

## SONYインフォリチウム 電池マネージャークिट

残容量・電圧・電流・内部温度を  
リアルタイムで液晶表示。  
PICマイコンを使用し電池内部から  
データを読み取ります。



# SONY インフォ リチウム infoLITHIUM 対応

## リチウム電池 マネージメントキット

残容量・電圧・電流・内部温度をリアルタイムで液晶表示

1370mAh	+0000mA
7812mV	23.5C

infoLITHIUM (インフォリチウム) はソニー株式会社の商標です。

- PICマイコンを使用し、インフォリチウム電池のデータ端子から電池内部のマイコンチップから、電池残量・現在の電流（放電電流、充電電流）・現在の電圧・内部温度をリアルタイムで16文字×2行液晶に表示します。
- パソコン通信機能付で表示データをそのままWINDOWS95, 98パソコンに表示・データファイル化することができます。
- 電流は充電時を+で、放電時を-で表示しますので、充放電のデータを取ることができます。
- 調整やめんどうな操作は一切有りません。電池を接続基板に差し込むだけで表示します
- キットの消費電流は10mA以下と小さく、電源不要でリチウム電池だけで表示します

### ■部品リスト■

品名	数	備考
AKI-PICユニバーサルボードキット	1	(リチウムMAプログラム書込済み)
AKI-PICキャリアボードキット	1	(S1~3, R1~6, LED1~3は使用しませんのでキットに入っていない場合があります)
B型ユニバーサル基板	1	
抵抗 100KΩ (茶黒黄金)	1	
みのむしクリップ	2	
14ピンピンヘッダ	1	(20ピンの場合有り)
14ピンピンフレーム	1	(20ピンの場合有り)

★注意 +-の電極のみで、データ接続電極の無い電池には、対応していません。

■製作■

①AKI-PICユニバーサルボードキット  
 キットの取説にしたがい、製作してください。  
 J1、J2はそれぞれを接続し、RS232C  
 通信が出来るようにしてください。

②AKI-PICキャリーボードキット  
 1、2ページの製作により、各部品を取り付けて  
 ください。  
 ただしS1~3、R1~6、LED1~3は  
 取り付けません。  
 VR1は時計方向回しておいてください。

2、図1にしたがい、ピンヘッダ、抵抗100kを  
 半田付けしてください。

3、CN1の4-6、7-8を基板上で、ビニ  
 ル線等で接続してください。

4、図2にしたがい半田面側で、CN2-2、1  
 00KΩ、ピンヘッダ7番目をビニル線ま  
 たは、メッキ線等で接続してください。

5、ICソケットにAKI-PICユニバーサル  
 ボードキットを差し込んでこの基板完成です。

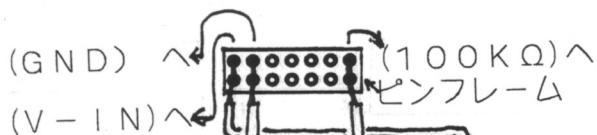
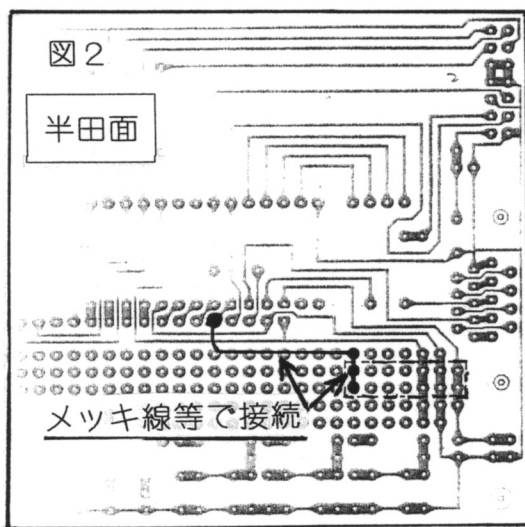
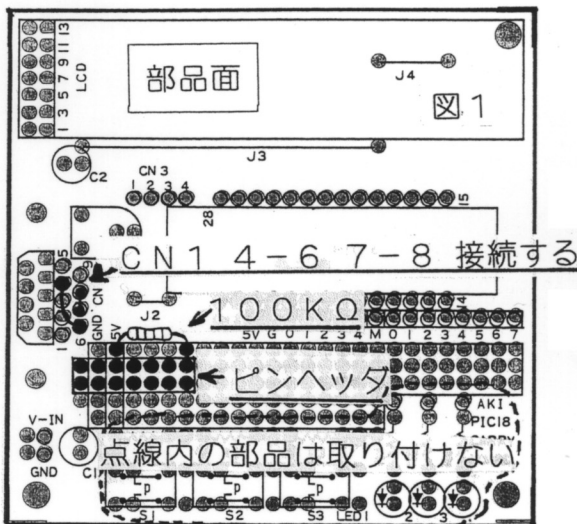


図3

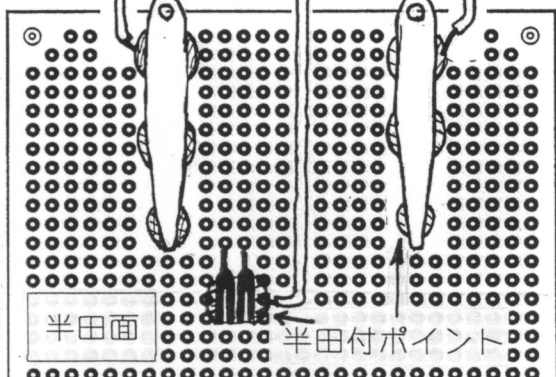


図4 半田付ポイント

ぬいたピン(2本)

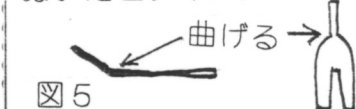


図5

③電池接続基板の製作

1、みのむしクリップのカバーをはずし、あらか  
 じめ、電池電極に入る様に、先をつぶし(図  
 4)、図3の位置に半田付けしてください。  
 基板はパターン(銅箔のある面)に半田付け  
 してください。  
 クリップはかみ合わせが半田付けされない様  
 ボール紙等をはさんで半田付けしてください。

