズの機器の台数を取得します。
接続情報には、*type* で指定された機種の接続台数を *num* に返します。

【制限事項】
同一機種での 2 台以上の接続の判定はおこないません。従って、必ず接続台数の値は 1 台以下となります。また、NORMAL モードと FWUP モードで動作している状態では、NORMAL モードが優先されます。
なお、動作モードの詳細については、YVC_CMD_GetSystemMode コマンドの頁を参照してください。

【パラメータ】

- **[IN] YVC_DEV_TYPE type**

YVC シリーズの機種コードを指定します。

パラメータ定義名	説明
YVC_DEV_TYPE_YVC1000	YVC-1000
YVC_DEV_TYPE_YVC300	YVC-300
YVC_DEV_TYPE_YVC200	YVC-200
YVC_DEV_TYPE_YVC330	YVC-330
YVC_DEV_TYPE_YVC331	YVC-331

- **[OUT] YVC_DEV_INFO *info**

接続情報を格納するためのバッファのポインタを指定します。

【戻り値】

エラーコード	内容
YVCAPI_ERROR_NOERROR	正常終了
YVCAPI_ERROR_INVALIDARGS	引数が不正

4.2-2 YVC_DeviceAttach

【書式】

```
int __stdcall YVC_DeviceAttach(
        YVC_DEV_HANDLE          *handle,
        YVC_DEV_TYPE             type,
        YVC_EVENT_CALLBACK       callback,
        void                     *user
);
```

【動作概要】

YVC シリーズの機器と HID 経由で通信をするためのハンドルを取得します。

【パラメータ】

- **[OUT] YVC_DEV_HANDLE *handle**

  取得したハンドルを格納する変数へのポインタを指定します。

- **[IN] YVC_DEV_TYPE type**

  YVC シリーズの機種タイプを指定します。

パラメータ定義名	説明
YVC_DEV_TYPE_YVC1000	YVC-1000
YVC_DEV_TYPE_YVC300	YVC-300
YVC_DEV_TYPE_YVC200	YVC-200
YVC_DEV_TYPE_YVC330	YVC-330
YVC_DEV_TYPE_YVC331	YVC-331

- **[IN] YVC_EVENT_CALLBACK callback**

  イベントが発生したときに通知先となるコールバック関数を指定します。使用しない場合は NULL を指定することができます。また、コールバック関数を指定した場合でも、YVC_CMD_SetNoticeEvent コマンドが未対応のファームウェアのバージョンで動作しているときにはイベントは通知されません。(YVC_EVENT_DEV_DISCONNECT イベントを除く)
  詳細については、「イベント詳細」の頁を参照してください。

  【注意事項】
  本 API でハンドルを取得後に USB ケーブルを抜かれるなどして機器が取り外された場合、YVC_DeviceAttach 関数で取得したハンドルを即座に(次に USB 接続されるまでに)解放する必要があります。
  機器の接続状態は、YVC_DeviceInfo 関数を定期的に呼び出して接続情報を監視することでも可能ですが、本パラメータでコールバック関数を指定しておくことで、YVC_EVENT_DEV_DISCONNECT イベント通知を受け取ることができます。そのイベントを受け取ったタイミングで取得したハンドルを解放する方法を推奨します。
  なお、コールバック関数内では、時間を要する処理や本 API 関数を実行しないようにしてください。

- **[IN] void *user**

  ユーザー管理用データのアドレスを指定します。不要な場合は NULL を設定してください。

【戻り値】

エラーコード	内容
YVCAPI_ERROR_NOERROR	正常終了
YVCAPI_ERROR_DEVNOTFOUND	指定された機器が見つからない
YVCAPI_ERROR_MEMEXHAUST	メモリが不足している
YVCAPI_ERROR_STATUS	状態エラーが発生した
YVCAPI_ERROR_INVALIDARGS	引数が不正
YVCAPI_ERROR_UNSUPPORTVER	未サポートのファームウェアバージョン

4.2-3 YVC_DeviceDetach

【書式】

```
int __stdcall YVC_DeviceDetach(
        YVC_DEV_HANDLE    handle
);
```

【動作概要】

YVC シリーズの機器と HID 経由で通信するためのハンドルを解放します。

【パラメータ】

- **[IN]　YVC_DEV_HANDLE　handle**

  YVC_DeviceAttach 関数で取得したハンドルを指定します。

【戻り値】

エラーコード	内容
YVCAPI_ERROR_NOERROR	正常終了
YVCAPI_ERROR_INVALIDHANDLE	ハンドルが不正
YVCAPI_ERROR_STATUS	状態エラーが発生した

4.2-4 YVC_DeviceControl

【書式】

```
int __stdcall YVC_DeviceControl(
        YVC_DEV_HANDLE      handle,
        YVC_CTRL_CMD          cmd,
        YVC_CTRL_ARG         *arg
);
```

【動作概要】

YVC シリーズの端末と HID 経由で通信し、機器の制御、およびパラメータの設定や取得をおこないます。

【パラメータ】

- **[IN] YVC_DEV_HANDLE handle**

  YVC_DeviceAttach 関数で取得したハンドルを指定します。

- **[IN] YVC_CTRL_CMD cmd**

  制御コマンドのコードを指定します。コマンドコードについては「制御コマンド一覧」を参照してください。

- **[IN/OUT] YVC_CTRL_ARG *arg**

  制御コマンドのパラメータを格納したバッファへのポインタを指定します。

【戻り値】

エラーコード	内容
YVCAPI_ERROR_NOERROR	正常終了
YVCAPI_ERROR_PENDING	要求受付（実行中）
YVCAPI_ERROR_DEVNOTFOUND	指定された機器が見つからない
YVCAPI_ERROR_MEMEXHAUST	メモリが不足している
YVCAPI_ERROR_STATUS	状態エラーが発生した
YVCAPI_ERROR_INVALIDCOMMAND	制御コマンドのコードが不正
YVCAPI_ERROR_INVALIDHANDLE	ハンドルが不正
YVCAPI_ERROR_INVALIDARGS	引数が不正
YVCAPI_ERROR_CTRLEXEC	制御の実行に失敗した
YVCAPI_ERROR_HIDTIMEOUT	HID通信において、タイムアウトが発生した
YVCAPI_ERROR_UNSUPPORTVER	未サポートのファームウェアバージョン
YVCAPI_ERROR_APPRUNNING	他の関連アプリケーションが起動中のため、実行できない
YVCAPI_ERROR_INVALIDPARAM	機器のパラメータ値が不正

4.2-5 YVC_APIInfo

【書式】

```
int __stdcall YVC_APIInfo(
        YVC_API_INFO    *info
);
```

【動作概要】

本 API のバージョン情報を取得します。
Version にバージョン番号が ASCII 文字列で格納されます。文字列の長さは 4 文字固定です。

例）Ver. 1.00 の場合

0	1	2	3	4
0x31	0x2E	0x30	0x30	0x00

【パラメータ】

- **[IN]** **YVC_API_INFO** ***info**

  API 情報を格納するためのバッファのポインタを指定します。

【戻り値】

エラーコード	内容
YVCAPI_ERROR_NOERROR	正常終了
YVCAPI_ERROR_INVALIDARGS	引数が不正

4.3 制御コマンド

制御コマンドには、設定型、取得型、実行型、および監視型の4つのタイプがあります。
以下にその概要を記述します。

コマンドタイプ	コマンド名定義	概要
取得型コマンド	YVC_CMD_**Get**<コマンド名>	機器の情報、状態、パラメータなどを取得します。
設定型コマンド	YVC_CMD_**Set**<コマンド名>	機器の動作、パラメータなどを設定します。 パラメータには、不揮発性パラメータと揮発性パラメータがあります。
実行型コマンド	YVC_CMD_**Exe**<コマンド名>	機器の各種機能を実行します。 システムの再起動、スタンバイ設定、工場出荷設定、自動音響調整などがあります。
監視型コマンド	YVC_CMD_**Mon**<コマンド名>	機器の動作状態を取得します。 このコマンドは、連続で発行するときの発行間隔に制限があります。 間隔時間については、各コマンドの頁を参照してください。

4.4 制御コマンド一覧

制御コマンドの一覧を記述します。

※表内の各欄の記号や記述ついて説明します。

「✔」　：コマンドサポートを示します
「X」　：コマンド非サポートを示します
「不揮発」　：設定値が電源 OFF 状態でも記憶されていることを示します
「揮発」　：設定値がシステムの再起動や電源 OFF により初期設定に戻ります
「なし」　：Set 系のコマンドで、設定されるパラメータが存在しないことを示します
「ー」　：Get/Exe/Mon 系コマンドのため、設定されるパラメータが存在しないことを示します

4.4-1 タイプ別一覧

1）取得型

コマンド定義名	概要	設定値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_CMD_GetSystemMode	システムの動作モードを取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetVersionInfo	バージョン情報を取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetModelName	モデル名称を取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetSerialNumber	シリアル No.を取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetMicNum	マイクユニットの接続台数を取得します	ー	✔	X	X	X	X
YVC_CMD_GetNoticeEvent	通知するイベントの設定を取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetBTUse	Bluetooth 機能の有効／無効の設定を取得します	ー	✔	✔	✔	X	✔
YVC_CMD_GetBTLocalName	Bluetooth の Local Name の設定を取得します	ー	✔	✔	✔	X	✔
YVC_CMD_GetBTOperation	Bluetooth 機能の状態を取得します	ー	✔	✔	✔	X	✔
YVC_CMD_GetNoticeSoundUse	通知音の有効／無効の設定を取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetNoticeSoundVolume	通知音の音量の設定を取得します	ー	X	✔	✔	✔	✔
YVC_CMD_GetVoiceGuideUse	音声ガイダンスの有効／無効の設定を取得します	ー	✔	X	X	X	X
YVC_CMD_GetVoiceGuideLang	音声ガイダンスの言語の設定を取得します	ー	✔	X	X	X	X
YVC_CMD_GetVoiceGuideVolume	音声ガイダンスの音量の設定を取得します	ー	✔	X	X	X	X
YVC_CMD_GetInitSpVolume	USB 接続時の再生音量の設定を取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetPowerOnState	電源投入時の動作の設定を取得します	ー	✔	X	X	X	X
YVC_CMD_GetMicMuteMode	マイクミュートの動作モードの設定を取得します	ー	✔	X	X	X	X
YVC_CMD_GetUsbSpeed	USB ポートの動作スピードの設定を取得します	ー	✔	X	X	X	X
YVC_CMD_GetExtTermMode	EXT 端子の接続モードの設定を取得します	ー	X	✔	✔	✔	X
YVC_CMD_GetSpOutSelect	スピーカー出力選択の設定を取得します	ー	✔	X	X	X	X
YVC_CMD_GetAudioInTerm	オーディオ入力端子の設定を取得します	ー	✔	X	X	X	X
YVC_CMD_GetAudioVolume	音量の設定を取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetAudioMute	ミュートの設定を取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetMicAgc	マイク AGC の設定を取得します	ー	✔	✔	✔	✔	✔
YVC_CMD_GetMicUnitTrack	マイクユニットの収音モードの設定を取得します	ー	✔	X	X	X	X
YVC_CMD_GetMicCapsuleTrack	マイクカプセルの収音モードの設定を取得します	ー	X	✔	✔	✔	X

YVC_CMD_GetMicFilter	マイクの周波数特性の設定を取得します	–	✔	✔	✔	✔	✔
YVC_CMD_GetSpFilter	スピーカーの周波数特性の設定を取得します	–	✔	✔	✔	✔	X
YVC_CMD_GetAATProgress	自動音響調整の進捗状況を取得します	–	✔	X	X	X	X
YVC_CMD_GetAATResult	自動音響調整の測定結果を取得します	–	✔	X	X	X	X

2）設定型

コマンド定義名	概要	設定値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_CMD_SetNoticeEvent	通知するイベントの設定をします	揮発	✔	✔	✔	✔	✔
YVC_CMD_SetBTUse	Bluetooth 機能を有効／無効にします	不揮発	✔	✔	✔	X	✔
YVC_CMD_SetBTLocalName	Bluetooth の Local Name を設定します	不揮発	✔	✔	✔	X	✔
YVC_CMD_SetBTOperation	Bluetooth 機能の操作をします	なし	✔	✔	✔	X	✔
YVC_CMD_SetNoticeSoundUse	通知音を有効／無効にします	不揮発	✔	✔	✔	✔	✔
YVC_CMD_SetNoticeSoundVolume	通知音の音量を調整します	不揮発	X	✔	✔	✔	✔
YVC_CMD_SetVoiceGuideUse	音声ガイダンスを有効／無効にします	不揮発	✔	X	X	X	X
YVC_CMD_SetVoiceGuideLang	音声ガイダンスの言語を変更します	不揮発	✔	X	X	X	X
YVC_CMD_SetVoiceGuideVolume	音声ガイダンスの音量を調整します	不揮発	✔	X	X	X	X
YVC_CMD_SetInitSpVolume	USB 接続時の再生音量を設定します	不揮発	✔	✔	✔	✔	✔
YVC_CMD_SetPowerOnState	電源投入時の動作を設定します	不揮発	✔	X	X	X	X
YVC_CMD_SetMicMuteMode	マイクミュートの動作モードの設定をします	不揮発	✔	X	X	X	X
YVC_CMD_SetUsbSpeed	USB ポートの動作スピードを設定します	不揮発	✔	X	X	X	X
YVC_CMD_SetExtTermMode	EXT 端子の接続モードの設定を変更します	不揮発	X	✔	✔	✔	X
YVC_CMD_SetSpOutSelect	スピーカー出力選択を設定します	不揮発	✔	X	X	X	X
YVC_CMD_SetAudioInTerm	オーディオ入力端子の設定を変更します	不揮発	✔	X	X	X	X
YVC_CMD_SetAudioVolume	音量の設定を調整します	不揮発(注)	✔	✔	✔	✔	✔
YVC_CMD_SetAudioMute	ミュートの設定を変更します	不揮発(注)	✔	✔	✔	✔	✔
YVC_CMD_SetMicAgc	マイク AGC の設定を変更します	不揮発	✔	✔	✔	✔	✔
YVC_CMD_SetMicUnitTrack	マイクユニットの収音モードを変更します	不揮発	✔	X	X	X	X
YVC_CMD_SetMicCapsuleTrack	マイクカプセルの収音モードを変更します	不揮発	X	✔	✔	✔	X
YVC_CMD_SetMicFilter	マイクの周波数特性を変更します	不揮発	✔	✔	✔	✔	✔
YVC_CMD_SetSpFilter	スピーカーの周波数特性を変更します	不揮発	✔	✔	✔	✔	X

(注) 一部、揮発パラメータが含まれます。

3）実行型

コマンド定義名	概要	設定値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_CMD_ExeSystemReboot	システムを再起動します	–	✔	✔	✔	✔	✔
YVC_CMD_ExeFactoryReset	パラメータを工場出荷時設定に戻します	–	✔	✔	✔	✔	✔
YVC_CMD_ExeStandbyMode	スタンバイ状態にします	–	✔	X	X	X	✔
YVC_CMD_ExeAAT	自動音響調整機能の実行を制御します	–	✔	X	X	X	X

4）監視型

コマンド定義名	概要	設定値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_CMD_MonMicCapsule	アクティブになっているマイクカプセルを取得します	ー	✓	✓	✓	✓	X
YVC_CMD_MonMicUnit	アクティブになっているマイクユニット ID を取得します	ー	✓	X	X	X	X
YVC_CMD_MonHeadsetPortState	ヘッドセット端子の接続状態を取得します	ー	X	X	X	X	✔
YVC_CMD_MonBatteryLevel	バッテリー残量を取得します。	ー	X	X	X	X	✔

4.4-2 機能別一覧

1）システム制御関連

コマンド定義名	概要	設定値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_CMD_GetSystemMode	システムの動作モードを取得します	―	✔	✔	✔	✔	✔
YVC_CMD_GetVersionInfo	バージョン情報を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_GetModelName	モデル名称を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_GetSerialNumber	シリアル No.を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_GetMicNum	マイクユニットの接続台数を取得します	―	✔	X	X	X	X
YVC_CMD_SetNoticeEvent	通知するイベントの設定をします	揮発	✔	✔	✔	✔	✔
YVC_CMD_GetNoticeEvent	通知するイベントの設定を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_ExeSystemReboot	システムを再起動します	―	✔	✔	✔	✔	✔
YVC_CMD_ExeFactoryReset	パラメータを工場出荷時設定に戻します	―	✔	✔	✔	✔	✔
YVC_CMD_ExeStandbyMode	スタンバイ状態にします	―	✔	X	X	X	✔
YVC_CMD_SetBTUse	Bluetooth 機能を有効／無効にします	不揮発	✔	✔	✔	X	✔
YVC_CMD_GetBTUse	Bluetooth 機能の有効／無効の設定を取得します	―	✔	✔	✔	X	✔
YVC_CMD_SetBTLocalName	Bluetooth の Local Name を設定します	不揮発	✔	✔	✔	X	✔
YVC_CMD_GetBTLocalName	Bluetooth の Local Name の設定を取得します	―	✔	✔	✔	X	✔
YVC_CMD_SetBTOperation	Bluetooth 機能の操作をします	なし	✔	✔	✔	X	✔
YVC_CMD_GetBTOperation	Bluetooth 機能の状態を取得します	―	✔	✔	✔	X	✔
YVC_CMD_SetNoticeSoundUse	通知音を有効／無効にします	不揮発	✔	✔	✔	✔	✔
YVC_CMD_GetNoticeSoundUse	通知音の有効／無効の設定を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_SetVoiceGuideUse	音声ガイダンスを有効／無効にします	不揮発	✔	X	X	X	X
YVC_CMD_GetVoiceGuideUse	音声ガイダンスの有効／無効の設定を取得します	―	✔	X	X	X	X
YVC_CMD_SetVoiceGuideLang	音声ガイダンスの言語を変更します	不揮発	✔	X	X	X	X
YVC_CMD_GetVoiceGuideLang	音声ガイダンスの言語の設定を取得します	―	✔	X	X	X	X
YVC_CMD_SetVoiceGuideVolume	音声ガイダンスの音量を調整します	不揮発	✔	X	X	X	X
YVC_CMD_GetVoiceGuideVolume	音声ガイダンスの音量の設定を取得します	―	✔	X	X	X	X
YVC_CMD_SetNoticeSoundVolume	通知音の音量を調整します	不揮発	X	✔	✔	✔	✔
YVC_CMD_GetNoticeSoundVolume	通知音の音量の設定を取得します	―	X	✔	✔	✔	✔
YVC_CMD_SetInitSpVolume	USB 接続時の再生音量を設定します	不揮発	✔	✔	✔	✔	✔
YVC_CMD_GetInitSpVolume	USB 接続時の再生音量の設定を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_SetPowerOnState	電源投入時の動作を設定します	不揮発	✔	X	X	X	X
YVC_CMD_GetPowerOnState	電源投入時の動作の設定を取得します	―	✔	X	X	X	X
YVC_CMD_SetMicMuteMode	マイクミュートの動作モードの設定をします	不揮発	✔	X	X	X	X
YVC_CMD_GetMicMuteMode	マイクミュートの動作モードの設定を取得します	―	✔	X	X	X	X
YVC_CMD_SetUsbSpeed	USB ポートの動作スピードを設定します	不揮発	✔	X	X	X	X
YVC_CMD_GetUsbSpeed	USB ポートの動作スピードの設定を取得します	―	✔	X	X	X	X
YVC_CMD_SetExtTermMode	EXT 端子の接続モードの設定を変更します	不揮発	X	✔	✔	✔	X
YVC_CMD_GetExtTermMode	EXT 端子の接続モードの設定を取得します	―	X	✔	✔	✔	X

2）オーディオ制御関連

コマンド定義名	概要	設定値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_CMD_SetSpOutSelect	スピーカー出力選択を設定します	不揮発	✔	X	X	X	X
YVC_CMD_GetSpOutSelect	スピーカー出力選択の設定を取得します	―	✔	X	X	X	X
YVC_CMD_SetAudioInTerm	オーディオ入力端子の設定を変更します	不揮発	✔	X	X	X	X
YVC_CMD_GetAudioInTerm	オーディオ入力端子の設定を取得します	―	✔	X	X	X	X
YVC_CMD_SetAudioVolume	音量の設定を調整します	不揮発(注)	✔	✔	✔	✔	✔
YVC_CMD_GetAudioVolume	音量の設定を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_SetAudioMute	ミュートの設定を変更します	不揮発(注)	✔	✔	✔	✔	✔
YVC_CMD_GetAudioMute	ミュートの設定を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_SetMicAgc	マイク AGC の設定を変更します	不揮発	✔	✔	✔	✔	✔
YVC_CMD_GetMicAgc	マイク AGC の設定を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_SetMicUnitTrack	マイクユニットの収音モードを変更します	不揮発	✔	X	X	X	X
YVC_CMD_GetMicUnitTrack	マイクユニットの収音モードの設定を取得します	―	✔	X	X	X	X
YVC_CMD_SetMicCapsuleTrack	マイクカプセルの収音モードを変更します	不揮発	X	✔	✔	✔	X
YVC_CMD_GetMicCapsuleTrack	マイクカプセルの収音モードの設定を取得します	―	X	✔	✔	✔	X
YVC_CMD_SetMicFilter	マイクの周波数特性を変更します	不揮発	✔	✔	✔	✔	✔
YVC_CMD_GetMicFilter	マイクの周波数特性の設定を取得します	―	✔	✔	✔	✔	✔
YVC_CMD_SetSpFilter	スピーカーの周波数特性を変更します	不揮発	✔	✔	✔	✔	X
YVC_CMD_GetSpFilter	スピーカーの周波数特性の設定を取得します	―	✔	✔	✔	✔	X

