

VERSION

1.0

KNPICC30

HID Library Manual .NET Framework 4

Kiko-Net Co., Ltd.

ご注意

本文書の著作権は(有)キコ・ネットが保有します。

本文書の内容を無断で転載することは一切禁止します。

本文書に記載されているサンプルプログラムの著作権は(有)キコ・ネットが保有します。

本文書の内容およびサンプルプログラムに基づき、アプリケーションを運用した結果、万一損害が発生しても、弊社では一切責任を負いませんのでご了承下さい。

本文書の内容については、万全を期して作成いたしました。が、万一ご不審な点、誤りなどお気づきの点がありましたら弊社までご連絡下さい。

本文書の内容は、将来予告なしに変更されることがあります。

商標について

PIC® microcontroller, MPLAB® IDEは、米国およびその他の国々におけるMicrochip Technology Incの登録商標です。

Windows®の正式名称はMicrosoft®Windows®Operating System です。

Microsoft、Windows、Windows NT は、Microsoft Corporation.の米国およびその他の国における商標または登録商標です。

Windows®7、Windows®Vista、Windows®XP、Windows®2000 Professional、

Windows®Millennium Edition、Windows®98 は、Microsoft Corporation の商品名称です。

その他、記載されている会社名、製品名、アプリケーションは各社の登録商標もしくは商標です。

本文書は®、©、™ を記述していない場合があります。

目次

概要.....	3
ファームウェアの配備	4
ライブラリの配備	4
HID ライブラリ	5
KNPIC30 ファームウェアの構造	5
HID 通信.....	6
KnHidControl クラス	6
コンストラクター	6
プロパティ	6
初期化.....	7
ReadDllVersion メソッド	7
Open メソッド	8
Close メソッド	8
ReadDeviceType メソッド	8
リセット/ブート	9
Reset メソッド	9
Boot メソッド	9
プログラム書き込み/読み込み.....	10
ReadProgram メソッド	11
ReadProgramCompleted イベント.....	11
WriteProgram メソッド.....	12
WriteProgramCompleted イベント	12
UDP 通信設定情報書き込み/読み込み	13
ReadUdpDescriptor メソッド	14
ReadUdpDescriptorCompleted イベント.....	14
WriteUdpDescriptor メソッド	15
WriteUdpDescriptorCompleted イベント.....	15
ReadMac メソッド.....	16
ReadMacCompleted イベント.....	16
LED コントロール	17

WriteLED メソッド	18
WriteLEDCompleted イベント	18
ReadLED メソッド	19
ReadLEDCompleted イベント	19
SW コントロール	20
ReadSW メソッド	21
ReadSWCompleted イベント	21
IO コントロール	22
WriteIOConfig メソッド	23
WriteIOConfigCompleted イベント	23
ReadIO メソッド	24
ReadIOCompleted イベント	24
WriteIO メソッド	25
WriteIOCompleted イベント	25
共通クラス	26
UdpDescriptor クラス	26
コンストラクター	26
プロパティ	26
IntelForm クラス	27
コンストラクター	27
プロパティ	27
IOConfigArg クラス	28
コンストラクター	28
プロパティ	28
エラー コード表	29
Bit-Map とコネクタの対応	30
IO ポート コネクタ マップ	30

概要

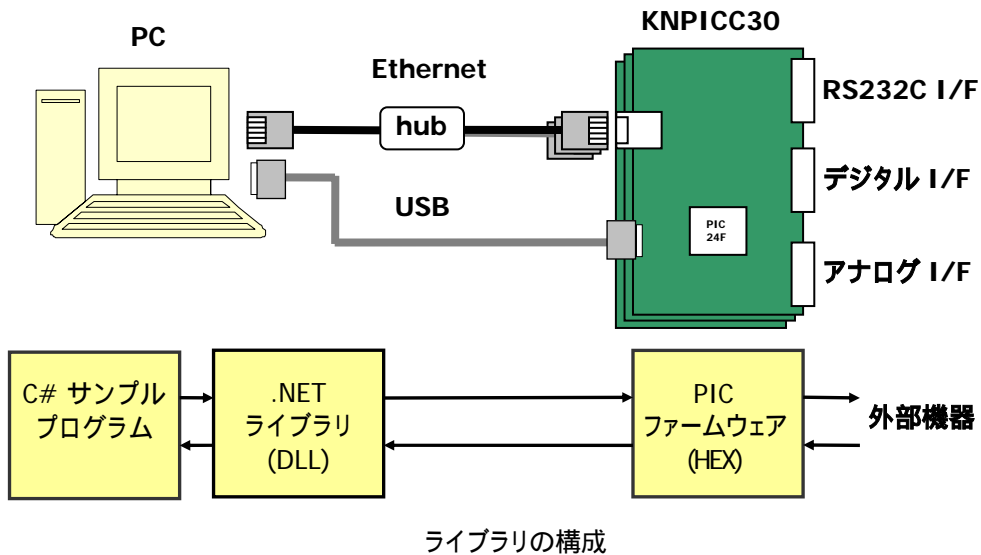
本書は、弊社通信端末CPUボードKNPIC30用HIDソフトウェアライブラリについて解説しています。KNPIC30のハードウェア仕様は、『KNPIC30 Hardware Manual』を参照してください。UDPソフトウェアライブラリの仕様は、『KNPIC30 UDP Library Manual』を参照してください。WinUSBソフトウェアライブラリの仕様は、『KNPIC30 WinUSB Library Manual』を参照してください。