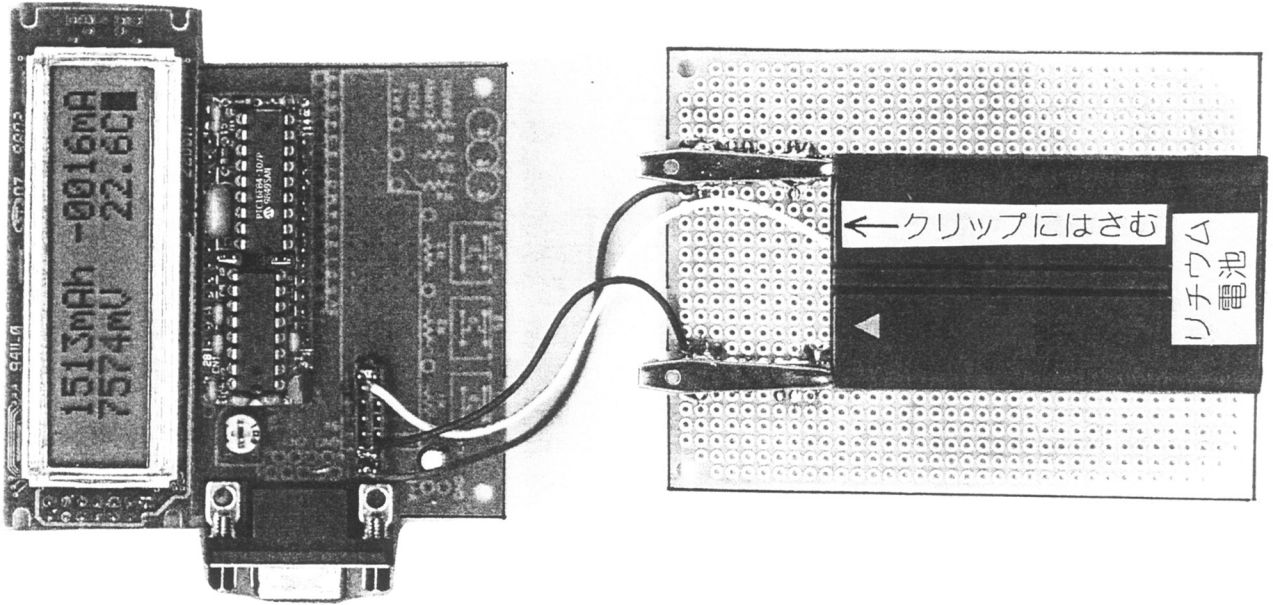
2、ピンフレームは、接続に使用しないピンをラ  
 ジオペンチなどでぬいてください。  
 抜いたピンを図5の様に曲げ、図3の位置に  
 半田付けしてください

3、ビニル線で接続基板とピンフレームを接続  
 してください。(図3)  
 完成した接続基板をキャリーボードのピンヘ  
 ッダに接続して、製作完了です。

## ■電池の接続■

あらかじめ充電されている電池をクリップではさむように電池接続基板に取り付けてください。

★注意 キットの電源は電池からとりますので、放電された電池や電圧が下がっている電池を接続した場合はなにも表示されません。その場合充電してから接続ください



## ■液晶表示の説明■

電池を接続するとはじめにバージョンが表示され、データ表示を行いません。データ表示は電池からの信号により、逐次最新データにかわります。

### 表示例

①電池残量	→	1370mAh	-0000mA	←	③電流
②電池電圧	→	7620mV	21.2C	←	④電池内部温度

### ①電池残量 (mAh) 表示

現在の電池の残量をmAhで表示します。

電池を使用しながらキットに接続した場合は表示数値が減っていき、充電しながら接続した場合は増えていきます。

Infolithiumバッテリーに対応していない機器で使用した場合などで電池残量が正しく表示されない場合があります。

その場合はInfolithiumバッテリー対応機種に取り付け、一度使いきってから満充電すると、正しく表示されます。

### ②電池電圧 (mV) 表示

現在の電池の電圧ををmVで表示します。

### ③電池電流 (mA) 表示

充電電流は+、電池からの放電電流は-で表示されます。

キットに接続しているだけの時は充電、放電ともありませんので、ほぼ0の値を表示します。

このキットでは±30mVのオフセット誤差（電流値をデジタル表示に変換するときの誤差）があり、キットに接続しているだけの状態で±30mV以内の表示をすることがあります。また、放電時±30mV以下の変化が表示にあらわれないことがあります

### ④電池内部温度 (C) 表示

電池内部のセンサーにより内部温度が表示されます。

## ■ W I N D O W S 9 5 , 9 8 接 続 ■

このキットは表示しているデータをリアルタイムで W I N D O W S パソコンに表示することが、出来ます。( C N 1 4-6 7-8 が接続してあることを確認してください。 ) このデータを利用して放電特性などのカーブを作ることができます。

### 1、 接 続

C N 1 から W I N D O W S パソコンの C O M ポート ( R S 2 3 2 C ) にストレートケーブルで接続してください。

### 2、 パソコンソフト

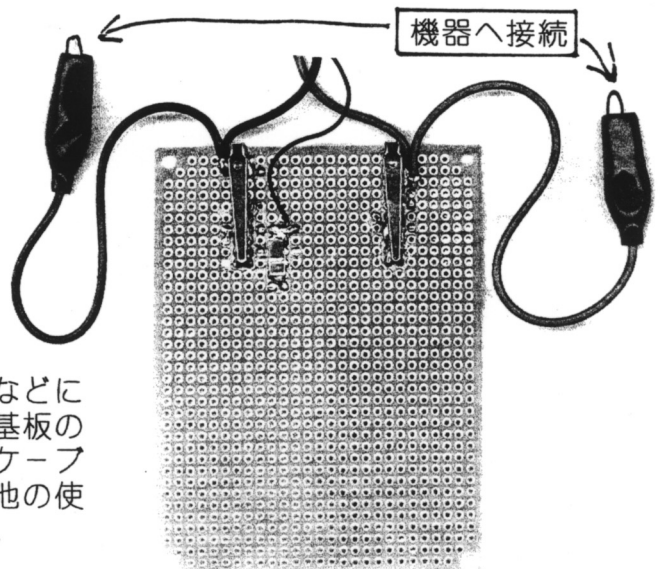
ソフトは W I N D O W S 付属のハイパーターミナルを使用します。ハイパーターミナルが入っていない場合はあらかじめ W I N D O W S で、ハイパーターミナルをインストールし、使用できるようにしてください。 W T E R M 等一般の D O S 版通信ソフトでも同様にしうできます。

### 3、 通信設定

○接続方法	C O M 1 へダイレクト
○モデムの設定	
ビット/秒	1 9 2 0 0
データビット	1
パリティ	なし
ストップビット	1
フロー制御	なし
エミュレーション	V T 1 0 0
○ A S C I I 設定	
受信データに改行文字を付けるを有効にする (チェックする)	