(注) 一部、揮発パラメータが含まれます。

3）自動音響調整制御関連

コマンド定義名	概要	設定値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_CMD_ExeAAT	自動音響調整機能の実行を制御します	―	✔	X	X	X	X
YVC_CMD_GetAATProgress	自動音響調整の進捗状況を取得します	―	✔	X	X	X	X
YVC_CMD_GetAATResult	自動音響調整の測定結果を取得します	―	✔	X	X	X	X

4）機器のモニタリング関連

コマンド定義名	概要	設定値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_CMD_MonMicCapsule	アクティブになっているマイクカプセルを取得します	―	✔	✔	✔	✔	X
YVC_CMD_MonMicUnit	アクティブになっているマイクユニット ID を取得します	―	✔	X	X	X	X
YVC_CMD_MonHeadsetPortState	ヘッドセット端子の接続状態を取得します	―	X	X	X	X	✔
YVC_CMD_MonBatteryLevel	バッテリー残量を取得します。	―	X	X	X	X	✔

4.5 制御コマンド詳細

4.5-1 YVC_CMD_GetSystemMode

【動作概要】

システムの動作モードを取得します。

動作モードには、NORMAL モードと FWUP モードの2つの動作モードがあります。
NORMAL モードは、マイクスピーカーシステム（スピーカーフォン）として起動するモードです。通常は、このモードで起動されます。
一方、FWUP モードは、機器のファームウェアを更新するときに使用するモードです。通常、このモードで起動されることはありません。
ファームウェアの更新を実行するときやシステムの問題により NORMAL モードで起動できないときに起動されます。

【書式】

```
typedef    struct {
        YVC_SYSTEM_MODE     mode;
} YVC_ARG_SystemMode;
```

【パラメータ】

- **[OUT]　YVC_SYSTEM_MODE　mode**

システムの動作モードを返します。

パラメータ定義名	説明
YVC_SYSTEM_MODE_FWUP	FWUP モードで動作しています
YVC_SYSTEM_MODE_NORMAL	NORMAL モードで動作しています

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	✔
YVC-300	Ver. 1.04 以降	✔	✔
YVC-330	Ver. 1.02 以降	✔	✔
YVC-331	Ver. 1.00 以降	✔	✔
YVC-200	Ver. 1.01 以降	✔	✔

4.5-2 YVC_CMD_GetVersionInfo

【動作概要】

バージョン情報を取得します。
成功した場合、*version* にバージョン番号が ASCII 文字列で格納されます。文字列の長さは 4 文字固定です。

例） Ver. 1.06 の場合

0	1	2	3	4
0x31	0x2E	0x30	0x36	0x00

【書式】

```
typedef    struct {
      char       version[YVC_LEN_VERSION_INFO+1] ;
} YVC_ARG_VersionInfo;
```

【パラメータ】

- **[OUT]　char　version[YVC_LEN_VERSION_INFO+1]**

バージョン情報を返します。
※終端文字(0x00)が含まれます。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	✔
YVC-300	Ver. 1.04 以降	✔	✔
YVC-330	Ver. 1.02 以降	✔	✔
YVC-331	Ver. 1.00 以降	✔	✔
YVC-200	Ver. 1.01 以降	✔	✔

4.5-3 YVC_CMD_GetModelName

【動作概要】

モデル名称を取得します。
成功した場合、*name* にモデル名称がASCII文字列で格納されます。文字列の長さは最大40文字で定義されていますが、実際に返す文字列は以下の通りです。

機種名	NORMAL モード	FWUP モード
YVC-1000	*“YVC-1000”*	*“YVC-1000 Updater”*
YVC-300	*“YVC-300”*	*“YVC-300 Updater”*
YVC-330	*“YVC-330”*	*“YVC-330 Updater”*
YVC-331	*“YVC-331”*	*“YVC-331 Updater”*
YVC-200	*“YVC-200”*	*“YVC-200 Updater”*

【書式】

```
typedef    struct {
      char       name[YVC_LEN_MODEL_NAME+1];
} YVC_ARG_ModelName;
```

【パラメータ】

- **[OUT] char name[YVC_LEN_MODEL_NAME+1]**

モデル名称を返します。
※終端文字(0x00)が含まれます。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	✔
YVC-300	Ver. 1.04 以降	✔	✔
YVC-330	Ver. 1.02 以降	✔	✔
YVC-331	Ver. 1.00 以降	✔	✔
YVC-200	Ver. 1.01 以降	✔	✔

4.5-4 YVC_CMD_GetSerialNumber

【動作概要】

シリアル No.を取得します。
成功した場合、*serialNumber* にシリアル No.が ASCII 文字列で格納されます。文字列の長さは最大 16 文字です。

YVC-1000 の場合で、接続されていないマイクユニット ID を指定した場合は、エラー(YVCAPI_ERROR_CTRLEXEC)を返します。また、FWUP モード時はマイクユニットのシリアル No.の取得はできません。指定された場合はエラー
(YVCAPI_ERROR_INVALIDARGS) を返します。

【書式】

```
typedef    struct {
        YVC_SERIALNUMBER_TYPE    type;
        char        serialNumber[YVC_LEN_SERIAL_NUMBER+1];
} YVC_ARG_SerialNumber;
```

【パラメータ】

- **[IN]　YVC_SERIALNUMBER_TYPE　type**

YVC-1000 の場合は、取得するユニットを指定します。
YVC-300、YVC-330、YVC-331、YVC-200 の場合は、指定は不要です。(無視されます)

パラメータ定義名	説明
YVC_SERIALNUMBER_TYPE_SPUNIT	本体を選択します
YVC_SERIALNUMBER_TYPE_MICUNIT0	マイクユニット 0 を選択します
YVC_SERIALNUMBER_TYPE_MICUNIT1	マイクユニット 1 を選択します
YVC_SERIALNUMBER_TYPE_MICUNIT2	マイクユニット 2 を選択します
YVC_SERIALNUMBER_TYPE_MICUNIT3	マイクユニット 3 を選択します
YVC_SERIALNUMBER_TYPE_MICUNIT4	マイクユニット 4 を選択します

- **[OUT]　char　serialNumber[YVC_LEN_SERIAL_NUMBER+1]**

シリアル No.を返します。
※終端文字(0x00)が含まれます。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	✔
YVC-300	Ver. 1.04 以降	✔	✔
YVC-330	Ver. 1.02 以降	✔	✔
YVC-331	Ver. 1.00 以降	✔	✔
YVC-200	Ver. 1.01 以降	✔	✔

4.5-5 YVC_CMD_GetMicNum

【動作概要】

マイクユニットの接続台数を取得します。

【書式】

```
typedef    struct {
        unsigned int        micNum;
} YVC_ARG_MicNum;
```

【パラメータ】

- **[OUT]　unsigned int　micNum**

  マイクユニットの接続台数(0～5)を返します。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-6 YVC_CMD_ExeSystemReboot

【動作概要】

システムを再起動します。

【書式】

```
typedef    struct {
      void          (*callback)(YVC_CTRL_CMD cmd, int err, void *user);
      void          *user;
} YVC_ARG_SystemReboot;
```

【パラメータ】

- **[IN]　void (*callback)(YVC_CTRL_CMD cmd, int err, void *user)**

コマンド完了通知用のコールバック関数アドレスを指定します。
NULL を設定した場合は、完了型として動作します。
NULL 以外を設定した場合は、非完了型として動作します。
非完了型では受付が正常完了すると、YVCAPI_ERROR_PENDING を返します。その後、当該処理を実行して、結果を本コールバック関数で通知します。
以下、非完了型として動作したときのパラメータについて説明します。

Cmd は、YVC_CMD_ExeSystemReboot を返します。
Err は、エラーコードを返します。
User は、指定したアドレスをそのまま返します。

【注意事項】
コールバック関数内では、時間を要する処理や本 API 関数を実行しないようにしてください。

- **[IN]　void　*user**

ユーザー管理用データのアドレスを指定します。
非完了型で実行する場合のみ、本パラメータの設定が有効となります。不要な場合は NULL を設定してください。
完了型で実行する場合は、無視されます。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	✔
YVC-300	Ver. 1.04 以降	✔	✔
YVC-330	Ver. 1.02 以降	✔	✔
YVC-331	Ver. 1.00 以降	✔	✔
YVC-200	Ver. 1.01 以降	✔	✔

4.5-7 YVC_CMD_ExeFactoryReset

【動作概要】

各種パラメータを工場出荷時の設定に戻します。

【書式】

```
typedef    struct {
      void          (*callback)(YVC_CTRL_CMD cmd, int err, void *user);
      void          *user;
} YVC_ARG_FactoryReset;
```

【パラメータ】

- **[IN]　void (*callback)(YVC_CTRL_CMD cmd, int err, void *user)**

コマンド完了通知用のコールバック関数アドレスを指定します。
NULL を設定した場合は、完了型として動作します。
NULL 以外を設定した場合は、非完了型として動作します。
非完了型では受付が正常完了すると、YVCAPI_ERROR_PENDING を返します。その後、当該処理を実行して、結果を本コールバック関数で通知します。
以下、非完了型として動作したときのパラメータについて説明します。

Cmd は、YVC_CMD_ExeFactoryReset を返します。
Err は、エラーコードを返します。
User は、指定したアドレスをそのまま返します。

【注意事項】
コールバック関数内では、時間を要する処理や本 API 関数を実行しないようにしてください。

- **[IN]　void　*user**

ユーザー管理用データのアドレスを指定します。
非完了型で実行する場合のみ、本パラメータの設定が有効となります。不要な場合は NULL を設定してください。
完了型で実行する場合は、無視されます。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-8 YVC_CMD_ExeStandbyMode

【動作概要】

YVC-1000、YVC-200 をスタンバイ状態にします。
ただし、スタンバイを解除して復帰させるコマンドはありません。スタンバイを解除するには、YVC-1000 の場合は本体の電源ボタンを押して解除してください。YVC-200 の場合は本体の電源ボタンを長押するか、USB ケーブルを接続して解除してください。

【書式】

パラメータはありません。*Arg* パラメータには NULL を指定してください。

【パラメータ】

パラメータはありません。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-9 YVC_CMD_SetNoticeEvent

【動作概要】

通知するイベントの設定をします。
イベント通知機能を利用する場合、通知先を YVC_DeviceAttach 関数で事前に登録しておく必要があります。
詳細については、「イベント詳細」の頁を参照してください。

【注意事項】
イベント設定は揮発性パラメータです。機器を再起動するとリセットされますので、必要に応じて再設定する必要があります。

【書式】

```
typedef    struct {
        unsigned int event;
} YVC_ARG_NoticeEvent;
```

【パラメータ】

- **[IN]　unsigned int　event**

通知するイベントを指定します。複数のイベントを指定することが可能です。

パラメータ定義名	値	説明	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_PRM_NOTICE_EVENT_MIC_UNIT_NUM	0x1	マイクユニット接続状態に変化があったことを通知します	✔	X	X	X	X
YVC_PRM_NOTICE_EVENT_MIC_MUTE_STATE	0x2	マイクミュートの状態を通知します。 ただし、ミュートボタンが押されたときのみに限ります。コマンド設定などで状態が変化した場合は通知されません。	✔	✔	✔	✔	✔
YVC_PRM_NOTICE_EVENT_HOOK_BTN	0x4	フックボタンの押下状態に変化があったことを通知します。ただし、Bluetooth の着信応答、終話（通話切断）の機能として動作する場合には通知されません。	X	✔	✔	✔	X

初期値は、0x0 (イベントなし) に設定されています。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver.2.10 以降	✔	X
YVC-300	Ver.1.09 以降	✔	X
YVC-330	Ver.1.02 以降	✔	X
YVC-331	Ver.1.00 以降	✔	X
YVC-200	Ver.1.01 以降	✔	X

4.5-10 YVC_CMD_GetNoticeEvent

【動作概要】

通知するイベントの設定を取得します。

【書式】

```
typedef    struct {
        unsigned int event;
} YVC_ARG_NoticeEvent;
```

【パラメータ】

- **[OUT]　unsigned int　event**

通知するイベントの設定を返します。
詳細は、YVC_CMD_SetNoticeEvent の頁を参照してください。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver.2.10 以降	✔	X
YVC-300	Ver.1.09 以降	✔	X
YVC-330	Ver.1.02 以降	✔	X
YVC-331	Ver.1.00 以降	✔	X
YVC-200	Ver.1.01 以降	✔	X

4.5-11 YVC_CMD_SetBTUse

【動作概要】

Bluetooth 機能を有効／無効にします。

【書式】

```
typedef    struct {
        BOOL        onoff;
} YVC_ARG_BTUse;
```

【パラメータ】

- **[IN] BOOL onoff**

Bluetooth 機能を有効にするか否かの設定を指定します。

値	説明	初期値
0	Bluetooth 機能を無効にします	
1	Bluetooth 機能を有効にします	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-12 YVC_CMD_GetBTUse

【動作概要】

Bluetooth 機能の有効／無効の設定を取得します。

【書式】

```
typedef    struct {
        BOOL        onoff;
} YVC_ARG_BTUse;
```

【パラメータ】

- **[OUT] BOOL onoff**

Bluetooth 機能の設定を返します。

値	説明	初期値
0	Bluetooth 機能が無効です	
1	Bluetooth 機能が有効です	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-13 YVC_CMD_SetBTLocalName

【動作概要】

Bluetooth の Local Name を設定します。
Local Name を設定すると Bluetooth 接続するスマートフォンやタブレット（以降、端末と呼ぶ）に表示される機器の名前を変更することができます。1 台の端末で複数の機器と接続するときに便利です。設定可能な文字コードは、ASCII コードで 0x20～0x7E までの範囲となります。また、設定可能な文字数は最大 11 文字(YVC_LEN_BT_LOCAL_NAME)となります。
なお、機種名（例. ”*YVC-1000*”）を変更・削除することはできません。

対応機種	初期値	機器に表示される名称
YVC-1000	*“Yamaha”*	*“YVC-1000 Yamaha”*
YVC-300	*“Yamaha”*	*“YVC-300 Yamaha”*
YVC-330	*“Yamaha”*	*“YVC-330 Yamaha”*
YVC-200	*“Yamaha”*	*“YVC-200 Yamaha”*

【注意事項】
端末によっては、ペアリング情報を一旦削除しないと表示名が更新されない場合があります。

【書式】

```
typedef    struct {
      char       name[YVC_LEN_BT_LOCAL_NAME_TOTAL+1];
} YVC_ARG_BTLocalName;
```

【パラメータ】

- **[IN]　char　name[YVC_LEN_BT_LOCAL_NAME_TOTAL+1]**

Bluetooth の Local Name を指定します。
文字列には終端文字(0x00)を含めるようにしてください。
なお、初期設定に戻す場合は、空文字(0x00)を設定してください。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver.2.09 以降	✔	X
YVC-300	Ver.1.08 以降	✔	X
YVC-330	Ver.1.02 以降	✔	X
YVC-200	Ver.1.01 以降	✔	X

4.5-14 YVC_CMD_GetBTLocalName

【動作概要】

Bluetooth の Local Name の設定を取得します。
成功した場合、*name* に Bluetooth の Local Name が ASCII 文字列で格納されます。文字列の長さは、最大 20 文字(YVC_LEN_BT_LOCAL_NAME_TOTAL)となります。

【書式】

```
typedef    struct {
      char       name[YVC_LEN_BT_LOCAL_NAME_TOTAL+1];
} YVC_ARG_BTLocalName;
```

【パラメータ】

- **[OUT] char name[YVC_LEN_BT_LOCAL_NAME_TOTAL+1]**

Bluetooth の Local Name 設定を返します。
※終端文字(0x00)が含まれます。

初期設定では、以下の文字列を返します。

対応機種	初期値
YVC-1000	*“YVC-1000 Yamaha”*
YVC-300	*“YVC-300 Yamaha”*
YVC-330	*“YVC-330 Yamaha”*
YVC-200	*“YVC-200 Yamaha”*

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver.2.09 以降	✔	X
YVC-300	Ver.1.08 以降	✔	X
YVC-330	Ver.1.02 以降	✔	X
YVC-200	Ver.1.01 以降	✔	X

4.5-15 YVC_CMD_SetBTOperation

【動作概要】

Bluetooth 機能の操作をします。
本操作は、Bluetooth 機能が有効に設定されている必要があります。無効に設定されている場合は、エラー(YVCAPI_ERROR_CTRLEXEC)を返します。
レディ状態、ペアリング、再接続の実行は、接続、またはペアリングが完了しないまま 90 秒経過すると処理が中止されます。

【書式】

```
typedef    struct {
      union {
            YVC_BT_OPERATION_REQ       req;
            YVC_BT_OPERATION_STAT      stat;
      } u;
} YVC_ARG_BTOperation;
```

【パラメータ】

- **[IN] YVC_BT_OPERATION_REQ req**

Bluetooth 機能の操作要求を指定します。

パラメータ定義名	説明	YVC-1000	YVC-300	YVC-330	YVC-200
YVC_BT_OPERATION_REQ_SLEEP	スリープ状態にします	✔	✔	✔	X
YVC_BT_OPERATION_REQ_READY	レディ状態にします	✔	✔	✔	X
YVC_BT_OPERATION_REQ_DISCONNECT	切断します	✔	✔	✔	✔
YVC_BT_OPERATION_REQ_PAIRING_ON	ペアリングを実行します	✔	✔	✔	✔
YVC_BT_OPERATION_REQ_PAIRING_OFF	ペアリングをキャンセルします	✔	✔	✔	✔
YVC_BT_OPERATION_REQ_LINKBACK	最後に接続したデバイスに再接続します	X	X	X	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-16 YVC_CMD_GetBTOperation

【動作概要】

Bluetooth 機能の状態を取得します。
本操作は、Bluetooth 機能が有効に設定されている必要があります。無効に設定されている場合は、エラー(YVCAPI_ERROR_CTRLEXEC)を返します。

【書式】

```
typedef    struct {
      union {
            YVC_BT_OPERATION_REQ      req;
            YVC_BT_OPERATION_STAT     stat;
      } u;
} YVC_ARG_BTOperation;
```

【パラメータ】

- **[OUT]　YVC_BT_OPERATION_STAT　stat**

Bluetooth 機能の操作状態を返します。

パラメータ定義名	説明	YVC-1000	YVC-300	YVC-330	YVC-200
YVC_BT_OPERATION_STAT_SLEEP	スリープ状態を示します	✔	✔	✔	✔
YVC_BT_OPERATION_STAT_READY	レディ状態を示します	✔	✔	✔	✔
YVC_BT_OPERATION_STAT_DISCONNECTING	切断中を示します	✔	✔	✔	✔
YVC_BT_OPERATION_STAT_PAIRING	ペアリング中状態を示します	✔	✔	✔	✔
YVC_BT_OPERATION_STAT_CONNECTED	接続状態を示します	✔	✔	✔	✔
YVC_BT_OPERATION_STAT_CONNECTING	接続中を示します	X	X	X	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-17 YVC_CMD_SetNoticeSoundUse

【動作概要】

通知音を有効／無効にします。
通知音は以下の操作をおこなったときに鳴動します。
YVC-1000 の場合は、音量調整、スピーカー選択、自動音響調整、Bluetooth/NFC、言語選択などの操作をおこなったとき。
YVC-300／YVC-330 の場合は、Bluetooth/NFC、設定モード／オーディオモード切替え、接続モード選択、Bluetooth 有効・無効設定、工場出荷設定、本体連結接続などの操作をおこなったとき。
YVC-331 の場合は、設定モード／オーディオモード切替え、接続モード選択、工場出荷設定、本体連結接続などの操作をおこなったとき。
YVC-200 の場合は、Bluetooth/NFC、音量調整、スピーカーやマイクのミュート操作、工場出荷設定などの操作をおこなったとき。

【書式】

```
typedef    struct {
        BOOL        onoff;
} YVC_ARG_NoticeSoundUse;
```

【パラメータ】

- **[IN]　BOOL　onoff**

通知音を有効にするか否かの設定を指定します。

値	説明	初期値
0	通知音を無効にします	
1	通知音を有効にします	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-18 YVC_CMD_GetNoticeSoundUse

【動作概要】

通知音の有効／無効の設定を取得します。

【書式】

```
typedef    struct {
        BOOL        onoff;
} YVC_ARG_NoticeSoundUse;
```

【パラメータ】

- **[OUT]　BOOL　onoff**

通知音の設定を返します。

値	説明	初期値
0	通知音が無効です	
1	通知音が有効です	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-19 YVC_CMD_SetNoticeSoundVolume

【動作概要】

通知音の音量を調整します。

【書式】

```
typedef    struct {
        short        volume;
} YVC_ARG_NoticeSoundVolume;
```

【パラメータ】

- **[IN]　short　volume**

通知音の音量値(dB)を指定します。

設定可能範囲(dB)	初期値(dB)	設定値
-50 ～ 0	0	不揮発

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-20 YVC_CMD_GetNoticeSoundVolume

【動作概要】

通知音の音量の設定を取得します。

【書式】

```
typedef    struct {
        short        volume;
} YVC_ARG_NoticeSoundVolume;
```

【パラメータ】

- **[OUT]　short　volume**

通知音の音量値(dB)の設定を返します。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-21 YVC_CMD_SetVoiceGuideUse

【動作概要】

音声ガイダンスを有効／無効にします。

【書式】

```
typedef    struct {
        BOOL        onoff;
} YVC_ARG_VoiceGuideUse;
```

