



iOS ComWireLite Library API Specification

V1.0.3 – Mar 27, 2013

Important Notice

GENERALPLUS TECHNOLOGY INC. reserves the right to change this documentation without prior notice. Information provided by GENERALPLUS TECHNOLOGY INC. is believed to be accurate and reliable. However, GENERALPLUS TECHNOLOGY INC. makes no warranty for any errors which may appear in this document. Contact GENERALPLUS TECHNOLOGY INC. to obtain the latest version of device specifications before placing your order. No responsibility is assumed by GENERALPLUS TECHNOLOGY INC. for any infringement of patent or other rights of third parties which may result from its use. In addition, GENERALPLUS products are not authorized for use as critical components in life support devices/ systems or aviation devices/systems, where a malfunction or failure of the product may reasonably be expected to result in significant injury to the user, without the express written approval of Generalplus.

Table of Content

1	Introduction	5
2	Library Module and Header Files.....	6
3	Notice for Developing app.....	8
3.1	ipod touch project	8
3.2	Developing app with game engine.....	8
3.3	Output audio on iPhone	9
4	Constrains and Limitations	10
5	API Calling Sequence.....	11
5.1	Decode Calling Sequence	11
6	API Specification	12
6.1	Exported APIs.....	12
6.1.1	InitComWireLite.....	12
6.1.2	UnitComWireLite	12
6.1.3	SetComWireLiteTxChannelMode	12
6.1.4	GetComWireLiteTxChannelMode.....	13
6.1.5	StartComWireLite.....	13
6.1.6	StopComWireLite	14
6.1.7	SendCommand	14
6.1.8	SetComWireLiteUserCallBack	15
6.1.9	GetLastCommand	15
6.1.10	EnableComWireLiteTx.....	16
6.1.11	EnableComWireLiteRx.....	16
6.1.12	SetSaveRawData	16
6.1.13	SetBitMode.....	17
6.2	Error Code	18
7	Reference.....	19

Revision History

Revision	Date	Revised By	Remark
1.0.3	03/27/2013	Shawn Liu	Support select bit mode. Remove SetUitone function.
1.0.2	11/20/2012	Shawn Liu	Fixed up API descriptions.
1.0.1	09/10/2012	Shawn Liu	Support Rx mode.
1.0.0	07/11/2012	Shawn Liu	First version.

1 Introduction

The ComWireLite library is to make it easily to interact with the devices which embed ComWireLite chips on iOS platform.

This library provides following functions:

- (1) Initialize audio record unit and ComWireLite encode function on iOS platform.
- (2) Encode ComWireLite sound by line-in on iOS platform.
- (3) Decode ComWireLite sound by line-in on iOS platform.

This API specification describes all the APIs provided by the ComWireLite library.

2 Library Module and Header Files

This ComWireLite library is in **static library on iOS** platform. This library is written by **C++**, before linking our library you must add the linker flag **-lstdc++** (Figure 2-1) or specify **C++/objective-C++** compiler (Figure 2-2) to compile target project. Static Library Module Name: **libComWireLiteLibrary.a**