### 4、 通信間隔 約 3 . 5 秒間隔で容量、電流、電圧、温度の順で表示します。

表示例  
1 5 1 3 m A h  
- 0 0 1 6 m V  
7 6 0 2 m V  
2 2 . 1 C  
以下同様につづく



## ■ 応 用 ■

I n f o L I T H I U M 電池をキットなどに応用する場合、右図のように電池接続基板のクリップの所に別のクリップを付けたケーブルを取り付けると簡単に接続でき、電池の使用状態をリアルタイムで監視できます。

I n f o L I T H I U M 電池マネージメントキット 秋月電子通商 KAKE 1999/3  
お問い合わせは往復はがきまたは返信用切手同封の封書でお願いいたします。  
電話・ファックス・Eメールでのお問い合わせは受け付けておりません。  
☎ 1 5 8 - 0 0 9 5 東京都世田谷区瀬田 5 - 3 5 - 6 秋月電子通商 質問係宛

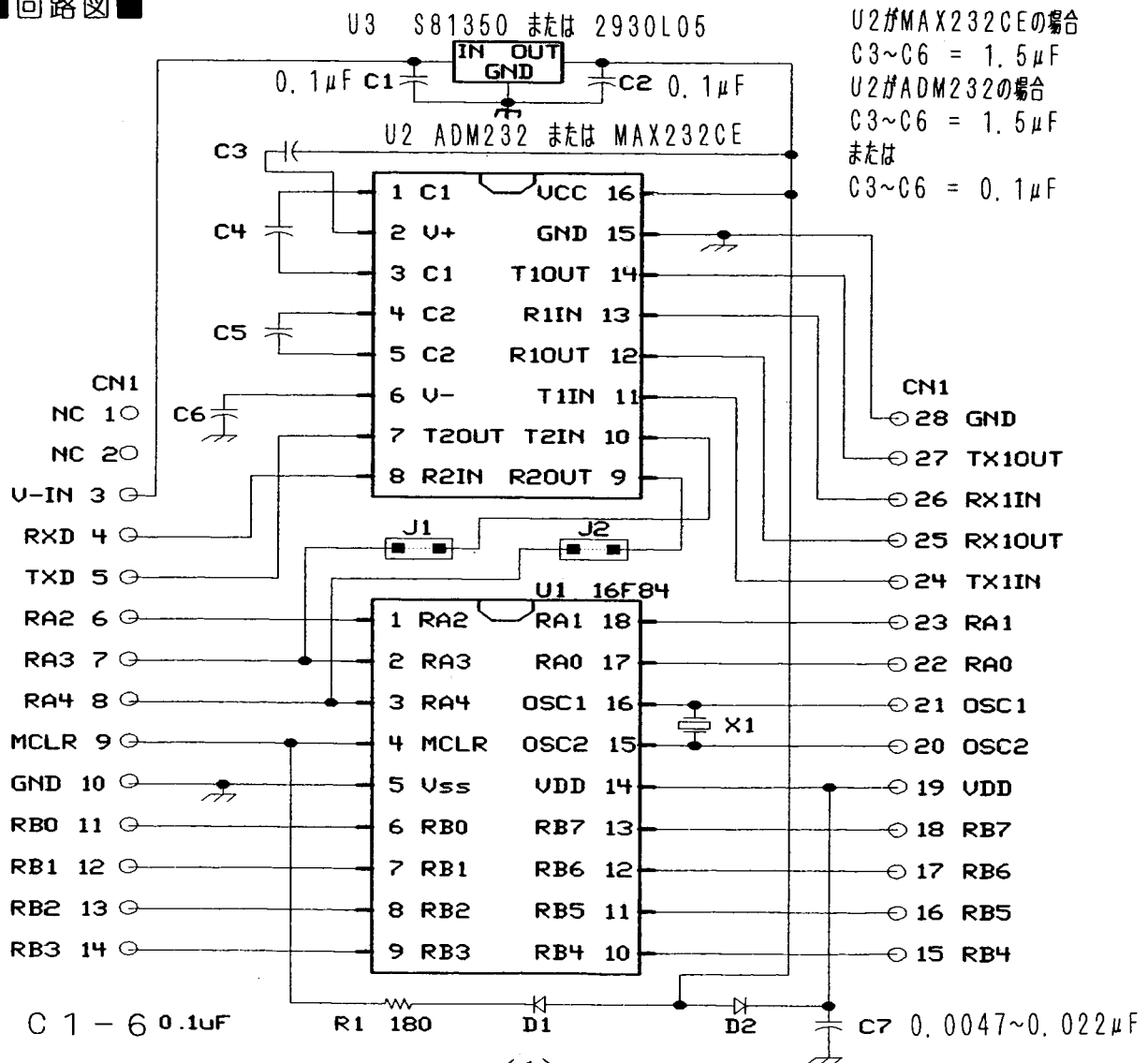
# AKI-PICユニバーサルボード

## 51×18mm超小型マイコンボード

### RS232搭載, 10MHz動作

- ご好評のPIC16F84が超小型マイコンボードになりました。
- プログラム書き込みは、この基板のままAKI-PICプログラマーの28ピンソケットで出来ますので、1つのモジュールとして、取り扱う事ができます
- PIC16F84はEEPROMですので、1000回以上の書き替えができてプログラム開発が容易にできます。
- RS232用ICを搭載していますので、パソコン等とのデータ通信などがこの基板のみで行なえます。
- 5VレギュレータIC, セラミック発振子が付いており、この基板に電源を接続するだけで動作します。

#### ■回路図■



■ 部品表 ■

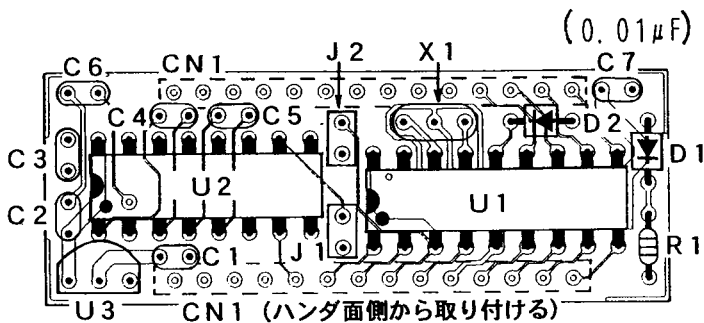
番号	名称	数	備考
C 1 - 2	コンデンサ 0.1 $\mu$ F	2	積層セラミックコンデンサ (104)
C 3 - 6	コンデンサ 0.1 $\mu$ F または 1.5 $\mu$ F	4	積層セラミックコンデンサ (104) または (155)
C 7	0.0047~0.022 $\mu$ F	1	コンデンサ (472~223)
D 1, 2	ダイオード	2	ショットキーダイオード
R 1	抵抗 180 $\Omega$	1	1/6W 茶灰茶金
U 1	16F84	1	サンプルソフト書き込み済み
U 2	ADN232 または MAX232CE	1	232レベルコンバータ
U 3	S81350 または 2930L05	1	低ドロップ5Vレギュレータ
X 1	セラロック	1	セラミック発振子 10MHz (コンデンサ内蔵)
基板	AE-PIG18	1	両面スルーホール基板
	連結ソケット	1	28PIN用
	ピンヘッダ		J 1、J 2用

