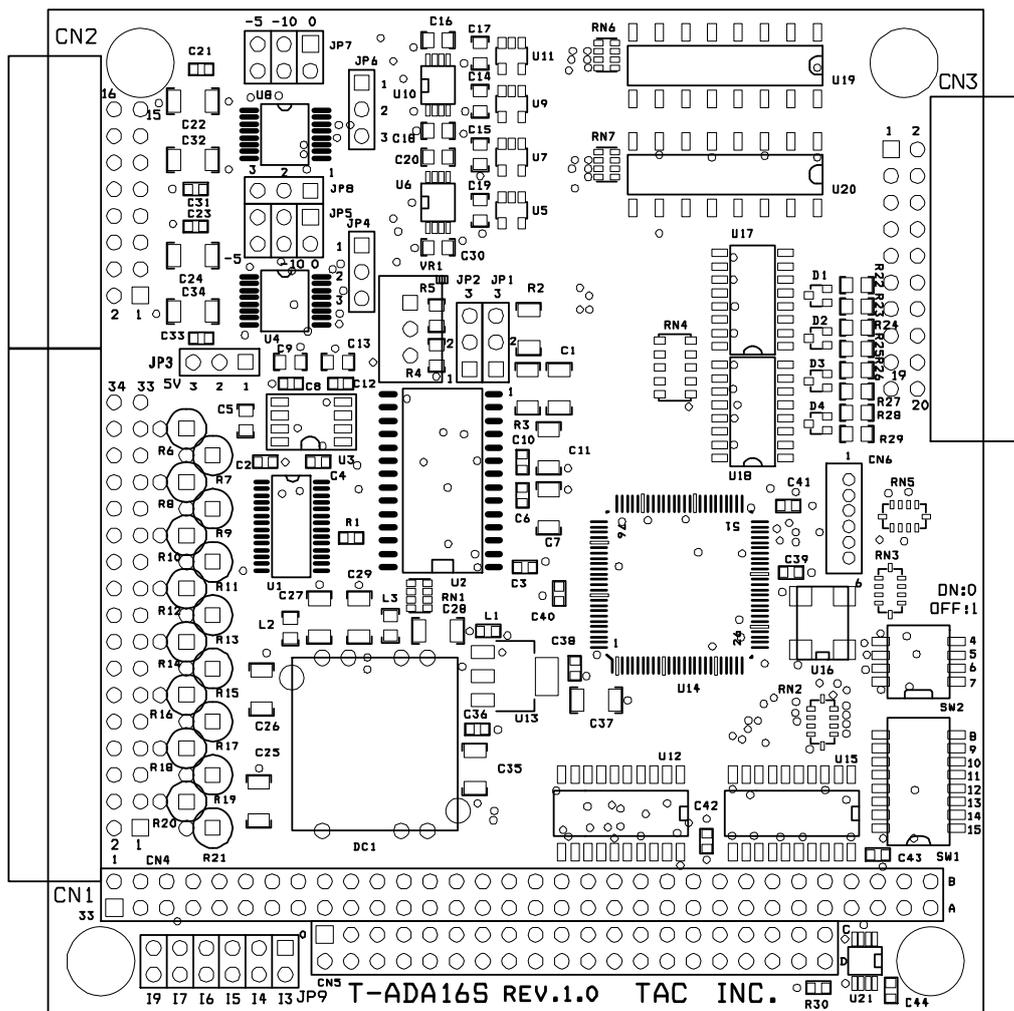


T-ADA16S rev. 1.0

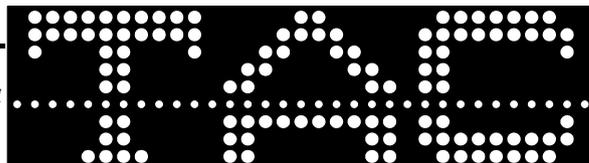
PC/104バスシリーズ 12ビットAD 12ビットDA 絶縁IO 取扱説明書



●各商品は、各社の商標・登録商標です。

●この製品の外観及び仕様は品質改善のため、予告無く変更することがありますのでご了承下さい。

(株)ティーエーシー
各種制御用マイクロコンピュータ
設計・製作・販売



〒600-8896
京都市下京区西七条西石ヶ坪町66
電話:075-311-7307 FAX:075-314-1174
<http://www.tacinc.jp>