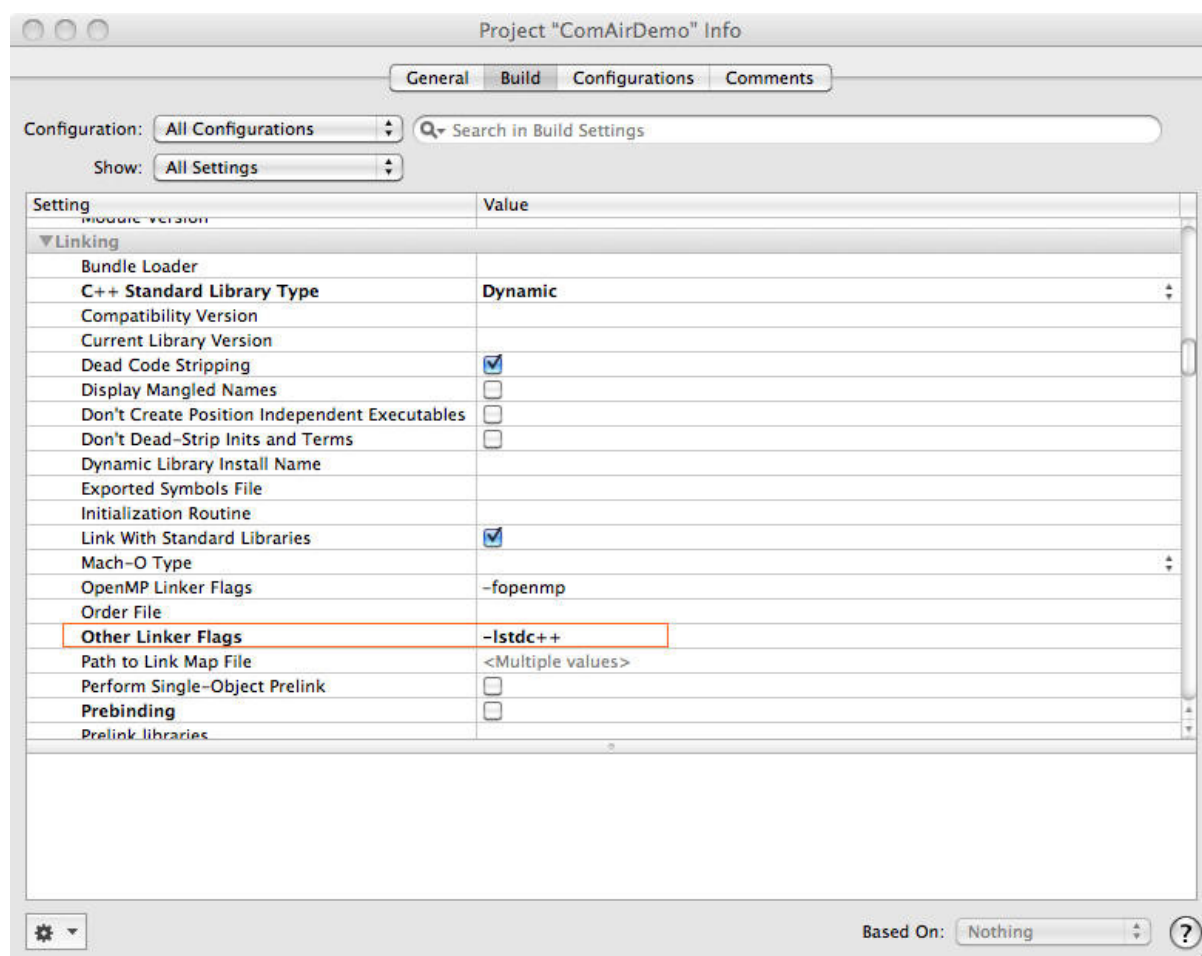


Figure 2-1 Linker Flags

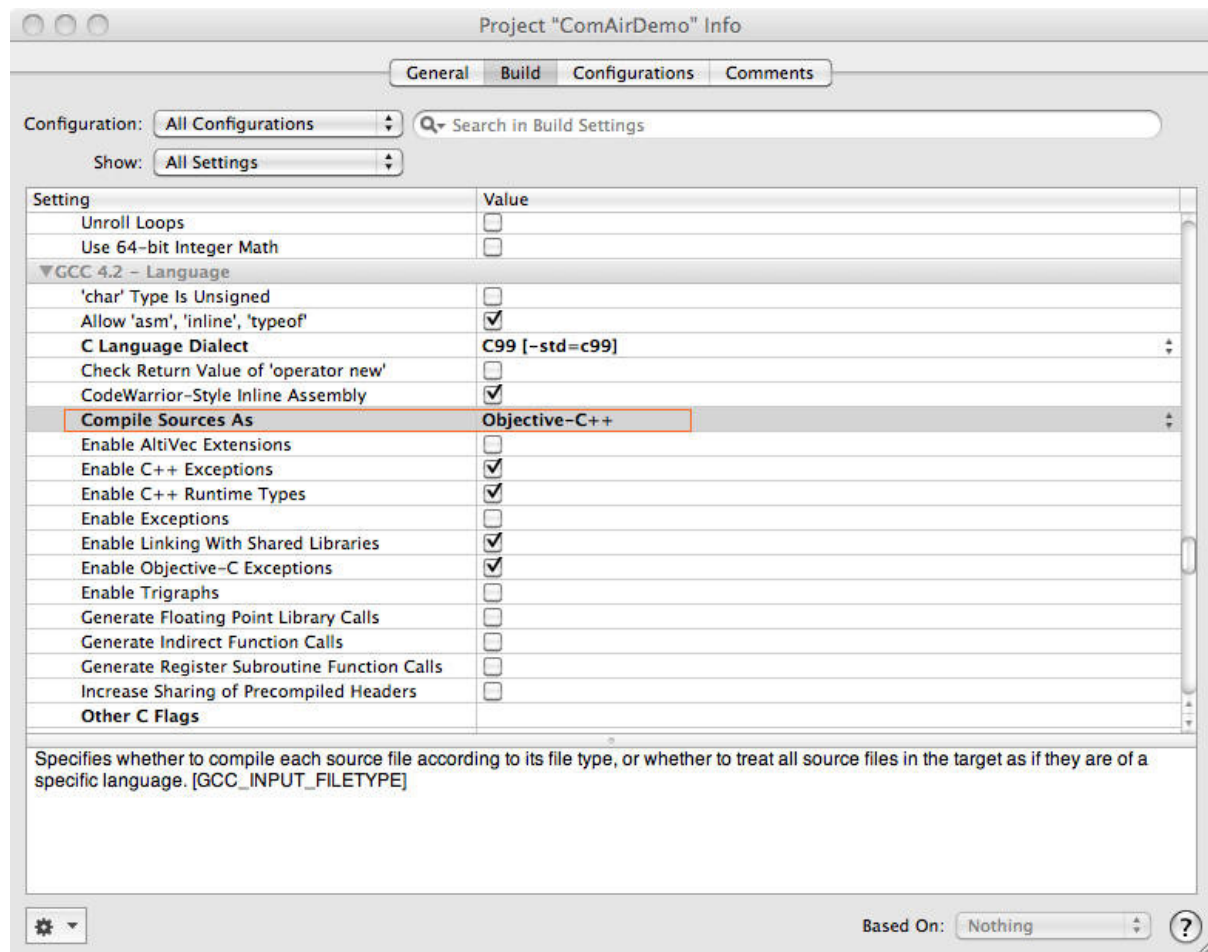


