

TUSB-0214DAM

USB インタフェース高分解能高速

DA コンバータユニット

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。



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。



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。

注意事項

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が、万が一ご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。

©2024 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。

注意事項

使用上の注意と警告



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。



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。

注意事項

1. はじめに	6
1.1 製品概要	6
1.2 製品構成	6
1.3 ご使用の前に	6
2. 各部の名称	7
2.1 フロント(前面)	7
2.2 リア(後面)	7
3. 各部説明	8
3.1 Ch1～Ch4	8
3.2 DIGITAL I/O	8
3.3 USB コネクタ	9

4. サポートソフト 10

4.1 ドライバとサンプルプログラム 10

5. DA コンバータの機能 11

5.1 基本的なドライバの操作手順 11

5.2 シングル出力モードの操作 11

5.3 周期波形モードの操作 12

5.4 FIFO 出力モードの操作 12

5.5 デジタル IO の操作 13

5.6 グループについて 13

5.7 同期機能 13

5.8 出力データと出力電圧の関係 14

5.9 周期波形出力でのトリガとゲートの違い 14

5.10 起動後の初期出力 14

5.11 同期信号の接続 15

5.12 音声出力として 15

6. プログラミング 17

目次

6.1 Visual Basic での使用 17

6.2 Visual C#での使用 17

6.3 関数説明 18

6.4 動作モード設定構造体 42

6.5 戻り値(エラーコード) 43

7. その他動作の補足 44

7.1 PC スリープについて 44

7.2 FIFO オーバーラン、アンダーランの時の動作 44

7.3 同期入力と出力タイミング 44

7.4 トリガ、ゲート信号と出力タイミング 44

7.5	同期出力のタイミング	45
-----	------------	----

8.	仕様	46
----	----	----

8.1	仕様概要	46
-----	------	----

8.2	寸法	47
-----	----	----

9.	お問い合わせ	48
----	--------	----

目次		
----	--	--

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 D/A コンバータユニット **TUSB-0214DAM** 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、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アナログ電圧信号を出力する事ができます。ドライバソフトウェアおよびサンプルプログラムをご用意しております。

※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。サポートソフト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。

- ①TUSB-0214DAM 本体
- ②USB 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。

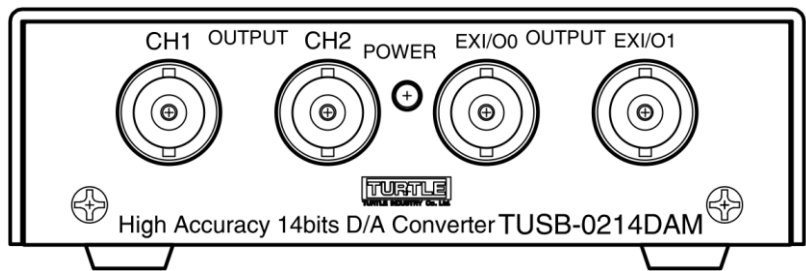
弊社 Web サイト(<https://www.turtle-ind.co.jp/>)にドライバおよびサンプルプログラムの入っているサポートソフトがございますのでダウンロードの上ご使用ください。

1.3 ご使用の前に

本製品のご使用前にはデバイスドライバのインストールが必要です。ドライバインストールの手順に従って正しくドライバインストールを行って下さい。

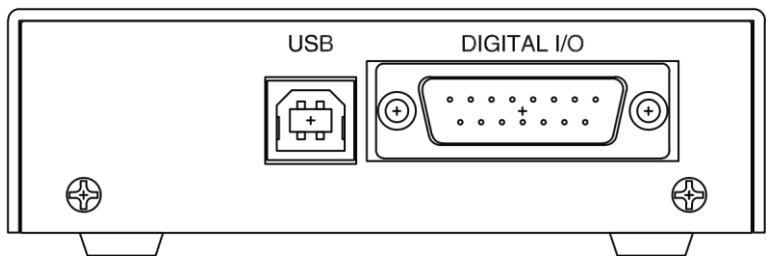
2. 各部の名称

2.1 フロント(前面)



コネクタ	説明
CH1	CH1 の DA 出力コネクタです。
CH2	CH2 の DA 出力コネクタです。
POWER	ユニットの電源が入っている時に点灯します。
EXI/O0 EXI/O1	双方向のデジタル I/O です。トリガ/ゲート入出力、同期クロック入出力、汎用入出力のいずれかを設定することができます。どの機能を割り当てるかはソフトウェア API から設定します。詳細はソフトウェア API の箇所を参照してください。

2.2 リア(後面)



USB ※	USB ケーブルを使用して PC と接続します。電源もここから供給されます。
DIGITAL I/O	入出力設定可能な 8 ビットの汎用デジタル入出力と、同期信号の入出力です。 << 入出力は TTL レベル互換です >>

※ 初めて PC に接続される時にはドライバのインストール作業が必要です。ドライバインストール説明書を参照の上ドライバをインストールしてください

3. 各部説明

3.1 EXI/00、EXI/01

正面パネルの左から 3 番目、4 番目はそれぞれ双方向 I/O が実装されており、同期用トリガ/ゲート、クロックの入出力、デジタル入出力のいずれかの機能をソフトウェア API で設定します。電圧レベルは以下のように規定されています。背面にも同じように外部機器との同期用信号の入出力があります。同期機能を有効/無効、どちら側を使用するかはソフトウェア API で設定します。

デジタル入出力

入力 0～0.81V : Low 1.8～5.5V : High
出力 Low : 0.7V 以下(10mA) High : 2.5V 以上(10mA) 同期入力
0～0.8V : Low 2.0～6.5V : High 同期出力
Low : 0.55V 以下(24mA) High : 2.3V 以上(24mA)

3.2 DIGITAL I/O

背面の Dusb15 ピンコネクタは汎用のデジタル入出力 8 ビットと同期信号の入出力です。入出力論理レベルは TTL です。

ピン番号	名称	機能
1	DIO0	デジタル入出力 0
2	DIO2	デジタル入出力 2
3	GND	信号グラウンド
4	DIO5	デジタル入出力 5

5	DI07	デジタル入出力 7
6	SYNCI0	同期入力 0
7	SYNCO0	同期出力 0
8	GND	信号グラウンド
9	DI01	デジタル入出力 1
10	DI03	デジタル入出力 3
11	DI04	デジタル入出力 4
12	DI06	デジタル入出力 6
13	GND	信号グラウンド
14	SYNCI1	同期入力 1
15	SYNCO1	同期出力 1

デジタル入出力

入力 0～0.81V : Low 1.8～5.5V : High
出力 Low : 0.7V 以下(10mA) High : 2.5V 以上(10mA) 同期入力
0～0.8V : Low 2.0～6.5V : High 同期出力
Low : 0.55V 以下(24mA) High : 2.3V 以上(24mA)

ケーブル側コネクタ

D サブ 15 ピンオスをご使用下さい。止めネジはインチ(#4-40UNC)です。

3.3 USB コネクタ

付属の USB ケーブルを使用して、ご利用の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ドライバのインストール作業が必要です。ドライバインストール説明書を参照してください。