■ 製作 ■

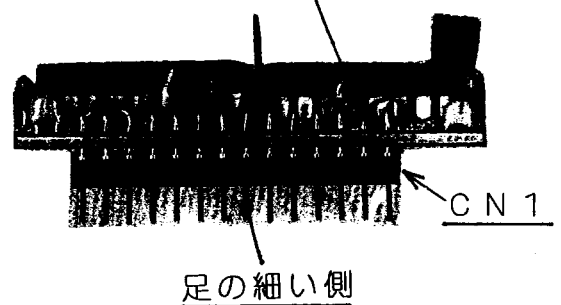
あらかじめ部品表と部品をてらしあわせ、数等をチェックしてから製作してください。連結ソケット以外の部品は部品面側（白い印刷のある面）にとりつけます。

- ① コンデンサ、抵抗、ダイオードの順で半田付けしていきます。  
コンデンサの足が5mmピッチになっている場合はあらかじめ、2.5mmピッチになるようにピンセット等で加工してください。  
ダイオードには極性がありますので基板印刷のマークにあわせてください。
- ② セラロック、ICをとります。  
セラロックは3本足のまんなかたがコモン端子ですので、向きはどちらの向きでも同じです。  
ICの取り付け方向は部品配置図を参考に付けてください。
- ③ J 1、J 2にピンヘッダをつけます。  
ピンヘッダは、あらかじめ2Pずつに切り離してください。  
ピンヘッダは取り付け穴の直径がぎりぎりの太さのため、入れにくい場合があります。その場合トントンとたたいて、入れてください。
- ④ ここまで半田付けを終了したところで、ハンダ面側のリードをニッパーで切って短くしてください。  
連結ソケットはハンダ面側から取り付けます。連結ソケットは、良く見ると足の太い側と細い側があります。太い側を基板にさして部品面側から半田付けしてください。  
細い側はICソケットに入る太さです。

■ 部品配置図 ■



部品面側から半田付け



♪ コーヒーブレイク 積層セラミックコンデンサ (Z品) は使用時間数千時間で容量が数十%減る場合があります。 (2)

■ CN1 接続表 ■

CN1	機能	CN1	機能
1	NC (無接続)	28	GND
2	NC (無接続)	27	TX1 (232)
3	V-IN (電源+)	26	RX1 (232)
4	RXD2 (232)	25	RX1 (TTL)
5	TXD2 (232)	24	TX1 (TTL)
6	RA2	23	RA1
7	RA3 (TXD2)	22	RA0
8	RA4 (RXD2)	21	OSC1
9	MCLR	20	OSC2
10	GND	19	VDD (5VOUT)
11	RBO	18	RB7
12	RB1	17	RB6
13	RB2	16	RB5
14	RB3	15	RB4

■ 回路の説明 ■ 回路図を参考にお読みください。

① CN1

外部に接続するコネクタです。28ピンICの形をしています。それぞれの機能はCN1接続表をごらんください。

② 16F84

16F84は全ピンがCN1に接続されています。OSC1, OSC2, MCLRは基板内で各部品に接続されています。RA3, RA4はJ1, J2で232に接続出来るようになっています。接続せずにそのまま通常のI/Oとして、使用する事もできます。RA0-2, RB0-7はそのままCN1に接続されています。

③ 電源

電源はCN1-3がV-IN, CN1-28がGNDです。電源電圧入力は5~12Vです。レギュレータS81350は自己のドロップが0.03Vと小さい為システム全体が5Vの場合でも、そのままV-INに5Vを入力して動作します。

④ 発振子

10MHzセラミックがついています。コンデンサ内蔵ですので、このままで10MHz動作します。

⑤ RS232

RS232は送信2CH、受信2CH分あります。1CH分はJ1, J2で16F84のRA3, RA4に接続できるようになっています。他の1CH分は入出力とも、CN1に出ていますので必要にあわせて、接続してください。

⑥ リセット

MCLRピンが抵抗でプルアップされ、電源リセットが働くようになっています。外部にリセット回路を接続することもできます。

■ PIC16F84の書き込み ■

プログラム書き込みは、この基板のままAKI-PICプログラマーキットの28ピンソケットで16F84として書き込みが出来ます。

● 書き込み時の注意

- ① J1, J2は書き込み時はオープンにしてください。接続したままですと書き込みできません。ユーザーソフト実行時にショートピン等で接続してください。
- ② 基板のサイズは28ピンより大きいですが、ライターソケットのレバーをよけるように反対側によせれば、そのまま書き込みソケットに入ります。
- ③ 16F84以外のPICマイコンを使用する場合は、基板での書き込みは出来ませんのであらかじめ書き込み済みのPICマイコンを取り付けてください。

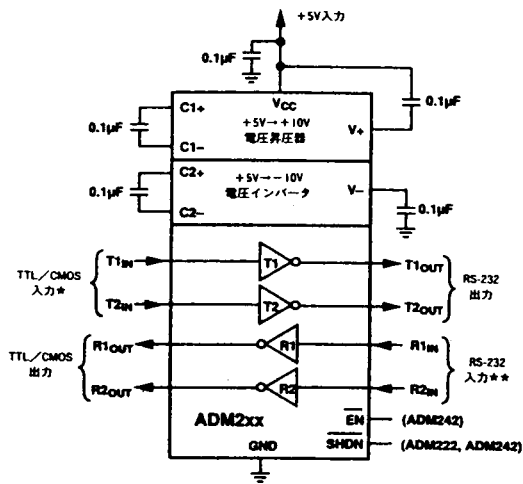




# ADM232AAN (MAX232CE)

## 高速、+5V、0.1 $\mu$ F CMOS RS-232ドライバ/レシーバ

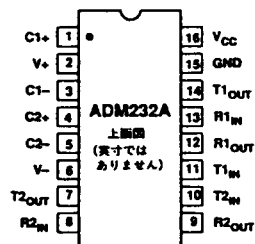
機能ブロック図



\*各TTL/CMOS入力上に400k $\Omega$ のプルアップ抵抗を内蔵  
\*\*各RS-232入力上に5k $\Omega$ のプルダウン抵抗を内蔵

