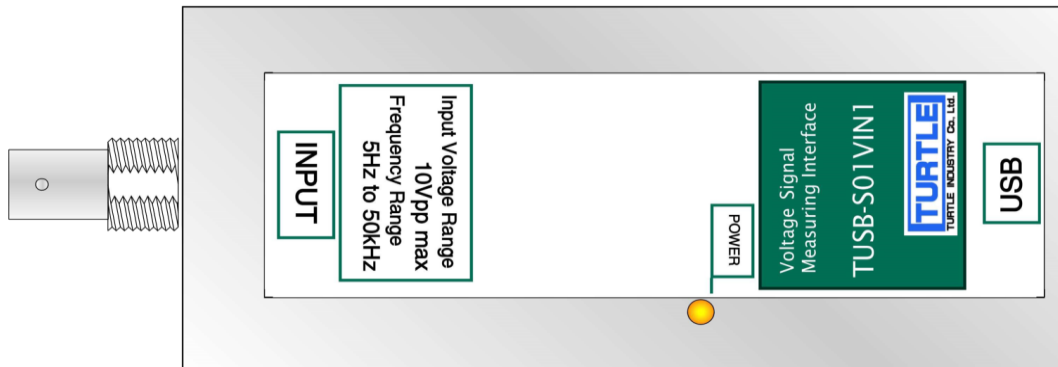


TUSB-S01VIN1

USB インタフェース付き電圧信号取り込みユニット



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。

©2017 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。

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 センサ入力.....	7
3.2 USB コネクタ.....	7
3.3 ID 選択スイッチ.....	7
5 付属のディスクについて.....	8
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.....	8
5.2 付属ソフトの実行について.....	9
6. 加速度サンプリングの方法.....	10
6.1 入力回路設定.....	10
6.2 サンプリング周波数設定.....	10
6.3 トリガについて.....	10
6.4 トリガ誤動作防止機能.....	11
7.プログラミング.....	13
7.1 Visual C++(C++/CLD) 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
7.2 Visual Basic 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.....	14
7.3 Visual C#での使用.....	14
7.3.1 使用準備.....	14
7.3.2 プロシージャの呼び出し方法.....	15
7.4 関数説明.....	16
7.5 エラーコード表.....	28
8. その他.....	29
8.1 USB について.....	エラー! ブックマークが定義されていません。
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電圧信号用データ収集ユニット TUSB-S01VIN1 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各種機器からの電圧波形データを取得し、記録する事ができます。ドライバソフトウェア、Visual C++ と Visual Basic のサンプルソフトウェアの他アプリケーションソフトウェアが付属しておりますので、これらの応用によって短時間に利用する事が可能です。

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

1.2 製品構成

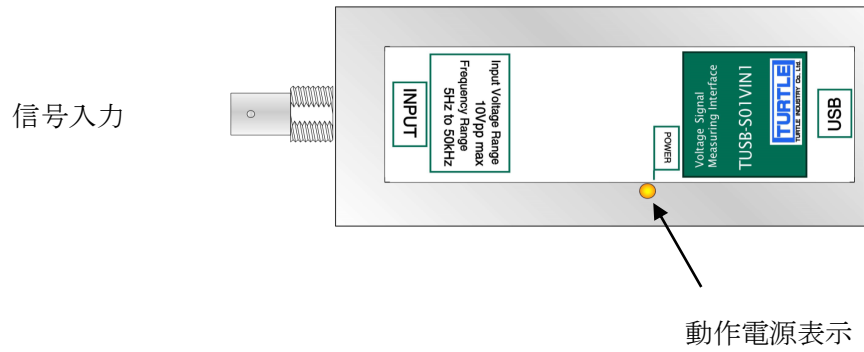
本製品には以下の物が含まれます。

- ① TUSB-S01VIN1 本体
- ② USB ケーブル(1m)
- ③ ゴム足(4 個)

不足品などがあれば、当社サービス課までご連絡下さい。

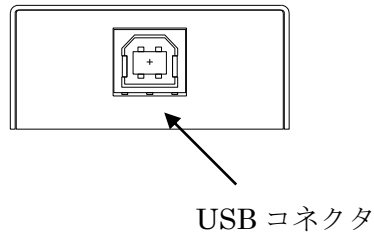
2. 各部の名称

2.1 上面



信号入力	計測する電圧信号を入力します。(BNC)
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
----------	-------------------------

