

# SDK Description

Ver.0.0.2e



**Probe Inc.**

16 MARCH 2011

# Orders

<b>Document History .....</b>	<b>4</b>
<b>1. Overview .....</b>	<b>5</b>
<b>2. SDK Connection Process .....</b>	<b>5</b>
<b>3. Interface .....</b>	<b>7</b>
<b>4. Check Point to FAQ .....</b>	<b>12</b>

## Document History

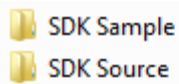
Date	Note	Author	Revision
2011-01-28	-Initial Version	Lee Sang Min	0.0.1
2011-03-16	-Comment for SDK file	Alex Oh	0.0.2

## 1. Overview

This document is for the brief explanation of SDK skim to be used with source program packaged together. this document can be referred with BASIC NETWORK PROTOCL for PNP which is detailed with PROBE Network Protocol.

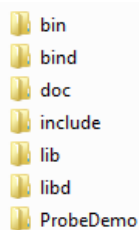
## 2. SDK Use

Source folders in SDK for respective use,

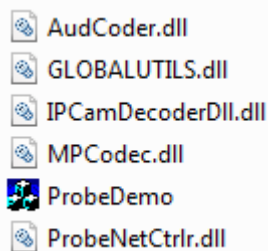


- 1) *SDK Source Folder: Full source file supporting the direct compile and debugging*

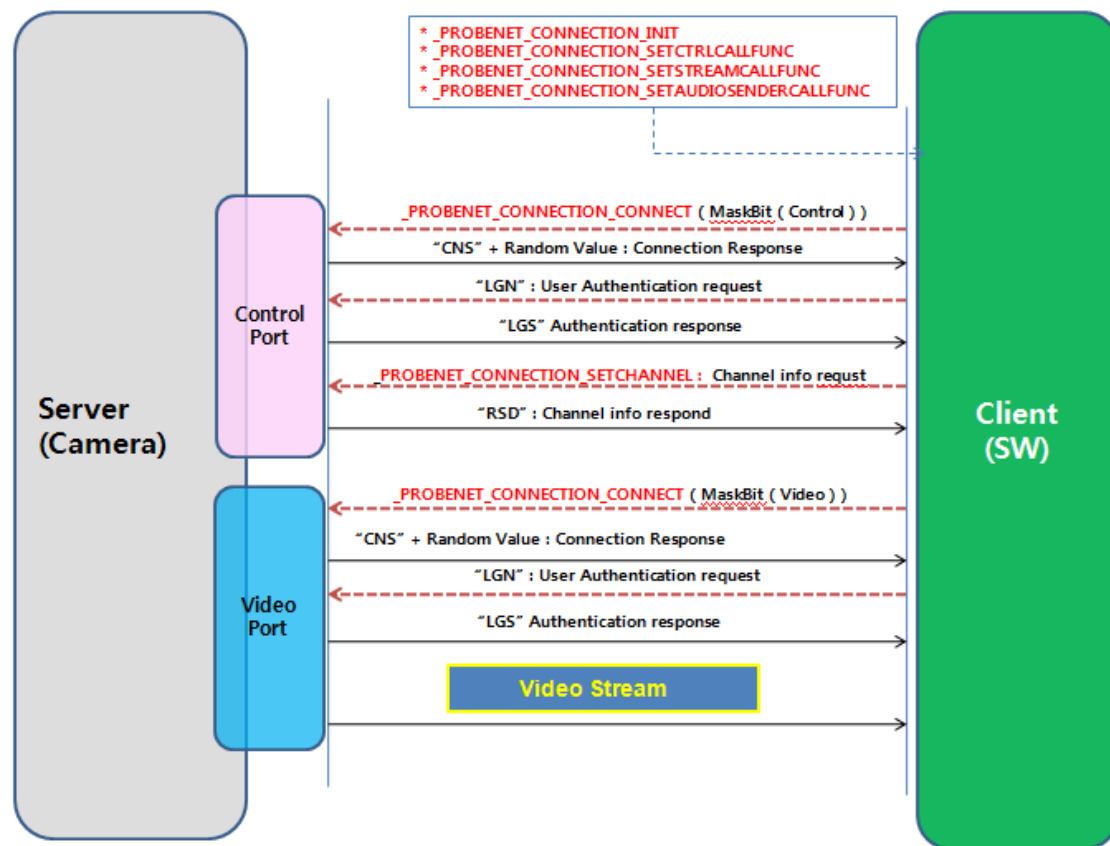
*\*Compile should be made to execute the source*



- 2) *SDK Sample Folder: Executable file (ProbeDemo) with corresponding “\*.dll” files.*