KNPIC30は、マイクロチップ社製ハーバードアーキテクチャ 16ビット PIC24F CPU を搭載した通信端末用CPUボードです。10/100BASE-TX対応Ethernetコントローラと、USB コントローラ(Full-Speed対応)を搭載しています。RS232Cライン レシーバ/トランシーバを各2チャンネル搭載し、最大2チャンネルのRS232C通信が可能です。デジタルI/Oポートを16本、AD変換器入力を6チャンネル搭載しています。

本ソフトウェアライブラリは、KNPIC30用のPICファームウェアを新たに開発することなく、パソコンからKNPIC30の機能 (LED, SW, デジタルI/Oポート)を使用することを目的としています。

本ソフトウェアライブラリは、マイクロソフト社.NET Framework 4 ライブラリ(dll) とマイクロチップ社PIC24Fのファームウェアで構成されています。本ソフトウェアライブラリを利用することで、NET Frameworkアプリケーションを作成するだけで、KNPIC30の機能を利用できるようになります。

PIC 通信制御ファームウェアは、マイクロチップ社が提供する TCP/IP Stack、USB Framework ファームウェアを使用しています。



ファームウェアの配備

KNPICC30 には、工場出荷時に、次のプログラムが事前にインストールされています。ファームウェアの再インストール方法/確認方法については、『**KNPICC30 Firmware Manual**』を参照してください。

- I. **HID ブートローダー**
UDP ブートローダーのインストール/通信パラメータの設定 (USB)
LED の設定, SW の取得, I/O ポートへのアクセス (USB)
- II. **UDP ブートローダー**
通信制御ファームウェアのインストール/通信パラメータの設定 (ネットワーク)
- III. **UDP 通信制御プログラム (ユーザプログラム)**
LED, SW, RS232C, デジタル I/O ポート, AD変換器の制御 (ネットワーク)

ライブラリの配備

KNPICC30 を制御する NET Framework アプリケーションを作成するには、KNPICC30 アクセス クラスライブラリ (**KnApiC30.dll**) をプロジェクトの参照に追加する必要があります。

KNPICC30 を HID 経由で制御する NET Framework アプリケーションを作成するには、HID アクセスライブラリ (**KnHidUsb.dll**) を実行ファイルと同じディレクトリに配置する必要があります。

KNPICC30 を WinUsb 経由で制御する NET Framework アプリケーションを作成するには、WinUsb ドライバをインストールし、WinUsb アクセス ライブラリ (**KnWinUsb.dll**) を実行ファイルと同じディレクトリに配置する必要があります。

ご注意

KNPICC30 クラスライブラリ、および サンプル プログラム内で、NET Framework 4 以前の NET Framework ライブラリを使用している場合があります。
サンプルを実行するためには、Visual C# 2010 Express(無償)をインストールしてください。実行に必要な NET Framework ライブラリがすべてインストールされます。

