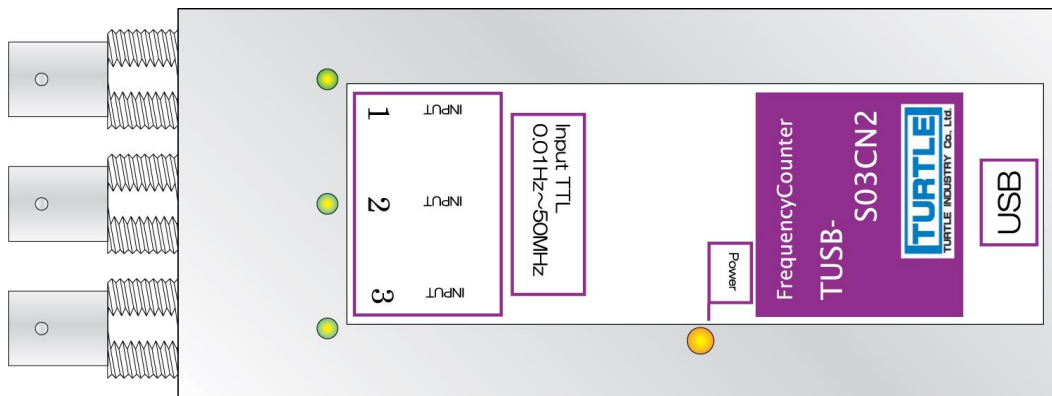


TUSB-S03CN2(B)(Z)

USB インタフェース付き周波数カウンタユニット



取扱説明書



本文中のマークについて(必ず始めにお読み下さい)

この取扱説明書には、あなたや他の人々への危害や財産への損害を未然に防ぎ、本製品を安全にお使いいただくために、守っていただきたい事項を示しています。

その表示と図記号の意味は次のようになっています。内容をよみ理解してから本文をお読み下さい。

 警告	この表示を無視して、誤った取扱をすると、人が死亡または重傷を負う可能性がある内容を示しています。
 注意	この表示を無視して、誤った取扱をすると、人が損害を負う可能性が想定される内容および物的損害のみの発生が想定される内容を示しています。

- ① 製品の仕様および取扱説明書の内容は予告なく変更することがあります。
- ② 本製品および本取扱説明書の一部または全部を無断転載することは禁じられています。
- ③ 本取扱説明書の内容は万全を期して作成いたしました。万が一不審な事やお気づきの事がございましたら、(株) タートル工業 サービス課までご連絡下さい。
- ④ 当社では、本製品の運用を理由とする損失、逸失利益等の請求につきましては、上記に関わらずいかなる責任も負いかねますので、予めご了承下さい。
- ⑤ 本製品は、人命に関わる設備や機器、高度な信頼性を必要とする設備や機器などへの組込や制御などへの使用は意図されておりません。これら設備や機器などに本装置を使用され人身事故、財産損害などが生じても、当社はいかなる責任も負いかねます。
- ⑥ 本製品およびソフトウェアが外国為替及び外国貿易管理法の規定により戦略物資（又は役務）に該当する場合には日本国外へ輸出する際に日本国政府の輸出許可が必要です。

©2019 Turtle Industry Co., Ltd. All rights reserved.