【パラメータ】

- **[IN]　BOOL　onoff**

音声ガイダンスを有効にするか否かの設定を指定します。

値	説明	初期値
0	音声ガイダンスを無効にします	
1	音声ガイダンスを有効にします	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-22 YVC_CMD_GetVoiceGuideUse

【動作概要】

音声ガイダンスの有効／無効の設定を取得します。

【書式】

```
typedef    struct {
        BOOL        onoff;
} YVC_ARG_VoiceGuideUse;
```

【パラメータ】

- **[OUT]　BOOL　onoff**

音声ガイダンスの設定を返します。

値	説明	初期値
0	音声ガイダンスが無効です	
1	音声ガイダンスが有効です	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-23 YVC_CMD_SetVoiceGuideLang

【動作概要】

音声ガイダンスの言語を変更します。

【書式】

```
typedef    struct {
        YVC_VOICE_GUIDE_LANG        type;
} YVC_ARG_VoiceGuideLang;
```

【パラメータ】

- **[IN]　YVC_VOICE_GUIDE_LANG　type**

音声ガイダンスの言語を指定します。

パラメータ定義名	説明	初期値
YVC_VOICE_GUIDE_LANG_EN	英語にします	✔
YVC_VOICE_GUIDE_LANG_JA	日本語にします	
YVC_VOICE_GUIDE_LANG_ZN	中国語にします	
YVC_VOICE_GUIDE_LANG_KO	韓国語にします	
YVC_VOICE_GUIDE_LANG_FR	フランス語にします	
YVC_VOICE_GUIDE_LANG_ES	スペイン語にします	
YVC_VOICE_GUIDE_LANG_DE	ドイツ語にします	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-24 YVC_CMD_GetVoiceGuideLang

【動作概要】

音声ガイダンスの言語の設定を取得します。

【書式】

```
typedef    struct {
        YVC_VOICE_GUIDE_LANG        type;
} YVC_ARG_VoiceGuideLang;
```

【パラメータ】

- **[IN]　YVC_VOICE_GUIDE_LANG　type**

音声ガイダンスの言語の設定を返します。

パラメータ定義名	説明	初期値
YVC_VOICE_GUIDE_LANG_EN	英語の設定です	✔
YVC_VOICE_GUIDE_LANG_JA	日本語の設定です	
YVC_VOICE_GUIDE_LANG_ZN	中国語の設定です	
YVC_VOICE_GUIDE_LANG_KO	韓国語の設定です	
YVC_VOICE_GUIDE_LANG_FR	フランス語の設定です	
YVC_VOICE_GUIDE_LANG_ES	スペイン語の設定です	
YVC_VOICE_GUIDE_LANG_DE	ドイツ語の設定です	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-25 YVC_CMD_SetVoiceGuideVolume

【動作概要】

音声ガイダンスの音量を調整します。

【書式】

```
typedef    struct {
        short        volume;
} YVC_ARG_VoiceGuideVolume;
```

【パラメータ】

- **[IN]　short　volume**

音声ガイダンスの音量値(dB)を指定します。

設定可能範囲(dB)	初期値(dB)	設定値
-50 ～ 0	0	不揮発

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-26 YVC_CMD_GetVoiceGuideVolume

【動作概要】

音声ガイダンスの音量の設定を取得します。

【書式】

```
typedef    struct {
        short        volume;
} YVC_ARG_VoiceGuideVolume;
```

【パラメータ】

- **[OUT]　short　volume**

音声ガイダンスの音量値(dB)の設定を返します。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-27 YVC_CMD_SetInitSpVolume

【動作概要】

USB 接続時の再生音量の設定をします。
ホストと機器を USB 接続した時の再生音量を機器が保持している値にするか、ホストから指定された値にするかを設定します。

以下に動作例を示します

type に YVC_INIT_SP_VOLUME_DEVICE を設定した場合

USB 接続状態	ホスト（PC）	機器
USB 接続前	-4.5dB	-10.0dB
USB 接続後	**-10.0dB**	-10.0dB

type に YVC_INIT_SP_VOLUME_HOST を設定した場合

USB 接続状態	ホスト（PC）	機器
USB 接続前	-4.5dB	-10.0dB
USB 接続後	-4.5dB	**-4.5dB**

【書式】

```
typedef    struct {
      YVC_INIT_SP_VOLUME       type;
} YVC_ARG_InitSpVolume;
```

【パラメータ】

- **[IN]　YVC_INIT_SP_VOLUME　type**

USB 接続時の再生音量を指定します。

パラメータ定義名	説明	初期値
YVC_INIT_SP_VOLUME_DEVICE	機器が保持している再生音量にします	✔
YVC_INIT_SP_VOLUME_HOST	ホスト(PC)から指定された再生音量にします	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-28 YVC_CMD_GetInitSpVolume

【動作概要】

USB 接続時の再生音量の設定を取得します。

【書式】

```
typedef    struct {
        YVC_INIT_SP_VOLUME        type;
} YVC_ARG_InitSpVolume;
```

【パラメータ】

- **[OUT]　YVC_INIT_SP_VOLUME　type**

USB 接続時の再生音量設定を返します。

パラメータ定義名	説明	初期値
YVC_INIT_SP_VOLUME_DEVICE	「機器が保持している再生音量」が設定されています	✔
YVC_INIT_SP_VOLUME_HOST	「ホスト(PC)から指定された再生音量」が設定されています	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01　以降	✔	X

4.5-29 YVC_CMD_SetPowerOnState

【動作概要】

電源投入時の動作を設定します。

【書式】

```
typedef    struct {
        YVC_POWER_ON_STATE        mode;
} YVC_ARG_PowerOnState;
```

【パラメータ】

- **[IN]　YVC_POWER_ON_STATE　mode**

電源投入時の動作を指定します。

パラメータ定義名	説明	初期値
YVC_POWER_ON_STATE_HOLD	電源が切断された時の状態に遷移します	
YVC_POWER_ON_STATE_STANDBY	スタンバイ状態に遷移します	✔
YVC_POWER_ON_STATE_RUN	起動状態に遷移します	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-30 YVC_CMD_GetPowerOnState

【動作概要】

電源投入時の動作の設定を取得します。

【書式】

```
typedef    struct {
        YVC_POWER_ON_STATE        mode;
} YVC_ARG_PowerOnState;
```

【パラメータ】

- **[OUT]　YVC_POWER_ON_STATE　mode**

電源投入時の動作設定を返します。

パラメータ定義名	説明	初期値
YVC_POWER_ON_STATE_HOLD	「電源が切断された時の状態に遷移する」が設定されています	
YVC_POWER_ON_STATE_STANDBY	「スタンバイ状態に遷移する」が設定されています	✔
YVC_POWER_ON_STATE_RUN	「起動状態に遷移する」が設定されています	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-31 YVC_CMD_SetMicMuteMode

【動作概要】

マイクミュートの動作モードを設定します。
ミュート状態のときに本コマンドで動作モードを変更した場合には、ミュート状態が解除されます。
なお、本操作によってミュート状態が変化しても(ミュートボタン操作ではないため)、イベント通知が有効の場合でもマイクミュートの状態通知イベント(YVC_EVENT_MIC_MUTE_STATE)は通知されません。

【書式】

```
typedef    struct {
      YVC_MIC_MUTE_MODE        mode;
} YVC_ARG_MicMuteMode;
```

【パラメータ】

- **[IN] YVC_MIC_MUTE_MODE mode**

マイクミュートの動作モードを指定します。

パラメータ定義名	説明	初期値
YVC_MIC_MUTE_MODE_GROUP	グループミュートモード(連動)で動作します	✔
YVC_MIC_MUTE_MODE_INDIVIDUAL	個別ミュートモード(非連動)で動作します	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver.2.10 以降	✔	X

4.5-32 YVC_CMD_GetMicMuteMode

【動作概要】

マイクミュートの動作モードの設定を取得します。

【書式】

```
typedef    struct {
        YVC_MIC_MUTE_MODE        mode;
} YVC_ARG_MicMuteMode;
```

【パラメータ】

- **[OUT]　YVC_MIC_MUTE_MODE　mode**

マイクミュートの動作モードの設定を返します。

パラメータ定義名	説明	初期値
YVC_MIC_MUTE_MODE_GROUP	「グループミュートモード（連動）」が設定されています	✔
YVC_MIC_MUTE_MODE_INDIVIDUAL	「個別ミュートモード（非連動）」が設定されています	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver.2.10 以降	✔	X

4.5-33 YVC_CMD_SetUsbSpeed

【動作概要】

USB ポートの動作スピードを設定します。
USB ポートの動作スピートの設定は、機器の起動時に有効となります。設定の変更を適用する場合は、*callback* パラメータを設定して再起動をするか、YVC_CMD_ExeSystemReboot コマンドで機器を再起動する必要があります。もしくは、機器の電源を一旦 OFF にしてから ON にする操作をおこないます。

【書式】

```
typedef    struct {
        YVC_USB_MODE    mode;
        struct {
            void                (*callback)(YVC_CTRL_CMD cmd, int err, void *user);
            void                *user;
        } set;
} YVC_ARG_UsbSpeed;
```

【パラメータ】

- **[IN]　YVC_USB_MODE　mode**

USB ポートの動作スピードを指定します。

パラメータ定義名	説明	初期値
YVC_USB_MODE_HI_SPEED	Hi-Speed で動作します	✔
YVC_USB_MODE_FULL_SPEED	Full-Speed で動作します	

- **[IN]　void (*callback)(YVC_CTRL_CMD cmd, int err, void *user)**

コマンド完了通知用のコールバック関数アドレスを指定します。
NULL を設定した場合は、設定のみおこない、再起動はおこないません。完了型として動作します。
NULL 以外を設定した場合は、設定後に再起動をします。非完了型として動作します。
非完了型では受付が正常完了すると、YVCAPI_ERROR_PENDING を返します。その後、当該処理を実行して、結果を本コールバック関数で通知します
以下、非完了型として動作したときのパラメータについて説明します。

cmd は、YVC_CMD_SetUsbSpeed を返します。
err は、エラーコードを返します。
user は、指定したアドレスをそのまま返します。

【注意事項】
コールバック関数内では、時間を要する処理や本 API 関数を実行しないようにしてください。

- **[IN]　void　*user**

ユーザー管理用データのアドレスを指定します。
非完了型で実行する場合のみ、本パラメータの設定が有効となります。不要な場合は NULL を設定してください。
完了型で実行する場合は、無視されます。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver.2.03 以降	✔	X

4.5-34 YVC_CMD_GetUsbSpeed

【動作概要】

USB ポートの動作スピードの設定を取得します。

【書式】

```
typedef    struct {
        YVC_USB_MODE    mode;
        struct {
            BOOL                reboot;
            void                (*callback)(YVC_CTRL_CMD cmd, int err, void *user);
            void                *user;
        } set;
} YVC_ARG_UsbSpeed;
```

【パラメータ】

- **[OUT]　YVC_USB_MODE　mode**

USB ポートの動作スピード設定を返します。

パラメータ定義名	説明	初期値
YVC_USB_MODE_HI_SPEED	「Hi-Speed」が設定されています	✔
YVC_USB_MODE_FULL_SPEED	「Full-Speed」が設定されています	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver.2.03 以降	✔	X

4.5-35 YVC_CMD_SetExtTermMode

【動作概要】

EXT 端子の接続モードの設定を変更します。
設定は、連結モード、またはビデオ会議システムモードの設定が可能です。

詳細は、「オーディオ・ブロック図」を参照してください。

【書式】

```
typedef    struct {
      YVC_EXTTERM_MODE       mode;
} YVC_ARG_ExtTermMode;
```

【パラメータ】

- **[IN]　YVC_EXTTERM_MODE　mode**

EXT 端子の接続モードを指定します。

パラメータ定義名	説明	初期値
YVC_EXTTERM_MODE_CASCADE	YVC-300／YVC-330／YVC-331 連結モードで動作します	✔
YVC_EXTTERM_MODE_CONFERENCE_SYSTEM	ビデオ会議システムモードで動作します	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X

4.5-36 YVC_CMD_GetExtTermMode

【動作概要】

EXT 端子の接続モードの設定を取得します。

【注意事項】
本体がメニューモードのときに返信する値は、メニューモードに入る前の値を返します。そのため、メニューモードでモード選択中は返信される値と LED 表示の設定が異なる場合があります。

【書式】

```
typedef    struct {
      YVC_EXTTERM_MODE        mode;
} YVC_ARG_ExtTermMode;
```

【パラメータ】

- **[OUT]　YVC_EXTTERM_MODE　mode**

EXT 端子の接続モード設定を返します。

パラメータ定義名	説明	初期値
YVC_EXTTERM_MODE_CASCADE	YVC-300／YVC-330／YVC-331 連結モードが設定されています	✔
YVC_EXTTERM_MODE_CONFERENCE_SYSTEM	「ビデオ会議システムモード」が設定されています	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X

4.5-37 YVC_CMD_SetSpOutSelect

【動作概要】

スピーカー出力選択を設定します。
「内蔵スピーカーのみ出力」、「外部スピーカーのみ出力」、「内蔵スピーカーと外部スピーカーの両方から出力」のいずれかを設定することができます。各設定における対象チャネルの音量値を以下に示します。
なお、YVC_CMD_SetAudioVolume コマンドで設定した値は本コマンドを発行すると更新されます。

スピーカー出力選択	VOLUME_CH_SP_BUILTIN	VOLUME_CH_SP_EXTERNAL
内蔵スピーカーのみ出力	0 dB	-128 dB
外部スピーカーのみ出力	-128 dB	0 dB
内蔵スピーカーと外部スピーカーの両方から出力	0 dB	0 dB

詳細は、「オーディオ・ブロック図」を参照してください。

【書式】

```
typedef    struct {
      YVC_SP_OUT_SELECT    select;
} YVC_ARG_SpOutSelect;
```

【パラメータ】

- **[IN]　YVC_SP_OUT_SELECT　select**

スピーカー出力選択を指定します。

パラメータ定義名	説明	初期値
YVC_SP_OUT_SELECT_BUILTIN	内蔵スピーカーのみ出力にします	
YVC_SP_OUT_SELECT_EXTERNAL	外部スピーカーのみ出力にします	
YVC_SP_OUT_SELECT_BOTH	内蔵スピーカーと外部スピーカーの両方から出力にします	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-38 YVC_CMD_GetSpOutSelect

【動作概要】

スピーカー出力選択の設定を取得します。
通常は、「内蔵スピーカーのみ出力」、「外部スピーカーのみ出力」、「内蔵スピーカーと外部スピーカーの両方から出力」のいずれかを返しますが、いずれにも該当しない場合にはエラー(YVCAPI_ERROR_INVALIDPARAM)を返します。
(YVC_CMD_SetAudioVolume コマンドで対象チャネルをそれぞれ-128dB に設定したときに発生します)

スピーカー出力選択	VOLUME_CH_SP_BUILTIN	VOLUME_CH_SP_EXTERNAL
内蔵スピーカーのみ出力	-128 dB 以外	-128 dB
外部スピーカーのみ出力	-128 dB	-128 dB 以外
内蔵スピーカーと外部スピーカーの両方から出力	-128 dB 以外	-128 dB 以外
エラー	-128 dB	-128 dB

【書式】

```
typedef    struct {
      YVC_SP_OUT_SELECT    select;
} YVC_ARG_SpOutSelect;
```

【パラメータ】

- **[OUT] YVC_SP_OUT_SELECT select**

スピーカー出力選択の設定を返します。

パラメータ定義名	説明	初期値
YVC_SP_OUT_SELECT_BUILTIN	「内蔵スピーカーのみ出力」が設定されています	
YVC_SP_OUT_SELECT_EXTERNAL	「外部スピーカーのみ出力」が設定されています	
YVC_SP_OUT_SELECT_BOTH	「内蔵スピーカーと外部スピーカーの両方から出力」が設定されています	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-39 YVC_CMD_SetAudioInTerm

【動作概要】

オーディオ入力端子の設定を変更します。
設定は、チャネル毎（L チャネル、R チャネル）に、ビデオ会議システムモード、または外部マイクモードの設定が可能です。
詳細は、「オーディオ・ブロック図」を参照してください。

【書式】

```
typedef    struct {
      struct {
            YVC_AUDIO_IN_MODE            mode;
            struct {
                  YVC_AUDIO_IN_LEVEL     level;
                  YVC_AUDIO_IN_EQ        eq;
                  YVC_AUDIO_IN_SP_OUT    spOut;
                  short                  gain;
            } paMic;
      } Lch, Rch;
} YVC_ARG_AudioInTerm;
```

【パラメータ】

- **[IN] Lch, Rch**

オーディオ入力端子の設定情報を格納するバッファのポインタを指定します。
設定は、L チャネル、R チャネルの順に設定してください。

- **[IN] YVC_AUDIO_IN_MODE mode**

パラメータ定義名	説明	初期値
YVC_AUDIO_IN_MODE_CONFERENCE	「ビデオ会議システム」モードにします	✔
YVC_AUDIO_IN_MODE_PA_MIC	「外部マイク」モードにします	

以下の *paMic* パラメータ(*level, eq, spOut, gain*) は、外部マイクを選択（*mode* に YVC_AUDIO_IN_MODE_PA_MIC）した場合に指定します。

- **[IN] YVC_AUDIO_IN_LEVEL level**

外部マイクの出力レベルを指定します。

パラメータ定義名	説明
YVC_AUDIO_IN_LEVEL_LINE	ラインレベルにします
YVC_AUDIO_IN_LEVEL_MIC	マイクレベルにします

- **[IN] YVC_AUDIO_IN_EQ eq**

外部マイクの音質の設定を指定します。

パラメータ定義名	説明
YVC_AUDIO_IN_EQ_THRU	音質を変更しません
YVC_AUDIO_IN_EQ_LOW_BOOST	低音を大きくします
YVC_AUDIO_IN_EQ_LOW_CUT	低音を小さくします

- **[IN] YVC_AUDIO_IN_SP_OUT spOut**

  外部マイクの自拠点拡声の設定を指定します。
  本パラメータは、L チャネル、R チャネルで共通です。各チャネルで YVC_AUDIO_IN_SP_OUT_OFF と YVC_AUDIO_IN_SP_OUT_ON が指定された場合には、YVC_AUDIO_IN_SP_OUT_ON として設定されます。

パラメータ定義名	説明
YVC_AUDIO_IN_SP_OUT_OFF	外部マイクの自拠点拡声を無効にします
YVC_AUDIO_IN_SP_OUT_ON	外部マイクの自拠点拡声を有効にします

- **[IN] short gain**

  外部マイクの感度を指定します。(-12dB ～ +12dB)

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-40 YVC_CMD_GetAudioInTerm

【動作概要】

オーディオ入力端子の設定を取得します。

【書式】

```
typedef    struct {
      struct {
            YVC_AUDIO_IN_MODE            mode;
            struct {
                  YVC_AUDIO_IN_LEVEL     level;
                  YVC_AUDIO_IN_EQ         eq;
                  YVC_AUDIO_IN_SP_OUT    spOut;
                  short                  gain;
            } paMic;
      } Lch, Rch;
} YVC_ARG_AudioInTerm;
```

【パラメータ】

- **[OUT]　Lch, Rch**

オーディオ入力端子の設定情報を返します。
設定情報は、L チャネル、R チャネルの順に格納されています。
外部マイクが設定（*mode* が YVC_AUDIO_IN_MODE_PA_MIC）されているとき、*paMic* パラメータ(*level, eq, spOut, gain*)が有効となります。
各パラメータについては、YVC_CMD_SetAudioInTerm コマンドの頁を参照してください。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-41 YVC_CMD_SetAudioVolume

【動作概要】

音量の設定を調整します。

スピーカーマスター出力チャネル、および USB 出力チャネルは、ホストからのボリューム制御で変更されるため、必ずしも本コマンドで設定した値になりません。また、変更した場合にはホストが保持している値と不整合が発生します。
YVC-300／YVC-330／YVC-331 の場合、AudioIn 入力チャネル、AudioOut 出力チャネルの設定は、接続モードの設定が「ビデオ会議システム」で動作しているとき、有効となります。ただし、「ビデオ会議システム」に設定されていない場合でも、設定した値は保存されます。
YVC-1000 の場合、内蔵スピーカー出力チャネル、および外部スピーカー出力チャネルの設定を変更すると、
YVC_CMD_SetSpOutSelect コマンドで設定した値が変わることがあります。

詳細は、「オーディオ・ブロック図」を参照してください。

【注意事項】
YVC-200 の場合、0dB を超えて設定をすると、入出力される信号によってはクリップする場合があります。

【書式】

```
typedef    struct {
      YVC_AUDIO_VOLUME_CH       ch;
      short                     volume;
} YVC_ARG_AudioVolume;
```

【パラメータ】

- **[IN]　YVC_AUDIO_VOLUME_CH　ch**

チャネルを指定します。

パラメータ定義名	説明	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_AUDIO_VOLUME_CH_SP	スピーカーマスター出力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_VOLUME_CH_MIC	マイクマスター入力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_VOLUME_CH_USB_IN	USB 入力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_VOLUME_CH_USB_OUT	USB 出力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_VOLUME_CH_BT_IN	Bluetooth 入力チャネル	✔	✔	✔	X	X
YVC_AUDIO_VOLUME_CH_BT_OUT	Bluetooth 出力チャネル	✔	✔	✔	X	✔
YVC_AUDIO_VOLUME_CH_AUDIO_IN	AudioIn 入力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_VOLUME_CH_AUDIO_OUT	AudioOut 出力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_VOLUME_CH_SP_BUILTIN	内蔵スピーカー出力チャネル	✔	X	X	X	X
YVC_AUDIO_VOLUME_CH_SP_EXTERNAL	外部スピーカー出力チャネル	✔	X	X	X	X
YVC_AUDIO_VOLUME_CH_SP_EXMIC	外部マイク拡声チャネル	✔	X	X	X	X
YVC_AUDIO_VOLUME_CH_EXMIC	外部マイクマスター入力チャネル	✔	X	X	X	X
YVC_AUDIO_VOLUME_CH_HS_IN	ヘッドセット入力チャネル	X	X	X	X	✔
YVC_AUDIO_VOLUME_CH_HP_OUT	ヘッドホン出力チャネル	X	X	X	X	✔