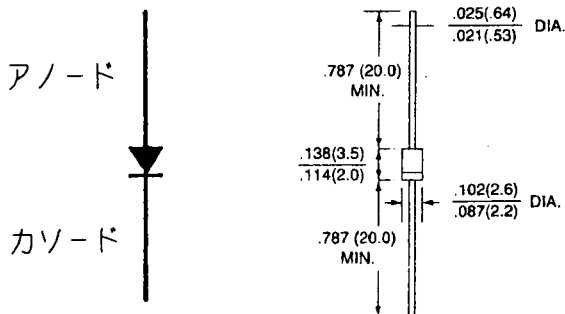
特長

- 200kB/秒の転送レート
- 小容量 (0.1 $\mu$ F) 値のチャージ・ポンプ用コンデンサ
- +5V単一電源動作 MAX232CEの場合1~1.5 $\mu$ F
- EIA-232-EおよびV.28規格に適合
- 2個のドライバと2個のレシーバ
- DC-DCコンバータを内蔵
- +5V電源で $\pm$ 9Vの出力幅
- $\pm$ 30Vのレシーバ入力レベル
- MAX222/MAX232A/MAX242とピン・コンパチブル



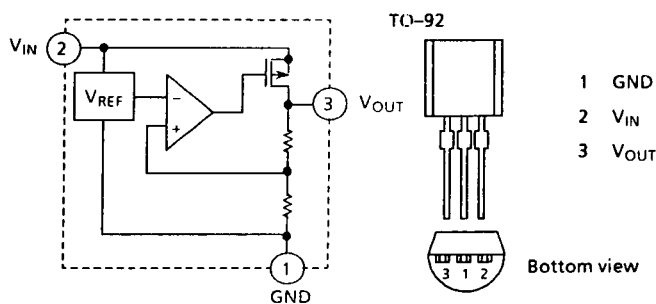
### 1S10 ショットキーダイオード

VOLTAGE 100 Volts  
CURRENT - 1.0 Ampere



### 高精度ボルテージレギュレータ S81350HG

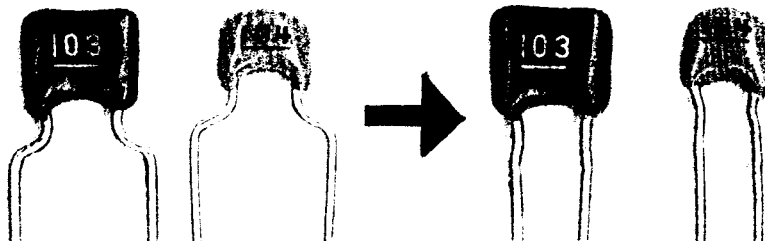
入出力電圧差が小さい (2930L05)  
S-81350HG: 0.12 V typ. I<sub>OUT</sub> = 40 mA



### 積層セラミックコンデンサ

0.1 $\mu$ F (表示104)  
0.01 $\mu$ F (表示103)

あらかじめリードを真っすぐ  
にのばし、2.5mmピッチ  
になるようにする。



### AKI-PI-Cユニバーサルボードキット 製作技術マニュアル

お問い合わせは往復はがきまたは返信用切手同封の封書にてお願いいたします。  
電話、ファックス、E-mailでのお問い合わせは受け付けておりません。  
当社ホームページに新製品情報、バージョンアップ情報等が掲載されることがご  
ざいます。ぜひご覧ください。(URL) <http://www.tomakomai.or.jp/akizuki>  
☎158-0095 東京都世田谷区瀬田5-35-6 秋月電子通商

## ■ サンプルプログラム ■

キット付属のPIC16F84にはサンプルプログラムが書き込まれています。  
 (AKIプラチナキットの付属16F84には、書き込まれていません。)  
 このプログラムで、パソコンからのRS232C信号をパラレルに変換することができます。