はじめに

このたびは、弊社 T-ADA16S をお買い上げ頂きましてまことにありがとうございます。このマニュアルは T-ADA16S の概要等について説明しています。各 LSI についての詳細は必要に応じてデータシートを参照してください。ハードウェアの不具合に関しましてのサポートはいたしますが、RTOS を含めたソフト面のサポートは基本的にはしていません。

【注意事項】

- (1) 本書の内容に関しては将来予告なしに変更することがあります。
- (2) 本書の一部または全部を無断で転載することは禁止されています。
- (3) 本書の内容については万全を期して作成いたしました。万が一不審な点や誤り、記載もれなどお気づきの点がございましたら、お買い求めの販売店、または当社技術部にご遠慮なくお申しつけください。

【本ボードご使用上の注意事項】

- (1) 本ボードは、静電気および衝撃などに十分注意して慎重にお取扱いください。
- (2) 外部入出力電圧、電流は、定格値を越えないよう注意してください。
- (3) コネクタの向き、ピン番号の誤りに注意してください。
- (4) 本ボードの改造及び、その使用にともなった弊害につきましては、当社は一切の責任を負いかねます。

1. 仕様概要

AD 変換 LSI アナログデバイセズ AD1674 非絶縁
スループット 100kSPS (CPU の処理能力に依存します)
シングルエンド 16ch マルチプレクサで切替
入力範囲 0 - 10V, ±5V, ±10V ジャンパーで設定
オプションで 0-5V も可能
オプションでボリュームによる可変ゲイン仕様も可能
変換開始ソフトウェアのみ
変換終了割り込み可能

DA 変換 LSI アナログデバイセズ AD5726 を 2 個使用 合計 8CH (オプション)
基準電圧 アナログデバイセズ ADR01, ADR02 使用
CPLD でパラレル→シリアル変換 27MHz パラレル→シリアル変換時間は 0.59 μsec
出力範囲 4CH 毎に 0 - 5V, 0 - 10V, ±5V, ±10V
電源 ON で出力 0V

絶縁 IO 出力 8 点 入力 8 点 (オプション)
入力フォトカプラ TLP281 または相当品 12V-24V
出力フォトカプラ TLP127 または相当品 最大コレクタ電流 150mA

PC104 バス 8ビット専用

2. IO アドレスの設定

ベース I/O アドレス (以下 BIO) は SW1 と SW2 で設定します。このスイッチは ON で 0, OFF で 1 です。
出荷時は SW1 の 1 番 (A7) と SW2 の 7 番 (A9) と 8 番 (A8) のみ OFF にしていますので、BIO は 380H になります。

アドレス	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4
SW1 番号	1	2	3	4	5	6	7	8				
SW2 番号									1	2	3	4
スイッチ状態	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF	ON	ON	ON