- **[IN]　short　volume**

音量値(dB)を指定します。

パラメータ定義名	設定可能範囲（dB）	初期値（dB）	設定値				
			YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_AUDIO_VOLUME_CH_SP	-50 ～ 0	-10	不揮発	不揮発	不揮発	不揮発	不揮発
YVC_AUDIO_VOLUME_CH_MIC	-128 ～ +12	0	揮発	揮発	揮発	揮発	不揮発
YVC_AUDIO_VOLUME_CH_USB_IN	-128 ～ +12	0	不揮発	不揮発	不揮発	不揮発	－
YVC_AUDIO_VOLUME_CH_USB_OUT	-20 ～ 0	0	揮発	揮発	揮発	揮発	揮発
YVC_AUDIO_VOLUME_CH_BT_IN	-128 ～ +12	0	不揮発	不揮発	不揮発	－	－
YVC_AUDIO_VOLUME_CH_BT_OUT	-128 ～ +12	0	不揮発	不揮発	不揮発	－	不揮発

YVC_AUDIO_VOLUME_CH_AUDIO_IN	-128 ～ +12	0	不揮発	不揮発	不揮発	不揮発	ー
YVC_AUDIO_VOLUME_CH_AUDIO_OUT	-128 ～ +12	0	不揮発	不揮発	不揮発	不揮発	ー
YVC_AUDIO_VOLUME_CH_SP_BUILTIN	-128 ～ 0	0	不揮発	ー	ー	ー	ー
YVC_AUDIO_VOLUME_CH_SP_EXTERNAL	-128 ～ 0	0	不揮発	ー	ー	ー	ー
YVC_AUDIO_VOLUME_CH_SP_EXMIC	-128 ～ 0	0	不揮発	ー	ー	ー	ー
YVC_AUDIO_VOLUME_CH_EXMIC	-128 ～ +12	0	不揮発	ー	ー	ー	ー
YVC_AUDIO_VOLUME_CH_HS_IN	-128 ～ +12	0	ー	ー	ー	ー	不揮発
YVC_AUDIO_VOLUME_CH_HP_OUT	-128 ～ +12	0	ー	ー	ー	ー	不揮発

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.09 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-42 YVC_CMD_GetAudioVolume

【動作概要】

音量の設定を取得します。

【書式】

```
typedef    struct {
      YVC_AUDIO_VOLUME_CH       ch;
      short                     volume;
} YVC_ARG_AudioVolume;
```

【パラメータ】

- **[IN]　YVC_AUDIO_VOLUME_CH　ch**

チャネルを指定します。

パラメータ定義名	説明	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_AUDIO_VOLUME_CH_SP	スピーカーマスター出力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_VOLUME_CH_MIC	マイクマスター入力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_VOLUME_CH_USB_IN	USB 入力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_VOLUME_CH_USB_OUT	USB 出力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_VOLUME_CH_BT_IN	Bluetooth 入力チャネル	✔	✔	✔	X	X
YVC_AUDIO_VOLUME_CH_BT_OUT	Bluetooth 出力チャネル	✔	✔	✔	X	✔
YVC_AUDIO_VOLUME_CH_AUDIO_IN	AudioIn 入力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_VOLUME_CH_AUDIO_OUT	AudioOut 出力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_VOLUME_CH_SP_BUILTIN	内蔵スピーカー出力チャネル	✔	X	X	X	X
YVC_AUDIO_VOLUME_CH_SP_EXTERNAL	外部スピーカー出力チャネル	✔	X	X	X	X
YVC_AUDIO_VOLUME_CH_SP_EXMIC	外部マイク拡声チャネル	✔	X	X	X	X
YVC_AUDIO_VOLUME_CH_EXMIC	外部マイクマスター入力チャネル	✔	X	X	X	X
YVC_AUDIO_VOLUME_CH_HS_IN	ヘッドセット入力チャネル	X	X	X	X	✔
YVC_AUDIO_VOLUME_CH_HP_OUT	ヘッドホン出力チャネル	X	X	X	X	✔

- **[OUT]　short　volume**

音量の設定を返します。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-43 YVC_CMD_SetAudioMute

【動作概要】

ミュートの設定を変更します。

YVC-1000、YVC-300、YVC-330、YVC-331 の場合、スピーカーマスター出力チャネル、および USB 出力チャネルは、ホストからのミュート制御で変更されるため、必ずしも本コマンドで設定した値になりません。同様に、YVC-200 の場合、スピーカーマスター出力チャネル、および、マイクマスター入力チャネルがホストから変更されます。また、変更した場合にはホストが保持している値と不整合が発生します。

マイクマスター入力チャネルに対してミュート設定をおこなったとき、ミュートボタン操作ではないため、イベント通知が有効の場合でもマイクミュートの状態通知イベント(YVC_EVENT_MIC_MUTE_STATE)は通知されません。

詳細は、「オーディオ・ブロック図」を参照してください。

【書式】

```
typedef    struct {
      YVC_AUDIO_MUTE_CH       ch;
      BOOL                    mute;
} YVC_ARG_AudioMute;
```

【パラメータ】

- **[IN]　YVC_AUDIO_MUTE_CH　ch**

チャネルを指定します。

パラメータ定義名	説明	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_AUDIO_MUTE_CH_SP	スピーカーマスター出力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_MUTE_CH_MIC	マイクマスター入力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_MUTE_CH_USB_IN	USB 入力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_MUTE_CH_USB_OUT	USB 出力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_MUTE_CH_BT_IN	Bluetooth 入力チャネル	✔	✔	✔	X	X
YVC_AUDIO_MUTE_CH_BT_OUT	Bluetooth 出力チャネル	✔	✔	✔	X	✔
YVC_AUDIO_MUTE_CH_AUDIO_IN	AudioIn 入力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_MUTE_CH_AUDIO_OUT	AudioOut 出力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_MUTE_CH_SP_BUILTIN	内蔵スピーカー出力チャネル	✔	X	X	X	X
YVC_AUDIO_MUTE_CH_SP_EXTERNAL	外部スピーカー出力チャネル	✔	X	X	X	X
YVC_AUDIO_MUTE_CH_EXMIC	外部マイクマスター入力チャネル	✔	X	X	X	X
YVC_AUDIO_MUTE_CH_HS_IN	ヘッドセット入力チャネル	X	X	X	X	✔
YVC_AUDIO_MUTE_CH_HP_OUT	ヘッドホン出力チャネル	X	X	X	X	✔

- **[IN]　BOOL　mute**

ミュート状態を指定します。

値	説明
0	ミュート状態を解除します
1	ミュート状態にします

以下に各ミュートチャネルの初期値、および設定の属性について示します。

パラメータ定義名	初期値	設定値				
		YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_AUDIO_MUTE_CH_SP	0	揮発	揮発	揮発	揮発	揮発
YVC_AUDIO_MUTE_CH_MIC	0	揮発	揮発	揮発	揮発	揮発
YVC_AUDIO_MUTE_CH_USB_IN	0	不揮発	不揮発	不揮発	不揮発	－
YVC_AUDIO_MUTE_CH_USB_OUT	0	揮発	揮発	揮発	揮発	不揮発
YVC_AUDIO_MUTE_CH_BT_IN	0	不揮発	不揮発	不揮発	－	－
YVC_AUDIO_MUTE_CH_BT_OUT	0	不揮発	不揮発	不揮発	－	不揮発

YVC_AUDIO_MUTE_CH_AUDIO_IN	0	不揮発	不揮発	不揮発	不揮発	ー
YVC_AUDIO_MUTE_CH_AUDIO_OUT	0	不揮発	不揮発	不揮発	不揮発	ー
YVC_AUDIO_MUTE_CH_SP_BUILTIN	0	不揮発	ー	ー	ー	ー
YVC_AUDIO_MUTE_CH_SP_EXTERNAL	0	不揮発	ー	ー	ー	ー
YVC_AUDIO_MUTE_CH_EXMIC	0	不揮発	ー	ー	ー	ー
YVC_AUDIO_MUTE_CH_HS_IN	0	ー	ー	ー	ー	不揮発
YVC_AUDIO_MUTE_CH_HP_OUT	0	ー	ー	ー	ー	不揮発

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-44 YVC_CMD_GetAudioMute

【動作概要】

ミュートの設定を取得します。

【書式】

```
typedef    struct {
        YVC_AUDIO_MUTE_CH        ch;
        BOOL                     mute;
} YVC_ARG_AudioMute;
```

【パラメータ】

- **[IN]　YVC_AUDIO_MUTE_CH　ch**

チャネルを指定します。

パラメータ定義名	説明	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_AUDIO_MUTE_CH_SP	スピーカーマスター出力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_MUTE_CH_MIC	マイクマスター入力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_MUTE_CH_USB_IN	USB 入力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_MUTE_CH_USB_OUT	USB 出力チャネル	✔	✔	✔	✔	✔
YVC_AUDIO_MUTE_CH_BT_IN	Bluetooth 入力チャネル	✔	✔	✔	X	X
YVC_AUDIO_MUTE_CH_BT_OUT	Bluetooth 出力チャネル	✔	✔	✔	X	✔
YVC_AUDIO_MUTE_CH_AUDIO_IN	AudioIn 入力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_MUTE_CH_AUDIO_OUT	AudioOut 出力チャネル	✔	✔	✔	✔	X
YVC_AUDIO_MUTE_CH_SP_BUILTIN	内蔵スピーカー出力チャネル	✔	X	X	X	X
YVC_AUDIO_MUTE_CH_SP_EXTERNAL	外部スピーカー出力チャネル	✔	X	X	X	X
YVC_AUDIO_MUTE_CH_EXMIC	外部マイクマスター入力チャネル	✔	X	X	X	X
YVC_AUDIO_MUTE_CH_HS_IN	ヘッドセット入力チャネル	X	X	X	X	✔
YVC_AUDIO_MUTE_CH_HP_OUT	ヘッドホン出力チャネル	X	X	X	X	✔

- **[OUT]　BOOL　mute**

ミュートの設定を返します。

値	説明
0	ミュート解除状態です
1	ミュート状態です

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-45 YVC_CMD_SetMicAgc

【動作概要】

マイク AGC の設定を変更します。

【書式】

```
typedef    struct {
        YVC_MIC_AGC_MODE        mode;
} YVC_ARG_MicAgc;
```

【パラメータ】

- **[IN]　YVC_MIC_AGC_MODE　mode**

マイク AGC の設定を指定します。

パラメータ定義名	説明	初期値
YVC_MIC_AGC_MODE_ON	マイク AGC を有効にします	✔
YVC_MIC_AGC_MODE_OFF	マイク AGC を無効にします	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-46 YVC_CMD_GetMicAgc

【動作概要】

マイク AGC の設定を取得します。

【書式】

```
typedef    struct {
        YVC_MIC_AGC_MODE        mode;
} YVC_ARG_MicAgc;
```

【パラメータ】

- **[OUT]　YVC_MIC_AGC_MODE　mode**

マイク AGC の設定を返します。

パラメータ定義名	説明	初期値
YVC_MIC_AGC_MODE_ON	マイク AGC が有効になっています	✔
YVC_MIC_AGC_MODE_OFF	マイク AGC が無効になっています	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-47 YVC_CMD_SetMicUnitTrack

【動作概要】

マイクユニットの収音モードを変更します。

収音モードには、TRACK モードと MIX モードの2つのモードがあります。
TRACK モードは、*unitSelect* で指定されたマイクユニットのうち、最も収音レベルが大きい信号を自動的に選択して出力します。
MIX モードは、*unitSelect* で指定されたマイクユニットの信号を全てミキシングして出力します。

【書式】

```
typedef    struct {
      YVC_MIC_MUX_MODE     mode;
      unsigned char                unitSelect;
} YVC_ARG_MicUnitTrack;
```

【パラメータ】

- **[IN]　YVC_MIC_MUX_MODE　 mode**

マイクのマルチプレクサの設定を指定します。

パラメータ定義名	説明	初期値
YVC_MIC_MUX_MODE_TRACK	TRACK モードで動作します	✔
YVC_MIC_MUX_MODE_MIX	MIX モードで動作します	

- **[IN]　unsigned char　 unitSelect**

マイクユニットの選択を指定します。
マイクユニットは、1 台以上選択する必要があります。いずれも選択されていない場合は、
エラー（YVCAPI_ERROR_INVALIDARGS）を返します。

パラメータ定義名	値	説明	初期値
YVC_PRM_MIC_UNIT0	0x01	マイクユニット 0 を選択します	✔
YVC_PRM_MIC_UNIT1	0x02	マイクユニット 1 を選択します	✔
YVC_PRM_MIC_UNIT2	0x04	マイクユニット 2 を選択します	✔
YVC_PRM_MIC_UNIT3	0x08	マイクユニット 3 を選択します	✔
YVC_PRM_MIC_UNIT4	0x10	マイクユニット 4 を選択します	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-48 YVC_CMD_GetMicUnitTrack

【動作概要】

マイクユニットの収音モードの設定を取得します。

【書式】

```
typedef    struct {
      YVC_MIC_MUX_MODE     mode;
      unsigned char                unitSelect;
} YVC_ARG_MicUnitTrack;
```

【パラメータ】

- **[OUT]　YVC_MIC_MUX_MODE　mode**

マイクのマルチプレクサの設定を返します。

パラメータ定義名	説明	初期値
YVC_MIC_MUX_MODE_TRACK	「TRACK モード」が設定されています	✔
YVC_MIC_MUX_MODE_MIX	「MIX モード」が設定されています	

- **[OUT]　unsigned char　unitSelect**

マイクユニットの設定を返します。

パラメータ定義名	値	説明	初期値
YVC_PRM_MIC_UNIT0	0x01	「マイクユニット 0」が設定されています	✔
YVC_PRM_MIC_UNIT1	0x02	「マイクユニット 1」が設定されています	✔
YVC_PRM_MIC_UNIT2	0x04	「マイクユニット 2」が設定されています	✔
YVC_PRM_MIC_UNIT3	0x08	「マイクユニット 3」が設定されています	✔
YVC_PRM_MIC_UNIT4	0x10	「マイクユニット 4」が設定されています	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X

4.5-49 YVC_CMD_SetMicCapsuleTrack

【動作概要】

マイクカプセルの収音モードを変更します。

収音モードには、TRACK モードと MIX モードの2つのモードがあります。
TRACK モードは、*capsuleSelect* で指定されたマイクカプセルのうち、最も収音レベルが大きい信号を自動的に選択して出力します。
MIX モードは、全てのマイクカプセルの信号を全てミキシングして出力します。

【書式】

```
typedef    struct {
      YVC_MIC_MUX_MODE     mode;
      unsigned char                capsuleSelect;
} YVC_ARG_MicCapsuleTrack;
```

【パラメータ】

- **[IN]　YVC_MIC_MUX_MODE　mode**

マイクのマルチプレクサの設定を指定します。

パラメータ定義名	説明	初期値
YVC_MIC_MUX_MODE_TRACK	「TRACK モード」が設定されています	✔
YVC_MIC_MUX_MODE_MIX	「MIX モード」が設定されています	

- **[IN]　unsigned char　capsuleSelect**

TRACK モードを選択した場合のみ、マイクカプセルの設定を指定します。
マイクカプセルは、1 つ以上選択する必要があります。いずれも選択されていない場合は、
エラー(YVCAPI_ERROR_INVALIDARGS)を返します。
なお、MIX モードを選択した場合は、無視されます。

パラメータ定義名	値	説明	初期値
YVC_PRM_MIC_CAPSULE_CENTER	0x01	マイクカプセル(中)を選択します	✔
YVC_PRM_MIC_CAPSULE_LEFT	0x02	マイクカプセル(左)を選択します	✔
YVC_PRM_MIC_CAPSULE_RIGHT	0x04	マイクカプセル(右)を選択します	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-300	Ver. 1.09 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X

4.5-50 YVC_CMD_GetMicCapsuleTrack

【動作概要】

マイクカプセルの収音モードの設定を取得します。

【書式】

```
typedef    struct {
      YVC_MIC_MUX_MODE     mode;
      unsigned char                capsuleSelect;
} YVC_ARG_MicCapsuleTrack;
```

【パラメータ】

- **[OUT]　YVC_MIC_MUX_MODE　mode**

マイクのマルチプレクサの設定を返します。

パラメータ定義名	説明	初期値
YVC_MIC_MUX_MODE_TRACK	「TRACK モード」が設定されています	✔
YVC_MIC_MUX_MODE_MIX	「MIX モード」が設定されています	

- **[OUT]　unsigned char　capsuleSelect**

マイクカプセルの設定を返します。

パラメータ定義名	値	説明	初期値
YVC_PRM_MIC_CAPSULE_CENTER	0x01	「マイクカプセル（中）」が設定されています	✔
YVC_PRM_MIC_CAPSULE_LEFT	0x02	「マイクカプセル（左）」が設定されています	✔
YVC_PRM_MIC_CAPSULE_RIGHT	0x04	「マイクカプセル（右）」が設定されています	✔

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-300	Ver.1.09 以降	✔	X
YVC-330	Ver.1.02 以降	✔	X
YVC-331	Ver.1.00 以降	✔	X

4.5-51 YVC_CMD_SetMicFilter

【動作概要】

マイクの周波数特性（マイク信号の低域をカット）を変更します。

【書式】

```
typedef    struct {
        YVC_MIC_FILTER_MODE        mode;
} YVC_ARG_MicFilter;
```

【パラメータ】

- **[IN]　YVC_MIC_FILTER_MODE　mode**

マイクの周波数特性を指定します。

パラメータ定義名	説明	初期値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_MIC_FILTER_MODE_OFF	低域をカットしないようにします	✔	✔	✔	✔	✔	✔
YVC_MIC_FILTER_MODE_LOW	200Hz 以下をカットします		✔	✔	✔	✔	X
YVC_MIC_FILTER_MODE_MID	315Hz 以下をカットします		✔	✔	✔	✔	✔
YVC_MIC_FILTER_MODE_HIGH	500Hz 以下をカットします		✔	✔	✔	✔	X

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.09 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-52 YVC_CMD_GetMicFilter

【動作概要】

マイクの周波数特性の設定を取得します。

【書式】

```
typedef    struct {
        YVC_MIC_FILTER_MODE        mode;
} YVC_ARG_MicFilter;
```

【パラメータ】

- **[OUT]　YVC_MIC_FILTER_MODE　mode**

マイクの周波数特性設定を返します。

パラメータ定義名	説明	初期値	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_MIC_FILTER_MODE_OFF	「低域をカットしない」が設定されています	✔	✔	✔	✔	✔	✔
YVC_MIC_FILTER_MODE_LOW	「200Hz 以下をカット」が設定されています		✔	✔	✔	✔	X
YVC_MIC_FILTER_MODE_MID	「315Hz 以下をカット」が設定されています		✔	✔	✔	✔	✔
YVC_MIC_FILTER_MODE_HIGH	「500Hz 以下をカット」が設定されています		✔	✔	✔	✔	X

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.09 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

4.5-53 YVC_CMD_SetSpFilter

【動作概要】

スピーカーの周波数特性（スピーカー信号の低域をカット）を変更します。

【書式】

```
typedef    struct {
        YVC_SP_FILTER_MODE        mode;
} YVC_ARG_SpFilter;
```

【パラメータ】

- **[IN]　YVC_SP_FILTER_MODE　mode**

スピーカーの周波数特性を指定します。

パラメータ定義名	説明	初期値
YVC_SP_FILTER_MODE_OFF	低域をカットしないようにします	✔
YVC_SP_FILTER_MODE_LOW	200Hz 以下をカットします	
YVC_SP_FILTER_MODE_MID	315Hz 以下をカットします	
YVC_SP_FILTER_MODE_HIGH	500Hz 以下をカットします	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.09 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X

4.5-54 YVC_CMD_GetSpFilter

【動作概要】

スピーカーの周波数特性の設定を取得します。

【書式】

```
typedef    struct {
        YVC_SP_FILTER_MODE        mode;
} YVC_ARG_SpFilter;
```

【パラメータ】

- **[OUT]　YVC_SP_FILTER_MODE　mode**

スピーカーの周波数特性設定を返します。

パラメータ定義名	説明	初期値
YVC_SP_FILTER_MODE_OFF	「低域をカットしない」が設定されています	✔
YVC_SP_FILTER_MODE_LOW	「200Hz 以下をカットする」が設定されています	
YVC_SP_FILTER_MODE_MID	「315Hz 以下をカットする」が設定されています	
YVC_SP_FILTER_MODE_HIGH	「500Hz 以下をカットする」が設定されています	

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.09 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X

4.5-55 YVC_CMD_ExeAAT

【動作概要】

自動音響調整機能の実行を制御(開始・停止)します。
YVC-1000 を設置した空間の音響特性や、マイクおよびスピーカーの設置位置などの音響状態を自動で測定し、音響設定を使用環境に合わせて最適な状態に調整します。