PIC16F84は、EEPROMタイプですので、このプログラムを使用せず  
 ユーザーが開発したプログラムを書き込んで、使用することができます。

### ① サンプルプログラム概要

RS232Cから1バイトデータを受信し、その内容をBポートに出力し、同  
 じ内容をRS232Cに送信します。

RS232Cの受信、送信はサブルーチン化されていますので、ユーザープロ  
 グラム等にご活用ください。

### ② 通信フォーマット

9600bps, 8ビット、ストップビット1

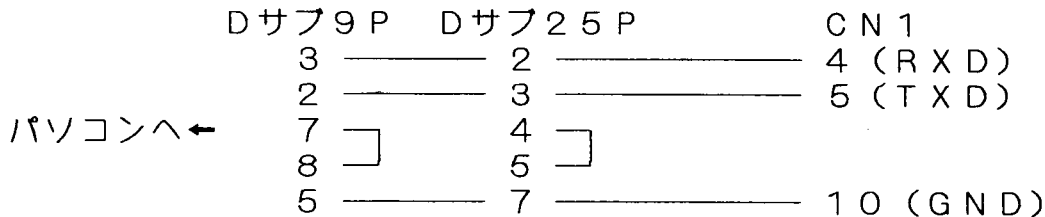
パリティ無し、フロー制御なし

フロー制御なしで1バイトずつ、受信送信しますので連続したデータを取るこ  
 とはできません。1バイトずつパソコンから送信してください。

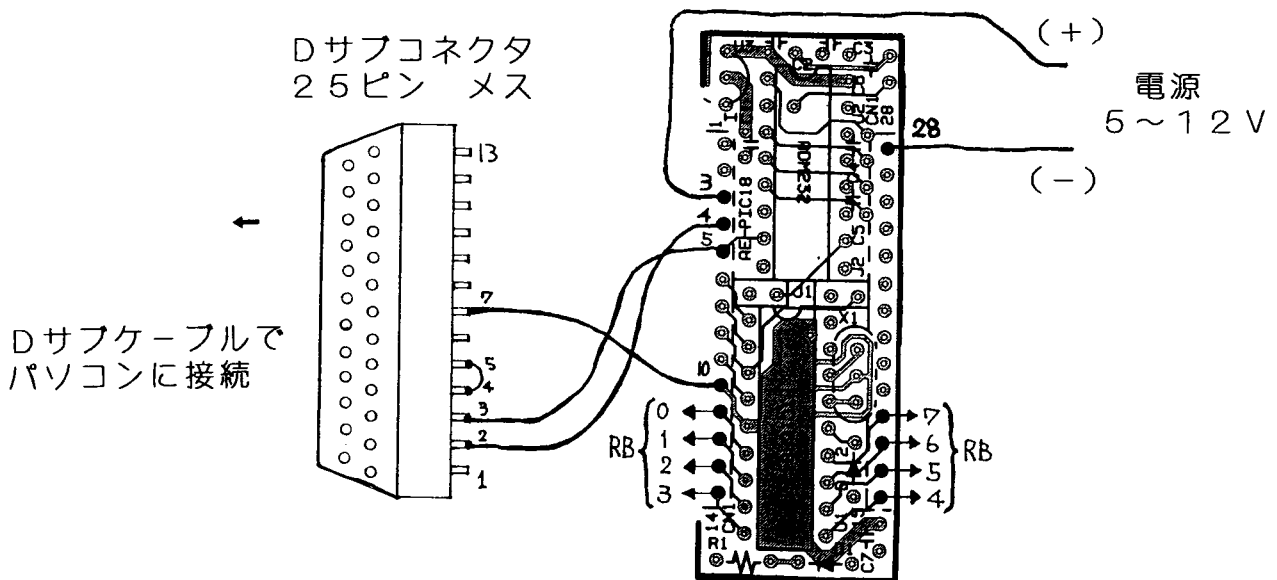
### ③ パソコンとの接続

パソコン側ソフトは、一般の通信ソフト(WTERM等)やWINDOWSの  
 ハイパーターミナル等がご使用になれます。

Dサブコネクタとパソコン間は通常のストレートケーブルで接続してください。



### ★ 接続図



### ④ 動作

パソコンから、1バイト(例'1')を送信するとBポートに00110001  
 (31H)を出力し、パソコンに同じ文字('1')を返します。

Bポートの値は、次のデータが受信されるまで保持されます。

最初の1バイト目はBポート出力、RS232Cの送信が文字化けすることが  
 あります。

(サンプル1)

## ■ サンプルプログラムソースリスト ■

PICプログラマキット用ですので、キット付属のP A. E X E でアセンブルしてください。

RS232C送受信サンプルプログラム  
AKI-PIC18基板用  
For PIC16F84

送信・受信フォーマット：  
9600bps, 8ビット, 1ストップビット  
パリティ無し, フロー制御なし

送信時：  
chに送信データをセットして  
transmitをcallしてください。

受信時：  
receiveをcallしてください。  
データが受信できるまでこのルーチンから戻ってきません。  
受信したデータはchに格納されます。

フロー制御はありません。又、受信割り込み等ありません。  
場合によっては、データの取りこぼしが発生することがあります。  
通信ポーレートはbltimeの値を変えることで変更できます。  
又、動作周波数でも異なります。

$$bltime = \{ (動作周波数(Hz) \div 転送スピード(bps) \div 4) - 10 \} \div 3$$

bltimeは四捨五入してください。動作周波数にもよりますが、あまり高速すると、ビット時間の誤差が大きくなり、エラーの原因になります。  
このプログラムでは、受信したデータをBポートに出力し、その後同じデータをRS232に送信します。  
フロー制御を行っていないので、1バイトづつしか、受信できません。

```

include 16f84.h

    .osc    hs
    .wdt    off
    .pwrt   off
    .protect off

bltime    equ    83        ; 9600bps @10MHz

txd        equ    ra, 3
rx         equ    ra, 4
led        equ    rb

    org    0ch
ch         ds     1        ; 送信受信データ(8ビット)
rs         ds     1        ; ウェイト時間調整用
cn         ds     1        ; ビット数

    org    0
goto      start
    org    4

start     mov    !ra, #10111b ; RA.3を出力ポートへ
                                ; RA.4を入力ポートへ

```

(サンプル2)

```

mov    !rb, #00000000b ; RBを出力ポートへ

endless

call    receive ; 1バイト受信(ch=受信データ)
mov    led, ch   ; 受信データをBポートに出力する
call    transmit ; その受信したデータを送信する
goto   endless

-----
RS232C送信サブルーチン
-----

transmit    bcf    txd
            mov    rs, #bltime
trans10     djnz   rs, trans10
            mov    cn, #8
            nop
transmit0   rrf    ch
            nop
            movb   txd, c ; データ出力(LSBから)

            mov    rs, #bltime
trans11     djnz   rs, trans11
            djnz   cn, transmit0
            nop
            nop
            nop
            nop
            nop
            nop
            nop
            bsf    txd
            mov    rs, #bltime
trans12     djnz   rs, trans12 ; STOPビット分ウェイト
            ret

-----
RS232C受信サブルーチン
-----

receive

blfsc    rxd
goto     receive ; STARTビットがくるまで待つ

            mov    rs, #bltime/2 ; 1/2ビット分待つ
recv10   djnz   rs, recv10
            mov    cn, #8
            nop

recv0    mov    rs, #bltime
recv11   djnz   rs, recv11
            nop

            movb   c, rxd ; データ入力
            rrf    ch

            djnz   cn, recv0
            ret

```

# PIC16F84

PIC16F84はSRAMが68×8ビットに強化されています。その他は16C84と同じです。

## 8-Bit CMOS EEPROMマイクロコントローラ

### 特長

- RISC-like高性能CPU
- 実行必要があるのは35個のシングルワード命令のみ
- 2サイクルのプログラム分岐を除いて、全てシングルサイクル(400ns)
- 動作スピード: DC-10MHz クロック入力 DC-400ns 命令サイクル
- 14ビット幅の命令
- 8ビット幅のデータバス
- 1024×14ビットの内蔵EEPROMプログラムメモリ
- 54×8ビットの内蔵SRAM (SRAM)
- 15個の特用途ハードウェアレジスタ
- 64×8ビットEEPROMデータメモリ
- 8レベルのハードウェアスタック
- ゲートドライバ、インダクタ(開閉)、リクティブ(相対)の各アドレスモード
- 4個の割り込み要因
- 外部INTピン
- TMROタイマオーバーフロー時の割込み
- PORTB<7:4> 低消費電力時の割込み
- データEEPROMライト終了時の割込み
- 1,000,000回のERASE/WRITEサイクル (標準)
- データ保持期間40年以上

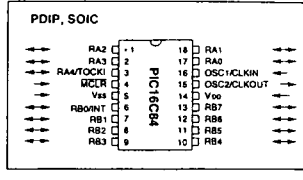
### 周辺回路の特長

- 個別に入出力制御ができる、13本のI/Oピン
- LEDを直接駆動できる、大出力ドライバ電流
- 各ピンの最大シンク電流25mA
- 各ピンの最大ソース電流20mA
- TMRO: 8ビットのリアルタイムクロック カウンタ (8ビットのプログラマブルプリスケール付き)