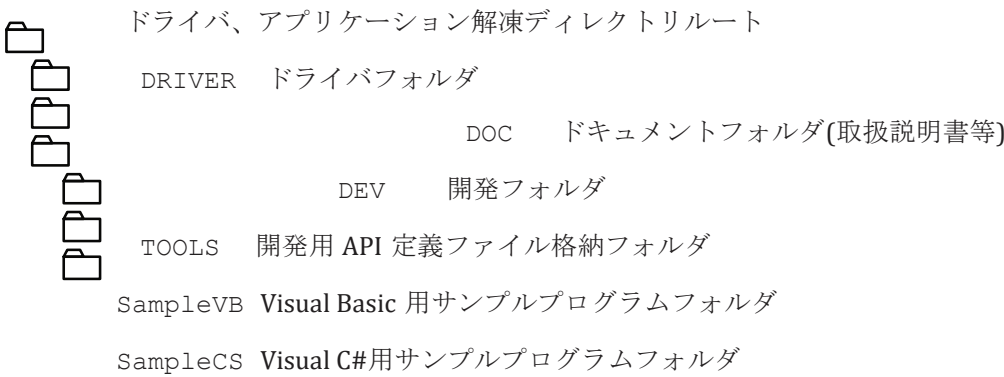
4. サポートソフト

4.1 ドライバとサンプルプログラム

ドライバおよびサンプルプログラムは弊社 Web サイトよりダウンロードしてください。

<https://www.turtle-ind.co.jp>

ダウンロードファイルを解凍すると下記構成のディレクトリにファイルが入っております。
このディレクトリ構成は予告なくソフトウェアのバージョンにより変わる事があります。



ディレクトリ名	説明
DRIVER	ドライバファイルが入っています。ドライバのインストールは DOC フォルダ内のドライバのインストール説明書に従ってインストールしてください。
DOC	取扱説明書等が入っています
TOOLS	開発時に必要な各種ファイルの入っているフォルダが入っています
SampleVB	Visual Basic のサンプルプログラムコードが入っています。サンプルプログラムは Visual Studio 2022 で作成されたもので、開発ツールのバージョンによってはそのままコンパイル実行できないコードがあるかもしれません。

SampleCS	Visual C#のサンプルプログラムコード入っています。 サンプルプログラムは Visual Studio 2022 で作成されましたので、開発ツールのバージョンによってはそのままコンパイル実行できないコードがあるかもしれません。
----------	---

サポートソフト

5. DA コンバータの機能

本 DA コンバータは指定電圧を出力(主に直流電圧)するシングル出力モード、指定波形を繰り返し出力する周期波形モード(主に任意波形発振器)、Pc から受信したデータを一定間隔で出し続ける FIFO 出力モード(音声や波形の記録データを再生)に対応しています。

シングル出力モード	周期波形モード	FIFO 出力モード
PC からのコマンドで指定電圧を出力します。次にコマンドを受け取るまで指定電圧を出力し続けます。直流電圧を出力します。	事前に波形イメージをメモリに書き込み、同一波形を繰り返し出し続けます。任意波形ファンクションジェネレータとなります。	PC から出力データ列を受信すると本体メモリに蓄積して指定周期で順次データを出し続けます。データを送信し続ける事で連続的に任意波形出力できます。

5.1 基本的なドライバの操作手順

ドライバ関数を使用して DA コンバータの基本的な操作手順は以下の通りです。

1. TUSB0214DAM_Connect を実行してデバイスと接続する。
2. ドライバ関数を操作して DA コンバータを使用する。
3. TUSB0214DAM_DisConnect を実行してデバイスとの接続を解除する。

通常、TUSB0214DAM_Connect と TUSB0214DAM_DisConnect はアプリケーションの開始時と終了時に実行してください。機能を実行する毎に Connect と Disconnect を繰り返す必要はありません。

5.2 シングル出力モードの操作

関連する操作関数

- TUSB0214DAM_DacCh_Set

- TUSB0214DAM_RangeSet
- TUSB0214DAM_RangeSet_R
- TUSB0214DAM_DcValSet
- TUSB0214DAM_DcValSet_R

TUSB0214DAM_DacCh_Set で指定チャンネルをシングル出力モードに設定、TUSB0214DAM_RangeSet で指定チャンネルの出力レンジを設定してください。TUSB0214DAM_DcValSet で指定チャンネルの出力電圧を設定してください。TUSB0214DAM_RangeSet_R で現在設定されている出力レンジ、TUSB0214DAM_DcValSet_R を使用して現在出力設定されている値を確認する事が出来ます。シングル出力モードに設定されている間、TUSB0214DAM_RangeSet や TUSB0214DAM_DcValSet 関数を使用して何度でも出力レンジや電圧を変更する事が出来ます。

5.3 周期波形モードの操作

関連する操作関数

- TUSB0214DAM_Exs_Sel(外部同期動作で同期信号として背面ポートを使用する場合)
- TUSB0214DAM_ExIO_Sel(外部同期動作で同期信号としてフロントパネルのBNCを使用する場合)
- TUSB0214DAM_Clock_Gen • TUSB0214DAM_DacCh_Set
- TUSB0214DAM_WavImgSet
- TUSB0214DAM_RangeSet
- TUSB0214DAM_RangeSet_R
- TUSB0214DAM_Trigger
- TUSB0214DAM_GateSet
- TUSB0214DAM_GateSet_R

TUSB0214DAM_DacCh_Set で指定チャンネル(後述)を周期波形モード(トリガ動作またはゲート動作)に設定してください。TUSB0214DAM_Clock_Gen で出力クロック周波数、TUSB0214DAM_RangeSet で出力電圧レンジ、TUSB0214DAM_WavImgSet で出力波形イメージを設定します。トリガ動作の場合は TUSB0214DAM_Trigger を使用して、ゲート動作の場合は TUSB0214DAM_GateSet を使用して波形の出力をコントロールします。

5.4 FIFO 出力モードの操作

関連する操作関数

- TUSB0214DAM_Exs_Sel(外部同期動作で同期信号として背面ポートを使用する場合)

- *TUSB0214DAM_ExIO_Sel*(外部同期動作で同期信号としてフロントパネルの BNC を使用する場合)
- *TUSB0214DAM_Clock_Gen*
- *TUSB0214DAM_DacCh_Set*
- *TUSB0214DAM_Fifo_ImgSet*
- *TUSB0214DAM_Fifo_Clr*
- *TUSB0214DAM_Fifo_Status*
- *TUSB0214DAM_RangeSet*
- *TUSB0214DAM_RangeSet_R*
- *TUSB0214DAM_Trigger*

TUSB0214DAM_DacCh_Set で指定チャンネル(後述)を FIFO 出力モードに設定してください。*TUSB0214DAM_Clock_Gen* で出力クロック周波数、*TUSB0214DAM_RangeSet* で出力電圧レンジを設定します。*TUSB0214DAM_Trigger* で出力動作を開始しますが、出力動作中は *TUSB0214DAM_Fifo_ImgSet* により FIFO メモリに書かれたデータを出力し続けます。*TUSB0214DAM_Fifo_Status* で FIFO メモリに残っているデータ量を確認しながら出力データを書き込み続けて下さい。通常は *TUSB0214DAM_Trigger* で出力動作を開始する前に開始直後に出力すべきデータ列を FIFO メモリに適当な長さ書き込んでおきます。

