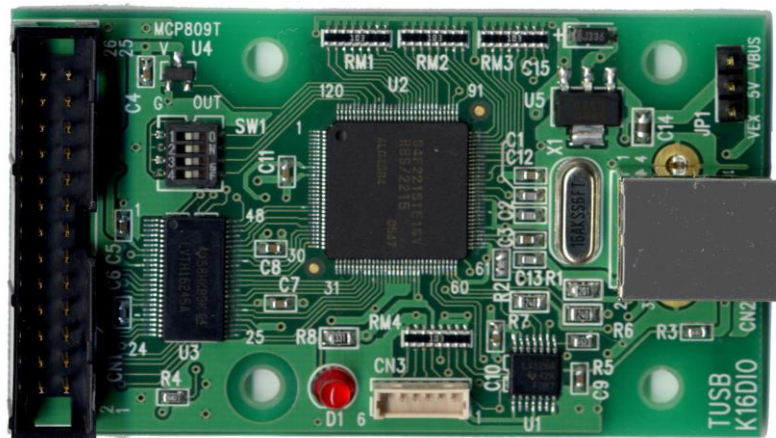


# TUSB-K16DIOZ

USB I/F 付き組込型デジタルIOボード

## 取扱説明書



(64bitOS 対応ドライバ版)



## 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。

 <b>警告</b>	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。
 <b>注意</b>	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付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
3. 各部説明.....	7
3.1 デジタル入出力(使用コネクタおよび電氣的仕様).....	7
3.1.1 使用コネクタ.....	7
3.1.2 ピン配置.....	7
3.1.3 デジタル出力について.....	8
3.1.4 デジタル入力について.....	8
3.1.5 電源出力について.....	8
3.1.6 電源入力について.....	8
3.2 電源選択ピン.....	9
3.3 USB コネクタ.....	9
3.4 電源ランプ.....	9
3.5 ID 設定.....	9
4. 機能解説.....	10
4.1 入出力設定.....	10
4.2 データ出力.....	10
4.3 データ読み込み.....	10
5. ソフトウェアについて.....	11
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.....	11
6. プログラミング.....	12
6.1 Visual C++(C++/CLD) での使用.....	12
6.1.1 使用準備.....	12
6.1.2 関数の呼び出し方法.....	12
6.2 Visual Basic での使用.....	13
6.2.1 使用準備.....	13
6.2.2 プロシージャの呼び出し方法.....	13
6.3 Visual C#での使用.....	13
6.3.1 使用準備.....	13
6.3.2 関数の呼び出し方法.....	13
6.4 関数説明.....	15
6.5 エラーコード表.....	24

---

7. その他 .....	25
7.1 USB について .....	25
7.2 連絡先 .....	26
8. 仕様 .....	27
8.1 仕様概要 .....	27
8.2 取り付け穴寸法図 .....	28

## 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デジタル入出力ボード TUSB-K16DIOZ 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。

本製品の使用には製品の性質上、電子回路の知識を必要とします。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フェースユニットです。コンピュータを使用して外部電の計測をする事が出来ます。ドライバソフトウェア、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。