以下、実行手順について説明します。

この測定は、本体スピーカー、および外部スピーカーで測定音を再生し、マイクユニットで収音して測定をするため、本体スピーカーや外部スピーカーから再生される音やマイクユニットに収音される音が遮蔽されますと測定誤差が生じます。また、測定音以外の大きな音が混入した場合にも同様に誤差が生じやすくなります。
これらの理由により、測定中は機器から離れて、静かにした状態で測定をおこなってください。

① 最初に、本コマンドで *action* パラメータに YVC_AAT_ACTION_START を指定して、自動音響調整を実行します。開始が受け付けられた場合、以下の音声ガイダンスが再生されます。
「*自動音響調整を開始します。機器から離れて静かにしてください。測定音を再生します*」
ガイダンスが終了すると、測定が開始されます。
自動音響調整を中止する場合は、*action* パラメータに YVC_AAT_ACTION_STOP を指定すると中止することができます。なお、コマンド以外の要因で自動音響調整が中止された場合、以下の音声ガイダンスが再生されます。
「*自動音響調整を中止ました*」
※YVC_CMD_SetVoiceGuideUse コマンドで、音声ガイダンスが無効に設定されている場合は、再生されません。

② 次に自動音響調整の終了確認をおこないます。
YVC_CMD_AATProgress コマンドを定期的に実行して進捗度を確認します。*progress* の値が 100 になった時点で終了となります。

③ 最後に YVC_CMD_GetAATResult コマンドで測定結果を取得します。
result パラメータに測定結果、*warning* パラメータに警告エラー情報が設定されます。
※警告エラーの音声ガイダンスは再生されません。

【注意事項】
自動音響調整の実行中は、AUDIO IN 端子、AUDIO OUT 端子、USB I/F や Bluetooth I/F の音声入出力は無効(無音)となります。また、自動音響調整の実行中は、コマンドの実行に制限があります。次頁の一覧表を参照してください。

【書式】

```
typedef    struct {
      YVC_AAT_ACTION       action;
} YVC_ARG_AAT;
```

【パラメータ】

- **[IN]　YVC_AAT_ACTION　action**

自動音響調整機能の操作を指定します。

パラメータ定義名	説明
YVC_AAT_ACTION_STOP	自動音響調整を停止(中止)します
YVC_AAT_ACTION_START	自動音響調整を実行します

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.10 以降	✔	X

自動音響調整実行中のコマンド実行可否の一覧

コマンドコード	内容	実行可否
YVC_CMD_GetSystemMode	システムの動作モードを取得します	✔
YVC_CMD_GetVersionInfo	バージョン情報を取得します	✔
YVC_CMD_GetModelName	モデル名称を取得します	✔
YVC_CMD_GetSerialNumber	シリアル No.を取得します	✔ (注)
YVC_CMD_GetMicNum	マイクユニットの接続台数を取得します	✔
YVC_CMD_ExeSystemReboot	システムを再起動します	✔
YVC_CMD_ExeFactoryReset	パラメータを工場出荷時設定に戻す	X
YVC_CMD_ExeStandbyMode	スタンバイ状態にします	✔
YVC_CMD_SetNoticeEvent	通知するイベントの設定をします	✔
YVC_CMD_GetNoticeEvent	通知するイベントの設定を取得します	✔
YVC_CMD_SetBTUse	Bluetooth 機能を有効／無効にします	X
YVC_CMD_GetBTUse	Bluetooth 機能の有効／無効の設定を取得します	X
YVC_CMD_SetBTLocalName	Bluetooth の Local Name を設定します	X
YVC_CMD_GetBTLocalName	Bluetooth の Local Name の設定を取得します	X
YVC_CMD_SetBTOperation	Bluetooth 機能の操作をします	X
YVC_CMD_GetBTOperation	Bluetooth 機能の状態を取得します	X
YVC_CMD_SetNoticeSoundUse	通知音を有効／無効にします	✔
YVC_CMD_GetNoticeSoundUse	通知音の有効／無効の設定を取得します	✔
YVC_CMD_SetVoiceGuideUse	音声ガイダンスを有効／無効にします	✔
YVC_CMD_GetVoiceGuideUse	音声ガイダンスの有効／無効の設定を取得します	✔
YVC_CMD_SetVoiceGuideLang	音声ガイダンスの言語を変更します	X
YVC_CMD_GetVoiceGuideLang	音声ガイダンスの言語の設定を取得します	X
YVC_CMD_SetVoiceGuideVolume	音声ガイダンスの音量を調整します	X
YVC_CMD_GetVoiceGuideVolume	音声ガイダンスの音量の設定を取得します	X
YVC_CMD_SetInitSpVolume	USB 接続時の再生音量を設定します	X
YVC_CMD_GetInitSpVolume	USB 接続時の再生音量の設定を取得します	X
YVC_CMD_SetPowerOnState	電源投入時の動作を設定します	✔
YVC_CMD_GetPowerOnState	電源投入時の動作の設定を取得します	✔
YVC_CMD_SetMicMuteMode	マイクミュートの動作モードの設定をします	X
YVC_CMD_GetMicMuteMode	マイクミュートの動作モードの設定を取得します	X
YVC_CMD_SetUsbSpeed	USB ポートの動作スピードを設定します	X
YVC_CMD_GetUsbSpeed	USB ポートの動作スピードの設定を取得します	X
YVC_CMD_SetSpOutSelect	スピーカー出力選択を設定します	X
YVC_CMD_GetSpOutSelect	スピーカー出力選択の設定を取得します	X
YVC_CMD_SetAudioInTerm	オーディオ入力端子の設定を変更します	X
YVC_CMD_GetAudioInTerm	オーディオ入力端子の設定を取得します	X
YVC_CMD_SetAudioVolume	音量の設定を調整します	X
YVC_CMD_GetAudioVolume	音量の設定を取得します	X
YVC_CMD_SetAudioMute	ミュートの設定を変更します	X
YVC_CMD_GetAudioMute	ミュートの設定を取得します	X

YVC_CMD_SetMicAgc	マイク AGC の設定を変更します	X
YVC_CMD_GetMicAgc	マイク AGC の設定を取得します	X
YVC_CMD_SetMicUnitTrack	マイクユニットの収音モードを変更します	X
YVC_CMD_GetMicUnitTrack	マイクユニットの収音モード設定を取得します	X
YVC_CMD_SetMicFilter	マイクの周波数特性を変更します	X
YVC_CMD_GetMicFilter	マイクの周波数特性の設定を取得します	X
YVC_CMD_SetSpFilter	スピーカーの周波数特性を変更します	X
YVC_CMD_GetSpFilter	スピーカーの周波数特性の設定を取得します	X
YVC_CMD_ExeAAT	自動音響調整機能の実行を制御します	✔
YVC_CMD_GetAATProgress	自動音響調整の進捗状況を取得します	✔
YVC_CMD_GetAATResult	自動音響調整の測定結果を取得します	X
YVC_CMD_MonMicCapsule	アクティブになっているマイクカプセルを取得します	X
YVC_CMD_MonMicUnit	アクティブになっているマイクユニット ID を取得します	X

（注）本体のシリアル No.のみ取得することができます。

4.5-56 YVC_CMD_GetAATProgress

【動作概要】

自動音響調整の進捗状況を取得します。

【書式】

```
typedef    struct {
        unsigned int    progress;
} YVC_ARG_AATProgress;
```

【パラメータ】

- **[OUT]　unsigned int　progress**

自動音響調整の進捗度を 100 分率で返します。(0 ～ 100)

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.10 以降	✔	X

4.5-57 YVC_CMD_GetAATResult

【動作概要】

自動音響調整の測定結果を取得します。

【書式】

```
typedef    struct {
        YVC_AAT_RESULT  result;
        unsigned int              warning;
} YVC_ARG_AATResult;
```

【パラメータ】

- **[OUT]　YVC_AAT_RESULT　result**

自動音響調整の測定結果を返します。

パラメータ定義名	内容
YVC_AAT_RESULT_OK	正常終了しました。 （問題は見つかりませんでした）
YVC_AAT_RESULT_CANCEL	キャンセルされました。 キャンセルにはコマンドからの実行中止以外に以下の操作も含まれます。 ・測定中にマイクユニットの接続台数が変わったとき ・ホスト(PC)からスピーカー音量が変更されたとき ・本体の音叉ボタンが押されたとき これらの場合、以下の音声ガイダンスが再生されます。 「*自動音響調整を中止しました*」 ※YVC_CMD_SetVoiceGuideUse コマンドで、音声ガイダンスが無効に設定されている場合は、再生されません。
YVC_AAT_RESULT_WARNING	警告エラーを検出しました。 詳細は、*warning* パラメータに設定されます。
YVC_AAT_RESULT_NO_MIC	マイクが接続されていないため、自動音響調整を中止しました。
YVC_AAT_RESULT_SP_VOL_SMALL	スピーカーの音量が小さすぎるため、自動音響調整を中止しました。 スピーカー音量が-16dB 以上に設定されている必要があります。 YVC_CMD_SetAudioVolume コマンドで、スピーカーマスター出力チャネルを指定して、音量を変更してください。
YVC_AAT_RESULT_EXT_SP_UNDETECT	外部スピーカーが検出できないため、自動音響調整を中止しました。（外部スピーカーのみを有効にしている場合に限ります）

- **[OUT]　unsigned int　warning**

自動音響調整の警告エラーを返します。*result* パラメータに YVC_AAT_RESULT_WARNING が設定されている場合に有効となります。
なお、警告エラーは、複数検出される場合もあります。

パラメータ定義名	値	内容
YVC_PRM_AAT_WARNING_OK	0x0	正常終了
YVC_PRM_AAT_WARNING_TOOCLOSE	0x1	エラーを検出しました 「マイクユニットと本体の距離が近すぎます」
YVC_PRM_AAT_WARNING_TOOFAR	0x2	エラーを検出しました 「マイクユニットと本体の距離が遠すぎます」
YVC_PRM_AAT_WARNING_LONGDELAY	0x4	エラーを検出しました 「外部スピーカーのディレイが大きすぎます」
YVC_PRM_AAT_WARNING_DISTORTION	0x8	エラーを検出しました 「外部スピーカーの歪が大きすぎます」

警告エラーの詳細内容につきましては、「ユニファイドコミュニケーション　マイクスピーカーシステム YVC-1000 取扱説明書」を参照してください。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.10 以降	✔	X

4.5-58 YVC_CMD_MonMicCapsule

【動作概要】

アクティブになっているマイクカプセルを取得します。
この値はマイクがミュートされている状態でも値を返します。用途に合わせて、マイクミュート状態のときはマスクするなどしてください。
本コマンドの実行間隔は、500ms 以上の時間を確保してください。500ms 未満でコマンドの実行要求が行われた場合には、エラー(YVCAPI_ERROR_CTRLEXEC)を返します。

各機器のマイクカプセルの位置については、「マイクカプセル位置」を参照してください。

【書式】

```
typedef    struct {
        unsigned char    trackInfo[YVC_LEN_MIC_UNIT_MAX];
} YVC_ARG_MonMicCapsule;
```

【パラメータ】

- **[OUT]　unsigned char　trackInfo[YVC_LEN_MIC_UNIT_MAX]**

アクティブになっているマイクカプセルを返します。

パラメータ定義名	値	内容
YVC_PRM_MIC_CAPSULE_NONE	0x00	すべてのマイクカプセルが非アクティブです
YVC_PRM_MIC_CAPSULE_CENTER	0x01	マイクカプセル（中）がアクティブです
YVC_PRM_MIC_CAPSULE_LEFT	0x02	マイクカプセル（左）がアクティブです
YVC_PRM_MIC_CAPSULE_RIGHT	0x04	マイクカプセル（右）がアクティブです

YVC-1000 の場合
　接続されているマイクユニットの台数が有効バイト数となります。
　　マイクユニット 0 の情報は、*trackInfo[0]*
　　マイクユニット 1 の情報は、*trackInfo[1]*
　　マイクユニット 2 の情報は、*trackInfo[2]*
　　マイクユニット 3 の情報は、*trackInfo[3]*
　　マイクユニット 4 の情報は、*trackInfo[4]*

YVC-300／YVC-330／YVC-331 の場合
　連結状態に関らず、最初の1バイトのみ（*trackInfo[0]*）が有効となります。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X

4.5-59 YVC_CMD_MonMicUnit

【動作概要】

アクティブになっているマイクユニット ID を取得します。
マイクユニット ID は、マイクがミュートされている状態でも値を返します。用途に合わせて、マイクミュート状態のときはマスクするなどしてください。
本コマンドの実行間隔は、500ms 以上の時間を確保してください。500ms 未満でコマンドの実行要求がおこなわれた場合には、エラー(YVCAPI_ERROR_CTRLEXEC)を返します。

各機器のマイクカプセルの位置については、「マイクカプセル位置」を参照してください。

【書式】

```
typedef    struct {
        int     unitId;
} YVC_ARG_MonMicUnit;
```

【パラメータ】

- **[OUT]　int　unitId**

アクティブになっているマイクユニット ID(0～4)を返します。
なお、マイクユニットが未接続の場合には、-1 を返します。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.10 以降	✔	X

4.5-60 YVC_CMD_MonHeadsetPortState

【動作概要】

ヘッドセット端子の接続状態を取得します。ヘッドセット端子が未接続の場合は、YVC-200 のマイク、スピーカーが有効ですが、ヘッドセット端子にヘッドホンやヘッドセットを接続した場合は、接続した機器のマイク、およびスピーカーが有効になります。
本コマンドの実行間隔は、500ms 以上の時間を確保してください。500ms 未満でコマンドの実行要求がおこなわれた場合には、エラー(YVCAPI_ERROR_CTRLEXEC)を返します。

詳細は、「オーディオ・ブロック図」を参照してください。

【書式】

```
typedef    struct {
        unsigned int     state;
} YVC_ARG_MonHeadsetPortState;
```

【パラメータ】

- **[OUT]　unsigned int　state**

ヘッドセット端子の接続状態を返します。

パラメータ定義名	値	内容
YVC_PRM_HEADSET_PORT_NONE	0x00	ヘッドセット端子は未接続状態です この接続状態ではスピーカーとマイクが有効です
YVC_PRM_HEADSET_PORT_HP	0x01	ヘッドセット端子にはヘッドホンが接続されています この接続状態ではヘッドホン出力とマイクが有効です
YVC_PRM_HEADSET_PORT_HS	0x02	ヘッドセット端子にはヘッドセットが接続されています この接続状態ではヘッドホン出力とヘッドセット入力が有効です

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-200	Ver. 1.01 以降	✔	X

4.5-61 YVC_CMD_MonBatteryLevel

【動作概要】

バッテリーの残量を取得します。
本コマンドの実行間隔は、500ms 以上の時間を確保してください。500ms 未満でコマンドの実行要求がおこなわれた場合には、エラー(YVCAPI_ERROR_CTRLEXEC)を返します。

【書式】

```
typedef    struct {
        unsigned int     level;
        unsigned int     charging;
} YVC_ARG_MonBatteryLevel;
```

【パラメータ】

- **[OUT] unsigned int level**

バッテリーの残量(0~3)を返します。

値	内容
0	バッテリーの残量は 25%未満です
1	バッテリーの残量は 25～40%です
2	バッテリーの残量は 40～80%です
3	バッテリーの残量は 80%以上です

- **[OUT] unsigned int charging**

バッテリーが充電中か否か(0/1)を返します。

値	内容
0	バッテリーを充電していません
1	バッテリーを充電中です

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-200	Ver. 1.01 以降	✔	X

4.6 イベント詳細

【動作概要】

イベントを通知します。
通知する場合は、通知先を YVC_DeviceAttach 関数で事前に登録しておく必要があります。また、各々のイベントを通知するか否かの設定は、YVC_CMD_SetNoticeEvent コマンドで設定します。
ただし、YVC_EVENT_DEV_DISCONNECT イベントは常に有効です。通知を無効にすることはできません。

【注意事項】
コールバック関数内では、時間を要する処理や本 API 関数を実行しないようにしてください。

【書式】

typedef　void (*YVC_EVENT_CALLBACK)(YVC_EVENT event, int data, void *user)

【パラメータ】

- **[IN]　YVC_EVENT　event**

イベントの種別を通知します。

イベント定義名	説明	YVC-1000	YVC-300	YVC-330	YVC-331	YVC-200
YVC_EVENT_DEV_DISCONNECT	機器が取り外されたことを通知します。 【注意事項】 本イベント通知を受け取った場合、YVC_DeviceDetach 関数を実行して、YVC_DeviceAttach 関数で取得したハンドルを解放する必要があります。ただし、コールバック関数内では、上記 API 関数を実行しないようにしてください。	✔	✔	✔	✔	✔
YVC_EVENT_MIC_UNIT_NUM	マイクユニットの接続状態に変化があったことを通知します。	✔	X	X	X	X
YVC_EVENT_MIC_MUTE_STATE	マイクミュートの状態に変化があったことを通知します。ただし、ミュートボタンが押されたときのみに限ります。コマンド設定などで状態が変化した場合は通知されません。	✔	✔	✔	✔	✔
YVC_EVENT_HOOK_BTN	フックボタンの押下状態に変化があったことを通知します。ただし、Bluetooth の着信応答、終話（通話切断）の機能として動作する場合には通知されません。	X	✔	✔	✔	X

- **[IN]　int　data**

イベントの詳細内容を通知します。

YVC_EVENT_DEV_DISCONNECT イベント

値	内容
0	意味を持ちません（0 固定）

YVC_EVENT_MIC_UNIT_NUM イベント

値	内容
0	マイクユニットが未接続となりました
1	マイクユニットが 1 台接続となりました
2	マイクユニットが 2 台接続となりました
3	マイクユニットが 3 台接続となりました
4	マイクユニットが 4 台接続となりました
5	マイクユニットが 5 台接続となりました

YVC_EVENT_MIC_MUTE_STATE イベント

YVC-300/YVC-330/YVC-331 と YVC-200、および
YVC-1000 の場合で YVC_MIC_MUTE_MODE パラメータの値が YVC_MIC_MUTE_MODE_GROUP のとき

値	内容
0	マイクミュートが OFF となりました
1	マイクミュートが ON となりました

YVC-1000 の場合で YVC_MIC_MUTE_MODE パラメータの値が YVC_MIC_MUTE_MODE_INDIVIDUAL のとき

いずれかのマイクユニットのミュート状態が変化したときに通知されます。*data* は、接続されているすべてのマイクユニットのミュート状態を示します。

パラメータ定義名	Bit	値	内容
YVC_EVENT_DATA_MICMUTE_UNIT0	0x1	0	マイクユニット 0 のミュート状態は OFF です
		1	マイクユニット 0 のミュート状態は ON です
YVC_EVENT_DATA_MICMUTE_UNIT1	0x2	0	マイクユニット 1 のミュート状態は OFF です
		1	マイクユニット 1 のミュート状態は ON です
YVC_EVENT_DATA_MICMUTE_UNIT2	0x4	0	マイクユニット 2 のミュート状態は OFF です
		1	マイクユニット 2 のミュート状態は ON です
YVC_EVENT_DATA_MICMUTE_UNIT3	0x8	0	マイクユニット 3 のミュート状態は OFF です
		1	マイクユニット 3 のミュート状態は ON です
YVC_EVENT_DATA_MICMUTE_UNIT4	0x10	0	マイクユニット 4 のミュート状態は OFF です
		1	マイクユニット 4 のミュート状態は ON です

例えば、マイクユニットが 2 台接続されている場合で、
　マイクユニット 0 のマイクをミュートしたとき、*data* の値は、0x01
　続いて、マイクユニット 1 のマイクをミュートしたとき、*data* の値は、0x03
となります。

YVC_EVENT_HOOK_BTN イベント

パラメータ定義名	値	内容
YVC_EVENT_DATA_HOOKBTN_SHORT	0	フックボタンの短押しが操作されました
YVC_EVENT_DATA_HOOKBTN_LONG	1	フックボタンの長押しが操作されました

- **[IN] void *user**

YVC_DeviceAttach 関数の第 4 パラメータ(*user*)で指定したアドレスをそのまま返します。

【対応機種】

対応機種	対応ファームウェア	対応モード	
		NORMAL	FWUP
YVC-1000	Ver. 2.00 以降	✔	X
YVC-300	Ver. 1.04 以降	✔	X
YVC-330	Ver. 1.02 以降	✔	X
YVC-331	Ver. 1.00 以降	✔	X
YVC-200	Ver. 1.01 以降	✔	X

【注意事項】
YVC_EVENT_DEV_DISCONNECT 以外のイベントについては、YVC_CMD_SetNoticeEvent コマンドで設定する必要がありますが、対応ファームウェアに制限があります。詳細は、YVC_CMD_SetNoticeEvent コマンドの頁を参照してください。

4.7 定数・型定義

本 API で使用する型定義、および定数について示します。

4.7-1 YVC_API_INFO

```
/* API 情報の定義 */
typedef    struct {
        char          version[YVC_LEN_API_VERSION+1];
} YVC_API_INFO;
```

4.7-2 YVC_DEV_HANDLE

```
/*  機器を制御するためのハンドル定義 */
typedef    void*    YVC_DEV_HANDLE;
```

4.7-3 YVC_DEV_TYPE

```
/* YVC シリーズの機種タイプの定義 */
typedef     enum     {
       YVC_DEV_TYPE_YVC1000 = 0,
       YVC_DEV_TYPE_YVC300,
       YVC_DEV_TYPE_YVC200,
       YVC_DEV_TYPE_YVC330,
       YVC_DEV_TYPE_YVC331,
       YVC_DEV_TYPE_NUM
} YVC_DEV_TYPE;
```