FIFO 出力は *TUSB0214DAM_Trigger* で停止できます。

[FIFO モードでの出力速度について]

FIFO 出力モードの動作では常にデータを送信しつづける必要があります。データ送信出力が間に合わないとアンダーランが発生します。PC の状況等にもよりますが、2ch 同時出力で 1MSPS で出力するとアンダーランが発生します。2ch 500kSPS 程度ですと安定して出力しましたが、PC 側の動作状態やアプリケーションの処理にもよりますので実機にて充分にご確認下さい。

5.5 デジタル IO の操作

関連する操作関数

- *TUSB0214DAM_Dio_Dir*
- *TUSB0214DAM_Dio_Dir_R*
- *TUSB0214DAM_Dio_In*
- *TUSB0214DAM_Dio_Out*
- *TUSB0214DAM_Dio_Out_R*
- *TUSB0214DAM_ExIO_Sel*

本装置は 8 ビットのデジタル IO を備えており、*TUSB0214DAM_Dio_Dir* によりビット毎に入出力方向を設定できます。*TUSB0214DAM_Dio_Out* により出力ビットの状態 High(1)、

Low(0)を設定し、TUSB0214DAM_Dio_In により入力ビットの状態を確認します。電源投入後の入出力方向初期値は全て入力です。全ビット 10k オームで 3.3V にプルアップされています。

5.6 チャンネルについて

Ch1 と Ch2 でそれぞれ個別に出力モードや出力クロック、トリガ、ゲート信号等を設定可能です。Ch1 と Ch2 で別々の出力モード、クロックやタイミングで動作させる事が可能です。

5.7 同期機能

チャンネル間や外部機器と同期動作を行う事が出来ます。

- Ch1 と Ch2 を同時に動作させる
- 本装置複数台の各チャンネルを同時に動作させる
- 外部機器の信号に同期して本装置の各チャンネルを動作させる

等を行う事が出来ます。

Ch グループの動作を外部同期するには TUSB0416DAM_DacGr_Set にてクロックやゲート/トリガ信号を外部に設定します。ディジタル入出力の同期入力 1 に外部からクロック信号を入力し、同期入力 0 にトリガまたはゲート信号を入力すると出力を開始します。クロックのみまたはゲート/トリガ信号のみを外部に設定する事も可能です。これらの同期動作時に使用するためグループ Ch1、Ch2 とは別にクロックとゲート/トリガ信号の生成回路があります。TUSB0416DAM_Exs_Sel を使用して同期出力 0 にゲート/トリガ信号、同期出力 1 にクロックを出力できます。同じ信号をフロントパネル側から送出することもできます。その場合は TUSB0416DAM_ExIO_Sel を使用して同期出力 0 にゲート/トリガ信号、同期出力 1 にクロックを出力できます。

5.8 出力データと出力電圧の関係

本装置はチャンネル毎に設定可能な 7 種類の出力レンジを持っており、各レンジにより出力される電圧の範囲が異なります。各レンジにおいて出力 0 が最も低い電圧、16383 が最も高い電圧になります。出力データと電圧の関係は以下の表の通りです。

レンジ	電圧範囲		出力データ (d) と出力電圧 [V]		
	最小 [V]	最大 [V]	0 (d)	8192 (d)	16383 (d)
±10V	-10	+10	-10.0000	0.0000	+9.9988
±5V	-5	+5	-5.0000	0.0000	+4.9994
±2.5V	-2.5	+2.5	-2.50000	0.00000	+2.49969
±1.25V	-1.25	+1.25	-1.25000	0.00000	+1.24985
10V	0	10	0.0000	+5.0000	+9.9994
5V	0	5	0.00000	+2.50000	+4.99969
2.5V	0	2.5	0.00000	+1.25000	+2.499847

- 表中の出力電圧は設計上の理想値であり精度を表したものではありません
- 電圧範囲の最大値は出力されません この電圧より各レンジの 1LSB だけ低い電圧が最大電圧とされます

5.9 周期波形出力でのトリガとゲートの違い

トリガ動作ではトリガ信号の立下り、信号レベル High(1) から Low(0) に下がったタイミングで波形出力を開始または停止 (外部同期時には外部から停止は出来ません) します。ゲート動作ではゲート信号レベルが High の時出力は停止で Low の時波形出力 (逆論理にも設定できます) となります。外部同期を使用しない場合はいずれも任意のタイミングで開始および停止できる事には変わりありませんが、動作には以下の違いがあります。

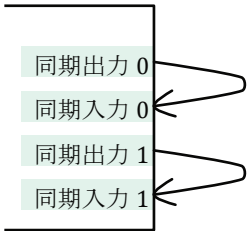
- トリガ動作では 1 回のトリガで事前に指定した回数だけ波形を出力する事ができます。ゲート動作では出力回数の設定が無いのでゲートが停止状態になるまで波形を出力し続けます。
- ゲート動作では一時停止が可能です。ゲートが停止した時の波形の最終電圧を維持して再度ゲートが波形出力開始した時に一時停止した波形の続きを出す事ができます。トリガではトリガ信号検出で波形イメージメモリの先頭から出力されません。
- 外部トリガの時には外部同期信号で波形出力を停止する事は出来ません。指定回数の出力で自動的に停止するか、TUSB0214DAM_Trigger で停止を指令すれば波形出力が停止されます。

5.10 起動後の初期出力

電源投入後は 5V(ユニポーラ)レンジで約 0V が出力されています。ソフトウェアでレンジ設定がされるまでは校正システムが動作せず 0V からは少しずれています。

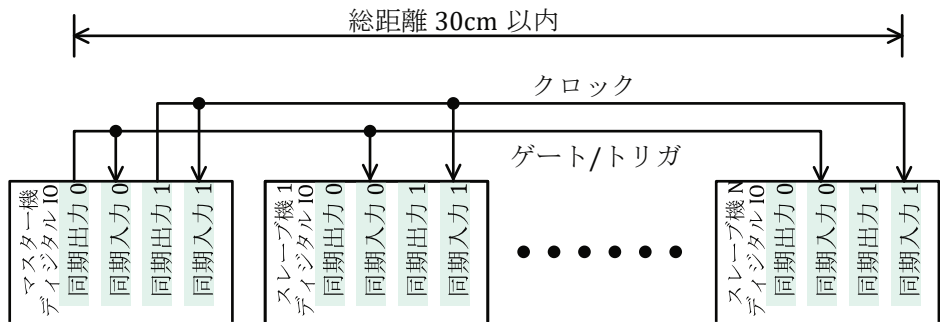
5.11 同期信号の接続

Ch1 と Ch2 を同期する場合、
デジタル IO の同期出力 0 と同期入力 0、
同期出力 1 と同期入力 1 を直近で接続してく
ださい。



□ 複数台同期使用する場合

1 台マスター機を決めてマスター機の同期出力からマスター機の同期入力およびスレーブ機の同期入力に信号を接続してください。スレーブ機の接続可能台数の規定はありませんが、接続総距離が 30cm 程度以内で繋いで下さい。それ以上の距離の場合は接続線や環境により同期信号が乱れて正常に動作しない可能性がありますので、信号波形をご確認の上ご使用下さい。