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

3. 各部説明

3.1 信号入力

計測する電圧信号を入力します。直流信号が乗っている信号も入力是可以しますが、直流分は遮断されます。信号は下記の条件でご使用下さい。

[絶対定格入力電圧範囲]

信号のピーク電圧 -50V～+50V 以内

交流振幅ピーク電圧※ -6.8V～+12.8V 以内

[有効入力電圧範囲] (変換可能な電圧範囲)

交流振幅ピーク電圧※ -5.0V～+5.0V 以内

※ ここで交流振幅ピーク電圧とは信号の平均値を基準とした時の信号のピーク電圧です。

3.2 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。付属のインストールマニュアルを参照してください。

3.3 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

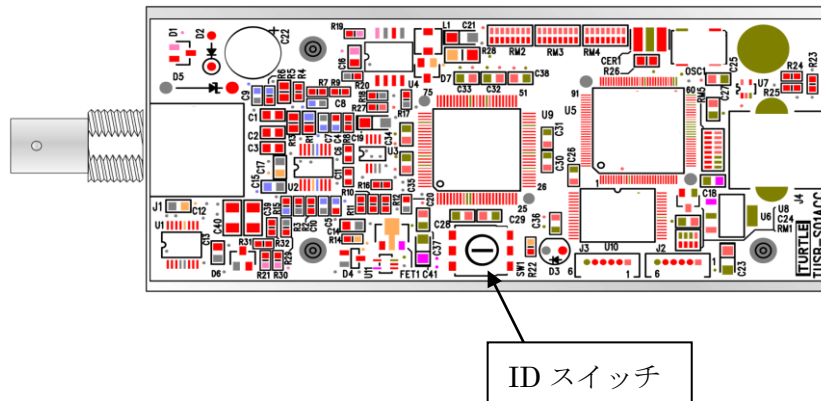
① 側面のネジを左右合わせて 2 個取り外します。



注意

ID 変更時は金属加工部で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。



5 付属のディスクについて

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

|-[TUSBSV11]

|-[DRIVER] : ドライバ

|-[DOC] : ドキュメント(取扱説明書等)

|-[DEV] :

|-[TOOLS] :開発用 API 定義ファイル等

|-[VB] :Visual Basic .NET 用 サンプルプロジェクト

|-[VCppCLI] :Visual C++(C++/CLI) 用 サンプルプロジェクト

|-[VCSharp] :Visual C#用 サンプルプロジェクト

○ [DRIVER]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。

○ [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。

○ [VB]ディレクトリ

Visual Basic 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。 ※

○ [VCppCLI]ディレクトリ

Visual C++(C++/CLI)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。

○ [VCSharp]ディレクトリ

Visual C#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。※

○ [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。

5.2 付属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. 信号サンプリングの方法

本ユニットは電圧信号を最速100kSPSのサンプリングレートで連続的に取得する事が可能です。

サンプリングされたデータは一旦本ユニット内のメモリに保存され、次にドライバソフトウェアによって順次先頭から読み出されPC上のメモリに蓄積されます。アプリケーションプログラムはそのドライバ上のバッファよりデータを読み出します。

6.1 入力回路設定

入力フィルタや増幅率の設定はサンプリング開始前に行います。入力部にはローパスフィルタとハイパスフィルタがあり、ローパスフィルタは50kHz、ハイパスフィルタは5Hzです。増幅率は×1、×10、×100から選択出来ます。これらの選択はTusbs01vin1_Input_Set関数で行います。

6.2 サンプリング周波数設定

サンプリング開始前に内部サンプリングクロックの設定を行います。この値は48MHzの割り算の除数を設定する事によって決まります。たとえば除数として480を設定すると

$$48\text{MHz} \div 480 = 100\text{kHz} \quad (100\text{kSPS})$$

となります。

除数の設定は次の式のMとNを指定します

$$\text{除数} = M \times 10^N \quad M:1 \sim 65536 \quad N:0 \sim 4$$

これらの指定はTusbs01vin1_Timer_Set関数で行います。

※ お使いのシステムによっては100kSPSでは取り込みが間に合わない場合もあります(特に複数台使用時)。実際の測定前にFifoPeak関数などを使用して十分に検証してください。

