

NET-FRM01 API Programming (Rev 1.1)



Windows, Windows2000, Windows NT and Windows XP are trademarks of **Microsoft**. We acknowledge that the trademarks or service names of all other organizations mentioned in this document as their own property.

Information furnished by DAQ system is believed to be accurate and reliable. However, no responsibility is assumed by DAQ system for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or copyrights of DAQ system.

The information in this document is subject to change without notice and no part of this document may be copied or reproduced without the prior written consent.

Copyrights © 2005 DAQ system, All rights reserved.

Board Level APIs

Overview

int	OpenNetDevice (const char* ip)
BOOL	ResetBoard (int nBoard)
BOOL	CloseNetDevice (void)
int	GetBoardNum (void)

OpenNetDevice

디바이스의 인터넷 프로토콜상의(Internet Protocol) 주소를 Open한다.
프로그램에서 초기에 반드시 한번 함수를 호출하여 디바이스를 Open하여야 한다.

Int **OpenNetDevice (const char* ip)**

Parameters:

ip : 장치의 I.P를 가져온다. (고정 I.P : 192.168.33.129)

Return Value:

함수 호출에 성공한 경우, "0"을 리턴한다.
함수 호출에 실패한 경우, "-1"을 리턴한다. 이것의 의미는 시스템에 장치가 없다는 의미이다.

ResetBoard

현재 시스템(PC)에 장착된 디바이스를 초기화 한다.

BOOL **ResetBoard (int nBoard)**

Parameters:

nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)

Return Value:

함수 호출에 성공일 경우 "TRUE"을 리턴함.
함수 호출에 실패한 경우, 어떤 API함수도 호출할 수 없고 CloseDAQDevice()를 부른다.

CloseNetDevice

Open된 디바이스를 Close한다.

장치의 사용이 끝나게 되면, 반드시 장치를 Close하여 다른 프로그램에서 사용할 수 있도록 한다.

BOOL CloseNetDevice (void)

Parameters:

없음.

Return Value:

디바이스 Open에 성공할 경우 TRUE를 실패할 경우 FALSE를 리턴한다.

GetBoardNum

시스템에 설치된 보드의 개수를 알려준다.

int GetBoardNum (void)

Parameters:

없음.

Return Value:

보드의 넘버. 보드 넘버는 장착되어있는 IP의 설정 값이 넘어온다.

LVDS APIs

Overview

BOOL	FRM_Init (int nBoard)
BOOL	FRM_Close (int nBoard)
BOOL	FRM_Start (int nBoard, int nBytes)
BOOL	FRM_GetFrame (int nBoard, DWORD* nCnt, unsigned char* buf)
BOOL	FRM_GetResolution (int nBoard, DWORD *xRes, DWORD *yRes)
BOOL	FRM_Stop (int nBoard)
BOOL	FRM_SetDataMode (int nBoard, int nMode)
BOOL	FRM_GetVersion (int nBoard, int *nVersion)
DWORD	FRM_GetError (int nBoard, DWORD *dwStatus)
BOOL	FRM_SetPktDelay (int nBoard, int nDelay)

FRM_Init

NET-FRM01보드의 sub-system의 자원, 예를 들어 interrupt와 LVDS control register을 초기화한다.

BOOL FRM_Init (int nBoard)

Parameters:

nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.

FRM_Close

함수에서 사용된 모든 자원을 되돌려준다. 어플리케이션 프로그램은 프로그램 종료 시 이 함수를 부른다.

BOOL FRM_Close (int nBoard)

Parameters:

nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.

FRM_Start

프레임 데이터 수집을 시작한다. 이 함수를 호출한 후에 FRM_GetFrame 함수를 호출하여 데이터가 완성이 되었는가를 확인할 수 있다.

BOOL FRM_Start (int nBoard, int nBytes)

Parameters:

nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)
nBytes : 가로해상도 x 세로해상도 x 1,2,4 바이트(8/16/32bit 선택 시)

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.

FRM_GetFrame

프레임 데이터가 완성이 되었는가를 검사하고 완성이 되었으면 프레임 데이터를 가져온다. 이때 데이터를 받아올 버퍼 크기를 반드시 알려주어야 한다.

BOOL FRM_GetFrame (int nBoard, DWORD* nCnt, unsigned char* buf)

Parameters:

nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)
nCnt : 프레임 데이터를 받아 올 버퍼의 크기가 저장 되어있는 변수의 번지이다.
함수를 호출할 때 버퍼크기를 지정하고 호출한 후에는 변수 값을 읽어서 실제로 읽어 온 개수를 확인한다. 데이터 크기는 바이트 단위이다.
buf : 프레임 버퍼 포인터.

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.
호출이 성공하면 nCnt 값을 확인하여 원하는 크기만큼 읽어 졌는가를 확인한다.
(주) 프레임 데이터가 완성이 되지 않았으면 바로 FALSE를 리턴하고 nCnt 값을 0으로 해서 리턴이 일어난다.

FRM_GetResolution

Video 입력의 해상도를 가져온다.