4.7-4 YVC_DEV_INFO

```
 /*  機器の接続情報の定義 */
 typedef    struct    {
        unsigned    int    num;
 } YVC_DEV_INFO;
```

4.7-5 YVC_EVENT

```
/*  イベント種別の定義 */
typedef    enum {
        YVC_EVENT_DEV_DISCONNECT = 0,
        YVC_EVENT_MIC_UNIT_NUM,
        YVC_EVENT_MIC_MUTE_STATE,
        YVC_EVENT_HOOK_BTN,
        YVC_EVENT_NUM
} YVC_EVENT;
```

4.7-6 YVC_EVENT_CALLBACK

```
/*  イベント通知用のコールバック関数の定義 */
typedef    void (*YVC_EVENT_CALLBACK)(YVC_EVENT event, int data, void *user);
```

4.7-7 YVC_CMD

```
/* コマンド種別の定義 */
  typedef    enum {
        YVC_CMD_GetSystemMode = 0,
        YVC_CMD_GetVersionInfo,
        YVC_CMD_GetModelName,
        YVC_CMD_GetSerialNumber,
        YVC_CMD_GetMicNum,
        YVC_CMD_ExeSystemReboot,
        YVC_CMD_ExeFactoryReset,
        YVC_CMD_ExeStandbyMode,
        YVC_CMD_SetNoticeEvent,
        YVC_CMD_GetNoticeEvent,
        YVC_CMD_SetBTUse,
        YVC_CMD_GetBTUse,
        YVC_CMD_SetBTLocalName,
        YVC_CMD_GetBTLocalName,
        YVC_CMD_SetBTOperation,
        YVC_CMD_GetBTOperation,
        YVC_CMD_SetNoticeSoundUse,
        YVC_CMD_GetNoticeSoundUse,
        YVC_CMD_SetNoticeSoundVolume,
        YVC_CMD_GetNoticeSoundVolume,
        YVC_CMD_SetVoiceGuideUse,
        YVC_CMD_GetVoiceGuideUse,
        YVC_CMD_SetVoiceGuideLang,
        YVC_CMD_GetVoiceGuideLang,
        YVC_CMD_SetVoiceGuideVolume,
        YVC_CMD_GetVoiceGuideVolume,
        YVC_CMD_SetInitSpVolume,
        YVC_CMD_GetInitSpVolume,
        YVC_CMD_SetPowerOnState,
        YVC_CMD_GetPowerOnState,
        YVC_CMD_SetMicMuteMode,
        YVC_CMD_GetMicMuteMode,
        YVC_CMD_SetUsbSpeed,
        YVC_CMD_GetUsbSpeed,
        YVC_CMD_SetExtTermMode,
        YVC_CMD_GetExtTermMode,
        YVC_CMD_SetSpOutSelect,
        YVC_CMD_GetSpOutSelect,
        YVC_CMD_SetAudioInTerm,
        YVC_CMD_GetAudioInTerm,
        YVC_CMD_SetAudioVolume,
        YVC_CMD_GetAudioVolume,
        YVC_CMD_SetAudioMute,
        YVC_CMD_GetAudioMute,
        YVC_CMD_SetMicAgc,
        YVC_CMD_GetMicAgc,
        YVC_CMD_SetMicUnitTrack,
        YVC_CMD_GetMicUnitTrack,
        YVC_CMD_SetMicFilter,
        YVC_CMD_GetMicFilter,
        YVC_CMD_SetSpFilter,
        YVC_CMD_GetSpFilter,
        YVC_CMD_ExeAAT,
        YVC_CMD_GetAATProgress,
        YVC_CMD_GetAATResult,
        YVC_CMD_MonMicCapsule,
        YVC_CMD_MonMicUnit,
        YVC_CMD_Reserved1,
        YVC_CMD_Reserved2,
        YVC_CMD_MonHeadsetPortState,
        YVC_CMD_MonBatteryLevel,
        YVC_CMD_NUM
  } YVC_CTRL_CMD;
```

4.7-8 YVC_CTRL_ARG

```
/* 制御パラメータの型定義 */
typedef   void*  YVC_CTRL_ARG;
```

以下に各コマンドの ARG の定義リストを記述します。

1）YVC_CMD_GetSystemMode

```
/* システムの動作モードのパラメータ定義 */
typedef   enum {
      YVC_SYSTEM_MODE_FWUP = 0,
      YVC_SYSTEM_MODE_NORMAL,
      YVC_SYSTEM_MODE_NUM
} YVC_SYSTEM_MODE;

typedef   struct {
      YVC_SYSTEM_MODE    mode;
} YVC_ARG_SystemMode;
```

2）YVC_CMD_GetVersionInfo

```
typedef   struct {
      char      version[YVC_LEN_VERSION_INFO+1];
} YVC_ARG_VersionInfo;
```

3）YVC_CMD_GetModelName

```
typedef   struct {
      char      name[YVC_LEN_MODEL_NAME+1];
} YVC_ARG_ModelName;
```

4）YVC_CMD_GetSerialNumber

```
/* シリアル番号のユニット選択リストの定義 */
typedef   enum {
    YVC_SERIALNUMBER_TYPE_SPUNIT = 0,
    YVC_SERIALNUMBER_TYPE_MICUNIT0,
    YVC_SERIALNUMBER_TYPE_MICUNIT1,
    YVC_SERIALNUMBER_TYPE_MICUNIT2,
    YVC_SERIALNUMBER_TYPE_MICUNIT3,
    YVC_SERIALNUMBER_TYPE_MICUNIT4,
    YVC_SERIALNUMBER_TYPE_NUM
} YVC_SERIALNUMBER_TYPE;

typedef   struct {
      YVC_SERIALNUMBER_TYPE type;
      char        serialNumber[YVC_LEN_SERIAL_NUMBER+1];
} YVC_ARG_SerialNumber;
```

5）YVC_CMD_GetMicNum

```
typedef   struct {
      unsigned int      micNum;
} YVC_ARG_MicNum;
```

6）YVC_CMD_ExeSystemReboot

```
typedef   struct {
      void      (*callback)(YVC_CTRL_CMD cmd, int err, void *user);
      void      *user;
} YVC_ARG_SystemReboot;
```

7）YVC_CMD_ExeFactoryReset

```
typedef    struct {
      void        (*callback)(YVC_CTRL_CMD cmd, int err, void *user);
      void        *user;
} YVC_ARG_FactoryReset;
```

8）YVC_CMD_SetNoticeEvent / YVC_CMD_GetNoticeEvent

```
typedef    struct {
      unsigned int    event;
} YVC_ARG_NoticeEvent;
```

9）YVC_CMD_SetBTUse / YVC_CMD_GetBTUse

```
typedef    struct {
      BOOL       onoff;
} YVC_ARG_BTUse;
```

10）YVC_CMD_SetBTLocalName / YVC_CMD_GetBTLocalName

```
typedef    truct {
      char       name[YVC_LEN_BT_LOCAL_NAME_TOTAL+1];
} YVC_ARG_BTLocalName;
```

11）YVC_CMD_SetBTOperation / YVC_CMD_GetBTOperation

```
/* BT 操作のパラメータ定義 */
typedef    enum {
      YVC_BT_OPERATION_REQ_SLEEP = 0,
      YVC_BT_OPERATION_REQ_READY,
      YVC_BT_OPERATION_REQ_DISCONNECT,
      YVC_BT_OPERATION_REQ_PAIRING_ON,
      YVC_BT_OPERATION_REQ_PAIRING_OFF,
      YVC_BT_OPERATION_REQ_LINKBACK,
      YVC_BT_OPERATION_REQ_NUM
} YVC_BT_OPERATION_REQ;

/* BT 動作状態のパラメータ定義 */
typedef    enum {
      YVC_BT_OPERATION_STAT_SLEEP = 0,
      YVC_BT_OPERATION_STAT_READY,
      YVC_BT_OPERATION_STAT_DISCONNECTING,
      YVC_BT_OPERATION_STAT_PAIRING,
      YVC_BT_OPERATION_STAT_CONNECTED,
      YVC_BT_OPERATION_STAT_CONNECTING,
      YVC_BT_OPERATION_STAT_NUM
} YVC_BT_OPERATION_STAT;

typedef    struct {
      union {
            YVC_BT_OPERATION_REQ       req;
            YVC_BT_OPERATION_STAT      stat;
      } u;
} YVC_ARG_BTOperation;
```

12）YVC_CMD_SetNoticeSoundUse / YVC_CMD_GetNoticeSoundUse

```
typedef    struct {
      BOOL       onoff;
} YVC_ARG_NoticeSoundUse;
```

13）YVC_CMD_SetNoticeSoundVolume / YVC_CMD_GetNoticeSoundVolume

```
typedef    struct {
      short      volume;
} YVC_ARG_NoticeSoundVolume;
```

14）YVC_CMD_SetVoiceGuideUse / YVC_CMD_GetVoiceGuideUse

```
typedef    struct {
      BOOL       onoff;
} YVC_ARG_VoiceGuideUse;
```

15）YVC_CMD_SetVoiceGuideLang / YVC_CMD_GetVoiceGuideLang

```
/*  音声ガイダンスの言語選択リストのパラメータ定義  */
typedef    enum {
      YVC_VOICE_GUIDE_LANG_EN = 0,
      YVC_VOICE_GUIDE_LANG_JA,
      YVC_VOICE_GUIDE_LANG_ZN,
      YVC_VOICE_GUIDE_LANG_KO,
      YVC_VOICE_GUIDE_LANG_FR,
      YVC_VOICE_GUIDE_LANG_ES,
      YVC_VOICE_GUIDE_LANG_DE,
      YVC_VOICE_GUIDE_LANG_NUM
} YVC_VOICE_GUIDE_LANG;

typedef    struct {
      YVC_VOICE_GUIDE_LANG       type;
} YVC_ARG_VoiceGuideLang;
```

16）YVC_CMD_SetVoiceGuideVolume / YVC_CMD_GetVoiceGuideVolume

```
typedef    struct {
      short       volume;
} YVC_ARG_VoiceGuideVolume;
```

17）YVC_CMD_SetInitSpVolume / YVC_CMD_GetInitSpVolume

```
/* USB 接続時の再生音量を設定するモードのパラメータ定義  */
typedef    enum {
      YVC_INIT_SP_VOLUME_DEVICE = 0,
      YVC_INIT_SP_VOLUME_HOST,
      YVC_INIT_SP_VOLUME_NUM
} YVC_INIT_SP_VOLUME;

typedef    struct {
      YVC_INIT_SP_VOLUME       type;
} YVC_ARG_InitSpVolume;
```

18）YVC_CMD_SetPowerOnState / YVC_CMD_GetPowerOnState

```
/*  電源投入時の動作モードのパラメータ定義  */
typedef    enum {
      YVC_POWER_ON_STATE_HOLD = 0,
      YVC_POWER_ON_STATE_STANDBY,
      YVC_POWER_ON_STATE_RUN,
      YVC_POWER_ON_STATE_NUM
} YVC_POWER_ON_STATE;

typedef    struct {
      YVC_POWER_ON_STATE       mode;
} YVC_ARG_PowerOnState;
```

19）YVC_CMD_SetMicMuteMode / YVC_CMD_GetMicMuteMode

```
/*  マイクのミュード動作モードのパラメータ定義  */
typedef    enum {
      YVC_MIC_MUTE_MODE_GROUP = 0,
      YVC_MIC_MUTE_MODE_INDIVIDUAL,
      YVC_MIC_MUTE_MODE_NUM
} YVC_MIC_MUTE_MODE;

typedef    struct {
```

```
    YVC_MIC_MUTE_MODE       mode;
} YVC_ARG_MicMuteMode;
```

20）YVC_CMD_SetUsbSpeed / YVC_CMD_GetUsbSpeed

```
/* USB ポートの動作スピードのパラメータ定義 */
typedef    enum {
    YVC_USB_MODE_HI_SPEED = 0,
    YVC_USB_MODE_FULL_SPEED,
    YVC_USB_MODE_NUM
} YVC_USB_MODE;

typedef    struct {
    YVC_USB_MODE    mode;
    struct {
            void        (*callback)(YVC_CTRL_CMD cmd, int err, void *user);
            void        *user;
    } set;
} YVC_ARG_UsbSpeed;
```

21）YVC_CMD_SetExtTermMode / YVC_CMD_GetExtTermMode

```
/* EXT 端子の接続モードの設定を変更する定義 */
typedef    enum {
    YVC_EXTTERM_MODE_CASCADE = 0,
    YVC_EXTTERM_MODE_CONFERENCE_SYSTEM,
    YVC_EXTTERM_MODE_NUM
} YVC_EXTTERM_MODE;

typedef    struct {
    YVC_EXTTERM_MODE       mode;
} YVC_ARG_ExtTermMode;
```

22）YVC_CMD_SetSpOutSelect / YVC_CMD_GetSpOutSelect

```
/* スピーカー出力選択リストのパラメータ定義 */
typedef    enum {
    YVC_SP_OUT_SELECT_BUILTIN = 0,
    YVC_SP_OUT_SELECT_EXTERNAL,
    YVC_SP_OUT_SELECT_BOTH,
    YVC_SP_OUT_SELECT_NUM
} YVC_SP_OUT_SELECT;

typedef    struct {
    YVC_SP_OUT_SELECT    select;
} YVC_ARG_SpOutSelect;
```

23）YVC_CMD_SetAudioInTerm / YVC_CMD_GetAudioInTerm

```
/* オーディオ入力端子のモードのパラメータ定義 */
typedef    enum {
    YVC_AUDIO_IN_MODE_CONFERENCE = 0,
    YVC_AUDIO_IN_MODE_PA_MIC,
    YVC_AUDIO_IN_MODE_NUM
} YVC_AUDIO_IN_MODE;

/* 外部マイクのレベルのパラメータ定義 */
typedef    enum {
    YVC_AUDIO_IN_LEVEL_LINE = 0,
    YVC_AUDIO_IN_LEVEL_MIC,
    YVC_AUDIO_IN_LEVEL_NUM
} YVC_AUDIO_IN_LEVEL;

/* 外部マイクの EQ モードのパラメータ定義 */
typedef    enum {
    YVC_AUDIO_IN_EQ_THRU = 0,
    YVC_AUDIO_IN_EQ_LOW_BOOST,
    YVC_AUDIO_IN_EQ_LOW_CUT,
```

```
    YVC_AUDIO_IN_EQ_NUM
} YVC_AUDIO_IN_EQ;

/* 外部マイクのスピーカー出力の設定のパラメータ定義 */
typedef  enum {
    YVC_AUDIO_IN_SP_OUT_OFF = 0,
    YVC_AUDIO_IN_SP_OUT_ON,
    YVC_AUDIO_IN_SP_OUT_NUM
} YVC_AUDIO_IN_SP_OUT;

typedef  struct {
    struct {
        YVC_AUDIO_IN_MODE          mode;
        struct {
            YVC_AUDIO_IN_LEVEL     level;
            YVC_AUDIO_IN_EQ        eq;
            YVC_AUDIO_IN_SP_OUT    spOut;
            short                  gain;
        } paMic;
    } Lch, Rch;
} YVC_ARG_AudioInTerm;
```

24）YVC_CMD_SetAudioVolume / YVC_CMD_GetAudioVolume

```
/* ボリューム操作のチャネル定義 */
typedef  enum {
    YVC_AUDIO_VOLUME_CH_SP = 0,
    YVC_AUDIO_VOLUME_CH_MIC,
    YVC_AUDIO_VOLUME_CH_USB_IN,
    YVC_AUDIO_VOLUME_CH_USB_OUT,
    YVC_AUDIO_VOLUME_CH_BT_IN,
    YVC_AUDIO_VOLUME_CH_BT_OUT,
    YVC_AUDIO_VOLUME_CH_AUDIO_IN,
    YVC_AUDIO_VOLUME_CH_AUDIO_OUT,
    YVC_AUDIO_VOLUME_CH_SP_BUILTIN,
    YVC_AUDIO_VOLUME_CH_SP_EXTERNAL,
    YVC_AUDIO_VOLUME_CH_SP_EXMIC,
    YVC_AUDIO_VOLUME_CH_EXMIC,
    YVC_AUDIO_VOLUME_CH_HS_IN,
    YVC_AUDIO_VOLUME_CH_HP_OUT,
    YVC_AUDIO_VOLUME_CH_NUM
} YVC_AUDIO_VOLUME_CH;

typedef  struct {
    YVC_AUDIO_VOLUME_CH    ch;
    short                  volume;
} YVC_ARG_AudioVolume;
```

25）YVC_CMD_SetAudioMute / YVC_CMD_GetAudioMute

```
/* ミュート操作のチャネル定義 */
typedef  enum {
    YVC_AUDIO_MUTE_CH_SP = 0,
    YVC_AUDIO_MUTE_CH_MIC,
    YVC_AUDIO_MUTE_CH_USB_IN,
    YVC_AUDIO_MUTE_CH_USB_OUT,
    YVC_AUDIO_MUTE_CH_BT_IN,
    YVC_AUDIO_MUTE_CH_BT_OUT,
    YVC_AUDIO_MUTE_CH_AUDIO_IN,
    YVC_AUDIO_MUTE_CH_AUDIO_OUT,
    YVC_AUDIO_MUTE_CH_SP_BUILTIN,
    YVC_AUDIO_MUTE_CH_SP_EXTERNAL,
    YVC_AUDIO_MUTE_CH_EXMIC,
    YVC_AUDIO_MUTE_CH_HS_IN,
    YVC_AUDIO_MUTE_CH_HP_OUT,
    YVC_AUDIO_MUTE_CH_NUM
} YVC_AUDIO_MUTE_CH;

typedef  struct {
    YVC_AUDIO_MUTE_CH    ch;
    BOOL                 mute;
```

```
} YVC_ARG_AudioMute;
```

26）YVC_CMD_SetMicAgc / YVC_CMD_GetMicAgc

```
/* マイク AGC 動作の選択リストのパラメータ定義 */
typedef    enum {
      YVC_MIC_AGC_MODE_ON = 0,
      YVC_MIC_AGC_MODE_OFF,
      YVC_MIC_AGC_MODE_NUM
} YVC_MIC_AGC_MODE;

typedef    struct {
      YVC_MIC_AGC_MODE       mode;
} YVC_ARG_MicAgc;
```

27）YVC_CMD_SetMicUnitTrack / YVC_CMD_GetMicUnitTrack

```
/* マイク MUX(マルチプレクサ)動作の選択リストの定義 */
typedef    enum {
    YVC_MIC_MUX_MODE_TRACK = 0,
    YVC_MIC_MUX_MODE_MIX,
    YVC_MIC_MUX_MODE_NUM
} YVC_MIC_MUX_MODE;

typedef    struct {
      YVC_MIC_MUX_MODE     mode;
      unsigned char                unitSelect;
} YVC_ARG_MicUnitTrack;
```

28）YVC_CMD_SetMicCapsuleTrack / YVC_CMD_GetMicCapsuleTrack

```
/* マイク MUX(マルチプレクサ)動作の選択リストの定義 */
typedef    enum {
    YVC_MIC_MUX_MODE_TRACK = 0,
    YVC_MIC_MUX_MODE_MIX,
    YVC_MIC_MUX_MODE_NUM
} YVC_MIC_MUX_MODE;

typedef    struct {
      YVC_MIC_MUX_MODE     mode;
      unsigned char                capsuleSelect;
} YVC_ARG_MicCapsuleTrack;
```

29）YVC_CMD_SetMicFilter / YVC_CMD_GetMicFilter

```
/* マイクの周波数特性の選択リストのパラメータ定義 */
typedef    enum {
      YVC_MIC_FILTER_MODE_OFF = 0,
      YVC_MIC_FILTER_MODE_LOW,
      YVC_MIC_FILTER_MODE_MID,
      YVC_MIC_FILTER_MODE_HIGH,
      YVC_MIC_FILTER_MODE_NUM
} YVC_MIC_FILTER_MODE;

typedef    struct {
      YVC_MIC_FILTER_MODE       mode;
} YVC_ARG_MicFilter;
```

30）YVC_CMD_SetSpFilter / YVC_CMD_GetSpFilter

```
/* スピーカーの周波数特性の選択リストのパラメータ定義 */
typedef    enum {
      YVC_SP_FILTER_MODE_OFF = 0,
      YVC_SP_FILTER_MODE_LOW,
      YVC_SP_FILTER_MODE_MID,
      YVC_SP_FILTER_MODE_HIGH,
      YVC_SP_FILTER_MODE_NUM
```

```
} YVC_SP_FILTER_MODE;

typedef    struct {
      YVC_SP_FILTER_MODE       mode;
} YVC_ARG_SpFilter;
```

31）YVC_CMD_ExeAAT

```
/* 自動音響調整の実行操作リストのパラメータ定義 */
typedef    enum {
      YVC_AAT_ACTION_STOP = 0,
      YVC_AAT_ACTION_START,
      YVC_AAT_NUM
} YVC_AAT_ACTION;

typedef    struct {
      YVC_AAT_ACTION       action;
} YVC_ARG_AAT;
```

32）YVC_CMD_GetAATProgress

```
typedef    struct {
      unsigned int     progress;
} YVC_ARG_AATProgress;
```

33）YVC_CMD_GetAATResult

```
/* 自動音響調整の測定結果のパラメータ定義 */
typedef enum {
    YVC_AAT_RESULT_OK = 0,
    YVC_AAT_RESULT_CANCEL,
    YVC_AAT_RESULT_WARNING,
    YVC_AAT_RESULT_NO_MIC,
    YVC_AAT_RESULT_SP_VOL_SMALL,
    YVC_AAT_RESULT_EXT_SP_UNDETECT,
    YVC_AAT_RESULT_NUM
} YVC_AAT_RESULT;

typedef    struct {
    YVC_AAT_RESULT  result;
    unsigned int              warning;
} YVC_ARG_AATResult;
```