6.3 トリガについて

Tusbs01vin1_Sampling_Start関数を実行するとサンプリングを開始しますが、サンプリングデータは一時的なメモリに保管されメモリ容量に応じて古い方から順次消去されます。有効なデータの開始点を決めるのがトリガです。トリガ条件が発生するとその時点からのデータが有効であるとし保管します(プレトリガ機能使用の場合はその時点より前になります)。トリガ条件としては次の6種類から選択できます。

[ソフトウェアトリガ]

PC側からのトリガコマンドTusbs01vin1_Sampling_Trigger関数によります

[プラス側レベル]

サンプリング値が指定レベルより大きい場合

[マイナス側レベル]

サンプリング値が指定レベルより小さい場合

[絶対値レベル]

サンプリング値の絶対値が指定レベルより大きい場合

[立上り]

サンプリング値が指定レベル以下から指定レベルより大きい値に変化した場合

[立下り]

サンプリング値が指定レベル以上から指定レベルより小さい値に変化した場合

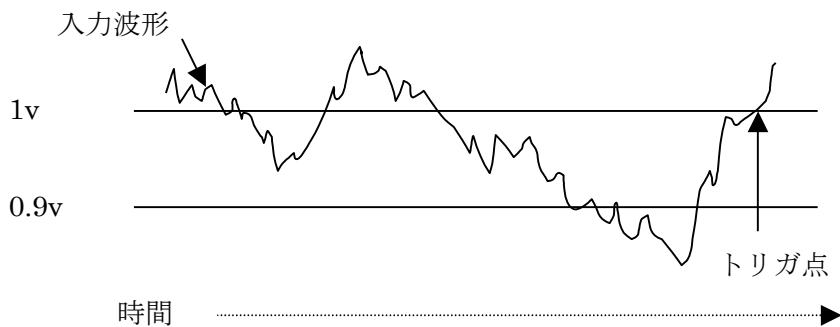
これらの指定は `Tusbs01vin1_Sampling_Start` 関数により行います。

6.4 トリガ誤動作防止機能

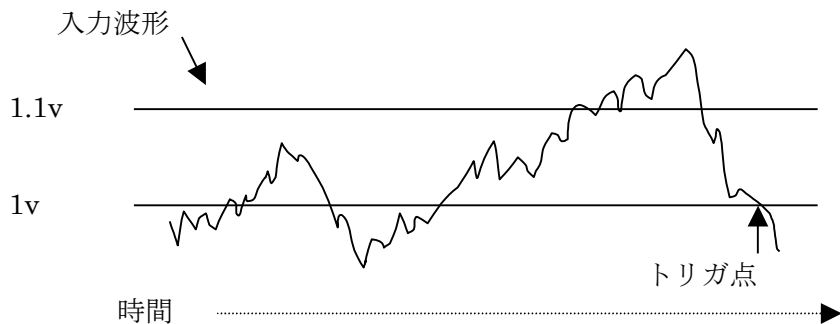
立上り、立下りトリガにはノイズによる誤動作防止帯が設けられています。×1レンジの時、立上りトリガは[指定レベル - 0.1v]未満の値を観測しないとトリガ検知を開始しません。立下りトリガは[指定レベル + 0.1v]より大きい値を観測しないとトリガ検知を開始しません。

※ ×10レンジの時誤動作防止電圧は0.01v ×100の時は0.001vとなります。

< ×1レンジで 1v に立ち上がりトリガ設定した時の例 >



< ×1レンジで 1v に立ち下がりトリガ設定した時の例 >



6.5 プレトリガ機能について

Tusbs01vin1_Sampling_Start 関数においてプレトリガ数を設定するとトリガ条件発生時より前のデータも取得可能です。ただし、Tusbs01vin1_Sampling_Start 関数前のデータは取得できません。

プレトリガ数を 0 以外の値に設定した場合、Tusbs01vin1_Sampling_Start 関数実行後プレトリガ数分のサンプリングが終了した時点でトリガ判定が有効となります。

6.6 データ取得の手順

Tusbs01vin1_Sample_Start によってサンプリング動作開始して下さい。取得完了したデータの個数は Tusbs01vin1_Status_Read によって確認できますので、必要に応じて Tusbs01vin1_Memory_Read 関数でデータを読み出して下さい。連続的にデータを取得する場合は Tusbs01vin1_Sample_Start で設定したバッファ個数に達する前に読み出してください。バッファ容量の空きが無くなるとサンプリングは停止します。