BOOL FRM_GetResolution (int nBoard, DWORD *xRes, DWORD *yRes)

Parameters:

- nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)
- *nXres : 수평해상도 값 즉, Frame의 Width를 가져온다.
- *nYres : 수직해상도 값 즉, Frame의 Height를 가져온다.

Return Value:

함수 호출에 실패한 경우 'FALSE', 성공일 경우 'TRUE'를 return한다.

FRM_Stop

프레임 데이터 Capture를 중지한다.

BOOL FRM_Stop (int nBoard)

Parameters:

- nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)

Return Value :

함수 호출에 실패한 경우 'FALSE', 성공일 경우 'TRUE'를 return한다.

FRM_SetDataMode

프레임(이미지) 데이터 모드를 선택한다.

BOOL FRM_SetDataMode (int nBoard, int nMode)

Parameters:

- nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)
- nMode : "0" 이면 8bit Mode, "1" 이면 16bit Mode
"2" 이면 24bit Mode, "3" 이면 32bit Mode

Return Value :

함수 호출에 실패한 경우 'FALSE', 성공일 경우 'TRUE'를 return한다.

FRM_GetVersion

현재 프로그램의 버전을 가져온다.

BOOL FRM_GetVersion (int nBoard, int *nVersion)

Parameters:

- nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)
- *nVersion : 현재 프로그램 버전 값.

Return Value :

함수 호출에 실패한 경우 'FALSE', 성공일 경우 'TRUE'를 return한다.

FRM_GetError

프레임(이미지) 에러를 가져온다.

DWORD FRM_GetError (int nBoard, DWORD *dwStatus)

Parameters:

- nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)
- *dwStatus : "1"이면 Overflow error
"2"이면 Undone Error
"4"이면 Size Error

Return Value :

함수 호출에 실패한 경우 'FALSE', 성공일 경우 'TRUE'를 return한다.

FRM_SetPktDelay

첫 번째 패킷을 전송 후 다음 패킷 네트워크 전송 지연을 설정한다.

DWORD FRM_SetPktDelay (int nBoard, int nDelay)

Parameters:

- nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)
- nDelay : 네트워크 Package 지연시간을 0 ~ 15 단계로 설정한다.

Return Value :

함수 호출에 실패한 경우 'FALSE', 성공일 경우 'TRUE'를 return한다.

UART APIs

Overview

BOOL	UART_Init (int nBoard)
BOOL	UART_GetData (int nBoard, DWORD* nCnt, unsigned char* buf)
BOOL	UART_SendData (int nBoard, DWORD* nCnt, unsigned char* buf)
BOOL	UART_Close (int nBoard)
BOOL	UART_SetBaud (int nBoard, DWORD nBaud)

UART_Init

UART sub-system의 자원, 예를 들어 interrupt와 UART control register를 초기화 한다.

BOOL **UART_Init (int nBoard)**

Parameters:

nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.

보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.

UART_GetData

차동 UART를 통하여 문자를 받는다.

BOOL **UART_GetData (int nBoard, DWORD* nCnt, unsigned char* buf)**

Parameters:

nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.

보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)

nCnt : 문자를 나타내는 어드레스를 바이트 사이즈로 받는다.

최대 받을 수 있는 문자는 1000 bytes 로 제한되어 있다.

buf : 버퍼 어드레스.

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.

UART_SendData

자동 UART를 통하여 문자를 보낸다.

BOOL **UART_SendData (int nBoard, DWORD* nCnt, unsigned char* buf)**

Parameters:

- nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
 보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)
- nCnt : 문자를 나타내는 어드레스를 바이트 사이즈로 받는다.
 최대 전송할 수 있는 문자는 1000 bytes 로 제한되어 있다.
- buf : 버퍼 어드레스.

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.

UART_Close

UART 함수에서 사용한 모든 자원을 되돌려 준다.

BOOL **UART_Close (int nBoard)**

Parameters:

- nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
 보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.

UART_SetBaud

UART Baud를 정한다.

BOOL **UART_SetBaud (int nBoard, DWORD nBaud)**

Parameters:

- nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.
 보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)
- nBaud : 0: 9600, 1: 19200, 2: 38400, 3:57600, 4:115200

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.

DIO(Digital Input Output) API Functions

Overview

BOOL DIO_Read (int nBoard, DWORD *dwval)

BOOL DIO_Write (int nBoard, DWORD dwval)

DIO_Read

카메라에서 제어 값을 읽어 온다.

BOOL DIO_Read (int nBoard, DWORD* dwval)

Parameters:

nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.

보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)

*dwval : 카메라 제어 값.

Return Value:

함수 호출에 실패한 경우 'FALSE', 성공일 경우 'TRUE'를 return한다.

DIO_Write

원하는 카메라 제어 값을 출력포트에 출력한다. (카메라 사양 참조)

BOOL DIO_Write (int nBoard, DWORD dwval)

Parameters:

nBoard : 현재 시스템에 장착되어 있는 보드 번호를 알려준다.

보드 번호는 IP 를 이용하여 설정한다. (현재는 0으로 설정)

dwval : 포트에 쓰여지는 값.

Return Value:

함수 호출에 실패할 경우 "FALSE" 성공일 경우 "TRUE"을 리턴함.