### 3. SDK Connection Process



```

1. _PROBENET_INIT();
2. HANDLE _PROBENET_CONNECTION_INIT(CONNECTION_INFO *lpConnectionInfo,
    DWORD dwMaskCreatingSocks, DWORD dwProductType=11);
3. _PROBENET_CONNECTION_SETCTRLCALLFUNC( m_hConnection,
    CLiveChannel::_ON_RECV_CTRL_DATA,
    CLiveChannel::_ON_ERROR_CTRL,
    (HANDLE)this);
    _PROBENET_CONNECTION_SETSTREAMCALLFUNC( m_hConnection,
    CLiveChannel::_ON_RECV_VIDEO_DATA,
    CLiveChannel::_ON_RECV_AUDIO_SPEAKER_DATA,
    CLiveChannel::_ON_RECV_VIDEO_CTRL_DATA,
    CLiveChannel::_ON_RECV_AUDIO_SPEAKER_CTRL_DATA,
    CLiveChannel::_ON_ERROR_VIDEO,
    CLiveChannel::_ON_ERROR_AUDIO_SPEAKER,

```

```

        (HANDLE)this);
        _PROBENET_CONNECTION_SETAUDIOSENDERCALLFUNC( m_hConnection,
        CLiveChannel::_ON_RECV_AUDIO_MIC_CTRL_DATA,
        CLiveChannel::_ON_ERROR_AUDIO_MIC,
        (HANDLE)this);

```

4.1 Actual connection is established once "CNS" will be received on Control Receive after requesting with `_PROBENET_CONNECTION_CONNECT( m_hConnection, _CONTROL_SOCKET);* _CONTROL_SOCKET`.

After this establishment, next step can be tried to t with `_VIDEO_SOCKET | _AUDUIO_SOCKET`.

```
* void _PROBENET_CONNECTION_SETCHANNEL(HANDLE handle, int nChannel);
```

```
4-2._PROBENET_CONNECTION_CONNECT( m_hConnection, _VIDEO_SOCKET );
```

```
4-3._PROBENET_CONNECTION_CONNECT( m_hConnection, _AUDIO_SOCKET );
```

```
5. void _PROBENET_CONNECTION_DISCONNECTALL( HANDLE handle );
```

```
6. void _PROBENET_CONNECTION_END(HANDLE handle);
```

```
7. void _PROBENET_END();
```

### 3. Interface

#### ☐ Registration for SDK use

Return Value	Function
Void	<code>_PROBENET_INIT</code>

Return Value	Function
Void	<a href="#">_PROBENET_END</a>

#### ❑ SDK Initialization

Return Value	Function
HANDLE	<a href="#">_PROBENET_CONNECTION_INIT</a>

#### Parameter

Index	Type	Name	Description
1	CONNECTION_INFO *	IpConnectionInfo	Connection info
2	DWORD	dwMaskCreatingSocks	Control, Video, Audio( Speaker )
3	DWORD	dwProductType=11	( DM365 Product = 11 )  *always setup with 11

Return Value	Function
Void	<a href="#">_PROBENET_CONNECTION_END</a>

#### Parameter

Index	Type	Name	Description
1	HANDLE	handle	Connected handle

#### ❑ Call-Back Function Registration ( Control )

Return Value	Function
void	<a href="#">_PROBENET_CONNECTION_SETCTRLCALLFUNC</a>

#### Parameter

Index	Type	Name	description
1	HANDLE	handle	SDK Initializing Handle
2	LPSOCKETFUNC_ON_RECV_CTRL_DATA	OnRecvCtrlMsg	Callback for CMD receive
3	LPSOCKETFUNC_ON_ERROR	OnError	Error Callback for Error receive
4	HANDLE	hCallback	Present Handle( this )

#### ❑ Call-Back Function Registration (Video, Audio ( Speaker ) )

Return Value	Function
void	<a href="#">_PROBENET_CONNECTION_SETSTREAMCALLFUNC</a>

#### Parameter

Index	Type	Name	description
1	HANDEL	handle	SDK Initializing Handle
2	LPSOCKETFUNC_ON_RECV_VIDEO_DATA	OnRecvVideo	Callback for Data command receive
3	LPSOCKETFUNC_ON_RECV_AUDIO_DATA	OnRecvAudio	Callback for Data command receive
4	LPSOCKETFUNC_ON_RECV_CTRL_DATA	OnRecvVideoMsg	Callback for Data command receive
5	LPSOCKETFUNC_ON_RECV_CTRL_DATA	OnRecvAudioMsg	Callback for Data command receive
6	LPSOCKETFUNC_ON_ERROR	OnErrorVideo	Callback for Video Error receive
7	LPSOCKETFUNC_ON_ERROR	OnErrorAudio	Callback for Speaker Error receive
8	HANDLE	hCallback	Present Handle( this )

#### ❑ Call-Back Function Registration ( Audio ( Microphone ) )

Return Value	Function
void	<a href="#">_PROBENET_CONNECTION_SETAUDIOSENDERCALLFUNC</a>