□ 外部機器からの信号供給

同期信号入力は TTL 論理レベルです。補足説明の同期信号のタイミングを参照の上接続してください。

5.12 音声出力として

本装置は PC のオーディオ出力用としては設計されておりませんのでコンピュータに接続しただけでは音声出力する事は出来ませんが、FIFO 出力機能を利用して音声ファイル等をアナログ出力する事は可能です。

サンプリングレートの設定例(TUSB0214DAM_Clock_Gen の設定)

レート[SPS]	DivideN	設定されるレート[SPS]	設定誤差[%]
32 k	3124	32 k	0

44.1k	2267	44.092k	-0.019
48 k	2082	48.008k	+0.016
96 k	1041	95.969k	-0.032
192 k	520	191.94 k	-0.032

クロックは PC に対して非同期です。装置内のクロックまたは外部クロック入力からのタイミングのみで出力されます。データ転送が間に合っている限りは USB の通信状態は出力に影響を与えません。

直流を遮断するフィルタがありませんので低周波側は直流までフラットです。使い方によって

直流が出力される可能性があります。接続する機器により問題が無いかご確認下さい。

ノイズレベル(参考値) 0.01mV
±2.5V レンジ 44.1kSPS で 0V を出力 聴感補正フィルタ A

6. プログラミング

Visual Basic、Visual C# で TUSB-0214DAM 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 Basic Visual C# 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。開発ツールの説明書、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。付属のドライバインストール説明書を参照してください。

6.1 Visual Basic での使用

Visual Basic で使用するためにライブラリファイルをサポートソフト内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。

TUSB0214DAM.vb

標準ライブラリファイル

ライブラリファイルはサポートソフト内の DEV\TOOLS フォルダに有ります。既存項目の追加でプロジェクトに追加してください。

□ 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。

TUSB0214DAM_Connect 以外のプロシージャは TUSB0214DAM_Connect 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

TUSB0214DAM_Connect を実行してプロシージャの使用が終了したら

TUSB0214DAM_DisConnect 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アプリケーションの初めに Connect し、アプリケーションの終了時に Disconnect します。

- 一つの TUSB-0214DAM 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 Connect～Disconnect 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。

6.2 Visual C#での使用

Visual C# で使用するためにライブラリファイルをサポートソフト内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。

TUSB0214DAM.cs

ライブラリファイル

ライブラリファイルはサポートソフト内の DEV\TOOLS フォルダにあります。既存項目の追加でプロジェクトに追加してください。

□ 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。

TUSB0214DAM_Connect 以外の関数は TUSB0214DAM_Connect 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 TUSB0214DAM_Connect を実行して機能関数の使用が終了したら TUSB0214DAM_DisConnect 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アプリケーションの初めに Connect し、アプリケーションの終了時に Disconnect します。

- 一つの TUSB-0214DAM 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 Connect～Disconnect 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。

6.3 関数説明

各関数(プロシージャ)のもつ機能の詳細を説明します。

□ 関数の引数で指定する SerialNumber(シリアル番号)文字列

AD コンバータの 4 桁のシリアル番号を指定してください。先頭のゼロは省略せずに 4 桁全て指定してください。例えばシリアル番号 0123 の場合、"123"ではなく"0123"という文字列で指定してください。

TUSB0214DAM_Connect

C 宣言	unsigned long TUSB0214DAM_Connect(const char* SerialNumber)
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してくだ

さい解説

指定シリアル番号の DA コンバータに接続して操作できるようにします。
DA コンバータのドライバ関数を使用する前に呼び出さなければなりません。アプリケーション開始時に実行してください。切断関数 TUSB0224ADM_DisConnect が実行されるまで接続状態を維持します。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
--------------	-------------------------

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_DisConnect

C 宣言	<code>unsigned long TUSB0214DAM_DisConnect(const char* SerialNumber)</code>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

指定シリアル番号の DA コンバータとの接続を解除します。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
---------------------	-------------------------

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Dio_Dir

C 宣言	<code>unsigned long TUSB0214DAM_Dio_Dir(const char* SerialNumber, unsigned char DirData)</code>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

ディジタル入出力 8 ビットの入出力方向を設定します

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
---------------------	-------------------------

DirData	入出力方向設定値 この引数の 8 ビット値の各ビットが入出力の 8 ビットに対応します。 0 のビットが入力、1 のビットが出力になります。 DirData の最下位ビットが DIO0 に対応し、順に並び最上位ビットが DIO7 に対応しています。例えば DIO1 だけ出力で他は入力とする場合にはこの引数に 02(HEX)を設定します。
----------------	--

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Dio_Dir_R

C 宣言	<code>unsigned long TUSB0214DAM_Dio_Dir_R(const char* SerialNumber, unsigned char* DirData)</code>
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

ディジタル入出力 8 ビットの入出力方向の設定値を読み出します
引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
DirData	設定値を格納する格納先変数への参照(アドレス、ポインタ) 格納されるデータ形式は TUSB0214DAM_Dir_Dir の DirData と同一です。

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Dio_In

C 宣言	<code>unsigned long TUSB0214DAM_Dio_In(const char* SerialNumber, unsigned char* InData)</code>
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

リアパネルの I/O 入出力 8 ビットのピンの状態とフロントパネルの I/O 入出力 2 ビット合計 10 ビットのピン状態を読み出します

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
InData	入力ピン状態を表す値を格納する格納先変数への参照(アドレス、ポインタ) 各桁ビット 0 が入力 Low、ビット 1 が入力 High を表します。InData の最下位ビットが DIO0 に対応し、順に並び最上位ビットが DIO7 に対応しています。例えば 02(HEX)が取得された場合は、DIO1 のみが High で他のビットは Low です。この値は入出力方向設定に関係なく IC ピンの状態を反映します。

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Dio_Out

C 宣言	unsigned long TUSB0214DAM_Dio_Out(const char* SerialNumber, unsigned char OutData)
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

ディジタル入出力 8 ビットの出力ピンの状態を設定します

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
OutData	出力設定値 この引数の 8 ビット値の各ビットが出力の 8 ビットに対応します。各 桁ビット 0 が出力 Low、ビット 1 が出力 Hig を表します。OutData の最下位ビットが DIO0 に対応し、順に並び最上位ビットが DIO7 に対応しています。例えば DIO1 だけ High 出力で他は Low 出力とする場合にはこの引数に 02(HEX)を設定します。入力に設定されているビットには本設定は効果はありませんが、入力ピンであっても設定値はレジスタに保持されますので出力ピンに設定すると即座に出力に反映します。

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Dio_Out_R

C 宣言	unsigned long TUSB0214DAM_Dio_Out_R(const char* SerialNumber, unsigned char *OutData)
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

ディジタル入出力 8 ビットの出力設定レジスタの状態を設定します

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
--------------	-------------------------

