

스캐너 결과 통지 API 사용자 메뉴얼

Project:

안드로이드 기반 스캐너 인식 결과 통지 API

Author:

조항복

Revision History:

Version	Date	Name	Comments
r01	2013/06/18	조항복	문서 최초 작성
R02	2014/06/17	조항복	API 함수와 함수형 정리
R03	2014/09/22	조항복	프로토콜 V1
R04	2014/09/24	조항복	EVN_GetDateTime 함수 추가
R05	2015/02/10	조항복	EVN_Connect 함수에 Timeout 설정

목차

개요	3
API 함수들	4
공통 API.....	4
Notice API.....	9
Reference	14
Header 구조.....	14
TYPE 값	15
PAYLOAD TYPE	15
ERROR CODE	17

개요

네트워크를 통해 연결된 스캐너의 인식 결과와 인식 인식 이미지를 수신하는 API.

API 함수들

공통 API

EVN_Init

Prototype

```
int EVN_Init();
```

Brief

Socket을 초기화 한다.

모든 함수들은 EVN_Init() 이후에 호출되어야 한다.

Return

만약 성공적으로 초기화하였다면 EVC_ERR_SUCCESS를 리턴하고, 실패한다면 그에 맞는 에러 값을 리턴한다

Example

```
// 함수 호출
```

```
int iErrCode = EVN_Init();
```

EVC_End

Prototype

```
int EVN_End();
```

Brief

Socket 사용을 종료한다.

모든 API사용이 끝나면 호출한다.

Return

만약 성공적으로 종료하였다면 EVN_ERR_SUCCESS를 리턴하고, 실패한다면 그에 맞는 에러 값을 리턴한다

Example

```
// 함수 호출  
  
int iErrCode = EVN_End();
```

EVN_GetVersion

Prototype

```
int EVN_GetVersion(int *pMajor, int *pMinor, int *pChange);
```

Brief

API의 버전을 리턴한다.

Parameter

int *pMajor	API의 Major 버전 번호
int *pMinor	API의 Minor 버전 번호
int *pChange	API의 Change 버전 번호

Return

만약 성공적으로 버전을 얻었다면 EVN_ERR_SUCCESS를 리턴하고, 실패한다면 그에 맞는 에러 값을 리턴한다

Example

```
// 함수 호출  
  
int iMajor, iMinor, iChange;  
  
int iErrCode = EVN_GetVersion(&iMajor, &iMinor, &iChange);  
  
if(iErrCode == EVC_ERR_SUCCESS)
```

```

{

    printf("NoticeApi Version v%d.%d.%d", iMajor, iMinor, iChage);

}

```

EVN_GetErrMsg

Prototype

```
char *EVN_GetErrMsg(int error_code);
```

Brief

API에서 리턴되는 error code에 해당하는 문자열을 얻는다.

Parameter

int error_code	API에서 리턴되는 error code값
----------------	------------------------

Return

error_code의 값이 리스트에 있다면 해당 문자열을 리턴하고, 없다면 "IT'S NOT IN ERROR LIST" 문자열을 리턴한다.

Example

```

// 함수 호출

iErrCode = EVN_Disconnect(m_hNoticeHandle);

if(iErrCode != EVN_ERR_SUCCESS)

{

    strErrMsg.Format("EVN_Disconnect() fail - %d(%s)", iErrCode, EVN_GetErrMsg(iErrCode));

    AfxMessageBox(strErrMsg);

    return -1;
}

```

```
}
```

EVN_GetDateTime

Prototype

```
char *EVN_GetDateTime(PNOTICE_HEADER pHeader, char *pBuf, int iBufLength);
```

Brief

헤더의 Timestamp와 Millisecond값을 Local Time 문자열로 리턴.

Parameter

pHeader	날짜 시간을 얻어낼 Header 포인터
pBuf	변환된 문자열이 들어갈 버퍼 포인터
iBufLength	버퍼의 크기

Return

변환에 성공하면 EVN_ERR_SUCCESS를 리턴하고, 실패한다면 그에 맞는 에러 코드 값을 리턴한다.

Example

```
// 함수 호출

char szDateTime[MAX_PATH];

iErrCode = EVN_GetDateTime(pHeader, szDateTime, MAX_PATH);

if(iErrCode != EVN_ERR_SUCCESS)

{

    strErrMsg.Format("EVN_GetDateTime() fail - %d(%s)", iErrCode, EVN_GetErrMsg(iErrCode));
```

```
AfxMessageBox(strErrMsg);
```

```
return -1;
```

```
}
```

```
TRACE("DATETIME: %s\\n", szDateTime);
```


Notice API

EVN_Connect

Prototype

```
int EVN_Connect(HANDLE *pHandle, const char *pAddress, int iTimeOut);
```

Brief

특정 IP 주소를 갖는 스캐너의 Notice Server로 연결한다.

Parameter

const char *pAddress 스캐너의 IP 주소

int iTimeOut TimeOut 설정. 단위는 초.