サンプリングを終了する場合は Tusbs01vin1_Sampling_Stop を実行してください。

6.7 取得データについて

取得されるデータは全て符号付整数値です。×1 レンジにおいて取得整数値を電圧に変換するには 6400 当たり 1V として変換してください。×10 レンジでは 6400 当たり 0.1V、×100 レンジでは 0.01V となっております。

変換表

	×1 レンジ			×10 レンジ			×100 レンジ		
整数値	-32000	0	32000	-32000	0	32000	-32000	0	32000
電圧	-5V	0V	5V	-0.5V	0	0.5V	-0.05V	0V	0.05V

7. プログラミング

ここでは、Visual C++、Visual Basic、Visual C# で TUSB-S01VIN1 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、ここでは添付ディスクに付属の定義ファイル(Visual C++、Visual Basic 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。

開発ツール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。

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

7.1 Visual C++(C++/CLI) での使用

7.1.1 使用準備

Visual C++で使用するために以下 2 つのファイルを付属ディスク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。

TUSBSVI1.H ヘッダファイル

※ 付属ディスク内の DEV\TOOLS フォルダに有ります。

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。

※ ネイティブコードで使用する場合は TOOLS フォルダ下の Native フォルダ内の TUSBSVI1 をインクルードし、TUSBSVI1.lib をプロジェクトに追加してください。

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。Tusbs01vin1_Device_Open 以外の関数は Tusbs01vin1_Device_Open 関数が正常に処理された後でないとはなりません。各機能関数を呼び出す前に Tusbs01vin1_Device_Open を実行して機能関数の使用が終了したら Tusbs01vin1_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-S01VIN1 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。

7.2 Visual Basic での使用

7.2.1 使用準備

Visual Basic で使用するために以下 1つのファイルを付属ディスク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。

TUSBSVI1.vb 標準ライブラリファイル

※ 付属ディスク内の **DEV\TOOLS** 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。

7.2.2 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。Tusbs01vin1_Device_Open 以外のプロシージャは Tusbs01vin1_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 Tusbs01vin1_Device_Open を実行してプロシージャの使用が終了したら Tusbs01vin1_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-S01VIN1 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 Visual C#での使用

7.3.1 使用準備

Visual C#で使用するために以下 1つのファイルを付属ディスク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。

TUSBSVI1.cs 標準ライブラリファイル

※ 付属ディスク内の **DEV\TOOLS** 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。

7.2.2 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関数を呼ぶ事によって実現されます。**Tusbs01vin1_Device_Open** 以外の関数は **Tusbs01vin1_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。各機能の関数を呼び出す前に **Tusbs01vin1_Device_Open** を実行して関数の使用が終了したら **Tusbs01vin1_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-S01VIN1** 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。

7.4 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。

Tusbs01vin1_Device_Open

C 宣言	short __ cdecl Tusbs01vin1_Device_Open(short id)
	各言語での宣言については付属の定義ファイルを参照してください。

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-F)
----	----------------------

戻り値

エラーコード(エラーコード表参照)

Tusbs01vin1_Device_Close

C 宣言	void Tusbs01vin1_Device_Close(short id)
	各言語での宣言については付属の定義ファイルを参照してください。

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-F)
----	----------------------

戻り値

なし

Tusbs01vin1_Timer_Set

C 宣言	short Tusbs01vin1_Timer_Set (short id ,char N,int M)
	各言語での宣言については付属の定義ファイルを参照してください。

解説

サンプリング動作のタイマーを設定します。

周波数 = $48 \times 10^6 \div (M \times 10^N)$ N : 0~4 M : 1~65536

必ず 100kHz 以下となる様に設定します。

引数

id	ユニット番号選択スイッチの番号(0-F)
N	タイマー設定値 N 0~4
M	タイマー設定値 M 1~65536

戻り値

エラーコード(エラーコード表参照)+

Tusbs01vin1_Input_Set

C 宣言	short Tusbs01vin1_Input_Set (short id ,char Range)
	各言語での宣言については付属の定義ファイルを参照してください。