OutData	出力ピンの設定レジスタの状態を表す値を格納する格納先変数への参照(アドレス、ポインタ) 格納されるデータ形式は TUSB0214DAM_Dio_Out の OutData と同一です。
---------	--

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Exs_Sel

C 宣言	<pre>unsigned long TUSB0214DAM_Exs_Sel(const char* SerialNumber, unsigned char OutSel0, unsigned char OutSel1)</pre>
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

ディジタル入出力の同期出力機能を設定します

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
OutSel0	同期出力 0 選択 0 : High レベル(起動後初期値) 1 : Low レベル 2 : ゲート/トリガ信号 3 : Ch1 周期波形同期出力 4 : Ch2 周期波形同期出力
OutSel1	同期出力 1 選択 0 : High レベル(起動後初期値) 1 : Low レベル 2 : クロック信号 3 : CH1 周期波形同期出力 4 : Ch2 周期波形同期出力

- 複数グループ、複数台同期動作時のゲート/トリガコントロールには **OutSel0** は 2 を選択してください。
- 複数グループ、複数台同期動作時のクロック出力には **OutSel1** は 2 を選択してください。

エラーコード(エラーコード表参照)

TUSB0214DAM_Clock_Gen

C 宣言	<code>unsigned long TUSB0214DAM_Clock_Gen(const char* SerialNumber, unsigned char Msel, unsigned int DivideN)</code>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

周期波形モード、FIFO 出力モード時の出力更新用クロック生成器の設定をします

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
Msel	設定モジュールの指定 0 : Ch1 1 : Ch2 2 : 同期出力用
DivideN	分周器設定値 99~4294967295 (2 の 32 乗マイナス 1)

周波数は 1MHz 以下となる様に設定してください

$$\text{周波数} = \frac{1 \times 10^8}{\text{DivideN} + 1 \text{ 戻り値}} \text{ [Hz]}$$

エラーコード(エラーコード表参照)

TUSB0214DAM_DacChSet

C 宣言	<code>unsigned long TUSB0214DAM_DacChSet(const char* SerialNumber, unsigned char Msel, stSettings *Damset)</code>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

指定グループの動作モードおよび関連設定をします

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
Msel	設定チャンネルの指定 0 : Ch1 1 : Ch2
DamSet	DA コンバータ動作モード設定構造体への参照(アドレス、ポインタ) 動作モード設定構造体の説明を参照してください。

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_DacChSet_R

C 宣言	<code>unsigned long TUSB0214DAM_DacChSet_R(const char* SerialNumber, unsigned char Msel, stSettings *Damset)</code>
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

指定グループに設定されている動作モードおよび関連設定を読み出します

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
MSe1	設定モジュールの指定 0 : Ch1 1 : Ch2
DamSet	DA コンバータ動作モード設定構造体への参照(アドレス、ポインタ) 動作モード設定構造体の説明を参照してください。

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_WavImgSet

C 宣言	unsigned long TUSB0214DAM_WavImgSet(const char* SerialNumber, unsigned char Msel, unsigned short *WImage, unsigned int Len)
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

周期波形モード(または FIFO 出力モード)時に波形イメージを書き込みます

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
MSe1	設定モジュールの指定 0 : Ch1 1 : Ch2
WImage	波形メモリへ書き込む波形データを格納した配列の先頭アドレス(ポインタ)

Len	波形配列のデータ個数 1~524288 周期波形出力モードでは出力波形の 1 周期のデータ個数になります。FIFO モードでは FIFO 出力メモリへ追加するデータ数です。
------------	--

周期波形出力モードと FIFO 出力モードでは波形メモリを共用しています。周期波形出力モードへのモード切り替えを行った時はこの API 関数でメモリのデータを書き換えてから波形出力を開始してください。FIFO 出力モードへのモード切替を行った時には FIFO メモリをクリアしてからこの API 関数または TUSB0214DAM_Fifo_ImgSet で波形を書き込んで下さい。

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Fifo_ImgSet

C 宣言	unsigned long TUSB0214DAM_Fifo_ImgSet(const char* SerialNumber, unsigned char Ch, unsigned short *WImage, unsigned int Len)
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

FIFO 出力モード時に波形イメージを書き込みます。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
Ch	イメージ書き込みチャンネルの指定 0 : Ch1 1 : Ch2

WImage	波形メモリへ書き込む波形データを格納した配列の先頭アドレス(ポインタ)
Len	波形配列のデータ個数 1~524288 周期波形出力モードでは出力波形の 1 周期のデータ個数になります。FIFO モードでは FIFO 出力メモリへ追加するデータ数です。

周期波形出力モードと FIFO 出力モードでは波形メモリを共用しています。FIFO 出力モードへのモード切替を行った時には FIFO メモリをクリアしてからこの API 関数または TUSB0214DAM_Fifo_ImgSet で波形を書き込んで下さい。周期波形出力モードではこの API 関数では正常にイメージメモリに書き込む事が出来ないので使用しないで下さい。

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Fifo_Clr

C 宣言	<code>unsigned long TUSB0214DAM_Fifo_ImgSet(const char* SerialNumber, unsigned char Ch)</code>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

FIFO 出力モード時に波形イメージをクリアします。FIFO 出力モードに切り替えた後や停止後に再度初めから新たな波形で開始する時に実行してください。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
---------------------	-------------------------

Ch	イメージをクリアするチャンネルの指定 0 : Ch1 1 : Ch2
----	--

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Fifo_Status

C 宣言	<pre>unsigned long TUSB0214DAM_Fifo_ImgSet(const char* SerialNumber, unsigned char *Over, unsigned char *Under, unsigned int *DataLen)</pre>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してくだ

さい解説

FIFO 出力モード時に波形イメージ FIFO メモリの状態を確認しま
す
引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
Over	バッファオーバーラン発生フラグを格納する配列の先頭ア ドレス (ポインタ) イメージメモリが埋まっている状態でさらにイメージデー タを書き込もうとするとオーバーランが発生します。通常 は 0 が格納され、オーバーランが発生すると 1 になりま す。この API により読み込まれるとフラグはクリアされま す。 Over[0] : ch1 フラグ Over[1] : ch2 フラグ

Under	バッファアンダーラン発生フグらを格納する配列の先頭アドレス (ポインタ) FIFO 出力中にイメージメモリにデータが無くなり出力が出来なくなるとアンダーランが発生します。通常は 0 が格納され、アンダーランが発生すると 1 になります。この API により読み込まれるとフラグはクリアされます。 Under[0] : ch1 フラグ Under[1] : ch2 フラグ
DataLen	イメージメモリ残データ個数を格納する配列の先頭アドレス(ポインタ) 書き込まれたイメージメモリでまだ出力されずにメモリ内に存在するデータの個数です。FIFO 出力モードで出力中はこの値が 0 にならない様に常にイメージデータを送信し続ける必要があります。 DataLen[0] : ch1 データ個数 DataLen[1] : ch2 データ個数 DataLen[2] : ch3 データ個数 DataLen[3] : ch4 データ個数

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Trigger

C 宣言	<code>unsigned long TUSB0214DAM_Trigger(const char* SerialNumber, unsigned char Msel, unsigned char Ssel)</code>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