株式会社タートル工業の許可なく、本書の内容の複製、改変などを行うことはできません。

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

その他、記載されている会社名、製品名は、各社の商標および登録商標です。

使用上の警告と注意



警告

接続機器の電源を全て切断してから端子台への接続および取り外しを行ってください。接続機器によっては感電の危険があります。



注意

端子台に印加する電圧、電流は仕様に規定された値を守ってください。過熱による火災や漏電のおそれがあります。

端子台のカバーを外したまま端子台に電圧を印加しないで下さい。接続端子に触ると感電の危険があります。

水や薬品のかかる可能性のある場所でご使用ならささないでください。火災やその他の災害の原因となる可能性があります。

発火性ガスの存在するところでご使用なさないでください。引火により火災、爆発の可能性があります。

不安定な所には設置しないでください。落下によりけがをする恐れがあります。

煙や異臭の発生した時は直ちにご使用をおやめ下さい。USB ケーブルを取り外し、当社サービス課までご相談下さい。

1. はじめに.....	5
1.1 製品概要.....	5
1.2 製品構成.....	5
2. 各部の名称.....	6
2.1 上面.....	6
2.2 リア(後部パネル).....	6
3. 各部説明.....	7
3.1 カウント入力.....	7
3.2 入力共通の仕様について.....	7
3.3 USB コネクタ.....	7
3.4 ID 選択スイッチ.....	8
5 ソフトウェアについて.....	9
5.1 ドライバ、アプリケーションディスクのディレクトリについて.....	9
5.2 アプリケーションソフトの実行について.....	10
6. カウント値と周波数、周期の関係.....	11
6.1 周波数モード.....	11
6.2 周期モード.....	11
6.3 低い周波数の信号測定をする方法.....	12
6.4 短い周期の信号測定をする方法.....	12
7. プログラミング.....	13
7.1 Visual C++ 6.0 での使用.....	13
7.1.1 使用準備.....	13
7.1.2 関数の呼び出し方法.....	13
7.1.3 サンプルコード.....	14
7.2 Visual Basic 6 での使用.....	14
7.2.1 使用準備.....	14
7.2.2 プロシージャの呼び出し方法.....	15
7.2.3 サンプルコード.....	15
7.3 Visual Basic .NET での使用.....	16
7.3.1 使用準備.....	16
7.3.2 プロシージャの呼び出し方法.....	16
7.3.3 サンプルコード.....	17
7.4 関数説明.....	18
7.5 エラーコード表.....	27
8. その他.....	28

8.1	USB について	28
8.2	連絡先.....	29
9.	仕様	30
9.1	仕様概要.....	30
9.2	取り付け穴寸法図.....	31

1. はじめに

この度は、(株)タートル工業製の USB インタフェース付き周波数カウンタユニット TUSB-S03CN2(B)(Z)をお買い求めいただき、誠にありがとうございます。

本書は、本製品の特徴、使用方法、取扱における注意事項、その他本製品に関する情報など、本製品をご使用される上で必要な事項について記述されております。

誤った使用をすると本製品の破損だけでなく重大な事故が発生する事も考えられます。本書の内容をよくご理解の上、正しくご使用下さる様をお願いします。

1.1 製品概要

本製品は、先進のインタフェースである USB(Universal Serial Bus)を使用したコンピュータインターフェースユニットです。コンピュータを使用してデジタルパルス信号の周波数および周期計測ができます。ドライバソフトウェア、Visual C++ 6.0 と Visual Basic 6.0、.NET のサンプルソフトウェアを利用できますので、これらの応用によって短時間に利用する事が可能です。

1.2 製品構成

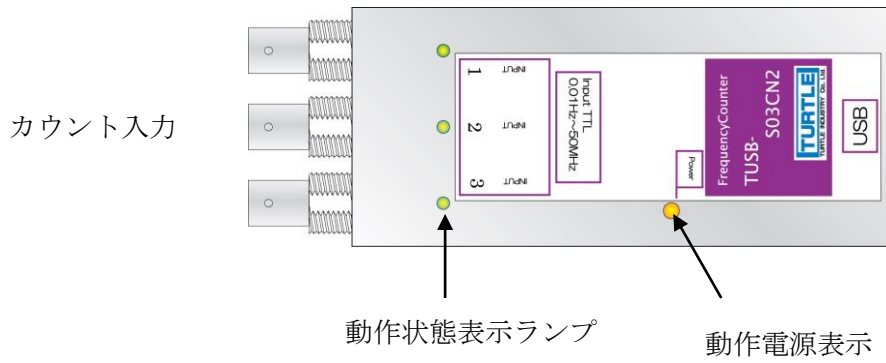
本製品には以下の物が含まれます。

- ① TUSB-S03CN2(B)(Z)本体
- ② USB ケーブル(1m)

不足品などがあれば、当社サービス課までご連絡下さい。

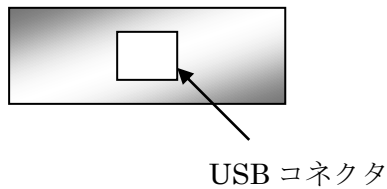
2. 各部の名称

2.1 上面



カウント入力	カウントする信号を入力します(TTL レベル)
動作状態表示ランプ	信号入力エッジ検出時点灯
動作電源表示	ユニットの電源が入っている時に点灯します

2.2 リア(後部パネル)



USB コネクタ	USB ケーブルを使用して PC と接続します
----------	-------------------------

3. 各部説明

3.1 カウント入力

この入力にカウントするべきデジタル信号を入力します。

この入力は TTL 論理レベル入力の立上り又は立下りエッジでカウントします。立上り、立下りの選択はソフトウェアから行います。

3.2 入力共通の仕様について

- 入力スレッシュホールドレベルについて
2V 以上で HIGH、0.6V 以下で LOW と判定します。
入力回路は 0.3V 以上のヒステリシス電圧を持っていますので、立上りの遅い波形でも安心して入力することが出来ます。
- パルスの周波数等について
入力信号は HIGH 区間 10nS 以上、LOW 区間 10nS 以上確保してください。
- 非接続時の入力端子状態について
入力端子は約 25k Ω でプルアップされており、何も接続しませんでした HIGH と判断されます。
- 入力容量について
入力容量は約 15pF です。