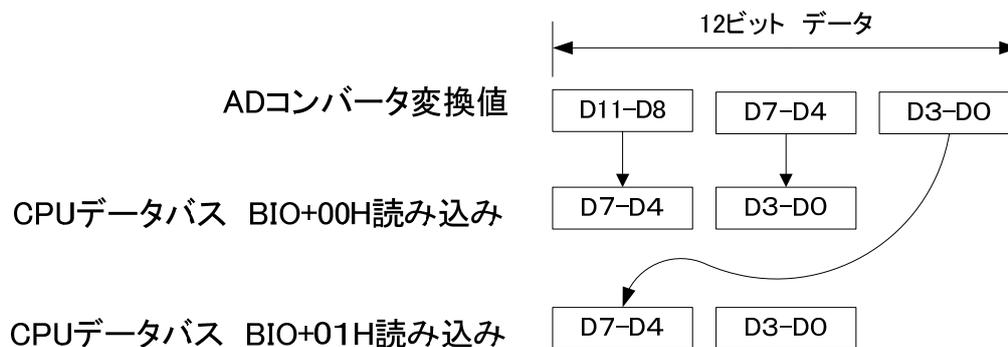
出荷時の状態

4. レジスタ一覧

アドレス	読み込み	書き込み
B10 + 00H	AD 変換データ 上位 8 ビット	書き込み動作で AD 変換開始
B10 + 01H	AD 変換データ 下位 4 ビット	
B10 + 02H	ビット 7 が 1 の時 AD 変換中	AD 入力 CH 選択
B10 + 03H		
B10 + 04H	絶縁 I/O 入力 フォトカプラ ON で 0	
B10 + 05H	絶縁 I/O 出力 1 でフォトカプラ ON	絶縁 I/O 出力 1 でフォトカプラ ON
B10 + 06H		DA の為の平行→シリアル変換 データ 下位 8 ビット データ
B10 + 07H		DA の為の平行→シリアル変換 データ 上位 8 ビット データ
B10 + 08H		書き込み動作で DA CH1-4 データラッチ
B10 + 09H		書き込み動作で DA CH5-8 データラッチ
B10 + 0AH		DA CH1-4 チップセレクト
B10 + 0BH		DA CH5-8 チップセレクト
B10 + 0CH	ビット 7 が 1 の時 平行→シリアル変換中	

5. AD コンバータ

A/DコンバータにはアナログデバイスAD1674を使用しています。変換速度は変換時10 μ secです。12ビットの変換データD0～D11の上位8ビット(D4～D11)がCPUのデータバスD0～D7に出力され、A/Dコンバータの下位4ビット(D0～D3)がCPUのデータバスD4～D7に2度に分けて出力されます。下図参照。したがって、I/OアドレスB10+00Hを読み込めば、A/Dコンバータの上位8ビットがCPUのデータバスD0～D7に、I/OアドレスB10+01Hを読み込めば、下位4ビットデータがCPUのデータバスD4～D7に出力されます。



A/Dの変換開始はOUTオペレーションで行われます。この場合、I/OアドレスB10+00Hに任意のデータを書き込めば変換が開始されます。変換中かどうかはI/OアドレスB10+02Hを読み込み、ビット7がHの時、変換中であることを確認できます。

マルチプレクサのチャンネル選択はI/OアドレスB10+02Hにチャンネル番号を書き込むことで選択出来ます。例えばマルチプレクサのS16のチャンネルを選びたい場合、B10+02Hに0FHを書き込みます。

A/Dコンバータへの入力電圧はユニポーラ0V～10V、またはバイポーラ-5V～+5V、-10V～+10Vの選択が可能です。これらはジャンパーピンの差し替えによって可能です。出荷時はユニポーラで0～10Vの入力範囲になっております。

オプションで0-5V 固定仕様、多回転ボリュームによる可変ゲイン仕様もあります。

AD コンバータ入力範囲 ジャンパー設定

入力範囲	JP1	JP2
0 - 10V	1 - 2	1 - 2
-5 - +5V	2 - 3	1 - 2
-10 - +10V	2 - 3	2 - 3

出荷時は 0-10V です。

AD 入力コネクタ CN1 ピン配列 HIROSE HIF3FC-34PA-2.54DS または互換品

ピン番号	信号名	ピン番号	信号名
1	S1	2	GND
3	S2	4	GND
5	S3	6	GND
7	S4	8	GND
9	S5	10	GND
11	S6	12	GND
13	S7	14	GND
15	S8	16	GND
17	S9	18	GND
19	S10	20	GND
21	S11	22	GND
23	S12	24	GND
25	S13	26	GND
27	S14	28	GND
29	S15	30	GND
31	S16	32	GND
33	+15V	34	-15V

33,34 番ピンは基板上の DCDC コンバータから出ています。

5mA 以上負荷をかけないで下さい。

● AD 変換手順

- ① BIO + 02H に変換するチャンネルを書き込む。例えば S16 の場合は Fh を書き込む。
- ② BIO + 00H に 0 を書き込み AD 変換開始
- ③ BIO + 02H を読み込み ビット 7 が H の時は変換中。
- ④ BIO + 02H のビット 7 が L を確認後、下位 4 ビット (BIO+01H)、上位 8 ビット (BIO+00H) を読み込む