34）YVC_CMD_MonMicCapsule

```
typedef    struct {
      unsigned char     trackInfo[YVC_LEN_MIC_UNIT_MAX];
} YVC_ARG_MonMicCapsule;
```

35）YVC_CMD_MonMicUnit

```
typedef    struct {
      int     unitId;
} YVC_ARG_MonMicUnit;
```

36） YVC_CMD_MonHeadsetPortState

```
typedef    struct {
      unsigned int      state;
} YVC_ARG_MonHeadsetPortState
```

37） YVC_CMD_MonBatteryLevel

```
typedef    struct {
      unsigned int      level;
```

```
    unsigned int      charging;
} YVC_ARG_MonBatteryLevel
```

4.7-9 定数

API バージョン定義

シンボル名	値	内容
YVC_LEN_API_VERSION	4	API バージョンの文字数

YVC シリーズ共通定義

シンボル名	値	内容
YVC_EVENT_DATA_MICMUTE_UNIT0	0x1	マイクユニット 0
YVC_EVENT_DATA_MICMUTE_UNIT1	0x2	マイクユニット 1
YVC_EVENT_DATA_MICMUTE_UNIT2	0x4	マイクユニット 2
YVC_EVENT_DATA_MICMUTE_UNIT3	0x8	マイクユニット 3
YVC_EVENT_DATA_MICMUTE_UNIT4	0x10	マイクユニット 4
YVC_EVENT_DATA_HOOKBTN_SHORT	0	フックボタン短押し
YVC_EVENT_DATA_HOOKBTN_LONG	1	フックボタン長押し
YVC_PRM_NOTICE_EVENT_MIC_UNIT_NUM	0x1	マイクユニット接続状態変化イベント
YVC_PRM_NOTICE_EVENT_MIC_MUTE_STATE	0x2	マイクミュート状態変化イベント
YVC_PRM_NOTICE_EVENT_HOOK_BTN	0x4	フックボタン押下状態変化イベント
YVC_PRM_MIC_UNIT0	0x1	マイクユニット 0
YVC_PRM_MIC_UNIT1	0x2	マイクユニット 1
YVC_PRM_MIC_UNIT2	0x4	マイクユニット 2
YVC_PRM_MIC_UNIT3	0x8	マイクユニット 3
YVC_PRM_MIC_UNIT4	0x10	マイクユニット 4
YVC_PRM_MIC_CAPSULE_NONE	0x0	マイクカプセル非アクティブ
YVC_PRM_MIC_CAPSULE_CENTER	0x1	マイクカプセル（中）
YVC_PRM_MIC_CAPSULE_LEFT	0x2	マイクカプセル（左）
YVC_PRM_MIC_CAPSULE_RIGHT	0x4	マイクカプセル（右）
YVC_PRM_AAT_WARNING_OK	0x0	自動音響調整は正常終了
YVC_PRM_AAT_WARNING_TOOCLOSE	0x1	自動音響調整エラー：マイクユニットと本体の距離が近すぎる
YVC_PRM_AAT_WARNING_TOOFAR	0x2	自動音響調整エラー：マイクユニットと本体の距離が遠すぎる
YVC_PRM_AAT_WARNING_LONGDELAY	0x4	自動音響調整エラー：外部スピーカーのディレイが大きすぎる
YVC_PRM_AAT_WARNING_DISTORTION	0x8	自動音響調整エラー：外部スピーカーの歪が大きすぎる
YVC_LEN_VERSION_INFO	4	ファームウェアバージョンの文字数
YVC_LEN_MODEL_NAME	40	モデル名称の最大文字数
YVC_LEN_SERIAL_NUMBER	16	シリアル No.の最大文字数
YVC_LEN_BT_LOCAL_NAME	11	Bluetooth の Local Name の文字数
YVC_LEN_BT_LOCAL_NAME_TOTAL	20	Bluetooth の Local Name の全文字数（機種名含む）
YVC_LEN_MIC_UNIT_MAX	5	マイクユニット接続可能最大数

YVC-1000 定義

シンボル名	値	内容
YVC1000_LEN_MIC_UNIT_MAX	5	マイクユニット接続可能最大数

YVC-300/YVC-330/YVC-331 定義

シンボル名	値	内容
YVC300_LEN_MIC_UNIT_MAX	2	マイクユニット（本体連結）接続可能最大数

4.8 エラー定義

API のエラー定義について示します。

4.8-1 エラーコード

エラーコード	値	内容
YVCAPI_ERROR_PENDING	1	要求受付（実行中）
YVCAPI_ERROR_NOERROR	0	正常終了
YVCAPI_ERROR_UNKNOWN	-1	異常終了（その他のエラー）
YVCAPI_ERROR_DEVNOTFOUND	-2	指定された機器が見つからない
YVCAPI_ERROR_MEMEXHAUST	-3	メモリが不足している
YVCAPI_ERROR_STATUS	-4	状態エラーが発生した
YVCAPI_ERROR_INVALIDCOMMAND	-5	制御コマンドのコードが不正
YVCAPI_ERROR_INVALIDHANDLE	-6	ハンドルが不正
YVCAPI_ERROR_INVALIDARGS	-7	引数が不正
YVCAPI_ERROR_CTRLEXEC	-8	制御の実行に失敗した
YVCAPI_ERROR_HIDTIMEOUT	-9	HID 通信において、タイムアウトが発生した
YVCAPI_ERROR_UNSUPPORTVER	-15	未サポートのファームウェアバージョン
YVCAPI_ERROR_APPRUNNING	-16	他の関連アプリケーションが起動中のため、実行できない
YVCAPI_ERROR_INVALIDPARAM	-17	機器のパラメータ値が不正

4.9 制限事項

本 API を使用するにあたっての制限事項を記述します。

- API のバージョン 1.01 以下では、YVC シリーズの機器をホスト PC に 1 台接続した場合のみ動作対象とします。
- ホスト PC 上に既に他の YVC シリーズのアプリケーションが起動されている状態で、YVC_DeviceAttach 関数や YVC_DeviceControl 関数を呼び出した場合は、エラー（YVCAPI_ERROR_APPRUNNING）を返します。

Windows

機種名	アプリケーション名	ファイル名
YVC-1000	YVC-1000 Configurator	*YVC-1000_Configurator.exe*
	YVC-1000 FW-Updater	*YVC-1000_Firmware_<version>.exe*
YVC-300	YVC-300 FW-Updater	*YVC-300_Firmware_<version>.exe*
YVC-330	YVC-330 FW-Updater	*YVC-330_Firmware_<version>.exe*
YVC-331	YVC-331 FW-Updater	*YVC-331_Firmware_<version>.exe*
YVC-200	YVC-200 FW-Updater	*YVC-200_Firmware_<version>.exe*

Mac

機種名	アプリケーション名	ファイル名
YVC-1000	YVC-1000 Configurator	*YVC-1000_Configurator.app*
	YVC-1000 FW-Updater	*YVC-1000_Firmware_<version>.app*
YVC-300	YVC-300 FW-Updater	*YVC-300_Firmware_<version>.app*
YVC-330	YVC-330 FW-Updater	*YVC-330_Firmware_<version>.app*
YVC-331	YVC-331 FW-Updater	*YVC-331_Firmware_<version>.app*
YVC-200	YVC-200 FW-Updater	*YVC-200_Firmware_<version>.app*

4.10 注意事項

本 API を使用するにあたっての注意事項を記述します。

- YVC-300/YVC-330/YVC-331 を連結接続で動作している場合、コマンドは親機のみに対して実行されます。

5. API 使用例

API の使用例を示します。

※Mac の場合、予め、プロジェクトに「YVCAPI.framework」を追加しておく必要があります。

```
#if (defined(_WIN32) || defined(_WIN64))
#include <windows.h>
#else /* MAC */
#import <Foundation/Foundation.h>
#endif
 :
#include <YVCAPI.h>

#if (defined(_WIN32) || defined(_WIN64))
typedef int     (__stdcall *FNC_YVC_DEVICE_INFO)(YVC_DEV_TYPE type,
                                                 YVC_DEV_INFO *info);
typedef int     (__stdcall *FNC_YVC_DEVICE_ATTACH)(YVC_DEV_HANDLE *handle,
                                                   YVC_DEV_TYPE type,
                                                   YVC_EVENT_CALLBACK callback,
                                                   void *user);
typedef int     (__stdcall *FNC_YVC_DEVICE_CONTROL)(YVC_DEV_HANDLE handle,
                                                    YVC_CTRL_CMD cmd,
                                                    YVC_CTRL_ARG *arg);
typedef int     (__stdcall *FNC_YVC_DEVICE_DETACH)(YVC_DEV_HANDLE handle);
typedef int     (__stdcall *FNC_YVC_API_INFO)(YVC_API_INFO *info);
HMODULE                         hModule
#else /* MAC */
typedef int     (*FNC_YVC_DEVICE_INFO)(YVC_DEV_TYPE type,
                                       YVC_DEV_INFO *info);
typedef int     (*FNC_YVC_DEVICE_ATTACH)(YVC_DEV_HANDLE *handle,
                                         YVC_DEV_TYPE type,
                                         YVC_EVENT_CALLBACK callback,
                                         void *user);
typedef int     (*FNC_YVC_DEVICE_CONTROL)(YVC_DEV_HANDLE handle,
                                          YVC_CTRL_CMD cmd,
                                          YVC_CTRL_ARG *arg);
typedef int     (*FNC_YVC_DEVICE_DETACH)(YVC_DEV_HANDLE handle);
typedef int     (*FNC_YVC_API_INFO)(YVC_API_INFO *info);
#endif
FNC_YVC_DEVICE_INFO             DeviceInfo;
FNC_YVC_DEVICE_ATTACH           DeviceAttach;
FNC_YVC_DEVICE_CONTROL          DeviceControl;
FNC_YVC_DEVICE_DETACH           DeviceDetach;
FNC_YVC_API_INFO                APIInfo;
```

```
/* YVCAPI モジュールハンドルを取得する（アドレス空間にマッピング） */
int YVCAPISample1(void)
{
#if (defined(_WIN32) || defined(_WIN64))
	hModule = LoadLibrary(_T("YVCAPI.dll"));
	if (hModule == NULL) {
		return -1;
	}
	DeviceInfo = (FNC_YVC_DEVICE_INFO)GetProcAddress(hModule, "YVC_DeviceInfo");
	if (DeviceInfo == NULL) {
		return -1;
	}
	DeviceAttach = (FNC_YVC_DEVICE_ATTACH)GetProcAddress(hModule, "YVC_DeviceAttach");
	if (DeviceAttach == NULL) {
		return -1;
	}
	DeviceDetach = (FNC_YVC_DEVICE_DETACH)GetProcAddress(hModule, "YVC_DeviceDetach");
	if (DeviceDetach == NULL) {
		return -1;
	}
	DeviceControl = (FNC_YVC_DEVICE_CONTROL)GetProcAddress(hModule, "YVC_DeviceControl");
	if (DeviceControl == NULL) {
		return -1;
	}
	APIInfo = (FNC_YVC_API_INFO)GetProcAddress(hModule, "YVC_APIInfo");
	if (APIInfo == NULL) {
		return -1;
	}
#else /* MAC */
	DeviceInfo = YVC_DeviceInfo;
	DeviceAttach = YVC_DeviceAttach;
	DeviceDetach = YVC_DeviceDetach;
	DeviceControl = YVC_DeviceControl;
	APIInfo = YVC_APIInfo;
	if (DeviceInfo == NULL || DeviceAttach == NULL || DeviceDetach == NULL || DeviceControl == NULL || APIInfo ==
NULL) {
		return -1;
	}
#endif
    return 0; /* 正常終了 */
}


#if (defined(_WIN32) || defined(_WIN64))
/* YVCAPI モジュールハンドルを解放する（マッピングを解除） */
int YVCAPISample2(void)
{
	if (hModule) {
		FreeLibrary(hModule);
	}
	return 0; /* 正常終了 */
}
#endif
```

```
YVC_DEV_HANDLE handle = NULL;

/* YVC-1000 との通信を開始する */
int YVCAPISample3(void)
{
	int ret;

	/* API 情報(バージョン)を取得する */
	YVC_API_INFO	apiInfo;
	ret = APIInfo(&apiInfo);
	if (ret != YVCAPI_ERROR_NOERROR) {
		return -1;
	}
	/* 取得したバージョンを表示する */
	printf("API Version : %s¥n", apiInfo.version);

	/* デバイス接続情報を取得する */
	YVC_DEV_INFO devInfo;
	/* YVC-1000 の接続台数を取得する */
	ret = DeviceInfo(YVC_DEV_TYPE_YVC1000, &devInfo);
	if (ret != YVCAPI_ERROR_NOERROR) {
		return -1;
	}
	if (devInfo.num < 1) {
		return -1;/* YVC-1000 が接続されるまで待つ */
	}

	/* YVC-1000 との通信ハンドルを取得する */
	ret = DeviceAttach(&handle, YVC_DEV_TYPE_YVC1000, YVCAPISample6, NULL);
	if (ret != YVCAPI_ERROR_NOERROR || handle == NULL) {
		return -1;
	}

	/* YVC-1000 の FW バージョンを取得する */
	YVC_ARG_VersionInfo VersionInfo;
	ret = DeviceControl(handle, YVC_CMD_GetVersionInfo, (YVC_CTRL_ARG *)&VersionInfo);
	if (ret != YVCAPI_ERROR_NOERROR) {
		goto END;
	}
	/* 取得したバージョンを表示する */
	printf("YVC-1000 FW Version : %s¥n", VersionInfo.version);

	/* モードを取得する */
	YVC_ARG_SystemMode	SystemMode;
	ret = DeviceControl(handle, YVC_CMD_GetSystemMode, (YVC_CTRL_ARG *)&SystemMode);
	if (ret != YVCAPI_ERROR_NOERROR) {
		goto END;
	}
	/* 取得したモードを表示する */
	printf("Mode : %s¥n", SystemMode.mode == YVC_SYSTEM_MODE_NORMAL ? "NORMAL" : "FWUP");

	:

END:
	return ret;
}


/* YVC-1000 との通信を終了する */
int YVCAPISample4(void)
{
	int ret;

	/* 通信終了 */
	if (handle != NULL) {
		ret = DeviceDetach(handle);
	}
	return ret;
}
```

```
/* YVC-1000 の音声ガイダンス言語を「日本語」に設定する */
int YVCAPISample5(void)
{
	int ret;

	YVC_ARG_VoiceGuideLang	VoiceGuideLang;
	/* 音声ガイダンス言語の設定を取得する */
	ret = DeviceControl(handle, YVC_CMD_GetVoiceGuideLang, (YVC_CTRL_ARG *)&VoiceGuideLang);
	if (ret != YVCAPI_ERROR_NOERROR) {
		return ret;
	}
	if (VoiceGuideLang.type != YVC_VOICE_GUIDE_LANG_JA) {
		/* 「日本語」に設定する */
		VoiceGuideLang.type = YVC_VOICE_GUIDE_LANG_JA;
		ret = DeviceControl(handle, YVC_CMD_SetVoiceGuideLang, (YVC_CTRL_ARG *)&VoiceGuideLang);
		if (ret == YVCAPI_ERROR_NOERROR) {
			printf("日本語に設定しました¥n");
		}
	}

	return ret;
}


/* イベント通知用コールバック関数 */
void YVCAPISample6(YVC_EVENT event, int data, void *user)
{
	WPARAM wParam;
	LPARAM lParam;

	switch(event) {
	case YVC_EVENT_DEV_DISCONNECT:
		/* 機器が取り外された */
		break;
	case YVC_EVENT_MIC_UNIT_NUM:
		/* マイクユニットの接続状態が変化 */
		break;
	case YVC_EVENT_MIC_MUTE_STATE:
		/* マイクミュートの状態が変化 */
		break;
	default:
		return;
	}

#if (defined(_WIN32) || defined(_WIN64))
	wParam = (WPARAM)event;
	lParam = (LPARAM)data;
	/* ポスト先ウィンドウ・ハンドルを指定して、メッセージをポストする */
	::PostMessage(hWnd, WM_APP, wParam, lParam);
#else /* MAC */
	NSDictionary *dict = [NSDictionary dictionaryWithObjectsAndKeys:
				[NSNumber numberWithInt: (int)event], @"event",
				[NSNumber numberWithInt: data], @"data",
				nil
			];
	/* ポスト先を指定して、メッセージをポストする */
	[[NSNotificationCenter defaultCenter] postNotificationName: @"SAMPLE"
							object: observer
						      userInfo: dict];
#endif

}
```

6. 付録

6.1 コマンドの対応モードと対応ファームウェアー覧

1）システム制御関連

コマンド定義名	YVC-1000		YVC-300		YVC-330		YVC-331		YVC-200	
	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP
YVC_CMD_GetSystemMode	2.00 以降	←	1.04 以降	←	1.02 以降	←	1.00 以降	←	1.01 以降	←
YVC_CMD_GetVersionInfo	↑	←	↑	←	↑	←	↑	←	↑	←
YVC_CMD_GetModelName	↑	←	↑	←	↑	←	↑	←	↑	←
YVC_CMD_GetSerialNumber	↑	←	↑	←	↑	←	↑	←	↑	←
YVC_CMD_GetMicNum	↑									
YVC_CMD_ExeSystemReboot	↑	←	1.04 以降	←	1.02 以降	←	1.00 以降	←	1.01 以降	←
YVC_CMD_ExeFactoryReset	↑		↑		↑		↑		↑	
YVC_CMD_ExeStandbyMode	↑								↑	
YVC_CMD_SetNoticeEvent	2.10 以降		1.09 以降		1.02 以降		1.00 以降		↑	
YVC_CMD_GetNoticeEvent	↑		↑		↑		↑		↑	
YVC_CMD_SetBTUse	2.00 以降		1.04 以降		↑				↑	
YVC_CMD_GetBTUse	↑		↑		↑				↑	
YVC_CMD_SetBTLocalName	2.09 以降		1.08 以降		↑				↑	
YVC_CMD_GetBTLocalName	↑		↑		↑				↑	
YVC_CMD_SetBTOperation	2.00 以降		1.04 以降		↑				↑	
YVC_CMD_GetBTOperation	↑		↑		↑				↑	
YVC_CMD_SetNoticeSoundUse	↑		↑		↑		1.00 以降		↑	
YVC_CMD_GetNoticeSoundUse	↑		↑		↑		↑		↑	
YVC_CMD_SetNoticeSoundVolume			↑		↑		↑		↑	
YVC_CMD_GetNoticeSoundVolume			↑		↑		↑		↑	
YVC_CMD_SetVoiceGuideUse	2.00 以降									
YVC_CMD_GetVoiceGuideUse	↑									
YVC_CMD_SetVoiceGuideLang	↑									
YVC_CMD_GetVoiceGuideLang	↑									
YVC_CMD_SetVoiceGuideVolume	↑									
YVC_CMD_GetVoiceGuideVolume	↑									
YVC_CMD_SetInitSpVolume	↑		1.04 以降		1.02 以降		1.00 以降		1.01 以降	
YVC_CMD_GetInitSpVolume	↑		↑		↑		↑		↑	
YVC_CMD_SetPowerOnState	↑									
YVC_CMD_GetPowerOnState	↑									
YVC_CMD_SetMicMuteMode	2.10 以降									
YVC_CMD_GetMicMuteMode	↑									
YVC_CMD_SetUsbSpeed	2.03 以降									
YVC_CMD_GetUsbSpeed	↑									
YVC_CMD_SetExtTermMode			1.04 以降		1.02 以降		1.00 以降			
YVC_CMD_GetExtTermMode			↑		↑		↑			

2）オーディオ制御関連

コマンド定義名	YVC-1000		YVC-300		YVC-330		YVC-331		YVC-200	
	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP
YVC_CMD_SetSpOutSelect	2.00 以降		1.04 以降		1.02 以降		1.00 以降			
YVC_CMD_GetSpOutSelect	↑		↑		↑		↑			
YVC_CMD_SetAudioInTerm	↑		↑		↑		↑			
YVC_CMD_GetAudioInTerm	↑		↑		↑		↑			
YVC_CMD_SetAudioVolume	↑		1.09 以降		↑		↑		1.01 以降	
YVC_CMD_GetAudioVolume	↑		1.04 以降		↑		↑		↑	
YVC_CMD_SetAudioMute	↑		↑		↑		↑		↑	
YVC_CMD_GetAudioMute	↑		↑		↑		↑		↑	
YVC_CMD_SetMicAgc	↑		↑		↑		↑		↑	
YVC_CMD_GetMicAgc	↑		↑		↑		↑		↑	
YVC_CMD_SetMicUnitTrack	↑									
YVC_CMD_GetMicUnitTrack	↑									
YVC_CMD_SetMicCapsuleTrack			1.09 以降		1.02 以降		1.00 以降			
YVC_CMD_GetMicCapsuleTrack			↑		↑		↑			
YVC_CMD_SetMicFilter	2.00 以降		↑		↑		↑		1.01 以降	
YVC_CMD_GetMicFilter	↑		↑		↑		↑		↑	
YVC_CMD_SetSpFilter	↑		↑		↑		↑			
YVC_CMD_GetSpFilter	↑		↑		↑		↑			

3）自動音響調整制御関連

コマンド定義名	YVC-1000		YVC-300		YVC-330		YVC-331		YVC-200	
	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP
YVC_CMD_ExeAAT	2.10 以降									
YVC_CMD_GetAATProgress	↑									
YVC_CMD_GetAATResult	↑									