※ 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。インストールマニュアルを参照してください。

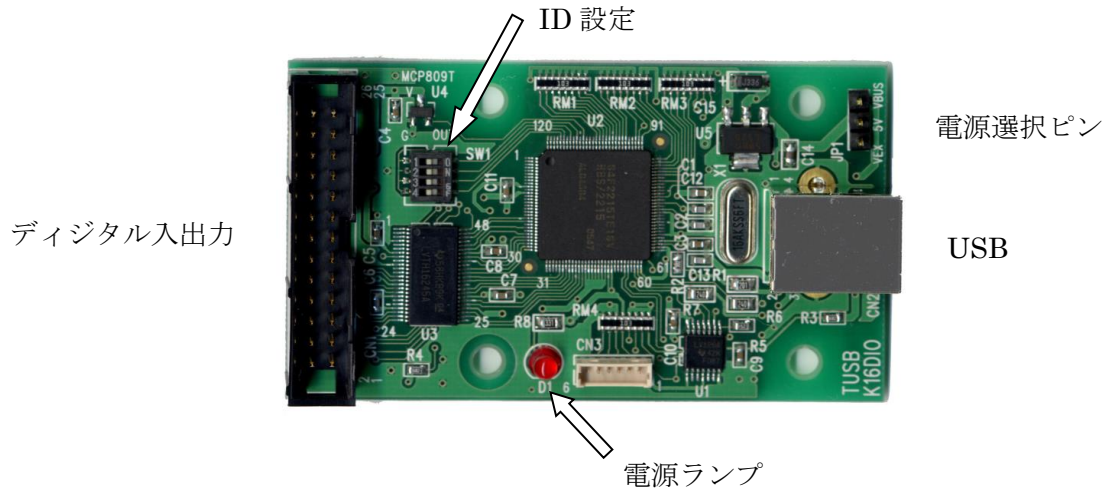
### 1.2 製品構成

本製品には以下の物が含まれます。

- ① TUSB-K16DIOZ 本体

## 2. 各部の名称

### 2.1 上面



デジタル入出力	16 ビットのデジタル入出力です。外部への電源出力および外部電源の入力端子もあります。
電源選択ピン	電源供給元の選択をします。
USB コネクタ	USB ケーブルでパソコンと接続します。
電源ランプ	ボードの電源がアクティブになると点灯します。
ID 設定(SW1)	複数枚使用する際に ID 設定をします。

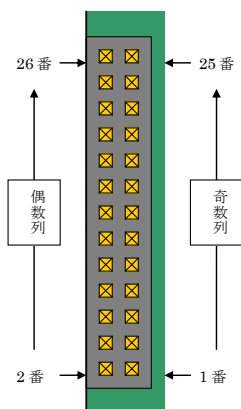
※ 初めて USB 接続される時にはドライバのインストール作業が必須です。インストールマニュアルを参照してください。

### 3. 各部説明

#### 3.1 デジタル入出力(使用コネクタおよび電氣的仕様)

16 ビットデジタル入出力および電源入出力コネクタです。デジタル入出力は 8 ビット毎に入出力設定可能です。入出力設定はソフトウェアにより行います。

##### 3.1.1 使用コネクタ



使用コネクタはフラットケーブル(又はリボンケーブル)用等と呼ばれているものです。

例) XG4C-2631(オムロン株式会社製)

ケーブル側は各社多数適合品種ありますが、一例としては以下の様なものがあります。

メーカー	型式
オムロン株式会社	XG4M-2630-T

##### 3.1.2 ピン配置

ピン番号	名称	ポート
1	D0	ポート 1
2	D1	
3	D2	
4	D3	
5	D4	
6	D5	
7	D6	
8	D7	
9	GND	
10		
11	D8	ポート 2
12	D9	
13	D10	

ピン番号	名称	ポート	
14	D11	ポート 2	
15	D12		
16	D13		
17	D14		
18	D15		
19	GND		
20			
21			
22	電源出力		
23			+5Vout
24			電源入力
25			+5Vin
26			



### 3.1.3 デジタル出力について

High 出力電圧の規定は 2V 以上ですが、出力電流によって電源電圧の約 3.3V に近くなります。又、Low 出力電圧は 0.8V 以下ですが、出力電流によって 0V(GND 電位)に近くなります。

いずれの論理レベルでも 1 ビット当たり 10mA 以上の電流を出力しないようにしてください。

### 3.1.4 デジタル入力について

入力電圧範囲は 0~5.5V 以内としてください。0.8V 以下で Low、2.0V 以上で High と認識します。

本ボードは入力がフローティングでも、前の論理状態を保持するホールド機能を有しています。入力の論理状態を切り替えるには外部ドライバに  $V_{th}$ (スレッシュホールド電圧)=1.5V 付近で  $750 \mu A$  以上の駆動能力が必要です。



**警告**

許容入力電圧を超える事の無い様にご使用下さい。  
感電や火災の原因となります。

### 3.1.5 電源出力について

内部電源 5V を外部に出力する端子です。電源容量は最大 100mA です。この電流は電源選択ピンの設定によって USB バス又は外部電源入力端子から供給されません。

### 3.1.6 電源入力について

外部電源を使用する場合(セルフパワー時)に電源を供給します。電圧は+5V ± 5%とってください。供給電流量は内部消費以外に各ビットからの出力電流や電源出力端子からの出力電流によります。

### 3.2 電源選択ピン

バスパワー(USB 電源)、セルフパワー(自己電源)の選択をします。工場出荷時はバスパワー設定です。次の場合等にセルフパワーを選択できます。

- (1) USB バスからの電流が不足の時。
- (2) 装置(システム)の電源 ON/OFF に連動させる必要がある場合。

※ バスパワー選択時には外部電源入力に電圧を印加しないで下さい。

※ セルフパワー選択時には外部電源入力に外部電源を接続してください。



### 3.3 USB コネクタ

USB ケーブルでパソコンと接続します。初めて接続する時にはドライバのインストール作業が必要です。接続前にドライバインストール説明書をご確認ください。

### 3.4 電源ランプ

ボードが動作状態になると点灯します。

### 3.5 ID 設定

1つのパソコンで本製品を複数台使用する場合には ID 設定が必要です。ID 設定は工場出荷時 0 です。その他の ID を設定する場合はスイッチで ID を設定してください。