3.3 USB コネクタ

付属の USB ケーブルを使用して、ご利用されるコンピュータまたはハブに接続してください。

※ 初めて接続される時にはインストール作業が必用です。インストールマニュアルを参照してください。

3.4 ID 選択スイッチ

同一の PC に本装置を複数台接続する時に ID スイッチを使用します（工場出荷時は 0 となっています）。ID スイッチは本体内部にありますので、変更する場合は天板を取り外します。

変更方法

- ① 側面のネジを左右合わせて 2 個取り外します。



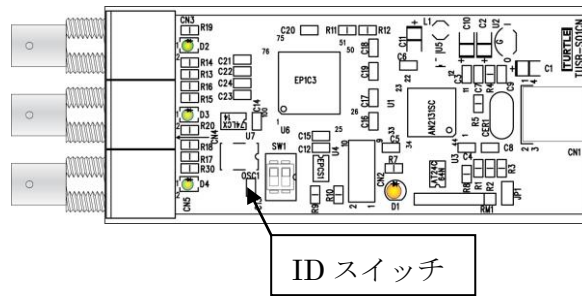
注意

ID 変更時は金属加工部だけがをしないように十分ご注意ください。内部には鋭利な部分があり、手などを切っけがをするおそれがあります。

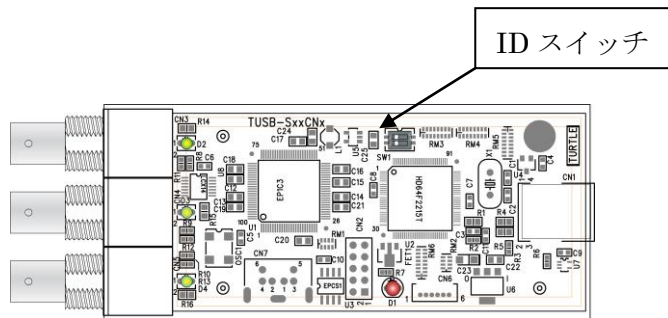
- ② 下記の場所にスイッチがありますので、精密ドライバーの先などで切り替えてください。

※ バージョンによってスイッチの場所と形状が異なりますので、ご注意ください。

TUSB-S03CN2



TUSB-S03CN2B(Z)



ID 番号	スイッチビット 1	スイッチビット 2
0	OFF	OFF
1	ON	OFF
2	OFF	ON
3	ON	ON

5 ソフトウェアについて

5.1 ドライバ、アプリケーションディスクのディレクトリについて

[ROOT]	: ドライバ、アプリケーションディスクルート
-[TUSBCN2]	
-[DRV]	: ドライバ
-[APP]	: アプリケーションインストーラ
-[DOC]	: ドキュメント(取扱説明書等)
-[DEV]	:
-[TOOLS]	:LIB,H,BAS,VB ファイル
-[VB6]	:Visual Basic 6 用 サンプルプロジェクト
-[VBNET]	:Visual Basic .NET 用 サンプルプロジェクト
-[VC]	:Visual C++ 用 サンプルプロジェクト

- [DRV]ディレクトリ
この階層にはドライバファイルが入っております。ドライバのインストール時にはこのディレクトリをご指定下さい。
- [APP]ディレクトリ
付属アプリケーションのインストーラです。
- [DOC]ディレクトリ
取扱説明書等が PDF 形式で入っております。
- [VB6]ディレクトリ
Visual Basic 6 のサンプルプログラムがソースファイルを含めたプロジェクトとして入っています。
- [VBNET]ディレクトリ
Visual Basic .NET のサンプルプログラムがソースファイルを含めたプロジェクトとして入っています。
- [VC]ディレクトリ
Visual C++のサンプルプログラムが Visual C++ 6.0 のソースファイルを含めたプロジェクトとして入っています。
- [TOOLS]ディレクトリ
開発時に必要な各種ファイルが入っています。

※ ソフトウェアディスクは付属しません。ソフトウェアは弊社 Web サイトよりダウンロードしてください。

5.2 アプリケーションソフトの実行について

プログラム取扱説明書を参照してください。

6. カウント値と周波数、周期の関係

本カウンタには周波数および周期の 2 つのカウントモードがあります。何れの結果も整数値として取得されます。ここでは取得された整数値から周期および周波数の算出の仕方について説明いたします。

※ 添付プログラム(サンプルではありません)では自動的に周波数および周期を算出しておりますのでそのまま直読できます。プログラミングされる場合には周波数および周期への変換が必要です。

6.1 周波数モード

周波数モードではゲートタイムで設定された時間毎に外部入力信号のエッジ発生件数の集計を行います。たとえばゲートタイムを 10ms とした時に外部より 1kHz の信号が入力していると 10 というカウント値が読み出されるはずですが、カウント値から周波数への変換式は以下の通りです。

$$\text{周波数 (Hz)} = \text{カウント値} / \text{ゲートタイム (S)}$$

先ほどの例をこの計算式に当てはめると

$$\text{周波数} = 10 / 0.01 = 1000 [\text{Hz}]$$

となり、入力信号が 1kHz である事が分かります。(有効数字に注意してください)