解説

入力信号増幅率の設定をします。

引数

id	ユニット番号選択スイッチの番号(0-F)
Range	増幅率の設定 0:×1 1:×10 2:×100

戻り値

エラーコード(エラーコード表参照)

Tusbs01vin1_Sample_Start

C 宣言	short Tusbs01vin1_Sample_Start(short id ,char TrType,short Level, int PreLen,int TotalLen)
	各言語での宣言については付属の定義ファイルを参照してください。

解説

データの取得を開始します。

引数

id	ユニット番号選択スイッチの番号(0-F)
TrType	トリガタイプの選択 0:ソフト 1:+側レベル 2:-側レベル 3:絶対値レベル 4:立上り 5:立下り
Level	トリガタイプ 2~6 の時の基準レベル。 取得されるデータと同じ整数値で設定して下さい。
PreLen	プレトリガ長を設定します。0~10,000,000 プレトリガ機能を使用しない場合は0 と設定してください。 この値より TotalLen は十分に大きくしてください。TotalLen が十分でない とトリガ発生時に直ぐにバッファが埋まり測定が強制終了します。
TotalLen	ドライバ内に確保するバッファ長を設定します。0~10,000,000 アプリケーションで取りこぼしが起こらない程度の十分なメモリを確保し ます。通常は数秒分も確保すれば十分です。たとえば 100kHz で取得する時 は 100,000 で 1 秒間です。

戻り値

エラーコード(エラーコード表参照)

Tusbs01vin1_Status_Read

C 宣言	short Tusbs01vin1_Status_Read (short id ,char *status, char *ovf, int *len)
	各言語での宣言については付属の定義ファイルを参照してください。

解説

現在の動作状態の確認を行います。

引数

id	ユニット番号選択スイッチの番号(0-F)
status	現在の取り込み状況 0:停止中 1:トリガ待ち 2:測定中
ovf	オーバーフロー発生状況 0:正常 1:オーバーフロー発生
len	現在バッファにあるデータの数

戻り値

エラーコード(エラーコード表参照)

Tusbs01vin1_Memory_Read

C 宣言	short Tusbs01vin1_Memory_Read (short id ,short *data,int *leng)
	各言語での宣言については付属の定義ファイルを参照してください。

解説

取得済みのデータを読み込みます。

引数

id	ユニット番号選択スイッチの番号(0-F)
data	データ格納用バッファへのポインタ
leng	取り込み数。バッファへの取り込み要求数を指定します。この関数(プロシージャ)から戻ると実際に読み込まれたデータ数が入ります。

戻り値

エラーコード(エラーコード表参照)

Tusbs01vin1_Sampling_Stop

C 宣言	short Tusbs01vin1_Sampling_Stop(short id)
	各言語での宣言については付属の定義ファイルを参照してください。

解説

連続取り込み動作を停止します。

引数

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

戻り値

エラーコード(エラーコード表参照)

Tusbs01vin1_Memory_Clear

C 宣言	short Tusbs01vin1_Memory_Clear (short id)
	各言語での宣言については付属の定義ファイルを参照してください。

解説

連続取り込み時に確保されたメモリを解放します。

引数

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

戻り値

エラーコード(エラーコード表参照)

Tusbs01vin1_Sampling_Trigger

C 宣言	short Tusbs01vin1_Sampling_Trigger (short id)
	各言語での宣言については付属の定義ファイルを参照してください。

解説

ソフトウェアトリガ使用時にソフトウェアトリガを発生させます。

引数

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

戻り値

エラーコード(エラーコード表参照)

Tusbs01vin1_FifoPeak

C 宣言	short Tusbs01vin1_FifoPeak (short id , char *Peak)
	各言語での宣言については付属の定義ファイルを参照してください。

解説

ユニット内のメモリ使用率ピークを取得します。

目安としてご使用下さい。ここで表示される値が 100%に到達しなくてもメモリ使用率が高い場合にはオーバーフローする場合があります。メモリオーバーフローの確認は Tusbs01vin1_Status_Read をご使用下さい。

引数

id	ユニット番号選択スイッチの番号(0-F)
Peak	使用率のピーク(0~100)%

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
10	メモリ作業領域が確保できません
11	連続取り込み動作中です
12	連続取り込みデータはありません
99	その他のエラー