HIDライブラリ

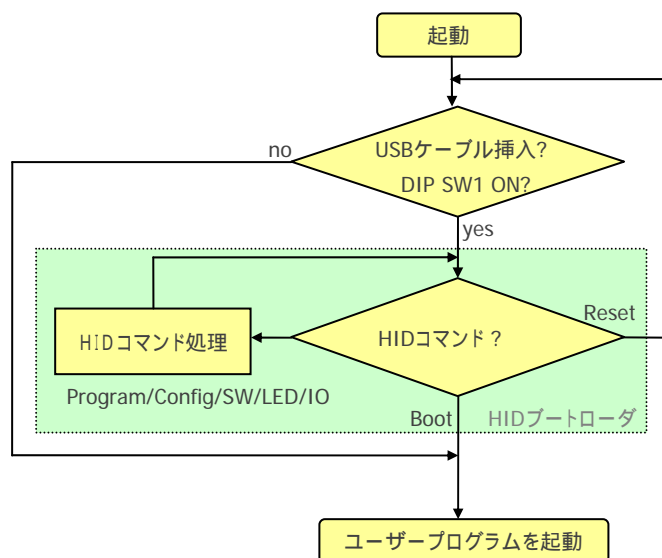
HIDライブラリは、USBのHIDクラスを使用して、ホスト(パソコン)とデバイス(KNPICC30)間の通信を行います。HIDクラスは、専用USBドライバをインストールする必要がなく、容易に使用できますが、通信速度が **64Kbyte/Sec** に制限されます (インターラプト転送)。HIDライブラリの機能は、LEDの設定、SWの取得、I/Oポートへのアクセス、ユーザープログラムの書き込み/読み込み、UDP通信設定の書き込み/読み込みに限定されています。現行のHIDライブラリは、複数のデバイス(KNPICC30)に対応していません。USB上に複数のデバイス(KNPICC30)が検出された場合、最初に見つかったデバイスのみが制御可能です。

(注) 本書では、KNPIC30を制御する装置 (主にPC)をホストと記述し、KNPIC30をデバイスと記述しています。

KNPIC30 ファームウェアの構造

KNPIC30には、工場出荷時にHIDブートローダがインストールされています。起動時に、USBケーブルが挿入されているか、DIP SW1 が ONであることを検知すると、HIDブートローダが起動し、HIDコマンド待ち状態になります。USBケーブルが挿入されておらず、DIP SW1 が OFFの場合、ユーザープログラム(工場出荷時はUDPブートローダ)が起動します。HIDブートローダで、LEDの設定、SWの取得、I/Oポートへのアクセス、ユーザープログラムの書き込み/読み込みが可能です。HIDブートローダの状態、Bootコマンドが入力されると、ユーザープログラムが起動します。

ファームウェアの詳細については、『KNPIC30 Firmware Manual』を参照してください。



HID 通信

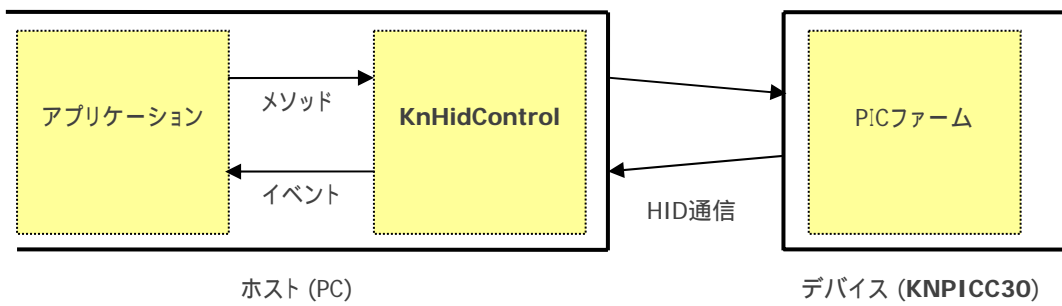
HID通信は、USB HIDクラスを利用し、ホストがデバイスを制御するためにパケットを送信し、その応答を受け取る往復通信です。

KnHidControl クラス

.NET Framework 4

名前空間: KnApis.Hid

説明: KnHidControl クラスは、HID通信を行うための手段を、メソッドとイベントのペアで提供しています。メソッドで、HIDパケットをデバイスに送信し、イベントで、デバイスからの応答結果をアプリケーションに通知します。HID通信メソッドを実行する前に、対応するイベントのイベントハンドラを登録する必要があります。



コンストラクター

名前	説明
KnHidControl	KnHidControl クラスの新しいインスタンスを初期化します。

プロパティ

名前	型	説明
IsBusy	bool	現在の通信状態を取得します。 true: 通信中 false: 通信可能 (通信停止状態)

初期化

機能: DLLのバージョンを取得します。

メソッド:

名前	説明
ReadDllVersion	DLLのバージョンを取得します。

機能: ドライバをオープンします。

メソッド:

名前	説明
Open	ドライバをオープンします。

機能: ドライバをクローズします。

メソッド:

名前	説明
Close	ドライバをクローズします。

機能: デバイス ファームウェアのタイプ/バージョンを取得します。

メソッド:

名前	説明
ReadDeviceType	ファームウェアのタイプ/バージョンを取得します。

ReadDllVersion メソッド

解説: DLLのバージョンを取得します。このメソッドに応答イベントはありません。

構文: void ReadDllVersion(ref int major, ref int minor)