※ 周波数の逆数をとると周期となります。(一定周期の波形と仮定した場合)

※ 計数は自動的に行われ、ソフトウェアから読み出した値は常に最新のものとなります。

6.2 周期モード

周期モードでは外部入力信号のエッジから次のエッジの区間に内部カウントクロックのエッジ発生件数の集計を行います。たとえばカウントクロックの設定を 100ns とした時に外部より 1kHz の信号が入力していると 10000 というカウント値が読み出されるはずですが、カウント値から周期への変換式は以下の通りです。

$$\text{周期 (S)} = \text{カウント値} \times \text{カウントクロック周期 (S)}$$

先ほどの例をこの計算式に当てはめると

$$\text{周期} = 10000 \times 100 \text{ (nS)} = 1.0000 \text{ (mS)}$$

となり、入力信号の周期が 1 ミリ秒であることが分かります。

入力分周器は入力信号を分周する事によってより低い周波数の信号に変えてからエッジ検出します。この機能により高い周波数の信号の周期をより高分解能で計測する事ができます。たとえば 1MHz を 100nS で周期測定するとカウント値はたった 10 ですが、1/1000 分周器を選択するとカウント値は 10000 となり、より高分解能に測定可能となります。

※ 周期の逆数をとると周波数となります。(一定周期の波形と仮定した場合)

※ 計数は自動的に行われ、ソフトウェアから読み出した値は常に最新のものとなります。

6.3 低い周波数の信号測定をする方法

低い周波数の測定を周波数モードで計測しても十分な分解能が得られない場合があります。たとえば 1Hz の周波数を測定する場合、ゲート時間を 10s としてもカウント値は 10 程度です。この様な場合には周期モードで周期を測定し、その逆数をとって下さい。

ただし、入力信号が一定周期である場合に限りです。

6.4 短い周期の信号測定をする方法

短い周期の信号を周期モードで計測しても十分な分解能が得られない場合があります。たとえば 100nS 周期の波形を測定する場合、カウントクロックを 100nS としてもカウント値は 1 程度です。この様な場合には周波数モードで周波数を測定し、その逆数をとって下さい。

ただし、入力信号が一定周期である場合に限りです。

※ 分周器を使う方法もあります。

7. プログラミング

ここでは、Visual C++ 6.0 以上 または Visual Basic 6.0 以上で TUSB-S03CN2 応用アプリケーションを開発する方法を説明します。本装置用のドライバをインストールするとドライバ操作用の DLL(ダイナミックリンクライブラリ)ファイルが同時にインストールされます。応用アプリケーションではこの DLL を介してドライバを操作します。

DLL を直接ロードして操作する方法もありますが、ここではソフトウェアセットに付属の定義ファイル(Visual C++6.0、Visual Basic 6.0 および Visual Basic .NET 用のものが入っております)を利用した方法を説明します。

両開発ツール(Visual C++、Visual Basic)の使用方法についてはご説明いたしません。それぞれに付属のマニュアルかその他の資料をご参照ください。

7.1 Visual C++ 6.0 での使用

7.1.1 使用準備

Visual C++で使用するために以下 2 つのファイルをソフトウェアセットから適当な場所にコピーしてください。コピー先は指定しませんが、通常はプロジェクトのフォルダで支障ありません。

TUSBSCN2.H	ヘッダファイル
TUSBSCN2.LIB	ライブラリファイル

※ ソフトウェアセット内の DEV¥TOOLS フォルダに有ります。

ライブラリファイルはプロジェクトに追加してください。

ヘッダファイルは関数を使用するソースコードファイルの適当な場所にインクルードしてください。

7.1.2 関数の呼び出し方法

デバイスドライバの操作は全て機能毎の関数を呼ぶ事によって実現されます。Tusbs03cn2_Device_Open 以外の関数は Tusbs03cn2_Device_Open 関数が正常に処理された後でないと有効にはなりません。各機能関数を呼び出す前に Tusbs03cn2_Device_Open を実行して機能関数の使用が終了したら Tusbs03cn2_Device_Close 関数を呼び出してデバイスを開放してください。デバイスを一つのアプリケーションで実行する場合には通常アプリケーションの初めに Open し、アプリケーションの終了時に Close すれば充分です。

1 つの TUSB-S03CN2 デバイスを 2 つのアプリケーションで同時にオープンする事は出来ません。1 つのアプリケーションでの Open~Close の間は他のアプリケーションで同じデバイスを操作する事はできません。また、3ch を別々に独立のアプリケーションから操作する事は出来ません。