● 変換終了割り込み

A/D の変換終了はソフトだけではなく割り込みでも知ることができます。
CPU 側はレベルではなくエッジで割り込みが発生する設定にしてください。
割り込 CH の設定は JP9 で設定可能です。下表を参照して下さい。

JP9 割り込み 設定

JP9 位置	PC104 バス 割り込み名称
I9	IRQ 9
I7	IRQ 7
I6	IRQ 6
I5	IRQ 5
I4	IRQ 4
I3	IRQ 3

6. DA コンバータ（オプション設定）

DA コンバータにはアナログデバイセズ AD5726 を 2 個使用し合計 8CH としています。
 この DA コンバータはシリアル入力ですが、CPLD でパラレル→シリアル変換しています。
 クロックは 27MHz ですので、パラレル→シリアル変換時間は 0.59 μ sec と高速です。
 外付け基準電圧はアナログデバイセズ ADR01,ADR02 使用しております。
 出力範囲 4CH 毎に 0 - 5V, 0 - 10V, $\pm 5V$, $\pm 10V$ の設定が可能です。
 電源 ON(リセット)で出力を 0V にクリア可能です。

● DA 変換手順 CH1-CH4

- ① BIO + 0AH に 0 を書き込み、CS0 を L に。
- ② BIO + 07H に DA データの上位 4 ビットをビット 3 ~ 0 に、CH 番号 - 1 をビット 7, 6 に書き込む。
- ③ BIO + 06H に DA データの下位 8 ビットを書き込み、パラレル→シリアル変換開始。
- ④ BIO + 0CH のビット 7 が L を確認後、BIO + 0AH に 1 を書き込み、CS0 を H に。
- ⑤ BIO + 08H に 0 を書き込み、ラッチ、これで電圧が出力される

● DA 変換手順 CH5-CH8

- ① BIO + 0BH に 0 を書き込み、CS1 を L に。
- ② BIO + 07H に DA データの上位 4 ビットをビット 3 ~ 0 に、CH 番号 - 1 をビット 7, 6 に書き込む。
- ③ BIO + 06H に DA データの下位 8 ビットを書き込み、パラレル→シリアル変換開始。
- ④ BIO + 0CH のビット 7 が L を確認後、BIO + 0BH に 1 を書き込み、CS1 を H に。
- ⑤ BIO + 09H に 0 を書き込み、ラッチ、これで電圧が出力される

● DA 出力範囲の設定

DA CH1 - CH4 出力範囲設定

入力範囲	JP3 ※1	JP4	JP5
0 - 5V	2 - 3	1 - 2	0V (1 - 2)
0 - 10V	1 - 2	1 - 2	0V (1 - 2)
-5 - +5V	2 - 3	2 - 3	-5V (5 - 6)
-10 - +10V	1 - 2	2 - 3	-10V (4 - 5)

出荷時は 0-10V です。

※1 JP3 の文字が基板上にありません、表紙のシルク図を参照して下さい。

DA CH5 - CH8 出力範囲設定

入力範囲	JP8	JP6	JP7
0 - 5V	2 - 3	1 - 2	0V (1 - 2)
0 - 10V	1 - 2	1 - 2	0V (1 - 2)
-5 - +5V	2 - 3	2 - 3	-5V (5 - 6)
-10 - +10V	1 - 2	2 - 3	-10V (3 - 4)

出荷時は 0-10V です。

● DA 出力コネクタ CN2 ピン配列

DA 出力コネクタ CN2 ピン配列 HIROSE HIF3FC-16PA-2.54DS または互換品

ピン番号	信号名	ピン番号	信号名
1	DA CH1	2	GND
3	DA CH2	4	GND
5	DA CH3	6	GND
7	DA CH4	8	GND
9	DA CH5	10	GND
11	DA CH6	12	GND
13	DA CH7	14	GND
15	DA CH8	16	GND