パラメーター:

名前	型	説明
major	int	DLLのバージョン (Major)
minor	int	DLLのバージョン (Minor)

戻り値: なし

Open メソッド

解説: デバイスをオープンします。このメソッドに応答イベントはありません。

構文: int Open(int vid, int pid)

パラメーター:

名前	型	説明
vid	int	デバイスのVID
pid	int	デバイスのPID

戻り値:

型	説明
int	デバイスの数

Close メソッド

解説: ドライバをクローズします。このメソッドに応答イベントはありません。

構文: void Close()

パラメーター: なし

戻り値: なし

ReadDeviceType メソッド

解説: ファームウェアのタイプ/バージョンを取得します。このメソッドに応答イベントはありません。

構文: int ReadDeviceType(ref int major, ref int minor)

パラメーター:

名前	型	説明
major	int	ファームウェアのバージョン (Major)
minor	int	ファームウェアのバージョン (Minor)

戻り値:

型	説明
int	ファームウェアのタイプ 0 : HIDブートローダ 1 : UDPブートローダ 2 : UDPユーザープログラム 3 : HIDユーザープログラム 4 : TCPユーザープログラム 5 : WinUSBユーザープログラム

リセット/ブート

機能: デバイスをリセットします。

メソッド:

名前	説明
Reset	デバイスをリセットします。

機能: ユーザー プログラムを起動します。

メソッド:

名前	説明
Boot	ユーザー プログラムを起動します。

Reset メソッド

解説: デバイスをリセットします。この通信に応答イベントはありません。

構文: int Reset()

パラメーター: なし

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

Boot メソッド

解説: デバイスをBOOTコマンドを送信します。この通信に応答イベントはありません。

構文: int Boot()

パラメーター: なし

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

ご注意

ユーザー プログラム領域 (004120h 番地)にジャンプします。
ユーザー プログラム領域にプログラムがインストールされている必要があります。
詳細は、『KNPIC30 Firmware Manual』を参照してください。

プログラム書き込み/読み込み

機能: デバイスからユーザー プログラムを読み込みます。

メソッド:

名前	説明
ReadProgram	デバイスからユーザー プログラムを読み込みます。

イベント:

名前	説明
ReadProgramCompleted	ReadProgramの操作の完了時、キャンセル時に発生します。

機能: デバイスにユーザー プログラムを書き込みます。

メソッド:

名前	説明
WriteProgram	WRITE_PROGRAM設定コマンドを送信し、応答を待ちます。

イベント:

名前	説明
WriteProgramCompleted	WriteProgramの操作の完了時、キャンセル時に発生します。

ReadProgram メソッド

解説: デバイスからプログラムを読み込みます。結果は、ReadProgramCompletedイベントで取得します。

構文: int ReadProgram(int saddr, int eaddr, bool blankterminate)

int ReadProgram(int saddr, int eaddr, bool blankterminate, object ID)

パラメーター:

名前	型	説明
saddr	int	読み込みを開始するアドレス
eaddr	int	読み込みを終了するアドレス
blankterminate	bool	true: ブランクが128バイト連続した場合読み込みを終了する false: ブランクが128バイト連続しても読み込みを終了しない
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

ReadProgramCompleted イベント

解説: プログラムの読み込み結果を取得します。ReadProgramメソッドの実行後に発生します。

定義: KnHidReadProgramCompletedEventHandler ReadProgramCompleted

デリゲート宣言: KnHidReadProgramCompletedEventHandler(int result, List<IntelForm> pgm, object ID)

パラメーター:

名前	型	説明
result	int	0: 読み込み成功 負の値: エラー コード表を参照
pgm	List<IntelForm>	Intel Hex形式(binary)の読み込みプログラム
ID	object	ReadProgramで指定されたID

WriteProgram メソッド

解説: デバイスにプログラムを書き込みます。結果は、WriteProgramCompletedイベントで取得します。

構文: int WriteProgram(List<IntelForm> hex, int saddr, int eaddr)

int WriteProgram(List<IntelForm> hex, int saddr, int eaddr, object ID)

パラメーター:

名前	型	説明
pgm	List<IntelForm>	Intel Hex形式(binary)の書き込みプログラム
saddr	int	消去開始アドレス
eaddr	int	消去終了アドレス
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

WriteProgramCompleted イベント

解説: ユーザー プログラムを読み込み結果を取得します。WriteProgramメソッドの実行後に発生します。

定義: event KnHidWriteProgramCompletedEventHandler WriteProgramCompleted

デリゲート宣言: delegate void KnHidWriteProgramCompletedEventHandler(int result, object ID)