ID 選択表

ID	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
ON BT	1		✓		✓		✓		✓		✓		✓		✓		✓
	2			✓	✓			✓	✓			✓	✓			✓	✓
	3					✓	✓	✓	✓					✓	✓	✓	✓
	4									✓	✓	✓	✓	✓	✓	✓	✓

例) ID = 5 に設定する場合 → 1 と 3 を ON その他は OFF

## 4. 機能解説

### 4.1 入出力設定

本ボードはソフトウェアによって入出力の設定が可能です。入出力の設定は 8 ビット単位で、D0～D7(ポート 1)、D8～D15(ポート 2)各々独立に設定できます。電源投入後の初期値は全て入力です。

主に使用する関数は

- TUSBK16DIO\_Dir
- TUSBK16DIO\_DirRead

となります。

### 4.2 データ出力

データの出力コマンドにより出力ポートの出力値を設定する事が出来ます。入力ポートに設定されている時にデータの出力コマンドを発行しても出力ポートにはなりません、内部に記憶されていますので出力設定にすると記憶されているデータが出力されます。電源が落ちると値は消えて全て **Low** となります。

主に使用する関数は

- TUSBK16DIO\_PortWrite [ポート毎に書き込む]
- TUSBK16DIO\_Write [2ポート共書き込む]
- TUSBK16DIO\_OutputRead [出力データを読み出す]

となります。

### 4.3 データ読み込み

ポートの入力状態を読み込みます。ポートが入力状態の時はポートの入力状態、ポートが出力状態の時は出力値が読み込まれます。

関連する関数は

- TUSBK16DIO\_Read

となります。

## 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

|-[TUSBKDIO]

|-[DRV] : 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

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

|-[CSharp] :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。

○ [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。 ※

○ [CppCLI]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。 ※

○ [CSharp]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。

※ 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。

## 6. プログラミング

ここでは、Visual C++、Visual Basic で TUSB-K16DIOZ 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、ここではソフトウェアセットに付属の定義ファイル(Visual C++および Visual Basic の両方が入っております)を利用した方法を説明します。

開発ツール(Visual C++、Visual Basic など)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。インストールマニュアルを参照してください。

注 1) 64bitOS 対応ドライバでは Visual Basic 6 は未対応です。

### 6.1 Visual C++(C++/CLI) での使用

#### 6.1.1 使用準備

Visual C++で使用するために以下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。

TUSBKDIO.H                      ヘッダファイル

※ 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。

※ 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 TUSKDIO.h をインクルードし、TUSKDIO.lib をプロジェクトに追加してください。

#### 6.1.2 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。TUSBK16DIO\_Device\_Open 以外の関数は TUSBK16DIO\_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 TUSBK16DIO\_Device\_Open を実行して機能関数の使用が終了したら TUSBK16DIO\_Device\_Close 関数を呼び出してドライバを開放してください。ドライバ関

数をアプリケーションで実行する場合には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** します。 **TUSB-K16DIOZ** を 2 つのアプリケーションで同時に操作する事は出来ません。

## 6.2 Visual Basic での使用

### 6.2.1 使用準備

Visual Basic で使用するために以下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。

**TUSBKDIO.vb**                      標準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既存項目の追加でプロジェクトに追加してください。

### 6.2.2 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。**TUSBK16DIO\_Device\_Open** 以外のプロシージャは **TUSBK16DIO\_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 **TUSBK16DIO\_Device\_Open** を実行してプロシージャの使用が終了したら **TUSBK16DIO\_Device\_Close** を呼び出してドライバを開放してください。デバイス機能をアプリケーションで実行する場合には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** します。

**TUSB-K16DIOZ** を 2 つのアプリケーションで同時に操作する事は出来ません。

## 6.3 Visual C#での使用

### 6.3.1 使用準備

Visual C#で使用するために以下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。

**TUSBKDIO.cs**                      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。

### 6.3.2 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。**TUSBK16DIO\_Device\_Open** 以外の関数は **TUSBK16DIO\_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

TUSBK16DIO\_Device\_Open を実行して機能関数の使用が終了したら TUSBK16DIO\_Device\_Close 関数を呼び出してドライバを開放してください。ドライバ関数をアプリケーションで実行する場合には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** します。 TUSB-K16DIOZ を 2 つのアプリケーションで同時に操作する事は出来ません。