周期波形出力モード(トリガ)、FIFO 出力モード時に波形出力の開始または停止を実行します。

波形出力停止時に Ssel に 1 を設定して実行すると波形出力が開始されます。波形の出力を停止したい時は Ssel に 0 を設定してください。同期出力コントロールには停止パルスがありませんので Ssel の値は無効です。同期開始した場合でも停止する場合は停止するグループを指定して停止してください。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
MSe1	設定モジュールの指定 0 : Ch1 1 : Ch2 2 : 同期出力用
SSe1	開始トリガまたは停止トリガの選択 0 : 停止トリガ 1 : 開始トリガ MSe1 = 2 の時はこの値は無効、開始しか実行できません

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_Trigger_Stat

C 宣言	unsigned long TUSB0214DAM_Trigger(const char* SerialNumber, unsigned char Mse1, unsigned char *Stat)
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

周期波形出力モード(トリガ)、FIFO 出力モード時に波形が出力中であるか確認します。開始トリガでステータスは出力中になり、指定回数の出力完了または停止トリガで出力停止中になります。引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
MSe1	設定モジュールの指定 0 : Ch1 1 : Ch2
Stat	ステータス格納先変数への参照(アドレス、ポインタ) 0 : 出力停止中 1 : 出力中

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_GateSet

C 宣言	<code>unsigned long TUSB0214DAM_GateSet(const char* SerialNumber, unsigned char Msel, unsigned char GateEn)</code>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

周期波形出力モード(ゲート)時に波形出力の開始または停止を指定します。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
Msel	設定モジュールの指定 0 : Ch1 1 : Ch2 2 : 同期出力用
GateEn	開始または停止の選択 0 : 出力停止 1 : 出力開始

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_GateSet_R

C 宣言	<code>unsigned long TUSB0214DAM_GateSet_R(const char *SerialNumber, unsigned char Msel, unsigned char *GateEn)</code>
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

ゲート信号の状態を表します。Ch1 および Ch2 の時は TUSB0214DAM_GateSet の設定ではなく実際に選択されているゲート信号の状態を表します。外部ゲートの時は
は外部ゲートが High(ゲート反転の時は Low)の時に GateEn が 1 となります。モードがゲート以外の時は無効です。
同期出力モジュールの場合は TUSB0214DAM_GateSet の設定値がそのまま返ります。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
MSEL	読み出すモジュールの指定 0 : Ch1 1 : Ch2 2 : 同期出力用
GateEn	ゲートの状態を格納する変数への参照(アドレス、ポインタ) 格納される値は TUSB0214DAM_GateSet の GateEn と同一です。

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_RangeSet

C 宣言	unsigned long TUSB0214DAM_RangeSet(const char *SerialNumber, unsigned char Ch, unsigned char Range)
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

指定チャンネルの出力レンジを設定します。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
Ch	レンジ設定チャンネルの指定 0 : Ch1 1 : Ch2
Range	レンジ指定値 0 : ±10V レンジ 1 : ±5V レンジ 2 : ±2.5V レンジ 3 : 0~10V レンジ 4 : 0~5V レンジ

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_RangeSet_R

C 宣言	<code>unsigned long TUSB0214DAM_RangeSet_R(const char *SerialNumber, unsigned char Ch, unsigned char Range)</code>
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

指定チャンネルの出力レンジ設定を読み出します

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
Ch	読み出すチャンネルの指定 0 : Ch1 1 : Ch2

Range	レンジ設定値を格納する変数への参照(アドレス、ポインタ) 格納される値は TUSB0214DAM_RangeSet の Range と同一です。
-------	---

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_DacValSet

C 宣言	<pre>unsigned long TUSB0214DAM_DacValSet(const char *SerialNumber, unsigned char Ch, unsigned short NVal)</pre>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

シングル出力モード時に指定チャンネルの出力レベル(電圧)を設定します。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
Ch	出力値設定チャンネルの指定 0 : Ch1 1 : Ch2
NVal	出力する値

戻り値

エラーコード(エラーコード表参照)

TUSB0214DAM_DacValSet_R

C 宣言	<pre>unsigned long TUSB0214DAM_DacValSet(const char *SerialNumber, unsigned char Ch, unsigned short *NVal)</pre>
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください



解説

シングル出力モード時に指定チャンネルに設定されている出力レベル(電圧)を読み出します。引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
Ch	読み出しチャンネルの指定 0 : Ch1 1 : Ch2
NVal	読み出した値を格納する変数への参照(アドレス、ポインタ)

戻り値

エラーコード(エラーコード表参照)

6.4 動作モード設定構造体

動作モード設定実行時に動作条件を設定する構造体です。構造体の宣言はヘッダやライブラリにあります。構造体メンバの説明をします。

Mode	Ch 設定する動作モードの選択 0 : シングル出力モード 1 : 周期波形出力モード ゲート動作 2 : 周期波形出力モード トリガ動作 3 : FIFO 出力モード
ClkSel	周期波形および FIFO 出力モードの時のクロックソース選択 0 : 内部クロック 1 : リアパネルのクロック入力 2 : フロントパネルのクロック入力
GTSe1	周期波形および FIFO 出力モードの時のゲートまたはトリガクロックソース選択 0 : 内部信号 1 : リアパネルのゲート/トリガ入力 2 : フロントパネルのゲート/トリガ入力
ExGtInv	周期波形ゲート動作時の外部ゲート信号論理反転 0 : 反転しない(外部入力 0 が Low の時波形出力) 1 : 反転する(外部入力 0 が High の時波形出力)
Rstc	ゲート再開時波形位置リセット選択 0 : リセットする 1 : リセットしない ゲート信号が一度ディセーブル(波形停止)した後、再度イネーブル(波形出力)になった時に前の波形の停止した位置から出力をする(1 を設定)か、波形メモリの先頭から出力(0 を設定)するかを選択です。
Option	予約機能 0 を設定してください。

Fstat	<p>波形停止時出力選択</p> <p>0 : 出力波形の停止時レベル</p> <p>1 : 指定停止出力レベル周期波形および FIFO 出力モードで波形を停止(ゲートがディセーブルになった、停止トリガが発行された、トリガ出力で指定回数波形出力を完了した)時に波形メモリの最終値を出し続ける(0 を選択)か、指定の出力値にする(1 を選択)かを選びます。</p> <p>(注) トリガ出力で指定回数の波形を出力している時、最後の波形データを出力した後 1 クロック後に指定停止出力レベルが出力されます。ゲート信号の停止または停止トリガにより停止した場合は停止信号を受け付けた後次のクロックタイミングで指定停止出力レベルが出力されます。波形出力中にモードが変更された場合(例えばゲート出力からトリガ出力に変更)波形は中断された時のレベルが維持され指定停止出力レベルは出力されません。</p>
FVal	<p>波形停止時指定レベル 0~16383</p> <p>Fstat の設定値が 1 の時の指定レベル</p>
RepCyc	<p>周期波形出力でトリガ動作の時の 1 トリガで出力する波形の回数</p> <p>0~65535</p> <p>0 を設定すると制限なく出し続けます。停止トリガで停止できます。</p>