8. その他

8.1 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(Windows 7 Home...など)
OS の Edition(Home Professional など)
OS のサービスパック 32bit/64bit の何れか
メモリ容量
ご使用のドライバのバージョン
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 ～

Web サイト	当社 Web サイト www.turtle-ind.co.jp からお問い合わせフォーム
E-mail	info@turtle-ind.co.jp
FAX	029-843-2024
郵送	〒300-0842 茨城県土浦市西根南 1-12-4

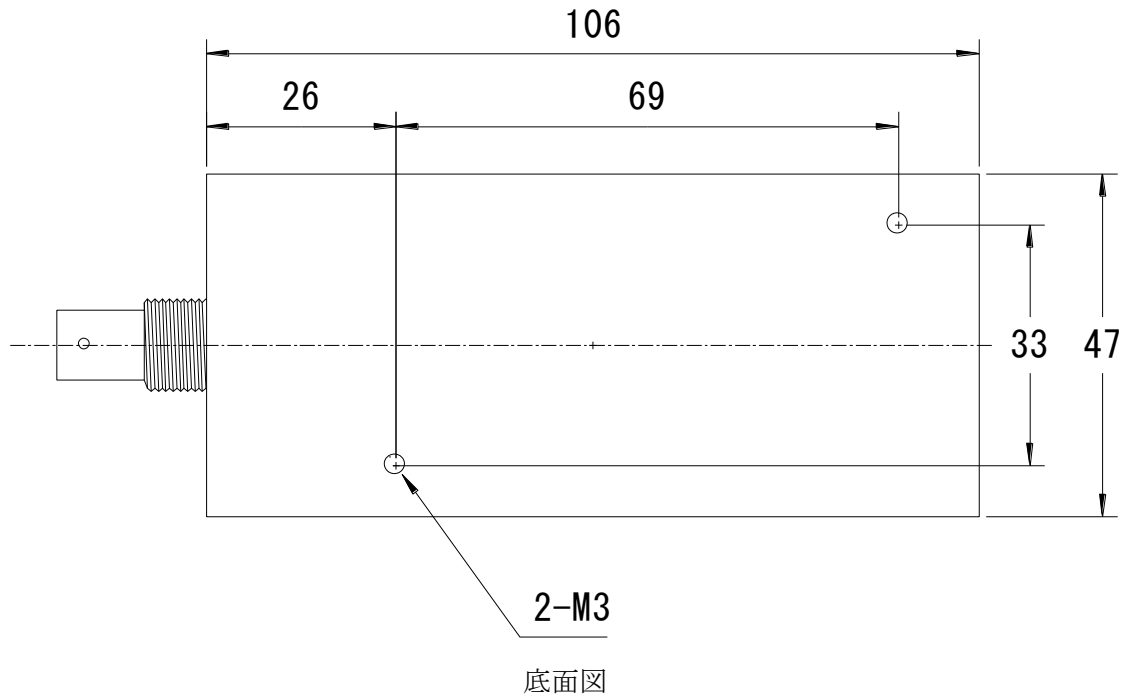
9. 仕様

9.1 仕様概要

入 力 信 号	電圧信号
入 力 数	1 チャンネル (BNC)
入 力 抵 抗	約 730k Ω
サンプリング周波数	最大 100kSPS (最小 0.1SPS)
ア ナ ロ グ 帯 域	5Hz~約 50kHz
ローパスフィルタ	50kHz 固定(24dB/oct 4 次バターワース)
ハイパスフィルタ	5Hz(1 次 CR フィルタ)
ダイナミックレンジ	70dB
入 力 電 圧 範 囲	$\pm 5Vp$
増 幅 率	1 倍 10 倍 100 倍 選択式
トリガ方式	1]信号レベル:プラス/マイナス/バイポーラ/立上り/立下り 2]コンピュータコマンド
メモリ容量	32k サンプル FIFO 方式による連続転送
測定誤差	0.5% of FS
使用温度範囲	5 $^{\circ}C$ ~45 $^{\circ}C$
消費電流	約 100mA(USB より供給)
大 き さ	47(W) \times 21.5(H) \times 126.5(D)mm (BNC コネクタ以外突起部含まず)
重 量	約 140g(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-S01VIN1 取扱説明書

発行年月 2018 年 12 月 第 2 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

©2018 株式会社 タートル工業