### マイクロコントローラの特長

- パワーマネージメント
- パワーマネージメント
- オシレータスタートアップタイム
- 確実な動作のために専用のRCオシレータを内蔵した、ウォッチドッグタイマー(WDT)
- コードプロテクションのための、セキュリティEEPROMヒューズ
- 消費電力を節約するSLEEPモード
- ユーザが選択できるオシレータオプション:
  - RCオシレータ: RC
  - セラミック/タリウム共振: XT
  - 高周波セラミック共振: HS
  - 消費電力を節約する低周波クリスタル: LP
- 2ビット幅のEEPROMプログラムおよびデータメモリのリアルタイムインシステムプログラミング(ISP)

### 図A ピン配置



- CMOSテクノロジ**
- 高速、低消費電力CMOS EEPROMテクノロジ
  - 完全スタティック設計
  - 余裕動作電圧範囲
  - 一消費: 20V - 6.0V
  - 一消費: 2.0V - 6.0V
  - 低消費電力
  - 2mA @ 5V, 4MHz
  - 60µA平均 @ 2V, 32KHz
  - 25µA平均スタンバイ電流 @ 2V

### 概要

PIC16C84は高性能、低価格、CMOS、完全スタティック、8ビットセキュリティのマイクロコントローラで、1K×14のEEPROMプログラムメモリと64バイトのEEPROMデータメモリを内蔵しています。このマイクロコントローラは、高性能を誇るPIC16CXXファミリの2番目の製品です。PIC16CXX製品をご使用中のユーザは付録Aのリストをご覧ください。

新しいPIC16C84は、プログラム分岐以外のすべての命令をシングルワード (14ビット幅) とし、各命令をシングルサイクル (10MHzで400ns) で実行することによって高性能を実現しています。プログラム分岐には2サイクルが必要ですが、PIC16C84には4個の割込み要因と8レベルのハードウェアスタックがあります。

周辺回路には8ビットプリスケール付8ビットタイマ/カウンタ (16bitタイマとして使用可能)、13本の両方向I/Oピンがあります。大駆動電流(最大シンク電流25mA、最大ソース電流20mA)を持つI/Oピンによって、外部駆動回路が必要なくシステムコストを節約できます。

PIC16C84製品にはセンサプ、インサネット、エミュレータ、重要なプログラマが用意されています。すべてのツールはIBM PCとその互換機でサポートされています。

### 3.0 アーキテクチャの概要

PIC16C84は一般的なRISCタイプマイクロプロセッサの設計思想に見られる多くの特徴を採用しています。まず、PIC16C84はハードウェアアーキテクチャを採用しており、それは、プログラムとデータがそれぞれ別のメモリからアクセスされるようになっています。これによってプログラムとデータが同じメモリからアクセスされるフェンリオンアーキテクチャよりも実行効率が改善されています。個々のプログラムとデータメモリによってさらに8ビット幅のデータワードとデータメモリを持つことができます。PIC16C84ではオペコードが14ビット幅でベクトル化されており、命令としてのものであります。14ビット幅のプログラムのメモリから14ビット幅の命令バスにアクセスし、シングルサイクルでそれをフェッチします。3線のバイラインによって命令のフェッチと実行を同時にします。したがって、プログラムの分岐を除いたすべての命令(35個)がシングルサイクルで実行されます。

PIC16C84は内蔵されている1K×14のプログラムメモリをアドレス指定し、プログラムの実行は内部のみで(マイクロコントローラモード)。

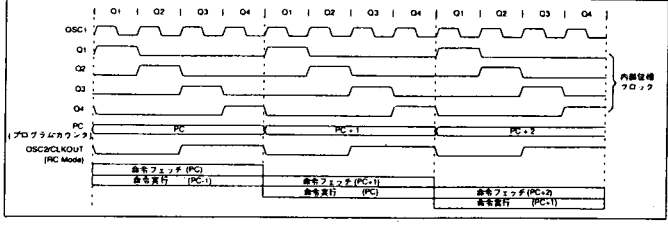
PIC16C84は48個のレジスタファイルまたはデータメモリを選択、または間接的にアドレス指定できます。プログラムカウンタを含むすべての特殊用途レジスタはデータメモリ内に配置されます。命令セットは非常に直交的 (対称的) に構成されており、この結果、アドレス指定モードやレジスタの種類に関わらず、任意のアドレス指定モードで任意のレジスタから任意の動作を実行することができます。このように命令セットの構成が対称的で、しかも「特定の最悪条件」に因りしないため、PIC16C84を使ったプログラミングが簡単にしかも効果的に行え、そのうえ学習効果がさらに高くなります。

### 3.1 PIC16C84 ピン信号の説明

ピン名称	ピンタイプ	通常動作	ピン機能	シリアルインシステムプログラミング (ISP) モード
VDD	P	電源	電源	
VSS	P	グランド	グランド	
OSC1/CLKIN	I	クロック入力/オシレータ接続		
OSC2/CLKOUT	I/O	オシレータ接続/CLKOUT出力 RCオシレータモードではCLKOUT、その他のモードではオシレータ接続		
MCLR/VPP	I/P	マスタクリア(外部リセット)入力、アクティブロー	マスタクリア、集電圧 (Vpp) 印加でプログラミングモードへ。	
RA4/T0CK1	I/O	オープンドレイン出力/入力ピン、TMROタイマ/カウンタへのクロック入力/シフトトリガ入力/パルファ		
RA0	I/O	双方向I/Oピン。TTL入力レベル		
RA1	I/O	双方向I/Oピン。TTL入力レベル		
RA2	I/O	双方向I/Oピン。TTL入力レベル		
RA3	I/O	双方向I/Oピン。TTL入力レベル		
RB0/INT	I/O	双方向I/Oピン/外部割込み入力。TTL入力レベル		
RB1	I/O	双方向I/Oピン。TTL入力レベル		
RB2	I/O	双方向I/Oピン。TTL入力レベル		
RB3	I/O	双方向I/Oピン。TTL入力レベル		
RB4	I/O	双方向I/Oピン。TTL入力レベル		
RB5	I/O	双方向I/Oピン。TTL入力レベル		
RB6	I/O	双方向I/Oピン。TTL入力レベル	クロック入力	
RB7	I/O	双方向I/Oピン。TTL入力レベル	クロック入力/出力	

説明: I = 入力, O = 出力, I/O = 入力/出力, P = 電源, - = 未使用

### 図3.2.1 クロック/命令サイクル



### 3.2 クロック波形/命令サイクル

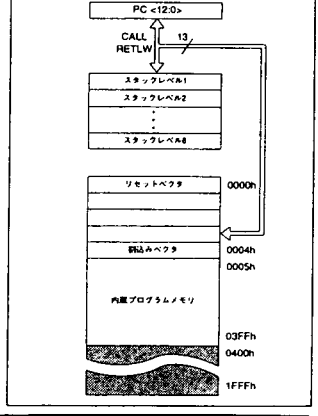
クロック入力(OSC1)からは内部で4分周され、Q1、Q2、Q3、Q4と呼ばれる4個のオーバーラップしない矩形波クロックを出力します。内部で、PCがQ1毎にインクリメントされ、命令がプログラムメモリからフェッチされQ4で命令レジスタにラッチされます。クロックと命令の実行フローを図3.2.1に示します。