Return

만약 성공적으로 연결되었다면 EVN_ERR_SUCCESS를 리턴하고, 실패한다면 그에 맞는 에러 값을 리턴한다

Example

```
// 함수 호출

HANDLE hNoticeHandle = NULL;

int iErrCode = EVN_Connect(&hHandle, m_strAddress);

if(iErrCode == EVN_ERR_SUCCESS && hHandle != NULL)

{

    printf("Connect to %s", m_strAddress);

}
```

EVN_Disconnect

Prototype

```
int EVN_Disconnect(HANDLE hHandle);
```

Brief

스캐너와의 연결을 끊는다.

Parameter

HANDLE hHandle EVN_Connect에서 받은 연결을 나타내는 HANDLE 값

Return

만약 성공적으로 연결 종료한다면 EVN_ERR_SUCCESS를 리턴하고, 실패한다면 그에 맞는 에러 값을 리턴한다

Example

```
// 함수 호출
```

```
int iErrCode = EVN_Disconnect(hHandle);
```

EVN_GetNotice

Prototype

```
int EVN_GetNotice(HANDLE hHandle, int iBlocked, int *pLength, PNOTICE_HEADER pHeader,  
unsigned char **ppPayload, int *pPayloadLength);
```

Brief

Header와 Payload로 구성된 하나의 패킷을 읽어온다.

Parameter

HANDLE hHandle	EVN_Connect에서 받은 연결을 나타내는 HANDLE 값.
int iBlocked	이 값이 1이면 이미지를 수신할 때까지 Block 되며, 0이면 수신 버퍼를 확인하고 바로 리턴.
int *pLength	수신된 Notice 수를 얻기 위한 int형 포인터.
PNOTICE_HEADER pHeader	수신된 Notice Header를 담을 포인터.
const char **ppPayload	수신된 Payload 데이터를 담은 메모리의 포인터를 얻어낼 포인터의 포인터 변수.
int *pPayloadLength	ppPayload 포인터로 넘겨지는 메모리의 길이를 얻어낼 포인터 변수.

Return

만약 성공적으로 연결 종료한다면 EVN_ERR_SUCCESS를 리턴하고, 실패한다면 그에 맞는 에러 값을 리턴한다.

Example

```
unsigned char *pPayload = NULL;

int iLength = 0, iPayloadLength = 0;

// 함수 호출

int iErrCode = EVN_GetNotice(m_hNoticeHandle, 1/*blocked*/, &iLength, &m_Header, &pPayload,
&iPayloadLength);

if(iErrCode == EVN_SUCCESS)

{

    UpdateResult(&m_Header, pPayload, iPayloadLength);

}
```

EVN_CancelNotice

Prototype

```
int EVN_CancelNotice(HANDLE hHandle);
```

Brief

Block된 GetNotice 루틴이 취소하고 빠져 나오도록 한다.

Parameter

HANDLE hHandle EVN_Connect에서 받은 연결을 나타내는 HANDLE 값

Return

만약 성공적으로 취소하였다면 EVN_ERR_SUCCESS를 리턴하고, 실패한다면 그에 맞는 에러 값을 리턴한다

Example

```
// 함수 호출  
  
int iErrCode = EVN_CancelNotice(hHandle);
```

EVN_IsConnect

Prototype

```
int EVN_IsConnect(HANDLE hHandle);
```

Brief

스캐너와 연결되어 있는지 확인 한다.

Parameter

HANDLE hHandle EVN_Connect에서 받은 연결을 나타내는 HANDLE 값

Return

만약 연결되어 있다면 EVN_ERR_SUCCESS를 리턴하고, 연결되어 있지 않거나 실패한다면 그에 맞는 에러 값을 리턴한다

Example

// 함수 호출

m_btnConnect.EnableWindow(EVN_IsConnect(m_hHandle) != EVN_ERR_SUCCESS);

Reference

Header 구조

고정 길이 헤더를 정의한 표. 모든 데이터는 big endian으로 저장된다.

type	size	Name	Comment
unsigned long	4	mStart	헤더의 앞 쪽 구분자(0x0000AA55)
unsigned long	4	mLength	헤더의 전체의 크기
unsigned long	4	mVersion	프로토콜 버전. 현재 1
unsigned long	4	mTimestamp	Unix time stamp
unsigned long	4	mMillisecond	millisecond
unsigned long	4	mPhaseNumber	패킷이 속한 Phase의 번호
unsigned long	4	mPhaseCount	Phase에 속한 패킷의 전체 갯수
unsigned long	4	mPhaseIndex	Phase에서 패킷의 순서
unsigned long	4	mType	패킷의 종류
unsigned long	4	mPayloadType	Payload 영역에 담기는 자료의 종류
unsigned long	4	mPayloadLength	Payload의 길이
unsigned long	4	mParameter1	Payload Type에 따라 사용할 수 있는 저장영역1
unsigned long	4	mParameter2	Payload Type에 따라 사용할 수 있는 저장영역2
unsigned long	4	mParameter3	Payload Type에 따라 사용할 수 있는 저장영역3
unsigned long	4	mParameter4	Payload Type에 따라 사용할 수 있는 저장영역4
unsigned long	4	mEnd	헤더의 뒤 쪽 구분자(0x55AA0000)

Table 1

TYPE 값

서버에 통지할 때는 설정에 따라 payload에 image binary 데이터가 올 수 있다. 서버에서 응답을 할 때는 문자열로 응답을 넣어주면 된다.