パラメーター:

名前	型	説明
result	int	0: 読み込み成功 負の値: エラー コード表を参照
ID	object	WriteProgramで指定されたID

UDP 通信設定情報書き込み/読み込み

機能: デバイスからUDP通信設定情報を読み込みます。

メソッド:

名前	説明
ReadUdpDescriptor	UDP通信設定情報読み込みコマンドを送信し、応答を待ちます。

イベント:

名前	説明
ReadUdpDescriptorCompleted	ReadUdpDescriptorの操作の完了時、キャンセル時に発生します。

機能: デバイスに通信設定情報を書き込みます。

メソッド:

名前	説明
WriteUdpDescriptor	UDP通信設定情報書き込みコマンドを送信し、応答を待ちます。

イベント:

名前	説明
WriteUdpDescriptorCompleted	WriteUdpDescriptorの操作の完了時、キャンセル時に発生します。

機能: デバイスのMacアドレスを読み込みます。

メソッド:

名前	説明
ReadMac	Macアドレスを読み込みみコマンドを送信し、応答を待ちます。

イベント:

名前	説明
ReadMacCompleted	ReadMacの操作の完了時、キャンセル時に発生します。

ReadUdpDescriptor メソッド

解説: デバイスから通信設定情報を読み込みます。結果は、ReadUdpDescriptorCompletedイベントで取得します。

構文: int ReadUdpDescriptor()

int ReadUdpDescriptor(object ID)

パラメーター:

名前	型	説明
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

ReadUdpDescriptorCompleted イベント

解説: 通信設定情報の読み込み結果を取得します。ReadUdpDescriptorメソッドの実行後に発生します。

定義: event KnHidReadUdpDescriptorCompletedEventHandler ReadUdpDescriptorCompleted

デリゲート宣言: delegate void KnHidReadUdpDescriptorCompletedEventHandler(int result, UdpDescriptor dsc, object ID)

パラメーター:

名前	型	説明
result	IPAddress	0: 読み込み成功 負の整数: エラー コード表を参照
Dsc	UdpDescriptor	設定情報クラス ReadOnly のパラメータは 常に 0 で意味をもちません
ID	object	ReadUdpDescriptorで指定されたID

WriteUdpDescriptor メソッド

解説: デバイスに通信設定情報を書き込みます。結果は、WriteUdpDescriptorCompletedイベントで取得します。

構文: int WriteUdpDescriptor(UdpDescriptor dsc)

int WriteUdpDescriptor(UdpDescriptor dsc, object ID)

パラメーター:

名前	型	説明
Dsc	UdpDescriptor	設定情報クラス ReadOnly のパラメータは書き込みません
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

WriteUdpDescriptorCompleted イベント

解説: 通信設定情報の書き込み結果を取得します。WriteUdpDescriptorメソッドの実行後に発生します。

定義: event KnHidWriteUdpDescriptorCompletedEventHandler WriteUdpDescriptorCompleted

デリゲート宣言: delegate void KnHidWriteUdpDescriptorCompletedEventHandler(int result, object ID)

パラメーター:

名前	型	説明
result	int	0: 読み込み成功 負の値: エラー コード表を参照
ID	object	WriteUdpDescriptorで指定されたID

ReadMac メソッド

解説: デバイスからMacアドレスを読み込みます。結果は、ReadMacCompletedイベントで取得します。

構文: int ReadMac()

int ReadMac(object ID)

パラメーター:

名前	型	説明
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

ReadMacCompleted イベント

解説: Macアドレスの読み込み結果を取得します。ReadMacメソッドの実行後に発生します。

定義: event KnHidReadMacCompletedEventHandler ReadMacCompleted

デリゲート宣言: delegate void KnHidReadMacCompletedEventHandler(int result, byte[] mac, object ID)

パラメーター:

名前	型	説明
result	IPAddress	0: 読み込み成功 負の整数: エラー コード表を参照
mac	byte[]	Macアドレス
ID	object	ReadMacで指定されたID

LED コントロール

機能: デバイスのLEDを設定します。

メソッド:

名前	説明
WriteLED	LED設定コマンドを送信し、応答を待ちます。

イベント:

名前	説明
WriteLEDCompleted	WriteLEDの操作の完了時、キャンセル時に発生します。

機能: デバイスのLEDの状態を取得します。

メソッド:

名前	説明
ReadLED	LED状態取得コマンドを送信し、応答を待ちます。

イベント:

名前	説明
ReadLEDCompleted	ReadLEDの操作の完了時、キャンセル時に発生します。

WriteLED メソッド

解説: デバイスのLEDを設定します。結果は、WriteLEDCompletedイベントで取得します。

構文: int WriteLED(int led1, int led2)

int WriteLED(int led1, int led2, object ID)

パラメーター:

名前	型	説明
led1	int	LEDの状態: 0: 消灯 1: 点灯
led2	int	LEDの状態: 0: 消灯 1: 点灯
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

WriteLEDCompleted イベント

解説: LED設定コマンドの通信結果を取得します。WriteLEDメソッドの実行後に発生します。

定義: event KnHidWriteLEDCompletedEventHandler WriteLEDCompleted

デリゲート宣言: delegate void KnHidWriteLEDCompletedEventHandler(int result)

パラメーター:

名前	型	説明
result	int	0: 書き込み成功 負の値: エラー コード表を参照
ID	object	WriteLEDで指定されたID

ReadLED メソッド

解説: デバイスのLEDの状態を取得します。結果は、ReadLEDCompletedイベントで取得します。

構文: int ReadLED()

int ReadLED(object ID)

パラメーター:

名前	型	説明
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

ReadLEDCompleted イベント

解説: LED状態取得コマンドの通信結果を取得します。ReadLEDメソッドの実行後に発生します。

定義: event KnHidReadLEDCompletedEventHandler ReadLEDCompleted

デリゲート宣言: delegate void KnHidReadLEDCompletedEventHandler(int result, int led1, int led2, object ID)

パラメーター:

名前	型	説明
result	int	0: 取得成功 負の値: エラー コード表を参照
led1	int	LED1の状態: 0: 消灯 1: 点灯
led2	int	LED2の状態: 0: 消灯 1: 点灯
ID	object	ReadLEDで指定されたID

SW コントロール

機能: デバイスのSWの状態を取得します。

メソッド:

名前	説明
ReadSW	SWの状態取得コマンドを送信し、応答を待ちます。

イベント:

名前	説明
ReadSWCompleted	ReadSWの操作の完了時、キャンセル時に発生します。

ReadSW メソッド

解説: デバイスのSWの状態を取得します。結果は、ReadSWCompletedイベントで取得します。

構文: int ReadSW()

int ReadSW(object ID)

パラメーター:

名前	型	説明
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

ReadSWCompleted イベント

解説: SW状態取得コマンドの通信結果を取得します。ReadSWメソッドの実行後に発生します。

定義: event KnHidReadSWCompletedEventHandler ReadSWCompleted

デリゲート宣言: delegate void KnHidReadSWCompletedEventHandler(int result, int sw1, int sw2, object ID)

パラメーター:

名前	型	説明
result	int	0: 読み込み成功 負の値: エラー コード表を参照
sw1	int	SW1の状態: 0: OFF 1: ON
sw2	int	SW2の状態: 0: OFF 1: ON
ID	object	ReadSWで指定されたID

IO コントロール

機能: デバイスのIOポートの構成を設定します。

メソッド:

名前	説明
WriteIOConfig	IOポートの構成設定コマンドを送信し、応答を待ちます。

イベント:

名前	説明
WriteIOConfigCompleted	WriteIOConfigの操作の完了時、キャンセル時に発生します。

機能: デバイスのIOポートからデータを読み取ります。

メソッド:

名前	説明
ReadIO	IOポートの読み込みコマンドを送信し、応答を待ちます。

イベント:

名前	説明
ReadIOCompleted	ReadIOの操作の完了時、キャンセル時に発生します。

機能: デバイスのIOポートにデータを書き込みます。

メソッド:

名前	説明
WriteIO	IOポートの書き込みコマンドを送信し、応答を待ちます。

イベント:

名前	説明
WriteIOCompleted	WriteIOの操作の完了時、キャンセル時に発生します。

WriteIOConfig メソッド

解説: デバイスに対してIOポートの構成設定コマンドを送信します。通信結果は、WriteIOConfigCompletedイベントで取得します。

構文: int WriteIOConfig(IOConfigArg param)
 int WriteIOConfig(IOConfigArg param, object ID)

パラメーター:

名前	型	説明
param	IOConfigArg	構成設定データ
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

ご注意

HIDライブラリには イベント通信はありません。IOConfigArgクラスのTrigger、Interval、Eventプロパティは意味をもちません。

WriteIOConfigCompleted イベント

解説: IOポート設定コマンドの通信結果を取得します。WriteIOConfigメソッドの実行後に発生します。

定義: event KnHidWriteIOConfigCompletedEventHandler WriteIOConfigCompleted

デリゲート宣言: delegate void KnHidWriteIOConfigCompletedEventHandler(int result, object ID)