Figure 2-2 Specify C++ Compile

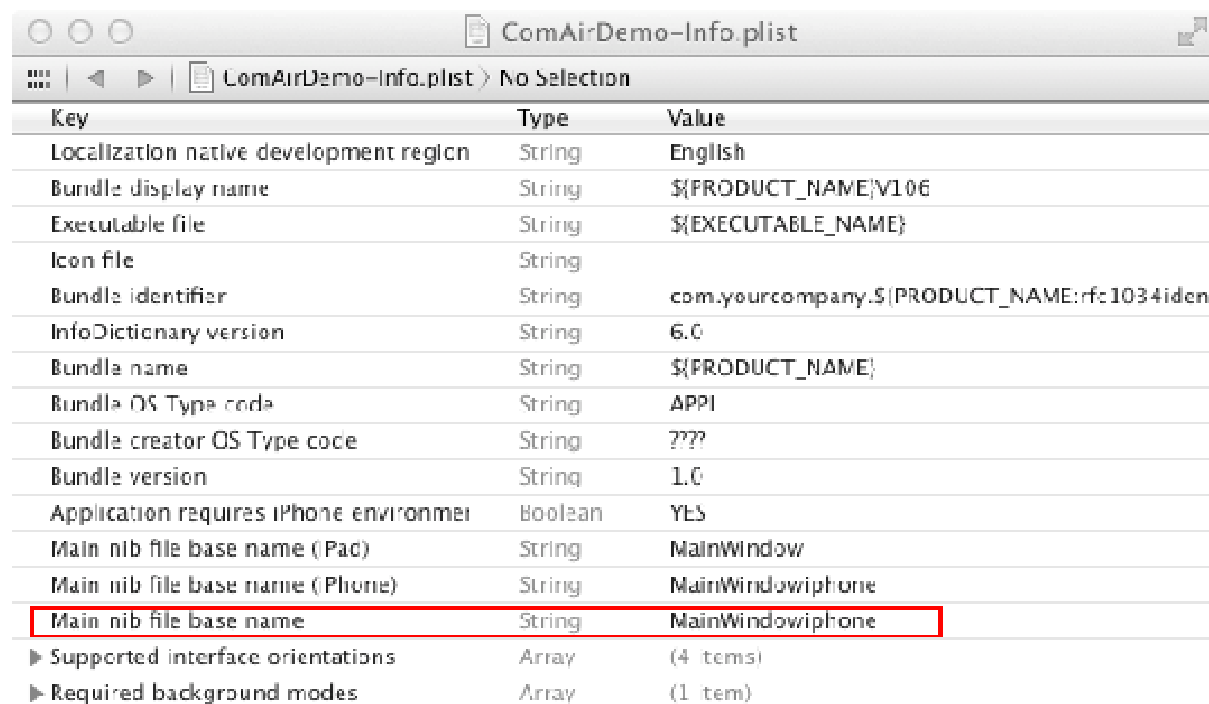
Header Files:

1. **ComWireLiteAPI.h:** Static library class definition which is used to manipulate ComWireLite library.

3 Notice for Developing app

3.1 ipod touch project

By default it will always loads the ipad main nib, this will causes the screen goes dark and no response when run app on ipod touch. Please assign the attribute **Main nib file base name** to the iphone main nib in your .plist file.



Key	Type	Value
Localization native development region	String	English
Bundle display name	String	\$(PRODUCT_NAME)V106
Executable file	String	\$(EXECUTABLE_NAME)
Icon file	String	
Bundle identifier	String	com.yourcompany.\$(PRODUCT_NAME):rfc1034iden
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	APPI
Bundle creator OS Type code	String	???
Bundle version	String	1.0
Application requires iPhone environment	Boolean	YES
Main nib file base name (Pad)	String	MainWindow
Main nib file base name (Phone)	String	MainWindowiphone
Main nib file base name	String	MainWindowiphone
Supported interface orientations	Array	(4 items)
Required background modes	Array	(1 item)

Figure 3-1 .plist file for ipod touch

3.2 Developing app with game engine

Some game engine will configure the audio session settings during their own initial process; these settings may causes the ComWireLite decoding process out of functional. Please make sure the audio session settings are set to appropriate value that are described below:

Settings	Appropriate Value
Audio Session Categories	AVAudioSessionCategoryPlayAndRecord

How to configure the Audio session please refers to the web site [Audio Session Programming Guide](#).

3.3 Output audio on iPhone

Audio will output from the receiver by default. If wants output audio from the speaker, it must route the audio from receiver to speaker. Following are the example that how to route audio:

```
UInt32 doChangeDefaultRoute = 1;
AudioSessionSetProperty ( kAudioSessionProperty_OverrideCategoryDefaultToSpeaker,
                          sizeof (doChangeDefaultRoute),
                          &doChangeDefaultRoute
                          );
[[AVAudioSession sharedInstance] setActive:YES error:nil];
```