No	항목	내용
0	PACKET_TYPE_COMMAND	Protocol 흐름 제어
1	PACKET_TYPE_RESULT	결과 전송
2	PACKET_TYPE_IMAGE	이미지 전송

Table 2

PAYLOAD TYPE

COMMAND TYPE

NO	항목	내용
1	PAYLOAD_TYPE_COMMAND_ACK	Phase를 정상 수신했을 때의 응답
2	PAYLOAD_TYPE_COMMAND_NACK	Phase를 정상 수신 하지 못했을 때의 응답
3	PAYLOAD_TYPE_COMMAND_RETRY	Phase 재전송을 요청함
3	PAYLOAD_TYPE_COMMAND_LIVE	장비상태를 알려주기 위해 1분마다 스캐너에서 보내지는 packet

Table 3

RESULT TYPE

NO	항목	내용
1	PAYLOAD_TYPE_RESULT_IMG	이미지만 전송
2	PAYLOAD_TYPE_RESULT_BGA_BAR	BGA 2D 인식 결과 전송(삭제 예정)
3	PAYLOAD_TYPE_RESULT_BGA_OCR	BGA OCR 인식 결과 전송(삭제 예정)
4	PAYLOAD_TYPE_RESULT_HDI	HDI OCR 인식 결과 전송
5	PAYLOAD_TYPE_RESULT_REEL	REEL 2D 인식 결과 전송
6	PAYLOAD_TYPE_RESULT_AFVI	AFVI 2D 인식 결과 전송(삭제 예정)
7	PAYLOAD_TYPE_RESULT_KPCB	KPCB OCR 인식 결과 전송
8	PAYLOAD_TYPE_RESULT_BGAB	BGA 2D 인식 결과 전송
9	PAYLOAD_TYPE_RESULT_BGAO	BGA OCR 인식 결과 전송
10	PAYLOAD_TYPE_RESULT_Hddb	HDD 2D&OCR 인식 결과 전송
11	PAYLOAD_TYPE_RESULT_HDDM	HDD Mark 인식 결과 전송

Table 4

IMAGE TYPE

NO	항목	내용
1	PAYLOAD_TYPE_IMAGE_Y	ROI 영역 이미지의 Y값. Header의 parameter1에는 0, parameter1에는 width, parameter2에는 height 정보가 담긴다

Table 5

ERROR CODE

Error Code	Value	Comment
EVN_ERR_SUCCESS	0	어떠한에러없이성공적으로호출되었을경우
EVN_ERR_CANT_INIT	-1	WSAStartup 함수호출에실패
EVN_ERR_CANT_CLOSE	-2	WSACleanup 함수호출에실패
EVN_ERR_CANT_CONNECT	-3	특정주소와포트로의연결이실패
EVN_ERR_NOT_CONNECT	-4	연결상태가아님
EVN_ERR_UNSUCCESSFUL	-5	함수의호출이유효하지않은인수나장치의상태이상으로인해실패하는경우
EVN_ERR_WINDOW_CANT_CREATE	-6	이벤트를받을자체윈도우를생성하는도중실패
EVN_ERR_INVALID_PARAMETER	-7	유효하지않은파라미터로함수를호출
EVN_ERR_INSUFFICIENT_RESOURCE	-8	메모리나리소스를할당하는데실패
EVN_ERR_BUFFER_EMPTY	-9	NOTICE 버퍼가비어있다
EVN_ERR_BUFFER_OVERFLOW	-10	NOTICE 버퍼를초과했다
EVN_ERR_OVERFLOW_PAYLOAD	-11	PAYLOAD 크기가지정된크기를초과
EVN_ERR_INVALID_HANDLE	-12	인수의핸들이유효한핸들이아님
EVN_ERR_NOTICEMGR_NOT_CREATE	-13	NoticeMgr의 Create되지 않은 상태에서 사용됨
EVN_ERR_NOTICEMGR_BUFFER_EMPTY	-14	NoticeMgr의 버퍼가 비었음.
EVN_ERR_DEVICE_CANT_THREAD	-15	Socket 전송 Thread 생성 실패
EVN_ERR_SOCKET_READ_ERROR	-16	Socket Read 실패
EVN_ERR_SOCKET_WRITE_ERROR	-17	Socket Write 실패
EVN_ERR_PACKET_PHASE_MISMATCH	-18	Packet 수신중 한 Phase중 다른 PhaseNumber의 패킷을 수신
EVN_ERR_PACKET_INVALID	-19	수신된 패킷의 delimiter가 규격에 맞지 않음
EVN_ERR_CANT_CREATE_EVENT	-20	이벤트 객체 생성에 실패

Table 6