7.1.3 サンプルコード

次に示すのは、TUSB-S03CN2(ID=0:デフォルト設定)のドライバをオープンし、カウント値を取得、表示しドライバをクローズするサンプルコードです。

```
#include "stdafx.h"
#include "tusbscn2.h"      //付属のヘッダファイル

void main(void)
{
    long Data;
    int id;                //装置の id 番号
    char ch;              //チャンネル番号
    char Valid;          //データの有効無効判定

    id = 0;
    ch = 0;
    if( Tusbs03cn2_Device_Open(id) ) //デバイスオープン
    {
        printf("オープンできませんでした");
        return 0;
    }
    //カウント値を一つ取得する
    if(Tusbs03cn2_Counter_Read(id, ch, &Data, &Valid) )
    {
        printf("関数の実行に失敗しました");
    }
    else
    {
        if( Valid )
        {
            printf("CNT = %u¥n",Data); //カウント値の表示
        }
        else
        {
            printf("有効データがありませんでした");
        }
    }

    Tusbs03cn2_Device_Close(id); //デバイスクローズ
}
```

7.2 Visual Basic 6 での使用

7.2.1 使用準備

Visual Basic 6で使用するために以下1つのファイルをソフトウェアセットから適当な場所にコピーしてください。コピー先は限定しませんが、通常はプロジェクトのフォルダで支障ありません。

TUSBSCN2.BAS 標準ライブラリファイル

※ ソフトウェアセット内の DEVTOOLS フォルダに有ります。

ライブラリファイルは標準ライブラリの追加でプロジェクトに追加してください。

7.2.2 プロシージャの呼び出し方法

デバイスドライバの操作は全て機能毎のプロシージャを呼ぶ事によって実現されます。

Tusbs03cn2_Device_Open 以外のプロシージャは Tusbs0cn2_Device_Open が正常に処理された後でないとは有効にはなりません。各機能のプロシージャを呼び出す前に Tusbs03cn2_Device_Open を実行してプロシージャの使用が終了したら Tusbs03cn2_Device_Close を呼び出してデバイスを開放してください。デバイスを一つのアプリケーションで実行する場合には通常アプリケーションの初めに Open し、アプリケーションの終了時に Close すれば充分です。

一つの TUSB-S03CN2 デバイスを 2 つのアプリケーションで同時にオープンする事は出来ません。一つのアプリケーションでの Open~Close の間は他のアプリケーションで同じデバイス进行操作する事はできません。

7.2.3 サンプルコード

次に示すのは、TUSB-S03CN2(ID=0:デフォルト設定)のドライバをオープンし、カウント値を取得、表示しドライバをクローズするサンプルコードです。

```
Private Sub GetData_Click()  
    Dim Data As Long  
    Dim Message As String  
    Dim Ch As Byte  
    Dim Valid As Byte  
  
    Id = 0 ' デバイスの ID  
    Ch = 0 ' 読み込みチャンネル  
  
    If Tusbs03cn2_Device_Open(Id) <> 0 Then 'デバイスオープン  
        MsgBox("デバイスのオープンに失敗しました");  
        Exit Sub  
    End If  
    'カウント値を一つ取得する  
    If Tusbs03cn2_Counter_Read (Id, Ch, Data, Valid) <> 0 Then  
        MsgBox("関数の実行に失敗しました");  
    Else  
        If Valid Then  
            Message = Format(Data,"S = 0000000000")  
            MsgBox(Message)  
        Else  
            MsgBox("有効なデータがありません")  
        End If  
    End If  
End Sub
```


7.3 Visual Basic .NET での使用

7.3.1 使用準備

Visual Basic .NET で使用するために以下1つのファイルをソフトウェアセットから適当な場所にコピーしてください。コピー先は指定しませんが、通常はプロジェクトのフォルダで支障ありません。

TUSBSCN2.vb 標準ライブラリファイル

※ ソフトウェアセット内の DEVTOOLS フォルダに有ります。
ライブラリファイルは既存項目の追加でプロジェクトに追加してください。

7.3.2 プロシージャの呼び出し方法

デバイスドライバの操作は全て機能毎のプロシージャを呼ぶ事によって実現されます。Tusbs03cn2_Device_Open 以外のプロシージャは Tusbs03cn2_Device_Open が正常に処理された後でないと有効にはなりません。各機能のプロシージャを呼び出す前に Tusbs03cn2_Device_Open を実行してプロシージャの使用が終了したら Tusbs03cn2_Device_Close を呼び出してデバイスを開放してください。デバイスを一つのアプリケーションで実行する場合には通常アプリケーションの初めに Open し、アプリケーションの終了時に Close すれば充分です。

一つの TUSB-S03CN2 デバイスを2つのアプリケーションで同時にオープンする事は出来ません。一つのアプリケーションでの Open~Close の間は他のアプリケーションで同じデバイスを操作する事はできません。