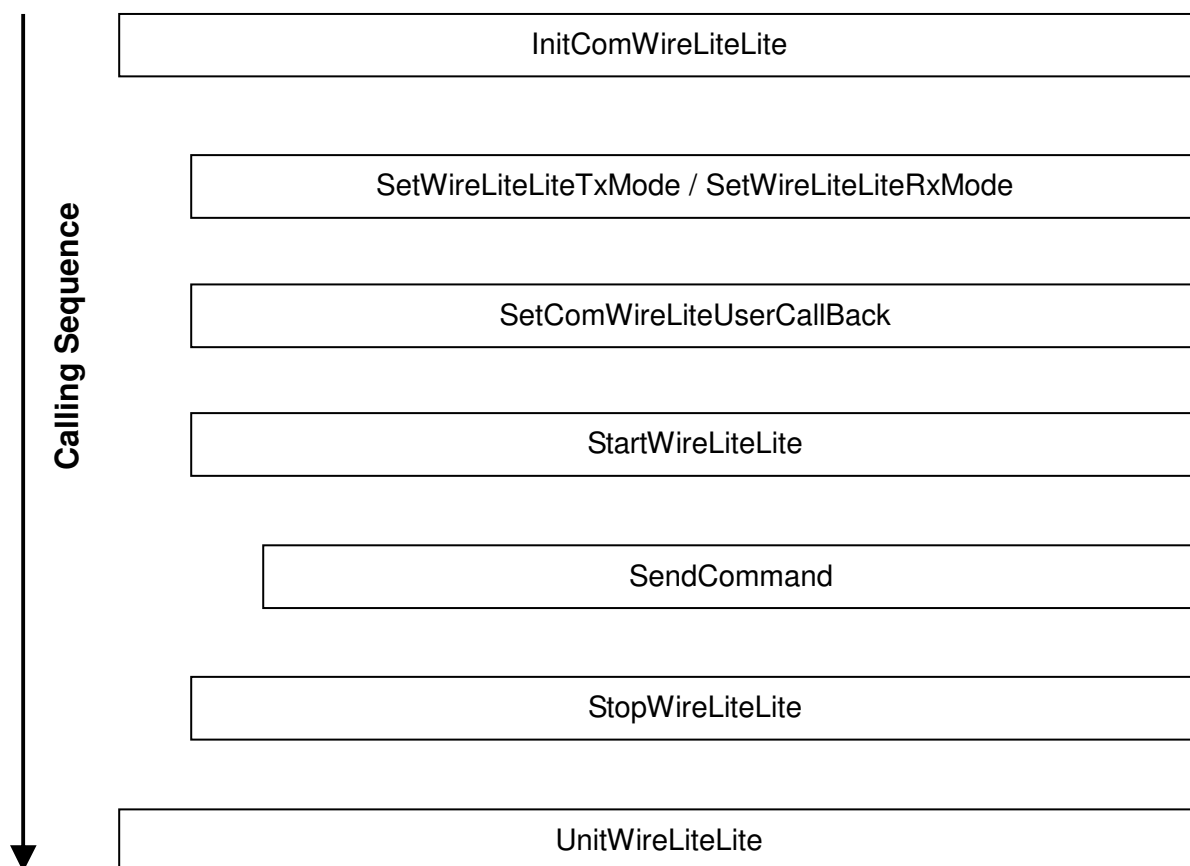
4 Constrains and Limitations

Constrains:

1. This library can only be used to interact with the the devices which embeds Generalplus ComWireLite chips.
2. This library supports only iOS platforms.

5 API Calling Sequence

5.1 Decode Calling Sequence



1. The API surrounded with dotted line can be optionally called.
2. The indented APIs should be called after its upper protrusive APIs were called.

6 API Specification

6.1 Exported APIs

6.1.1 InitComWireLite

Prototype:

```
int          InitComWireLite (  
    );
```

Description:

Initialize ComWireLite Audio unit for encoding command.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:

Remark:

This API should be called before any other ones.

6.1.2 UnitComWireLite

Prototype:

```
int          UnitComWireLite (  
    );
```

Description:

Uninitialize ComWireLite Audio unit.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:

Remark:

This API should be called before exit program.

6.1.3 SetComWireLiteTxChannelMode

Prototype:

```
int          SetComWireLiteTxChannelMode (  
    E_ComWireLiteTxChannelMode eMode  
);
```

Description:

Set ComWireLite Tx Channel mode.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:

[in] eMode:

Set ComWireLite Tx Channel mode.

Remark:

eComWireLiteTxChannelMode_Default is for output ComWireLite in both channel; eComWireLiteTxChannelMode_Left is for left channel only; eComWireLiteTxChannelMode_Right is for right channel only.

6.1.4 GetComWireLiteTxChannelMode

Prototype:

```
E_ComWireLiteTxChannelMode GetComWireLiteTxChannelMode (  
);
```

Description:

Get ComWireLite Tx Channel mode.

Return Value:

The ComWireLite Tx Channel mode.

Parameters:

Remark:

6.1.5 StartComWireLite

Prototype:

```
int          StartComWireLite (  
);
```

Description:

Start ComWireLite service.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:**Remark:**

Before calling this API, it must call *SetComWireLiteUserCallBack()* to set user call back first.

6.1.6 StopComWireLite

Prototype:

```
int          StopComWireLite (  
                                     ) ;
```

Description:

Stop ComWireLite service.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:**Remark:**

6.1.7 SendCommand

Prototype:

```
int          SendCommand (  
                                     BYTE      *pi8CommandArray,  
                                     int        i32ArryCmdSize  
                                     );
```

Description:

Send the ComWireLite command sound.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:

[in] pi8CommandArray:

The byte array that want to send.