## 6.4 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。



## TUSBK16DIO\_Device\_Open

C 宣言	short TUSBK16DIO_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

0 の場合、オープン出来た事を表します。  
それ以外の場合エラーコード表参照してください。

## TUSBK16DIO\_Device\_Close

C 宣言	void TUSBK16DIO_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

## TUSBK16DIO\_Dir

C 宣言	short TUSBK16DIO_Dir( short Id , BYTE Port , BYTE Dir )
	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

各ポートの入出力を設定します。出力に設定すると内部出力データバッファのデータが出力データとなります。

### 引数

Id	ユニット番号選択スイッチの番号(0-F)
Port	設定するポート 0:ポート 1 1:ポート 2
Dir (Direction)	入出力設定値 0:入力 1:出力

### 戻り値

エラーコード(エラーコード表参照)+

## TUSBK16DIO\_PortWrite

C 宣言	short TUSBK16DIO_PortWrite( short Id , BYTE Port , BYTE Data )
	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

指定のポートの出力データバッファにデータを書き込みます。書き込みポートが出力設定の場合は直ちに出力され、入力設定の場合はバッファに記録されます。

### 引数

Id	ユニット番号選択スイッチの番号(0-F)
Port	書き込むポート 0:ポート 1 1:ポート 2
Data	出力データを設定します。書き込むデータはバイナリで1のビットが <b>High</b> となります。 例えば、ポート 1 に 10 進数の 5 を書き込むと、2 進数では 0000 0101 なので 0 番面と 2 番目が 1 となるので D0 と D2 が <b>High</b> となり、残りのビットは全て <b>Low</b> となります。

### 戻り値

エラーコード(エラーコード表参照)

## TUSBK16DIO\_Write

C 宣言	short TUSBK16DIO_Write ( short Id , BYTE Data1 , BYTE Data2 )
	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

両ポートの出力データバッファにデータを書き込みます。書き込みポートが出力設定の場合は直ちに出力され、入力設定の場合はバッファに記録されます。

### 引数

Id	ユニット番号選択スイッチの番号(0-F)
Data1	ポート 1 の出力データを設定します。書き込むデータはバイナリで 1 のビットが High となります。 例えば、5 を書き込むと、2 進数では 0000 0101 なので 0 番面と 2 番目が 1 となるので D0 と D2 が High となり、残りのビットは全て Low となります。
Data2	ポート 2 の出力データを設定します。書き込むデータはバイナリで 1 のビットが High となります。 例えば、5 を書き込むと、2 進数では 0000 0101 なので 0 番面と 2 番目が 1 となるので D8 と D10 が High となり、残りのビットは全て Low となります。

### 戻り値

エラーコード(エラーコード表参照)

## TUSBK16DIO\_Read

C 宣言	short TUSBK16DIO_Read( short Id , BYTE *Data1 , BYTE *Data2 )
	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

ポートの状態を読み込みます。入力設定の場合は入力値、出力設定の場合は出力値が読み込まれます。

### 引数

Id	ユニット番号選択スイッチの番号(0-F)
Data1	<p>ポート 1 の状態を格納するためのバッファですデータはバイナリで <b>High</b> のビットが 1 となります。</p> <p>例えば、D0 と D2 が <b>High</b> で、残りのビットは全て <b>Low</b> の場合。</p> <p>2 進数では 0000 0101 なので 0 番面と 2 番目が 1 となり、5 が読み込まれます。</p>
Data2	<p>ポート 2 の状態を格納するためのバッファですデータはバイナリで <b>High</b> のビットが 1 となります。</p> <p>例えば、D8 と D10 が <b>High</b> で、残りのビットは全て <b>Low</b> の場合。</p> <p>2 進数では 0000 0101 なので 0 番面と 2 番目が 1 となり、5 が読み込まれます。</p>

### 戻り値

エラーコード(エラーコード表参照)

## TUSBK16DIO\_DirRead

C 宣言	short TUSBK16DIO_DirRead (short Id , BYTE *Dir1 , BYTE *Dir2 )
	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

設定されている入出力状態を確認します。

### 引数

Id	ユニット番号選択スイッチの番号(0-F)
Dir1	ポート 1 の入出力状態 0:入力 1:出力
Dir2	ポート 2 の入出力状態 0:入力 1:出力

### 戻り値

エラーコード(エラーコード表参照)

## TUSBK16DIO\_OutputRead

C 宣言	short TUSBK16DIO_OutputRead (short Id , BYTE *Data1 , BYTE *Data2)
	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