7.3.3 サンプルコード

次に示すのは、TUSB-S03CN2(ID=0:デフォルト設定)のドライバをオープンし、カウント値を取得、表示しドライバをクローズするサンプルコードです。

```
Private Sub GetData_Click()  
    Dim Data As Integer  
    Dim Id As Short  
    Dim Ch As Byte  
    Dim Valid As Byte  
  
    Id = 0 ' デバイスの ID  
    Ch = 0 ' 読み込むチャンネル  
  
    If Tusbs03cn2_Device_Open(Id) <> 0 Then ' デバイスオープン  
        MsgBox("デバイスのオープンに失敗しました");  
        Exit Sub  
    End If  
    ' ひずみ量を一つ取得する  
    If Tusbs03cn2_Counter_Read(Id, Ch, Data, Valid) <> 0 Then  
        MsgBox("関数の実行に失敗しました");  
    Else  
        If Valid Then  
            MsgBox(CStr(Data))  
        Else  
            MsgBox("有効なデータがありません")  
        End If  
    End If  
End Sub
```

7.4 関数説明

ここでは、各関数(プロシージャ)のもつ機能などの詳細を説明します。

Tusbs03cn2_Device_Open

C,C++宣言	short __stdcall Tusbs03cn2_Device_Open(short id)
VB6 宣言	Declare Function Tusbs03cn2_Device_Open Lib "TUSBSCN2.DLL" (ByVal id As Integer) As Integer
VB.NET 宣言	Declare Function Tusbs03cn2_Device_Open Lib "TUSBSCN2.DLL" (ByVal id As Short) As Short

解説

指定 ID(ユニット番号選択スイッチの値)のデバイスをオープンします。
このデバイスに関する各種関数を使用する前に必ず呼び出す必要が有ります。

引数

id	ユニット番号選択スイッチの番号(0-3)
----	----------------------

戻り値

エラーコード(エラーコード表参照)

Tusbs03cn2_Device_Close

C,C++宣言	void __stdcall Tusbs03cn2_Device_Close(short id)
VB6 宣言	Declare Sub Tusbs03cn2_Device_Close Lib "TUSBSCN2.DLL" (ByVal id As Integer)
VB.NET 宣言	Declare Sub Tusbs03cn2_Device_Close Lib "TUSBSCN2.DLL" (ByVal id As Short)

解説

指定 ID(ユニット番号選択スイッチの値)のデバイスをクローズします。

引数

id	ユニット番号選択スイッチの番号(0-3)
----	----------------------

戻り値

なし

Tusbs03cn2_Counter_Read

C,C++宣言	short Tusbs03cn2_Counter_Read (short id ,char ch,int *Data, char *Valid)
VB6 宣言	Declare Function Tusbs03cn2_Counter_Read Lib "TUSBSCN2.DLL" (ByVal id As Integer, ByVal ch As Byte, ByRef Data As Long, ByRef Valid As Byte) As Integer
VB.NET 宣言	Declare Function Tusbs03cn2_Counter_Read Lib "TUSBSCN2.DLL" (ByVal Id As Short, ByVal ch As Byte, ByRef Data As Integer, ByRef Valid As Byte) As Short

解説

指定ID(ユニット番号選択スイッチの値)のデバイスから現在のカウンタの値を取得します。
有効なデータがある時は Valid に 1、有効なデータが無い時には 0 が入ります。
※ 有効なデータが無い時とは、ゲート時間 1 秒と設定した後 1 秒未満の時など。

引数

id	ユニット番号選択スイッチの番号(0-3)
ch	カウント値を読み込むチャンネル指定(0-2)
Data	カウント値格納用バッファ
Valid	1 の時データ有効、0 の时无効

戻り値

エラーコード(エラーコード表参照)

Tusbs03cn2_Ext_Input_Set

C,C++宣言	short Tusbs03cn2_Ext_Input_Set (short id ,char ch,char UpDown)
VB6 宣言	Declare Function Tusbs03cn2_Ext_Input_Set Lib "TUSBSCN2.DLL" (ByVal id As Integer, ByVal ch As Byte, ByVal UpDown As Byte) As Integer
VB.NET 宣言	Declare Function Tusbs03cn2_Ext_Input_Set Lib "TUSBSCN2.DLL" (ByVal id As Short, ByVal ch As Byte, ByVal UpDown As Byte) As Short

解説

外部入力の有効なエッジ方向を設定します。

引数

id	ユニット番号選択スイッチの番号(0-3)
ch	設定をするチャンネル(0-2)
UpDown	0:立上りエッジ有効 1:立下りエッジ有効

戻り値

エラーコード(エラーコード表参照)

Tusbs03cn2_Counter_Mode

C,C++宣言	short Tusbs03cn2_Counter_Mode(short id ,char ch,char mode)
VB6 宣言	Declare Function Tusbs03cn2_Counter_Mode Lib "TUSBSCN2.DLL" (ByVal id As Integer, ByVal ch As Byte, ByVal mode As Byte) As Integer
VB.NET 宣言	Declare Function Tusbs03cn2_Counter_Mode Lib "TUSBSCN2.DLL" (ByVal id As Short, ByVal ch As Byte, ByVal mode As Byte) As Short