TUSB0214DAM_ExIO_Sel

C 宣言	<pre>unsigned long TUSB0214DAM_ExIO_Sel(const char* SerialNumber, unsigned char ExIOSel0, unsigned char ExIOSel1)</pre>
------	---

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

フロントパネルの EXI/O0 と EXI/O1 の機能を設定します

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
ExIOSe1	入力 0:ゲート/トリガ、I/O 入力 出力 1: High レベル出力(起動後初期値) 2: Low レベル出力 3: ゲート/トリガ出力 4: ch1 周期波形同期出力 5: ch2 周期波形同期出力
ExIOSe1	入力 0:クロック、I/O 入力 出力 1: High レベル出力(起動後初期値) 2: Low レベル出力 3:クロック出力 4: ch1 周期波形同期出力 5: ch2 周期波形同期出力

エラーコード(エラーコード表参照)

TUSB0214DAM_ExIO_SeL_R

C 宣言	unsigned long TUSB0214DAM_ExIO_Set_R(const char* SerialNumber, unsigned char* ExIOSe10, unsigned char* ExIOSe11)
------	--

各言語での呼び出し宣言については付属のライブラリファイルを参照してください

解説

フロントパネルの EXI/00 と EXI/01 に何が設定されているかを読み出します。

引数

SerialNumber	使用する DA コンバータのシリアル番号文字列
SelNum	どちらの EXIO の設定をするか 0:EXI/O0 1:EXI/O1
*EXIOSel	TUSB0214DAM_EXIO_Set の EXIOSel0、EXIOSel1 を参照してください。

エラーコード(エラーコード表参照)

6.5 戻り値(エラーコード)

各機能関数(プロシージャ)から戻る処理結果コード値表です

戻り値	結果
0	正常終了
1	引数エラー
2	デバイスが接続されていない
3	予約
4	指定のデバイスが見つからない
5	USB 通信エラー
6	デバイスは既に接続されている

7. その他動作の補足

7.1 PC スリープについて

PC をスリープ状態にすると DA 変換コントロール回路への電源供給が止まります。
スリープから回復した後は電源起動した後と同じ状態になります。

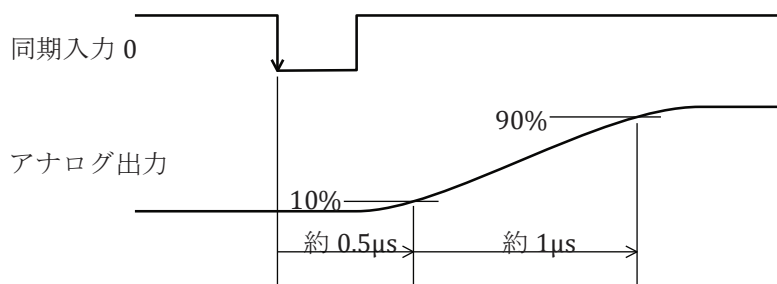
7.2 FIFO オーバーラン、アンダーランの時の動作

FIFO データの空容量より多いデータを書き込むと FIFO バッファオーバーランが発生します。書き込めなかったデータは捨てられて書き込めた分のデータだけが出力されます。オーバーランの発生は TUSB0214DAM_Fifo_Status で確認できます。

FIFO 出力中に FIFO メモリ内の残データが無くなり FIFO 出力に失敗すると FIFO バッファアンダーランが発生します。データは更新されません。新たなデータが書き込まれるまで最後に出力されたレベルが出続けます。アンダーランの発生は TUSB0214DAM_Fifo_Status で確認できます。

7.3 同期入力と出力タイミング

クロック選択を同期入力 0(CLK)としている時、外部入力の立下りに同期して DA コンバータからデータが出力されます。外部入力の立下りから約 500ns 程度後から出力が変化し始め変化量の 10%から 90%までおおよそ 1 μ 秒程度かかります。(セトリグタイム)

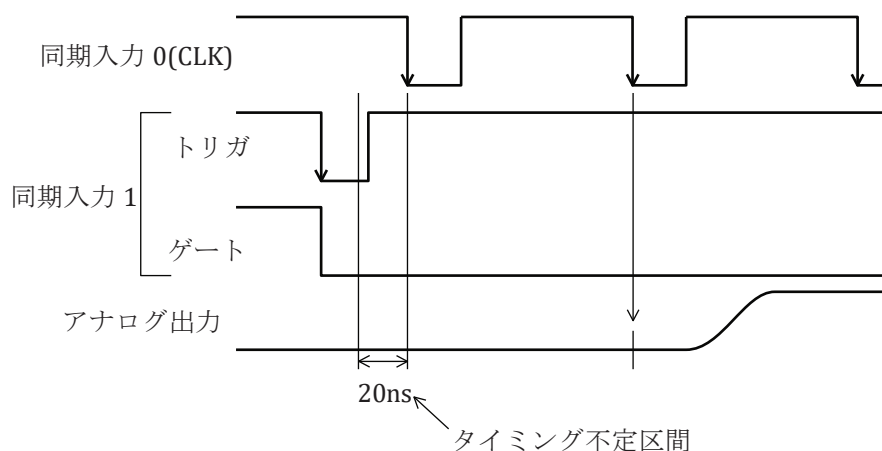


7.4 トリガ、ゲート信号と出力タイミング

○ 同期入力 0 と同期入力 1 併用で動作させている時同期入力 1 のトリガおよびゲート信号の出力開始条件が発生してから同期入力 0 の 2 クロック後に出力が開始されます。ただし、同期入力 0 のクロックの立下り前 20ns 以内にトリガおよびゲート変化を発生させるとタイミング不定となり 2 クロックまたは 3 クロック後に出力開始されます。複数グループを同時に開始するためにはトリガおよびゲートの開始をこの 20ns(タイミング不定区間)で発生させないようにしてください。

- 装置内蔵の外部同期タイマ、トリガ、ゲート信号発生器はタイミング不定区間に開始しない様に出力信号をコントロールしています。
- ゲート入力により停止タイミングも同様です。停止条件成立後 1 クロックは出力動作が継続され、2 クロック後に停止です。

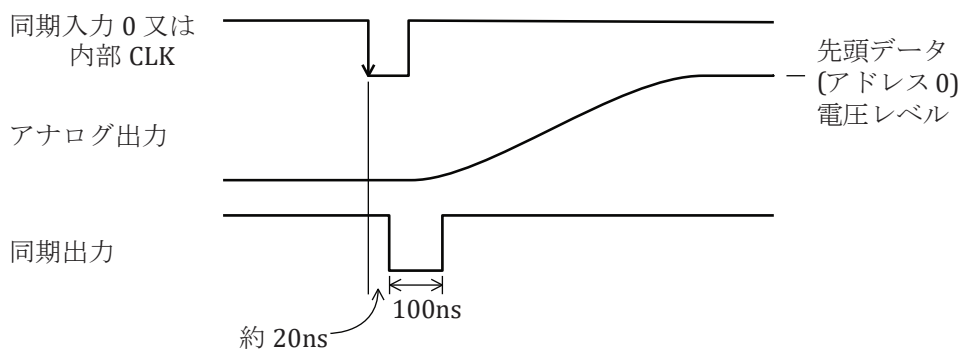
その他動作の補足



○ 内部クロックと同期入力 1 で動作させている時クロックと開始信号(同期入力 1)は同期していないため開始タイミングは曖昧になります。トリガ、ゲート開始条件からおおよそ 1 クロック周期～2 クロック周期で出力が開始されます。ゲート入力の停止時も同様に 1 クロックから 2 クロック周期後に停止されます。

