

GSCTI ActiveX

機能説明書

第一版 2014/9/5

目次

・ インターフェース一覧.....	3
・ メソッド説明.....	5
・ イベント説明.....	12
・ 注意事項.....	19

メソッド一覧

No.	メソッド	内容	Page
1.1	Init	CTIServiceの初期化をします。	5
1.2	Term	CTIServiceのTaskの終了処理をします。	5
1.3	Dial	架電します。	6
1.4	ButtonEvent	電話機にボタンイベントを送ります。	6
1.5	SendDTMF	DTMFを送信します。	6
1.6	ParkHold	パーク保留コマンドを送信します。	7
1.7	ParkRetrieve	パーク保留解除コマンドを送信します。	7
1.8	EndCall	終話します。	7
1.9	StartRecord	録音を開始します。	8
1.10	StopRecord	録音を終了します。	8
1.11	setAutoRec	自動録音、手動録音の切り替えをします。	9
1.12	getRecFiles	全ての録音ファイルを電話機からPCへ移動します。	9
1.13	getRecFile	特定の録音ファイルを電話機からPCへ移動します。	9
1.14	moveRecFiles	全ての録音ファイルを保存先に移動します。	10
1.15	moveRecFile	特定の録音ファイルを保存先に移動します。	10
1.16	setTalkDeviceState	使用するオーディオデバイスを変更します。	11
1.17	setCallBlocking	アカウントの状態を変更します。	11
1.18	getCallLog	通話履歴の取得を開始します。	12
1.19	getCallLogNumber	通話履歴の連番を元に電話番号を取得します。	12
1.20	getCallLogDate	通話履歴の連番を元に日時を取得します。	12
1.21	getCallLogType	通話履歴の連番を元に履歴の種別を取得します。	13
1.22	getFunctionName	エラーコードを元にエラーが起こったメソッド名を取得します。	13
1.23	getErrorStr	エラーコードを元にどのようなエラーが起こったかを取得します。	13

イベント一覧

No.	イベント	内容	Page
2.1	DetectedCallStatusChange	呼状態の変化(通話相手番号、通話相手番号の登録名、使用アカウント名)を通知します。	14
2.2	DetectedDTMF	DTMF取得イベント。	14
2.3	DetectedParkCmdStateChanged	パーク保留状態変更イベント。	15
2.4	DetectedRecStatusChange	録音の状態変化を通知します。	15
2.5	DetectedGetRecordFiles	録音ファイル取得イベント。	15
2.6	DetectedMute	電話機のミュートの状態が変化した事を通知します。	16
2.7	DetectedRemoteOnMute	会議通話時、他の電話のミュートの状態が変化した事を通知します。	16
2.8	DetectedConference	会議通話の状態が変化した事を通知します。	17
2.9	DetectedVideoCall	ビデオ通話を発信した事を通知します。	17
2.10	DetectedVideoComing	ビデオ通話の要求が来た事を通知します。	17
2.11	DetectedTalkDeviceStatusChange	オーディオデバイスの状態変化を通知します。	18
2.12	DetectedGetCallLog	通話履歴の取得完了イベント。	18
2.13	DetectedFunctionError	メソッドの実行エラー取得イベント。	18

1.1 Init

用途

GrandStreamIP電話機の常駐サービスのステータスを初期化し、電話機の状態の取得を開始します。
Init()をしないと全ての関数はエラーを返す。イベントも上がらない。

構文

Long Init(LONG iAutoRec)

引数

iAutoRec ...0:手動録音 1:自動録音

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

1.2 Term

用途

GrandStreamIP電話機の常駐サービスの電話機の状態の取得を停止する。
Term()後、全ての関数はエラーを返す。イベントも上がらない。

構文

Long Term()

引数

なし

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

アプリケーション終了時には必ず本処理を行ってください。

1.3 Dial

用途

GrandStreamIP電話機から指定のアカウント番号でcNumberに発信します。

構文

Long Dial (CHAR* cNumber, LONG iAccountID)

引数

cNumber ...発信番号

iAccountID ...使用するアカウント番号(0~5)

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

Lineを指定することはできません。

使用するLineは未使用のLineの内一番小さいものになります。

1.4 ButtonEvent

用途

電話機にボタンイベントを送ります。

構文

Long ButtonEvent (LONG iKeyCode)

引数

iKeyCode ...91:ミュートボタン 1801:保留ボタン 1802:ヘッドセットボタン 1807:スピーカボタン
0~9:テンキー 11:* 12:#

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

現在使用できる引数は4種類です。

1.5 SendDTMF

用途

DTMFを送信します。

構文

LONG SendDTMF(CHAR* cDTMF)

引数

cDTMF ...送信DTMF(24桁まで)

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

1.6 ParkHold

用途

パーク保留コマンドを送信します。

構文

LONG ParkHold()

引数

なし

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

特定のIP PBX(DTMFでパーク保留をする物)で使用できます。

1.7 ParkRetrieve

用途

パーク保留解除コマンドを送信します。

構文

LONG ParkRetrieve(CHAR* cNumber)