解説

カウンタの動作モードを周波数又は周期の何れかから選択します。

引数

id	ユニット番号選択スイッチの番号(0-3)
ch	設定をするチャンネル(0-2)
mode	0:周期モード 1:周波数モード

戻り値

エラーコード(エラーコード表参照)

Tusbs03cn2_Gate_Time

C,C++宣言	short Tusbs03cn2_Gate_Time(short id ,char ch,char time)
VB6 宣言	Declare Function Tusbs03cn2_Gate_Time Lib "TUSBSCN2.DLL" (ByVal id As Integer, ByVal ch As Byte, ByVal time As Byte) As Integer
VB.NET 宣言	Declare Function Tusbs03cn2_Gate_Time Lib "TUSBSCN2.DLL" (ByVal id As Short, ByVal ch As Byte, ByVal time As Byte) As Short

解説

周波数モード時のゲート時間を設定します。

引数

id	ユニット番号選択スイッチの番号(0-3)
ch	設定をするチャンネル(0-2)
time	0:10mS 1:100mS 2:1S 3:10S

戻り値

エラーコード(エラーコード表参照)

Tusbs03cn2_Cnt_Clock_Set

C,C++宣言	short Tusbs03cn2_Cnt_Clock_Set(short id ,char ch,char clk)
VB6 宣言	Declare Function Tusbs03cn2_Cnt_Clock_Set Lib "TUSBSCN2.DLL" (ByVal id As Integer, ByVal ch As Byte, ByVal clk As Byte) As Integer
VB.NET 宣言	Declare Function Tusbs03cn2_Cnt_Clock_Set Lib "TUSBSCN2.DLL" (ByVal id As Short, ByVal ch As Byte, ByVal clk As Byte) As Short

解説

周期モード時のカウントクロックを設定します。

引数

id	ユニット番号選択スイッチの番号(0-3)
ch	設定をするチャンネル(0-2)
clk	0:100nS 1:1 μ S 2:10 μ S 3:100 μ S

戻り値

エラーコード(エラーコード表参照)

Tusbs03cn2_Divider_Set

C,C++宣言	short Tusbs03cn2_Divider_Set(short id ,char ch,char div)
VB6 宣言	Declare Function Tusbs03cn2_Cnt_Clock_Set Lib "TUSBSCN2.DLL" (ByVal id As Integer, ByVal ch As Byte, ByVal div As Byte) As Integer
VB.NET 宣言	Declare Function Tusbs03cn2_Cnt_Clock_Set Lib "TUSBSCN2.DLL" (ByVal id As Short, ByVal ch As Byte, ByVal div As Byte) As Short

解説

周期モード時の分周器を設定します。

引数

id	ユニット番号選択スイッチの番号(0-3)
ch	設定をするチャンネル(0-2)
div	0:1/1 1:1/10 2:1/100 3:1/1000

戻り値

エラーコード(エラーコード表参照)

7.5 エラーコード表

各機能関数(プロシージャ)から戻る処理結果コードの値の表です。

戻り値	状態
0	正常終了
1	ID 番号が異なる
2	ドライバが正常にインストールされていない
3	このデバイスは既にオープンされている
4	接続台数が多すぎる
5	デバイスをオープンできなかった
6	指定のデバイスが見つからない
7	指定のデバイスはオープンされてない
8	指定パラメータのエラー
9	USB 通信エラー

8. その他

8.1 USB について

USB とは Universal Serial Bus の頭文字の略で、新しいコンピュータのインターフェースバスです。インターフェースのコストが低く使い易い事などからパーソナルコンピュータを中心に普及しました。USB1.1 の仕様では、1.5Mbps ロースピードデバイスおよび 12Mbps ハイスピードデバイスがあります。

※ ここでの記述は USB の一般的な記述となっております。

USB の主な特長	
高速	12Mbps のバススピード(USB 2.0 では 480Mbps)
接続が容易	ISA や PCI などの拡張バスと違いケーブル 1 本で接続可能。コンピュータの動作中でも抜き差し可能。
多数接続可能	ハブの利用により最高 127 台(ハブを含む)のデバイスを接続可能。
バス電源供給可能	標準で 100mA、最大で 500mA の電源をバスで供給可能。
低コスト	多くのパーソナルコンピュータに標準で装備されており、安価なケーブル 1 本で接続可能。ただし、標準装備のポート数より多くのデバイスを接続するにはハブが必要。