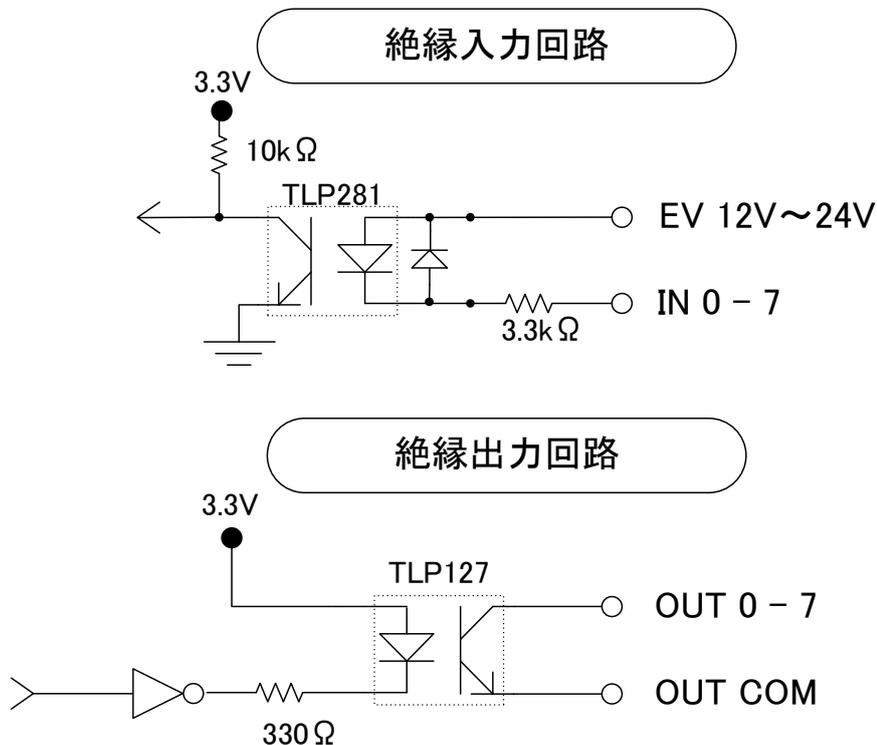
7. 汎用 絶縁入出力 (オプション設定)

入力 8 点、出力 8 点のフォトカプラ絶縁入出力をオプションで追加可能です。

入力フォトカプラは TLP281 または相当品を使用しております。電圧は 12V~24V に対応した抵抗値に設定しています。出力フォトカプラは TLP127 または相当品を使用しております。最大コレクタ電流 150mA です。コレクタ・エミッタ間飽和電圧が最大 1.2V(Ic=100mA) ありますので受け側の L の範囲に注意が必要です。

入力は BIO + 04H を読み込むことで状態を知ることが可能です。フォトカプラ ON で 0 です。ビット 7~0 が IN7~0 に対応します。アノード共通です。

出力は BIO + 05H にデータを書き込むことでフォトカプラを ON/OFF することが可能です。1 でフォトカプラ ON です。ビット 7~0 が OUT7~0 に対応します。エミッタ共通です。



CN3 ピン配列 使用コネクタ ヒロセ電機 HIF3FC-20PA-2.54DS または相当品

ピン番号	信号名	ピン番号	信号名
1	OUT COM	2	OUT COM
3	OUT 0 (ビット 0)	4	OUT 1 (ビット 1)
5	OUT 2 (ビット 2)	6	OUT 3 (ビット 3)
7	OUT 4 (ビット 4)	8	OUT 5 (ビット 5)
9	OUT 6 (ビット 6)	10	OUT 7 (ビット 7)
11	EV 12 - 24 V	12	EV 12 - 24 V
13	IN 0	14	IN 1
15	IN 2	16	IN 3
17	IN 4	18	IN 5
19	IN 6	20	IN 7