引数

cNumber ... MPLServer使用:保留解除する番号 MPLServer未使用:ボタン番号

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

特定のIP PBX(DTMFでパーク保留をする物)で使用できます。

1.8 EndCall

用途

指定したLine番号の通話を終了します。

構文

Long EndCall (LONG iLine)

引数

iLine ... Line番号(0~7)

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

1.9 StartRecord

用途

通話中のLineの録音開始をします。

構文

LONG StartRecord()

引数

なし

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

Lineを指定することはできません。

1.10 StopRecord

用途

録音を停止します。

構文

LONG StopRecord ()

引数

なし

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

1.11 setAutoRec

用途

自動録音、手動録音の切り替えをします。

構文

LONG setAutoRec(Long iAutoRec)

引数

iAutoRec ... 0:手動録音 1:自動録音

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

起動時の設定は、GVVCTIhostの設定ファイルで設定してあります。

1.12 getRecFiles

用途

全ての録音ファイルを電話機からPCへ移動します。

構文

LONG getRecFiles()

引数

なし

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

1.13 getRecFile

用途

特定の録音ファイルを電話機からPCへ移動します。

構文

LONG getRecFile(CHAR* cRecFileName)

引数

cRecFileName ... 取得する録音ファイル名

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

1.14 moveRecFiles

用途

一時フォルダから、保存先に全ての録音ファイルを移動します。

構文

LONG moveRecfiles(CHAR* cPath)

引数

cPath ...移動先のフルパス。

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

移動先のフォルダには、Everyone、フルコントロールの権限が必要です。

1.15 moveRecFile

用途

一時フォルダから、保存先に特定の録音ファイルを移動します。

構文

LONG moveRecFile(CHAR* cRecFileName, CHAR* cMoveToFolder)

引数

cRecFileName ...移動するファイル名。

cMoveToFolder ...移動先のフルパス。

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

移動先のフォルダには、Everyone、フルコントロールの権限が必要です。

1.16 setTalkDeviceState

用途

通話中に、使用するデバイスを変更するのに使用します。

構文

LONG setTalkDeviceState (LONG iHookType)

引数

iHookType ... 0:SpeakerPhone 1:WiredHeadset 2:HandSet

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

通話中に、変更する場合に使用してください。

1.17 setCallBlocking

用途

アカウントの状態を変更します。

構文

LONG setCallBlocking (LONG iActive)

引数

iActive ... 0:Active 1:Block

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

1.18 getCallLog

用途

電話機に保存されている通話履歴の取得を開始します。

構文

LONG getCallLog()

引数

なし

戻り値

-1...GSCTIhostとの通信エラー 0...GSCTIhostが受け取った

備考

取得が完了すると、イベントDetectedGetCallLogが上がる。

1.19 getCallLogNumber

用途

通話履歴の連番を元に電話番号を取得します。

構文

BSTR getCallLogNumber(LONG iIndex)

引数

iIndex...0からDetectedGetCallLogイベントの引数iCallLogCount-1までの実数。

戻り値

通話履歴の連番に基づいた電話番号。

備考

DetectedGetCallLogイベント時に実行してください。

1.20 getCallLogDate

用途

構文

DATE getCallLogDate(LONG iIndex)

引数

iIndex...0からDetectedGetCallLogイベントの引数iCallLogCount-1までの実数。

戻り値

通話履歴の連番に基づいた日付 (DATE型)。

備考

DetectedGetCallLogイベント時に実行してください。

1.21 getCallLogType

用途

構文

LONG getCallLogType(LONG iIndex)

引数

iIndex...0からDetectedGetCallLogイベントの引数iCallLogCount-1までの実数。

戻り値

- 1...着信
- 2...発信
- 3...不在着信

備考

DetectedGetCallLogイベント時に実行してください。

1.22 getFunctionName

用途

エラーが起きた関数名を返します。

構文

BSTR getFunctionName(LONG iFunctionCode)

引数

iFunctionCode... DetectedFunctionErrorの引数 iFunctionCode

戻り値

エラーが起きた関数名 (BSTR型)。

備考

DetectedFunctionErrorイベント時に実行してください。

1.23 getErrorStr

用途

GSCTIhostのどの内部関数でどのようなエラーが起きたかの説明を返します。

構文

BSTR getErrorStr(LONG iInnerFunctionCode, LONG iErrorCode)

引数

iInnerFunctionCode ... DetectedFunctionErrorの引数 iInnerFunctionCode

iErrorCode ... DetectedFunctionErrorの引数 iErrorCode

戻り値

内部関数で起きたエラーの説明 (BSTR型)。

備考

DetectedFunctionErrorイベント時に実行してください。

2.1 DetectedCallStatusChange

用途

呼状態の変化を通知します。

構文

DetectedCallStateChange(LONG iLine, LONG iStatusCode)

引数

iLine ... イベントが上がったLine番号。

iStatusCode ... 呼状態を示す値です。(下記参照)