出力データバッファに設定されている値を読み込みます。

### 引数

Id	ユニット番号選択スイッチの番号(0-F)
Data1	ポート 1 出力データバッファの値
Data2	ポート 2 出力データバッファの値

### 戻り値

エラーコード(エラーコード表参照)



## 6.5 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 番号が不正です
2	ドライバが初期化されていない
3	すでにデバイスはオープンされています
4	接続されている台数が多すぎます
5	オープンできませんでした
6	指定のデバイスが見つからない
8	指定パラメータのエラー
9	USB 通信エラー
99	その他のエラー

## 7. その他

### 7.1 USBについて

※ ここでの記述は USB の一般的な記述となっております。

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フェースバスです。インタ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。USB2.0 の仕様では、1.5Mbps ロースピードデバイスおよび 12Mbps フルスピードデバイス、480Mbps のハイスピードデバイスが定義されています。

USB(フルスピード)の主な特長	
高速	12Mbps のバススピード(ハイスピードでは 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際にはハブが必要。)

#### 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。

## 7.2 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、メーカー  
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  
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	support@turtle-ind.co.jp
FAX	029-843-2024
郵送	〒300-0842 茨城県土浦市西根南 1-12-4

## 8. 仕様

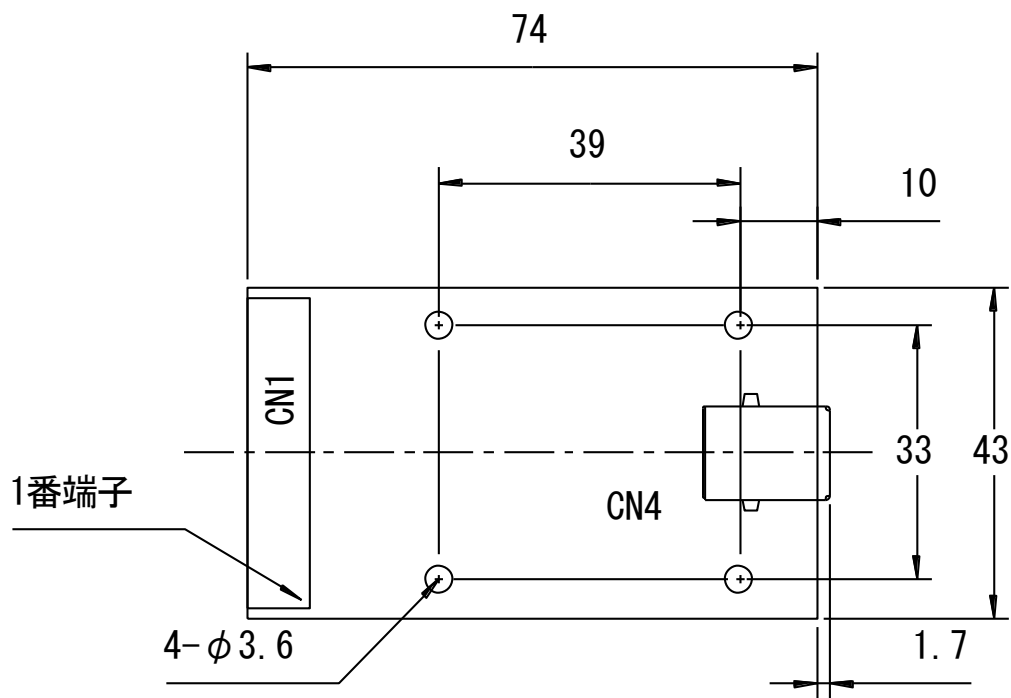
### 8.1 仕様概要

#### 仕様概要

入出力ビット数	16 ビット
入出力レベル	TTL 互換
入力 Low レベル	0.8V 以下
入力 High レベル	2V 以上
入力電流	100 $\mu$ A 未満(入力 0 又は 3.3V 以上の時)
出力 Low レベル	0.8V 以下
出力 High レベル	2V 以上
出力電流	10mA/ビット (シンク/ソース共) 全ポート合計 160mA
入出力選択	1 バイト(8 ビット)単位
使用温度範囲	5~45°C
外形寸法	74×43×約 14[半田面突起から一番背の高い部品の高さまで] (各 mm)
質量	約 20g(本体)
電源電圧と消費電流	USB バス又は+5V $\pm$ 5% 約 60mA(出力電流、外部電源出力含まず)

8.2 取り付け穴寸法図

取り付け穴図です。部品実装面から見た図(TOP VIEW)です。



**TUSB-K16DIOZ 取扱説明書**  
**(64bitOS 用ドライバ対応版)**

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