パラメーター:

名前	型	説明
result	int	0: 設定成功 負の値: エラー コード表を参照
ID	object	WriteIOConfigで指定されたID

ReadIO メソッド

解説: デバイスに対してIOポート読み込みコマンドを送信します。結果は、ReadIOCompletedイベントで取得します。

構文: int ReadIO()

int ReadIO(object ID)

パラメーター:

名前	型	説明
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

ReadIOCompleted イベント

解説: IOポート読み込みコマンドの通信結果を取得します。ReadIOメソッドの実行後に発生します。

定義: event KnHidReadIOCompletedEventHandler ReadIOCompleted

デリゲート宣言: delegate void KnHidReadIOCompletedEventHandler(int result, int data, object ID)

パラメーター:

名前	型	説明
result	int	0: 読み込み成功 負の値: エラー コード表を参照
data	int	読み込みデ - タ
ID	object	ReadIOで指定されたID

WriteIO メソッド

解説: デバイスに対してIOポートへの書き込みコマンドを送信します。結果は、WriteIOCompletedイベントで取得します。

構文: int WriteIO(int data)

パラメーター:

名前	型	説明
data	int	書き込みデ - タ
ID	object	メソッドの識別子

戻り値:

型	説明
int	0: メソッド受付成功 負の値: エラー コード表を参照

WriteIOCompleted イベント

解説: IOポートの書き込みコマンドの通信結果を取得します。WriteIOメソッドの実行後に発生します。

定義: event KnHidWriteIOCompletedEventHandler WriteIOCompleted

デリゲート宣言: delegate void KnHidWriteIOCompletedEventHandler(int result, object ID)

パラメーター:

名前	型	説明
result	int	0: 書き込み成功 負の値: エラー コード表を参照
ID	object	WriteIOで指定されたID

共通クラス

この項には、各通信で使用される汎用的なクラスが含まれます。

UdpDescriptor クラス

.NET Framework 4

名前空間: KnApis.Common

説明: UdpDescriptor クラスは、デバイス(KNPICC30)の通信設定情報を指定するクラスです。

コンストラクター

名前	説明
UdpDescriptor	UdpDescriptor クラスの新しいインスタンスを初期化します。

プロパティ

名前	型	説明
DeviceIP	IPAddress	現在のIPアドレス (ReadOnly)
DeviceSubnet	IPAddress	現在のサブネットマスク (ReadOnly)
DeviceMac	byte[6]	Macアドレス (ReadOnly)
DefaultIP	IPAddress	IPアドレスの初期値
DefaultSubnet	IPAddress	サブネットマスクの初期値
DefaultGateway	IPAddress	デフォルト ゲートウェイ
PrimaryDNS	IPAddress	プライマリDNSサーバ
SecondaryDNS	IPAddress	セカンダリDNSサーバ
DHCPEnable	bool	True: DHCP有効 false: DHCP無効
NetBIOSName	byte[16]	NetBIOS名
HostIP	IPAddress	ホストのIPアドレス (ReadOnly)
HostMac	byte[6]	ホストのMacアドレス (ReadOnly)
AnnouncePort	int	Announce通信用ポート
DiscoveryPort	int	ディスカバリ通信用ポート
ControlPort	int	コントロール通信用ポート
EventPort	int	イベント通信用ポート
AnnounceneEnable	bool	True: Announce通信有効 false: 無効
AnnounceInterval	int	Announce通信インターバル (mS Max 65536)

名前	型	説明
EventTimeOut	int	イベント通信タイムアウト時間 (mS Max 65536)
EventRetry	int	イベント通信リトライ回数 (Max 255)
DeviceMode	int	1: UDP ブートローダ 2: ユーザー プログラム (ReadOnly)
StackVersion	byte[16]	TCP/IPプロトコル スタックのバージョン (ReadOnly)
FirmMajor	int	ファームウェア バージョン (Major) (ReadOnly)
FirmMinor	int	ファームウェア バージョン (Minor) (ReadOnly)

IntelForm クラス

.NET Framework 4

名前空間: KnApis.Common

説明: IntelForm クラスは、IntelHex形式の1レコード(128byte)を表現するクラスです(Binary形式)。デバイスにプログラムを読み込み、または 書き込みをおこなう場合に使用します。

コンストラクター

名前	説明
IntelForm	IntelForm クラスの新しいインスタンスを初期化します。

プロパティ

名前	型	説明
Type	int	レコードのタイプ (通常: 0)。
Length	int	データ長
Offset	int	データの開始アドレス
bData	byte[]	データ (128byte)