0:Idle	1:Dialing
2:Ringing	3:Calling
4:Connect	5:Hold
6:Transferred	7:End
8:Failed	9:Transfer
10:Conference	11:Paging
12:Ringback	13:IP Call

備考

Init()していないとイベントは上がりません。

2.2 DetectedDTMF

用途

DTMF取得イベント。

構文

DetectedDTMF(LONG iLine, CHAR* cDTMF)

引数

iLine ... イベントが上がったLine番号。

cDTMF ... 取得したDTMF。

備考

Init()していないとイベントは上がりません。

2.3 DetectedParkCmdStateChanged

用途

パーク保留状態変更イベント。

構文

DetectedParkCmdStateChanged(LONG iParkCmdNumber, CHAR* cParkedNumber)

引数

iParkCmdNumber ...0~N-1。(Nはパーク保留コマンドの数)

cParkedNumber ...パーク保留されている番号。(番号が入っていない場合は、解除)

備考

Init()していないとイベントは上がりません。

このイベントは、特定のIP PBX(DTMFでパーク保留をする物)で使用できます。

2.4 DetectedRecStatusChange

用途

録音の状態変化を通知します。

構文

DetectedRecStatusChange(LONG isRec)

引数

isRec ...0:停止 1:開始

備考

Init()していないとイベントは上がりません。

2.5 DetectedGetRecordFiles

用途

録音ファイル取得イベント。

構文

DetectedGetRecordFiles(LONG iFileCount)

引数

iFileCount ...取得した録音ファイルの数。

備考

Init()していないとイベントは上がりません。

2.6 DetectedMute

用途

電話機のミュートの状態が変化した事を通知します。

構文

DetectedMute(LONG iLine, LONG iMute)

引数

iLine ... イベントが上がったLine番号。

iMute ... 0:停止 1:ミュート

備考

Init()していないとイベントは上がりません。

2.7 DetectedRemoteOnMute

用途

会議通話時、他の電話のミュートの状態が変化した事を通知します。

構文

DetectedRemoteOnMute(LONG iLine, LONG iMute)

引数

iLine ... イベントが上がったLine番号。

iMute ... 0:停止 1:ミュート

備考

Init()していないとイベントは上がりません。

2.8 DetectedConference

用途

会議通話の状態が変化した事を通知します。

構文

DetectedConference(LONG iLine, LONG iStart)

引数

iLine ... イベントが上がったLine番号。

iStart ... 0:停止 1:開始

備考

Init()していないとイベントは上がりません。

2.9 DetectedVideoCall

用途

ビデオ通話発信した事を通知します。

構文

DetectedVideoCall(LONG iLine, LONG isVideo)

引数

iLine ... イベントが上がったLine番号。

isVideo ... 0:通常 1:ビデオ通話

備考

Init()していないとイベントは上がりません。

2.10 DetectedVideoComing

用途

ビデオ通話の要求が来た事を通知します。

構文

DetectedVideoComing(LONG iLine, LONG isVideo)

引数

iLine ... イベントが上がったLine番号。

isVideo ... 0:通常 1:ビデオ通話

備考

Init()していないとイベントは上がりません。

2.11 DetectedTalkDeviceStatusChange

用途

オーディオデバイスの状態が変化したことを通知します。

構文

DetectedTalkDeviceStatusChange(LONG iHandset, LONG iSpeakerPhone, LONG iWiredHeadset)

引数

iHandset ...0:false 1:true

iSpeakerPhone ...0:false 1:true

iWiredHeadset ...0:false 1:true

備考

Init()していないとイベントは上がりません。

2.12 DetectedGetCallLog

用途

通話履歴の取得完了イベント。

構文

DetectedGetCallLog(LONG iCallLogCount)

引数

iCallLogCount ...取得した通話履歴の数。

備考

Init()していないとイベントは上がりません。

iCallLogCountを元にgetCallLogNumber()、getCallLogDate()、getCallLogType()を実行してください。

2.13 DetectedFunctionError

用途

メソッドの実行エラー取得イベント。

構文

DetectedFunctionError(LONG iFunctionCode, LONG iInnerFunctionCode, LONG iErrorCode)

引数

iFunctionCode ...エラーが起きたメソッドに対応した実数。

iInnerFunctionCode ...エラーが起きたGSCTIhost.exe内部の関数に対応した実数。

iErrorCode ... GSCTIhost.exe内部で起きたエラーに対応した実数。

備考

Init()していないとイベントは上がりません。

getFunctionName()にiFunctionCodeを渡せば、エラーが起こったメソッド名が取得できます。

また、getErrorStr()にiInnerFunctionCode、iErrorCodeを渡せば、エラーの理由が取得できます。

注意事項

1. GSCTIhost.exeが起動していない状態でのGSCTI.ocxの動作は保証しかねますのでご了承ください。
2. GSCTI.ocxを同時に2つ以上使用した場合の動作は保証しかねますのでご了承ください。