7.5 同期出力のタイミング

Ch1 および Ch2 周期波形同期出力(周期波形の先頭データ出力時にパルスを出す機能)は周期波形メモリのアドレス 0 のデータをセットする時に 100ns 幅のパルスを出力します。



その他動作の補足

8. 仕様

8.1 仕様概要

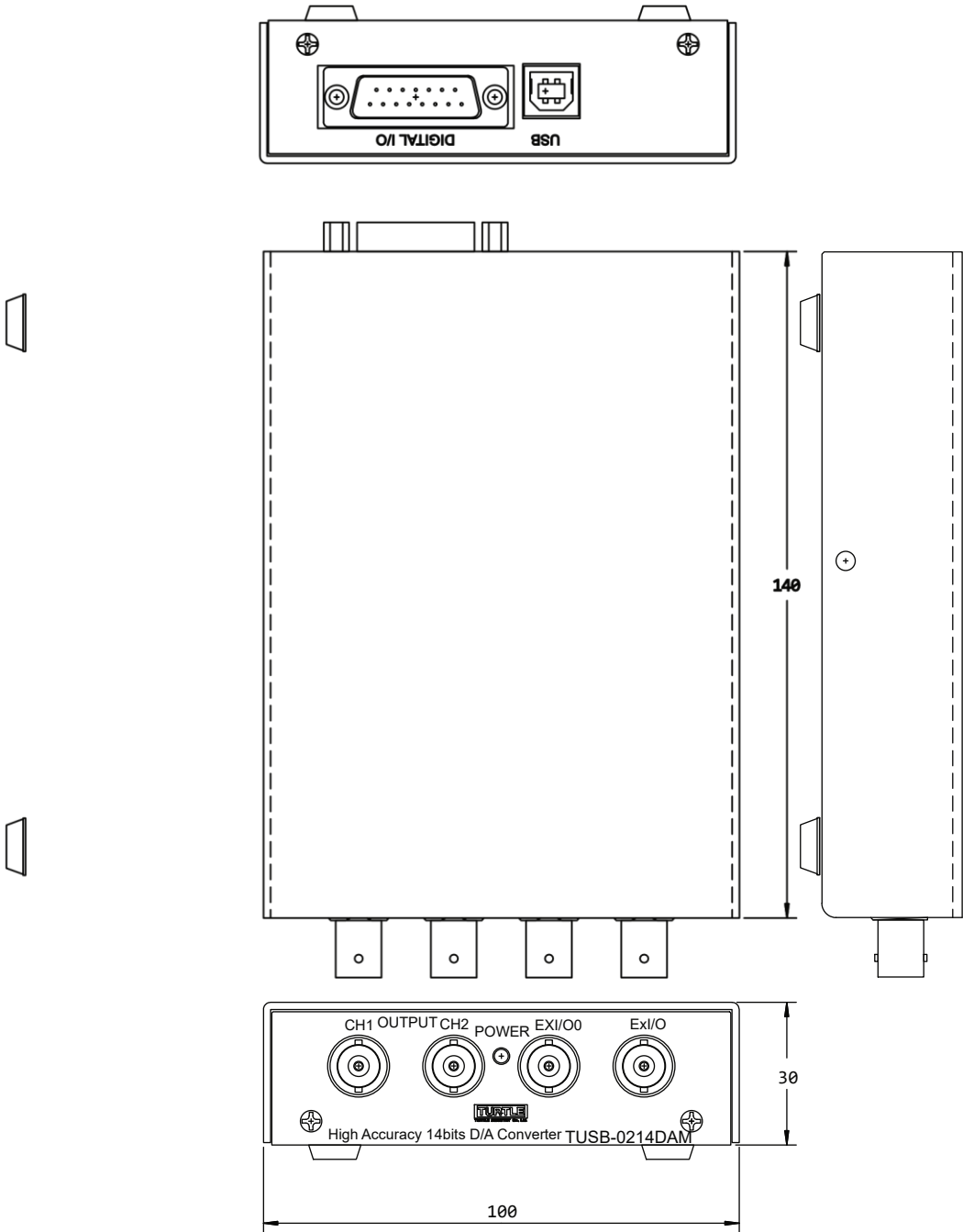
出力チャンネル数	2ch
量子化分解能	14 ビット
非直線性	1LSB 以内
雑音	約 0.25mV(レンジが±10V の時) 約 0.15mV(レンジが±5V、±1.25V、0~+10V、 0~+5V のとき) 約 0.2mV(レンジが±2.5V の時) 約 0.05mV(レンジが 0~1.25V のとき)
オフセット温度ドリフト ※2	0.25LSB/°C
最大出力電圧	±10V
出力レンジ	±10V,±5V,±2.5V,±1.25V,0~10V,0~5V,0~2.5V
出力抵抗	約 50Ω
出力電流	各チャンネル最大±10mA
変換レート	最速 1MSPS
クロックモジュール数	3 個(2 グループ+1 外部同期補助用)
クロック設定周期	最速 1μs (設定分解能 10ns)
メモリ容量	512k ワード×2ch
同期機能	複数台同期機能あり
ディジタル入出力	8 ビット(ビット毎入出力可変)
同期入出力	入力 2 ビット/出力 2 ビット
使用温度範囲	5°C~10°C

電源電圧	5V(USB から供給)
電源電流	最大 500mA
コンピュータ IF	USB-B
外形寸法	30H×100W×140D mm(突起部含まず)
質量	約 370g(本体のみ)

“約”と書かれている数値は製造時検査対象外で試験試作器の実測値

※1 製造時検査対象外で部品の特性から想定される計算値

8.2 寸法



9. お問い合わせ

動作上の問題および不明な点などご使用中のお問い合わせは下記までお願いします。確認の上、ご連絡いたします。弊社 **Web** サイトのお問い合わせフォームをご利用下さい。

ご質問の際には動作環境や不具合状況などできるだけ詳細な情報をお送りください。

- ご使用のコンピュータの機種
- ご使用の OS(Pro/Home の違いやビット数なども)
- メモリの容量
- 本製品以外で使用されている USB 機器
- 回答ご連絡先

株式会社タートル工業サービス係

〒300-0842 茨城県土浦市西根南 1-12-4

FAX 029-843-2024 <https://www.turtle-ind.co.jp/>

support@turtle-ind.co.jp

お問い合わせ

TUSB-0214DAM 取扱説明書 v1

2025 年 7 月 28 日 発行

発行所 株式会社 タートル工業
〒300-0842 茨城県土浦市西根南 1-12-4
<https://www.turtle-ind.co.jp>

本書を無断で複写・複製するを禁じます。

©2025 Turtle Industry Co.,Ltd.