IOConfigArg クラス

.NET Framework 4

名前空間: KnApis.Common

説明: IOConfigArgクラスは、デバイス(KNPIC30)のIOポート構成を指定するクラスです。

コンストラクター

名前	説明
IOConfigArg	IOConfigArgクラスの新しいインスタンスを初期化します。

プロパティ

名前	型	説明
Direction	int	IOポート入出力をbit単位で指定します。 0: 出力 1: 入力
OpenDrain	int	IOポート出力形式をbit単位で指定します。 0: トーテン ポール 1: オープン ドレイン (Directionで、出力を指定した場合の意味をもちます)
Trigger	int	イベント通信のトリガーをbit単位で指定します。 0: 状態変化した場合にイベント通信を発生しない 1: 状態変化した場合にイベント通信を発生する
Interval	int	イベント通信 を発生させるためのIOポートのポーリング周期を指定します。(mS: 1 < Interval < 65534)
Event	int	イベント通信の有効化/無効化。 0: イベント通信無効 1: イベント通信有効
Readed	int	IOポートの読み込み値 (ReadOnly)

*IOポートのbitとコネクタの対応は、『Bit-Mapとコネクタの対応』の「IOポート コネクタ マップ」を参照してください。

エラーコード表

名前空間: KnApis.Common

列挙子宣言 enum ERRORTYPE

列挙子	数値	説明
SUCCESS	0	成功
TIMEOUT	-1	タイムアウト
CANCEL	-2	キャンセル
EXCEPTION	-3	例外発生
BUSY	-4	スレッドがビジー
NOT_BUSY	-5	スレッドがビジーでない
NOCALLBACK	-6	コールバックが未登録
PARAMETER	-7	パラメータ エラー
UNKNOWN	-8	未知のエラー
DEVICE_NOT_SUPPORT	-20	デバイスがサポートしていない
DEVICE_DENY	-21	デバイスが拒否
DEVICE_INTERNAL	-22	デバイス内部エラー
DEVICE_ERASE	-23	フラッシュ イレース エラー
DEVICE_SEND	-24	送信エラー
COMMAND	-30	コマンド エラー
LENGTH	-31	レングス エラー
SEQUENCENUM	-32	シーケンス番号 エラー
DEVICEMODE	-33	モード エラー
CTRLCODE	-34	コントロールコード エラー
ERRORCODE	-35	エラーコード エラー
HEADERSIZE	-36	ヘッダーサイズ エラー
DATASIZE	-37	データサイズ エラー

Bit-Map とコネクタの対応

IO ポート コネクタ マップ

Bit-Map	コネクタ	ピン番号	CPU ピン番号	CPUピン機能
bit23	CN7	1	24	VREF+/PMA6/CN42/RA10
bit22	CN7	2	23	VREF-/PMA7/CN41/RA9
bit21	CN7	5	20	PGED1/AN0/RP0/CN2/RB0
bit20	CN7	6	19	PGEC1/AN1/RP1/CN3/RB1
bit19	CN7	9	18	AN2/C2INB/VMIO/ RP13 /CN4/RB2
bit18	CN7	10	17	AN3/C2INA/VPIO/CN5/RB3
bit17	CN7	13	16	PGED3/AN4/C1INB/USBOEN/RP28/CN6/RB4
bit16	CN7	14	15	PGEC3/AN5/C1INA/VBUSON/RP18/CN7/RB5
bit15	CN5	1	38	RP5/CN21/RD15
bit14	CN5	2	37	RPI43/CN20/RD14
bit13	CN5	3	65	CN19/RD13
bit12	CN5	4	64	RPI42/CN57/RD12
bit11	CN5	5	56	SCL1/RP3/PMCS2/CN55/RD10
bit10	CN5	6	55	SDA1/DPLN/RP4/CN54/RD9
bit9	CN5	7	54	DMLN/RTCC/RP2/CN53/RD8
bit8	CN5	8	63	RP22/PMBE/CN52/RD3
bit7	CN5	9	62	DPH/RP23/CN51/RD2
bit6	CN5	10	61	VCPCON/RP24/CN50/RD1
bit5	CN5	11	58	DMH/RP11/INT0/CN49/RD0
bit4	CN5	12	53	SDA2/RPI35/CN44/RA15
bit3	CN5	13	52	SCL2/RPI36/CN43/RA14
bit2	CN5	14	73	VCMPST2/CN69/RF1
bit1	CN5	15	74	CN78/RG1
bit0	CN5	16	75	CN77/RG0

*bit23 – bit16 は、ADCを無効化した場合に使用できます。