[in] i32ArryCmdSize:

The size of byte array.

Remark:

This API must be called after StartComWireLite ().

6.1.8 SetComWireLiteUserCallBack

Prototype:

```
int          SetComWireLiteUserCallBack (  
            PFN_ComWireLiteUserCallBack  pfnUserCallBack,  
            );
```

Description:

Set the user call back function. User call back will be called when catch the command during decoding ComWirelite sound.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:

[in] pfnUserCallBack:

This is the pointer of user call back function.

Remark:

This API must be called after StartComWireLite ().

6.1.9 GetLastCommand

Prototype:

```
int          GetLastCommand (  
            int          *i32Command  
            );
```

Description:

Get last ComWireLite command.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:

[out] i32Command:

The last ComWireLite command.

Remark:

This API must be called after StartComWire ().

6.1.10 EnableComWireLiteTx

Prototype:

```
int          EnableComWireLiteTx (  
                                Bool          bEnable  
                                );
```

Description:

Set enable ComWireLite Tx mode.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:

[in] bEnable:
Enable flag.

Remark:

6.1.11 EnableComWireLiteRx

Prototype:

```
int          EnableComWireLiteRx (  
                                Bool          bEnable  
                                );
```

Description:

Set enable ComWireLite Rx mode.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:

[in] bEnable:
Enable flag.

Remark:

6.1.12 SetSaveRawData

Prototype:

```
int          SetSaveRawDate (  

```



```
        bool  bEnable  
    );
```

Description:

Set save raw PCM data to file. The PCM file will store in applications document folder and can be downloaded by iTunes.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:**[in] bEnable:**

Enable flag.

Remark:

6.1.13 SetBitMode

Prototype:

```
    int          SetBitMode (  
        E_ComWireLiteBitMode  eMode  
    );
```

Description:

Set the command bit width.

Return Value:

Return 0 (*COMWIRELITE_NOERR*) if this function succeeded. Otherwise, other value returned. Please refer to section 6.2 for error code definition.

Parameters:**[in] eMode:**

The bit width.

Remark:

Increase the bit width will also increase the command latency. The transmission time of ComWireLiteBitMode_8Bit mode is about 20 ms/command, and ComWireLiteBitMode_16Bit is about 32 ms/command.

6.2 Error Code

Symbol Name	Value	Description
<i>COMWIRELITE_NOERR</i>	0x0000	No error. API works normally.
<i>COMWIRELITE_AUDIOUNITFAILED</i>	0x0001	Get audio unit failed.
<i>COMWIRELITE_ENABLEIORECF FAILED</i>	0x0002	Enable Record failed.
<i>COMWIRELITE_SETFORMATFAILED</i>	0x0003	Set audio failed.
<i>COMWIRELITE_SETRECCALLBACKFAILED</i>	0x0004	Set call back function for recording failed.
<i>COMWIRELITE_ALLOCBUFF FAILED</i>	0x0005	Try to allocate buffer for recording failed.
<i>COMWIRELITE_AUDIONOTINIT</i>	0x0006	ComWireLite audio had not been initialized.
<i>COMWIRELITE_SETDECODEMODE FAILED</i>	0x0007	Set ComWireLite mode failed.
<i>COMWIRELITE_SENDCMDQUEUEBUSY</i>	0x0008	Command queue is full.
<i>COMWIRELITE_SENDCMDSIZELIMIT</i>	0x0009	Reach max command queue size.

7 Reference

The Objective-C Programming Language

<http://developer.apple.com/library/mac/#documentation/Cocoa/Conceptual/ObjectiveC/Introduction/introObjectiveC.html>

iOS Developer library

<http://developer.apple.com/library/ios/navigation/>

iOS Application Programming Guide

<http://developer.apple.com/library/ios/#documentation/iPhone/Conceptual/iPhoneOSProgrammingGuide/Introduction/Introduction.html>

Multimedia Programming Guide

<http://developer.apple.com/library/ios/#documentation/AudioVideo/Conceptual/MultimediaPG/Introduction/Introduction.html>

Audio Session Programming Guide

<http://developer.apple.com/library/ios/#documentation/Audio/Conceptual/AudioSessionProgrammingGuide/Introduction/Introduction.html>