4）機器のモニタリング関連

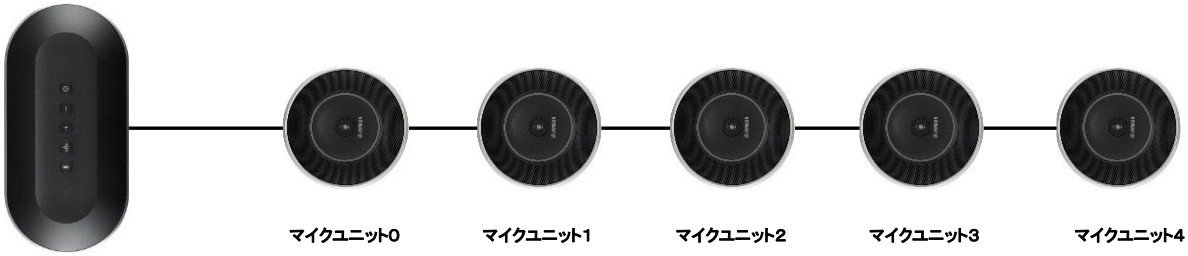
コマンド定義名	YVC-1000		YVC-300		YVC-330		YVC-331		YVC-200	
	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP	NORMAL	FWUP
YVC_CMD_MonMicCapsule	2.00 以降		1.04 以降		1.02 以降		1.00 以降			
YVC_CMD_MonMicUnit	2.10 以降									
YVC_CMD_MonHeadsetPortState									1.01 以降	
YVC_CMD_MonBatteryLevel									↑	

6.2 マイクカプセル位置

各機器のマイクカプセルの位置を示します。

1）YVC-1000

以下の図は、YVC-1000 のマイクユニット ID を示します。
本体側に接続されたマイクユニットから順番に 0 から割り振られます。



以下の図は、YVC-1000 のマイクユニットのマイクカプセルの位置を示します。



2）YVC-300/YVC-330/YVC-331

以下の図は、YVC-300/YVC-330/YVC-331 のマイクカプセルの位置を示します。



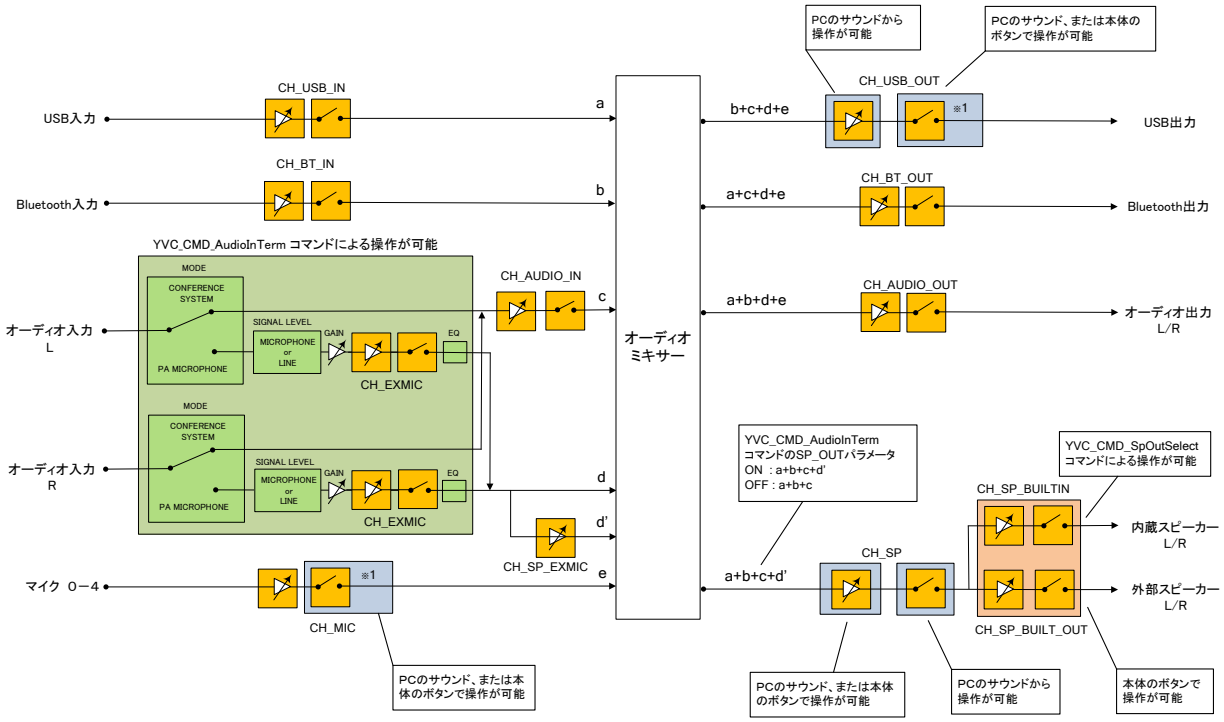
6.3 オーディオ・ブロック図

各機器のオーディオ・ブロックを示します。

1）YVC-1000

以下に YVC-1000 のオーディオ・ブロック図を示します。

【Ver 3.00 以降】

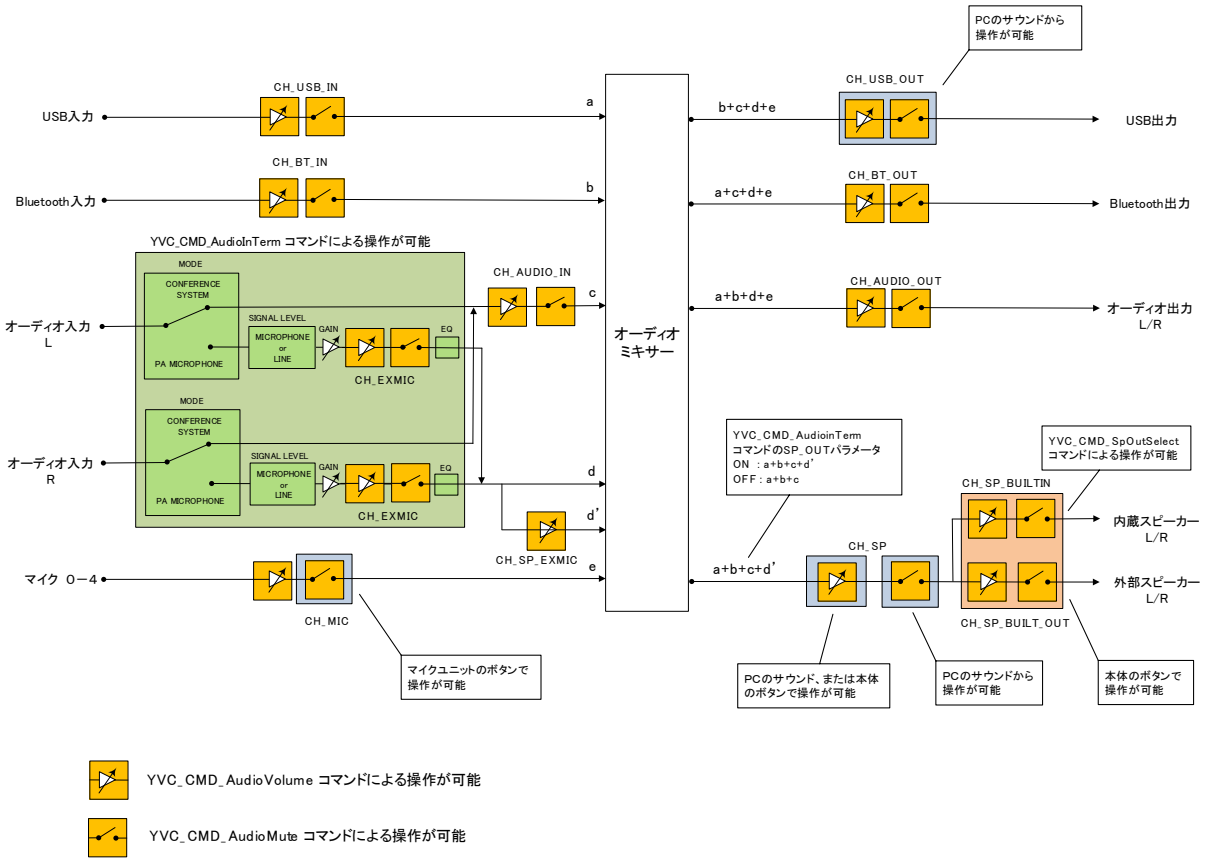


YVC_CMD_AudioVolume コマンドによる操作が可能

YVC_CMD_AudioMute コマンドによる操作が可能

※1
２つのマイクミュートは連動します。
ただし、YVC_CMD_AudioMuteコマンドで操作した場合は、連動しません。
また、YVC-1000 Configurator で非連動に設定した場合の動作は、ファームウェア Ver.2.11 と同等になります。

【Ver 2.11 以前】



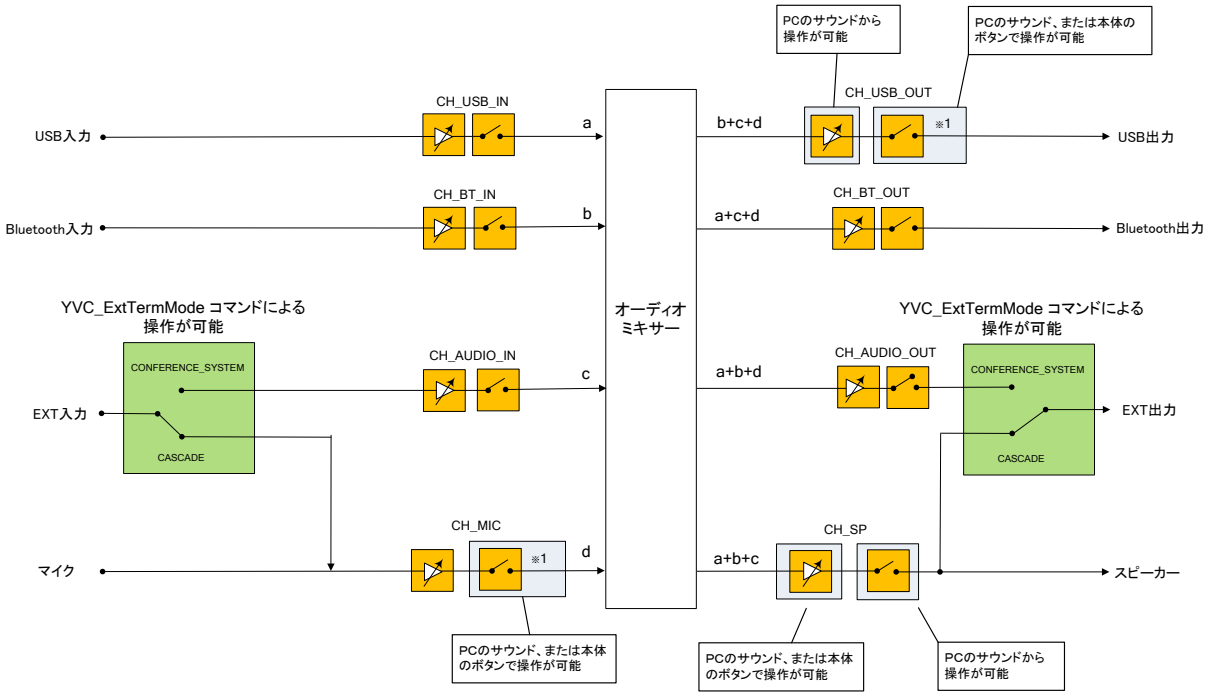
YVC_CMD_AudioVolume コマンドによる操作が可能

YVC_CMD_AudioMute コマンドによる操作が可能

2）YVC-300

以下に YVC-300 のオーディオ･ブロック図を示します。

【Ver2.00 以降】

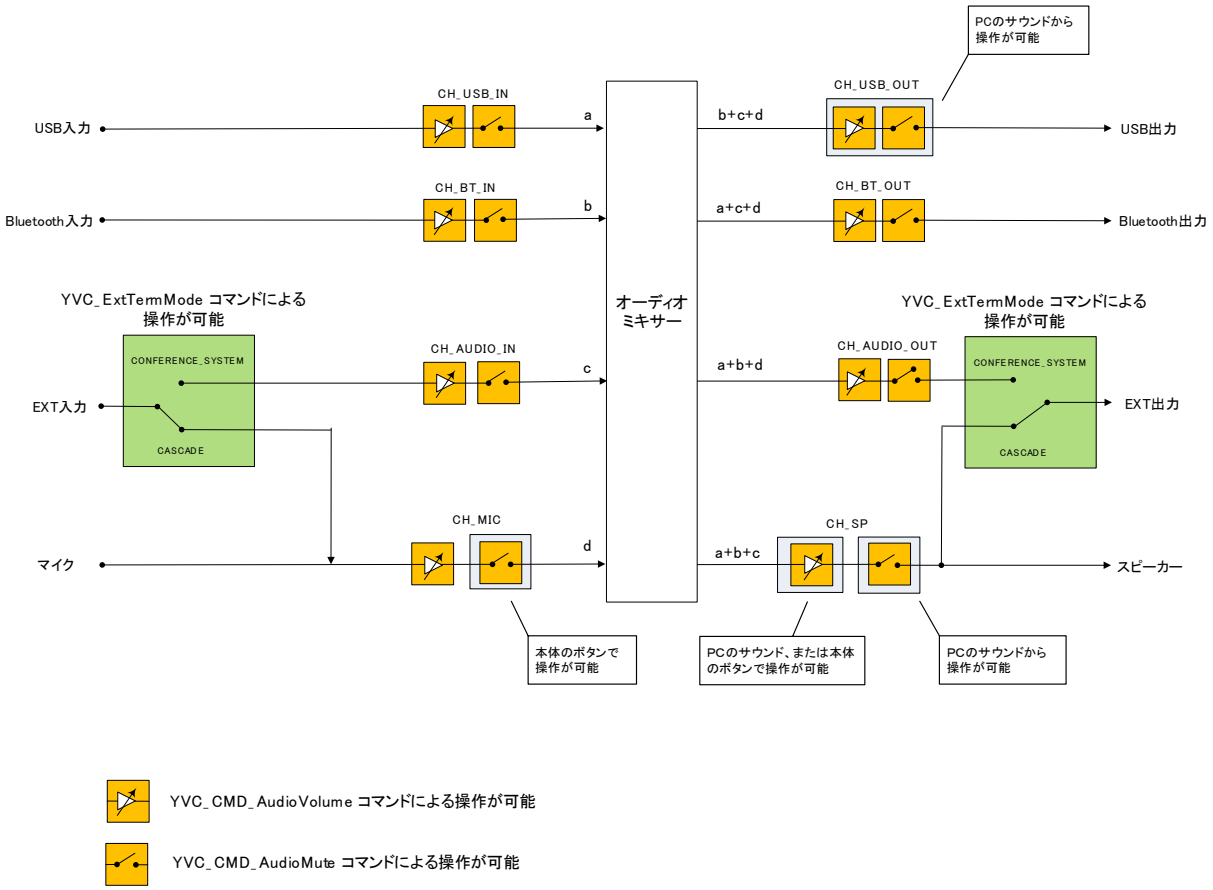


YVC_CMD_AudioVolume コマンドによる操作が可能

YVC_CMD_AudioMute コマンドによる操作が可能

※1
２つのマイクミュートは連動します。
ただし、YVC_CMD_AudioMuteコマンドで操作した場合は、連動しません。

【Ver 1.09 以前】

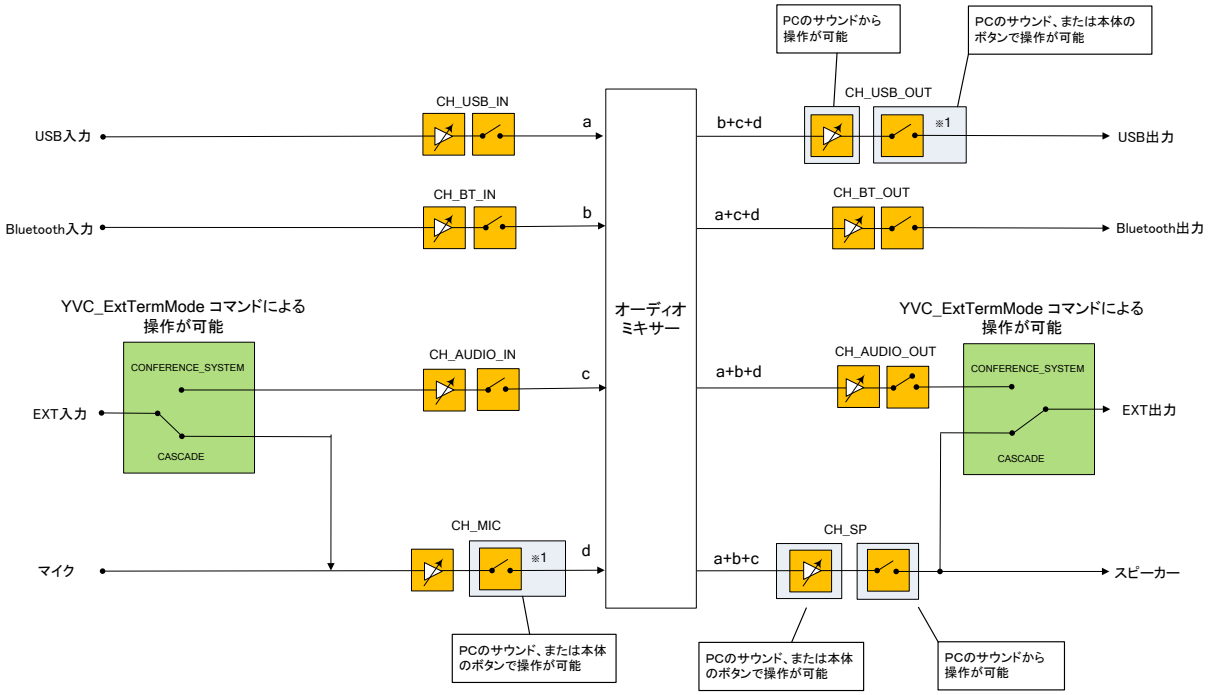


YVC_CMD_AudioVolume コマンドによる操作が可能

YVC_CMD_AudioMute コマンドによる操作が可能

3 〉YVC-330

以下に YVC-330 のオーディオ･ブロック図を示します。



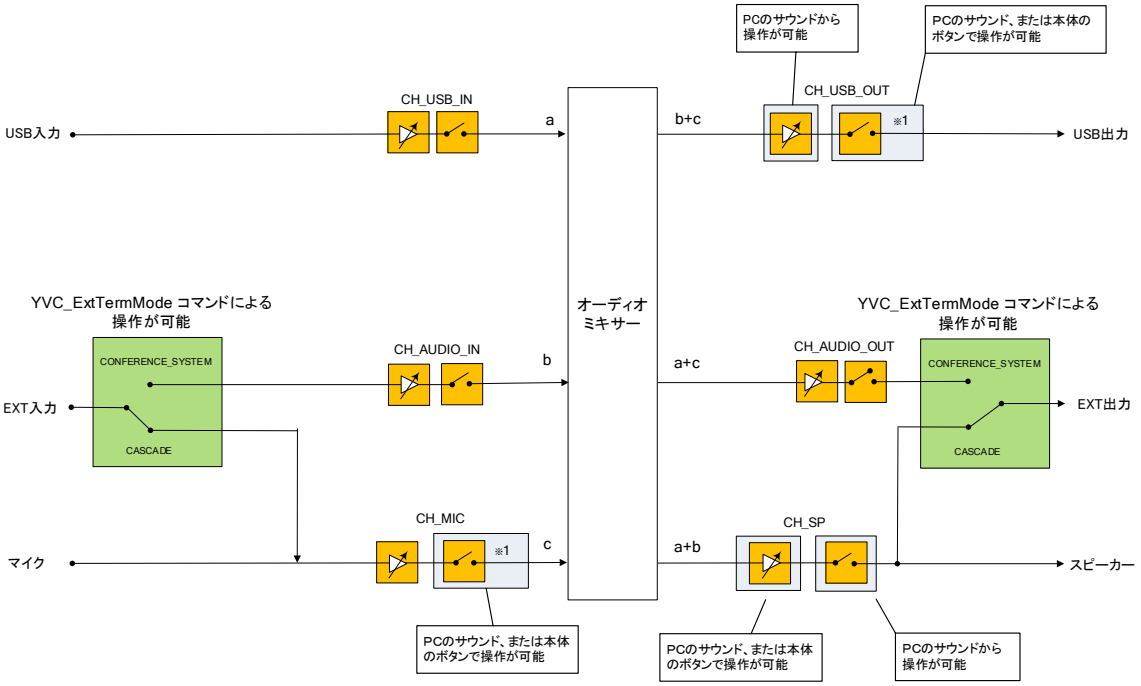
YVC_CMD_AudioVolume コマンドによる操作が可能

YVC_CMD_AudioMute コマンドによる操作が可能

※1
２つのマイクミュートは連動します。
ただし、YVC_CMD_AudioMuteコマンドで操作した場合は、連動しません。

4） YVC-331

以下に YVC-331 のオーディオ・ブロック図を示します。



YVC_CMD_AudioVolume コマンドによる操作が可能

YVC_CMD_AudioMute コマンドによる操作が可能

※1
２つのマイクミュートは連動します。
ただし、YVC_CMD_AudioMuteコマンドで操作した場合は、連動しません。

5） YVC-200

以下に YVC-200 のオーディオ・ブロック図を示します。