ハブについて

多数の USB を接続するにはハブデバイスが必要です。ハブは 1 本の USB 線(上流側)を複数の USB 線(下流側)に分岐します。ハブにはバスパワーハブとセルフパワーハブがあり、前者は上流側の電源により動作しますが、後者は外部電源により動作します。ホストのポートからは標準で 100mA、最大 500mA の電流を供給する事が出来ます。バスパワーハブでは通常 100mA 未満の電流を消費するため、このハブに接続されたデバイスはバスから 500mA を供給される事は出来ません。100mA 以上の電流を消費するデバイスをバスパワーハブに接続する場合には注意が必要です。

ケーブルについて

USB ケーブルは A タイプと B タイプに分かれます。ホストのポートは A タイプ、デバイス側は B タイプとなっており、誤挿入が起こらない仕様となっております。

転送速度について

USB の転送速度はきわめて高速ですが、接続されたデバイスの単位時間当たりのデータ転送量総合計が最高転送量を超える事はありません。あるデバイスで大量のデータ転送を行うと他のデバイスの転送速度に影響の出る可能性があります。

8.2 連絡先

動作上の問題点および不明な点などのお問い合わせは下記までお願いします。
調査の上、当社よりご連絡差し上げます。

ご質問の際には動作環境等、なるべく詳細な情報を下さい。
特に次の情報は必ず記載してください。

ご使用のコンピュータの機種
ご使用 OS(Windows98 , Windows98 SE など)
メモリ容量
ハードディスクの容量
本ユニット以外でご使用されている USB 装置
こちらからご連絡差し上げる場合の貴ご連絡先

株式会社タートル工業

～ 技術部 技術課 サービス係 ～

E-mail	info@turtle-ind.co.jp
FAX	0298-43-2024
郵送	〒300-0842 茨城県土浦市西根南 1-12-4

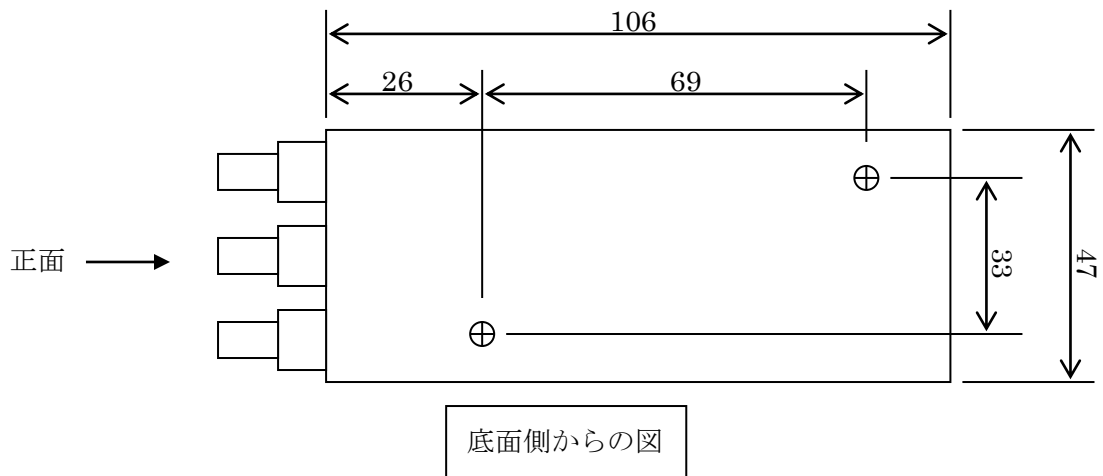
9. 仕様

9.1 仕様概要

入力	3点独立(BNC) TTLエッジ(立上り、立下りを選択)
カウントモード	周波数カウント(内部ゲートによるコントロール) 内部ゲート時間 10mS 100mS 1S 10S 周期カウント(内部クロックのカウント) 内部クロック周波数 10MHz 1MHz 100kHz 10kHz
入力分周器	1/1 1/10 1/100 1/1000
最大カウント数	20ビット(1048575)
入力周波数範囲	0.01Hz~50MHz(Hi,Lo 区間共に 10nS 以上)
最小時間分解能	100pS(周期カウント、100nS クロック、1/1000 分周時)
周波数精度	±25ppm 以内(5~45℃において)
インタフェース	USB1.1
接続台数	4台まで、(内部スイッチでID選択、HUBが必要)
使用温度範囲	5℃~45℃
消費電流	約 60mA(USBより供給)
大きさ	47(W)×21.5(H)×126.5(D)mm (BNCコネクタ以外突起部含まず)
重量	約 160g(ケーブル含まず)

9.2 取り付け穴寸法図

取り付け穴図です。底面から見た図(BOTTOM VIEW)です。取り付け穴は M3(ミリネジ)のタップです。



TUSB-S03CN2(B)(Z)取扱説明書

発行年月 2019年4月 第6a版

発 行 株式会社 タートル工業

編 集 株式会社 タートル工業

©2019 株式会社 タートル工業