**Parameter**

Index	Type	Name	description
1	HANDLE	handle	SDK Initializing Handle
2	LPSOCKETFUNC_ON_RECV_CTRL_DATA	OnRecvCtrlMsg	Callback for CMD receive
3	LPSOCKETFUNC_ON_ERROR	OnError	Callback for Error Receive
4	HANDLE	hCallback	Present Handle( this)

☐ **Connect, Disconnect ( Control, Video, Audio( Speaker ), Audio( Microphone ) )**

Return Value	Function
void	<a href="#">_PROBENET_CONNECTION_CONNECT</a> <a href="#">_PROBENET_CONNECTION_DISCONNECT</a>

## Parameter

Index	Type	Name	Description
1	HANDLE	handle	SDK Initializing Handle
2	DWORD	dwMaskCreatingSocks	Control, Video, Audio( Speaker, Microphone )

Return Value	Function
void	<a href="#">_PROBENET_CONNECTION_DISCONNECTALL</a>

## Parameter

Index	Type	Name	Description
1	HANDEL	handle	SDK Initializing Handle

☐ **Control Command**

Return Value	Function
void	<a href="#">_PROBENET_CONNECTION_SETCCHANNEL</a>

**Parameter**

Index	Type	Name	description
1	HANDEL	handle	SDK Initializing Handle
2	Int	nChannel	Channel to be changed

Return Value	Function
int	<a href="#">_PROBENET_CONNECTION_SENDCOMMAND</a>

**Parameter**

Index	Type	Name	description
1	HANDLE	handle	SDK Initializing Handle
2	char *	szCmd	Command
3	BYTE *	lpParam	Command Parameter Info
4	DWORD	dwParamSize	Command Parameter Size

Return Value	Function
int	<a href="#">_PROBENET_CONNECTION_SENDAUDIOPACK</a>

**Parameter**

Index	Type	Name	Description
1	HANDEL	handle	SDK Initializing Handle
2	AUDIO_DATA_PACK *	lpData	Speaker information

☐ Encoder Setup

Return Value	Function
BOOL	<a href="#">_PROBENET_CONNECTION_GETENCODERSETUPINFO</a>

**Parameter**

Index	Type	Name	description
-------	------	------	-------------

1	HANDLE	handle	SDK Initializing Handle
2	Int	nChannel	Channel
3	ENCODE_SETUP_INFO *	lpData	Encoder Setup info
4	BOOL	bRefreshData=TRUE	Encoder Setup Update

Return Value	Function
int	<a href="#">_PROBENET_CONNECTION_SETENCODESETUPINFO</a>

**Parameter**

Index	Type	Name	Description
1	HANDEL	handle	SDK Initializing Handle
2	ENCODE_SETUP_INFO *	lpData	Encoder Setup info

**4. Check Point to FAQ**

- ❑ it takes "1" second to create the random code(value) in CNS command so it's recommended not to send the connection request to the camera within 1 second
- ❑ it's one of specific skim from IOCP to create threads internally with same number like the number of CPU core for the improvement of streaming performance
- ❑ AlarmIn : Only "Alarm ON" status, Camera will send the alarm signal(alarm ON event) to SDK so, SDK will send the same "ON" event notification to Client(SW). in the case of "Alarm Off", SDK detects its change after 5 seconds if there is no further "Alarm ON" notification from camera and sends the "off event" to Client(SW)
- ❑ Connection status check
  - : "STA" event can be used for this purpose
  - : Server --("STA")--> Client --("STA")--> Client ( every 10 seconds )
- ❑ In order to get the name of camera name and MAC address, SW can get the information from camera through the channel information from "RSD" command.

[\(\\_PROBENET\\_CONNECTION\\_SETCCHANNEL\(\)\)](#)

- ❑ Audio setting value [ Audio( Speaker & Mic ) ]

\* Set

```

WAVEFORMATEX wfx;

wfx.wFormatTag      = WAVE_FORMAT_PCM;           // 1
wfx.nChannels       = AT_MONO;                   // 1
wfx.nSamplesPerSec  = AUDIO_SAMPLE_PER_SEC;     // 16,000
wfx.wBitsPerSample  = AUDIO_BITS_FOR_SAMPLE;    // 16
wfx.nBlockAlign     = (wfx.wBitsPerSample / 8) * wfx.nChannels;
wfx.nAvgBytesPerSec = wfx.nSamplesPerSec * wfx.nChannels * wfx.nBlockAlign;
wfx.cbSize          = 0;

```

#### ❑ GOS size change with Frame-Rate setting

\* The change of GOP value will be not applied if frameRate value will be same without change

\* If value of FrameRate is changed, then following table will be applied.

FrameRate	Changed GOP	GOP (H.264)	GOP (MJPEG)	GOP (MPEG4)
<10	Any value	Same value with FrameRate (follow FrameRate)		
=,>10	<10	15 (always)		
=,>10	=,>10	Follow GOP setup(changed) value accordingly.		
No change	1~30	No change	No change	No change