8. サンプルプログラム

OS : MES 2.6

CPU : T-SH2MB

ソフトウェア変換、割り込み未使用。

```
#include <mes2.h>
#include <h8/reg704x.h>

//T-ADA16S BASE 0x380
#define ADU (*(volatile unsigned char *)0xc00380)
#define ADL (*(volatile unsigned char *)0xc00381)
#define ADSTS (*(volatile unsigned char *)0xc00382)

#define DA_D_L (*(volatile unsigned char *)0xc00386)
#define DA_D_H (*(volatile unsigned char *)0xc00387)
#define LDAC_0 (*(volatile unsigned char *)0xc00388)
#define LDAC_1 (*(volatile unsigned char *)0xc00389)
#define DA_CS0 (*(volatile unsigned char *)0xc0038a)
#define DA_CS1 (*(volatile unsigned char *)0xc0038b)
#define SPL_STS (*(volatile unsigned char *)0xc0038c)

#define ISO_IN (*(volatile unsigned char *)0xc00384)
#define ISO_OUT (*(volatile unsigned char *)0xc00385)

void set_da(int ch,int data){
    if(ch < 4) DA_CS0 = 0; //cs0:L
    else      DA_CS1 = 0; //cs1:L

    DA_D_H = ((ch << 6) & 0xc0) | ((data >> 8) & 0xf);
    DA_D_L = data & 0xff; //start spi send
    while(SPL_STS & 0x80) ; //0.59usec

    if(ch < 4){
        DA_CS0 = 1; //cs0:H
        LDAC_0 = 0; //latch
    }
    else{
        DA_CS1 = 1; //cs1:H
        LDAC_1 = 0; //latch
    }
}
```

```

int addata(int a) //0 - 15
{
    ADSTS = a; /* select ch. */
    ADSTS = a; /* select ch. */
    ADSTS = a; /* select ch. */
    ADU = 0; //Start ADconvert
    while((ADSTS & 0x80){ ; } /* converting */
    return( (ADU << 4) | ( (ADL >> 4) & 0xf) );
}

void adin(void){
    int ch;
    unsigned char c;

    while( !read(0, &c, 1) ){
        for(ch = 0 ; ch < 16 ; ch ++ ){
            // printf("%2.3f ",(float)addata(ch) * 10 / 4096 );
            printf("%03x ",addata(ch));
        }
        printf("%02x ",ISO_IN);
        printf("¥r");

        for(ch = 0 ; ch < 8 ; ch ++ ){
            set_da(ch,addata(ch));
        }
        ISO_OUT = ISO_IN;

        sleep(20);
    }
}

void adin2(void){
    int ch;
    unsigned char c;

    while( !read(0, &c, 1) ){
        printf("%04d ",addata(0));
        printf("¥r");
    }
}

void test_da(void){
    int ch;
    int data;
    int i;
    unsigned char c;

    //while(1){
    while( !read(0, &c, 1) ){
        for(data = 0 ; data < 0x1000 ; data ++ ){
            for(i=0;i<8;i++){
                set_da(i,data);
            }
            //set_da(0,data);
            /*
            set_da(1,data / 2);
            set_da(2,data / 3);
            set_da(3,data / 4);
            set_da(4,data / 5);
            set_da(5,data / 6);
            set_da(6,data / 7);
            set_da(7,data / 8);
            */
        }
    }
}

```

```

void test_da2(void){
    int data = 0xffff;

    set_da(0,(data * 1) / 8 );
    set_da(1,(data * 2) / 8 );
    set_da(2,(data * 3) / 8 );
    set_da(3,(data * 4) / 8 );
    set_da(4,(data * 5) / 8 );
    set_da(5,(data * 6) / 8 );
    set_da(6,(data * 7) / 8 );
    set_da(7,(data * 8) / 8 );
}
int main(int argc, char **argv) {
    int    year, month, week, day, hour, min, sec;
    int    i, value;
    int n;
    unsigned char c;

    PAIORH |= 0x30; //PA21,PA20:out

    BCR1 &= ~0x8; //A3SZ = 0 CS3:8bitbus for PC/104
    PACRL2 = ((PACRL2 & 0x3fff) | 0x8000) ; //PA7MD = 2 CS3 出力 PIN

    DA_CS0 = 1; //DA CS:H
    DA_CS1 = 1; //DA CS:H

    adin();
    adin2();
    test_da();
    //test_da2();
}

```

改訂履歴

2011/10/17 初版