PIC16C84は、ユーザのアプリケーションによってプログラムメモリが頻りに更新されるような用途には適しません。プログラムメモリは2つのデータ・クロックピンを使って逐次的にプログラムすることができ (6.0動作モード)、これを行うインシステムプログラミング (ISP) が可能になります。インシステムプログラミングを利用すると、最終試験中にシステムをカスタマイズしたり、現場でシステムをアップグレードしたりすることができます。

### 3.3 命令のフロー/バイライン動作

命令サイクルはQ1、Q2、Q3、Q4でできています。命令のフェッチと実行サイクルはフェッチに1命令サイクル、デコードと実行に1命令サイクルかかるようにバイライン方式で行われます。しかしバイライン方式によって、それ以外の命令は効率よく1サイクルで実行します。もし命令によってプログラムカウンタが変化した場合(GOTOなどは)、その命令を完全に実行するためには2サイクルが必要ですが、フェッチサイクルはQ1でプログラムカウンタ(PC)のインクリメントと同時に始まります。フェッチされた命令は命令レジスタ(IR)にラッチされて、Q2、Q3、Q4の間にデコードされ実行されます。データメモリに対してはQ2の読みが行われ(オペランドリード)、Q4の読みが行われます(ディスティネーションリード)。

### 図3.4.1 プログラムメモリマップとスタック



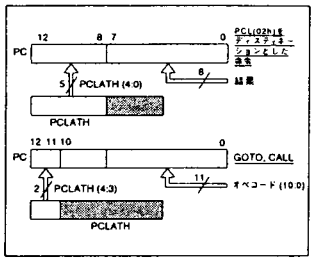
### 3.4 プログラムメモリ構成

PIC16C84には8K×14のプログラムベースをアドレス指定できる13ビット幅のプログラムカウンタがあります。 (図3.4.1) 最初の幅の1K×14(0000h-03FFh)のみ物理的に設定されています。3FFFh以上のローケーションをアクセスするとその最初の1K×14のスペース内で循環することになります。リセットベクタは0000hに、割り込みベクタは0004hにあります。PIC16C84のEEPROMプログラムメモリは、イレース・ライトサイクルが可能な設定になっています。プログラムメモリをプログラムしたいときは、まずMCLRピンをハイに上げて特殊モードをオンにしてください。プログラミングの仕様については6.0章をご覧ください。また、プログラミングにはVDDを4.5Vから5.5Vに保つ必要があります。

### 3.5 プログラムカウンタモジュール

プログラムカウンタ(PC)は13ビット幅です。その下位8ビット、PCLはリード、ライト可能なレジスタです。PCの上位5ビット、PCHは直接リード、ライトできません。PCの上位5ビットはPCLATH(H0Ah)を通してライトされます。CALL、GOTOまたはPCLATHへのライトの際にPCHに新しい値がロードされたときは、図3.5.1に示すようにPCLATHからPCの上位5ビットにロードされます。

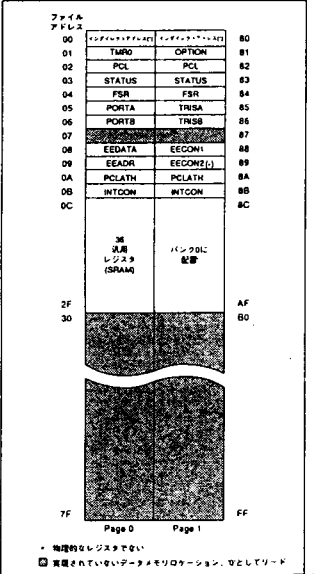
### 図3.5.1 異なる条件でのロードPC



### 3.6 スタック

PIC16C84には8レベル×13ビット幅のハードウェアスタックがあります。スタックのスペースはプログラムあるいはデータスペースの一部ではなくスタックポインタはリード、ライトできません。PCはCALL命令が実行されたとき、または割込みが認識されたとき、スタックにプッシュされます。そのスタックはRETURN、RETLW、RETFB命令の実行されたときポップされます。PCLATH(0Ah)はプッシュあるいはポップ動作に影響を受けません。

### 図3.7.1 レジスタファイルマップ



### 3.7 レジスタファイル構成

PIC16C84のレジスタファイルは128×8で構成されており、ファイルセレクトレジスタFSRを通して直接または間接にアクセスされ、データメモリとしても参照されます。STATUSレジスタにレジスタファイルのページセレクトビットがあり、4ページまで選択できます。しかし、PIC16C84のデータメモリは32hまでしか内蔵していません。最初の12ローケーションは特殊用途レジスタを配置するために使われます。ローケーション0Ch-2FhはスタックRAMで実装されている汎用レジスタです。ほとんどの特殊用途レジスタはページ1に配置されます。ページ1の時に、ローケーション8Ch-AFhをアクセスするとページ0のRAMをアクセスすることになります (図3.7.1)。

### 3.7.1 レジスタファイルのアドレス指定モード

レジスタファイルに対しては、直接的または間接的なアドレス指定が可能です。どちらのモードでも、最大512個のレジスタローケーションにアドレスを指定できます。直接アドレス指定モード: 図3.7.1.1に示す通り、OPCODEからの直接アドレスビットとステータスレジスタからの2ビット (RPI、RPO) を連結させることによって、有効な9ビットの直接アドレスを作成することができます。間接アドレス指定モード: 上記の他に、ファイルアドレス00h (INDF) を使用して間接的にアドレスを指定することもできます。ファイルレジスタとしてINDFを使用したときの命令は、実際には、ファイルセレクトレジスタ (FSR) が指示するファイルレジスタをアクセスすることになります。INDF自体を間接的にリードすると00hが読み出されます。INDFに間接的なライトを実行しても、(ステータスビットが変化する場合) 書き込みは行われません。FSRレジスタの8ビットとステータスレジスタのIRPビットを連結させることによって、有効な9ビットのアドレスを作成することができます。特殊機能をもつレジスタの中にはページ1にマップされているものがあります。したがって、RPOビットをセットしてこれらにアドレス指定する必要があります。RPIビットとIRPビットは実際には使用されません。使いやすさを考えて、汎用レジスタはページ0とページ1の両方にマップしてあります。

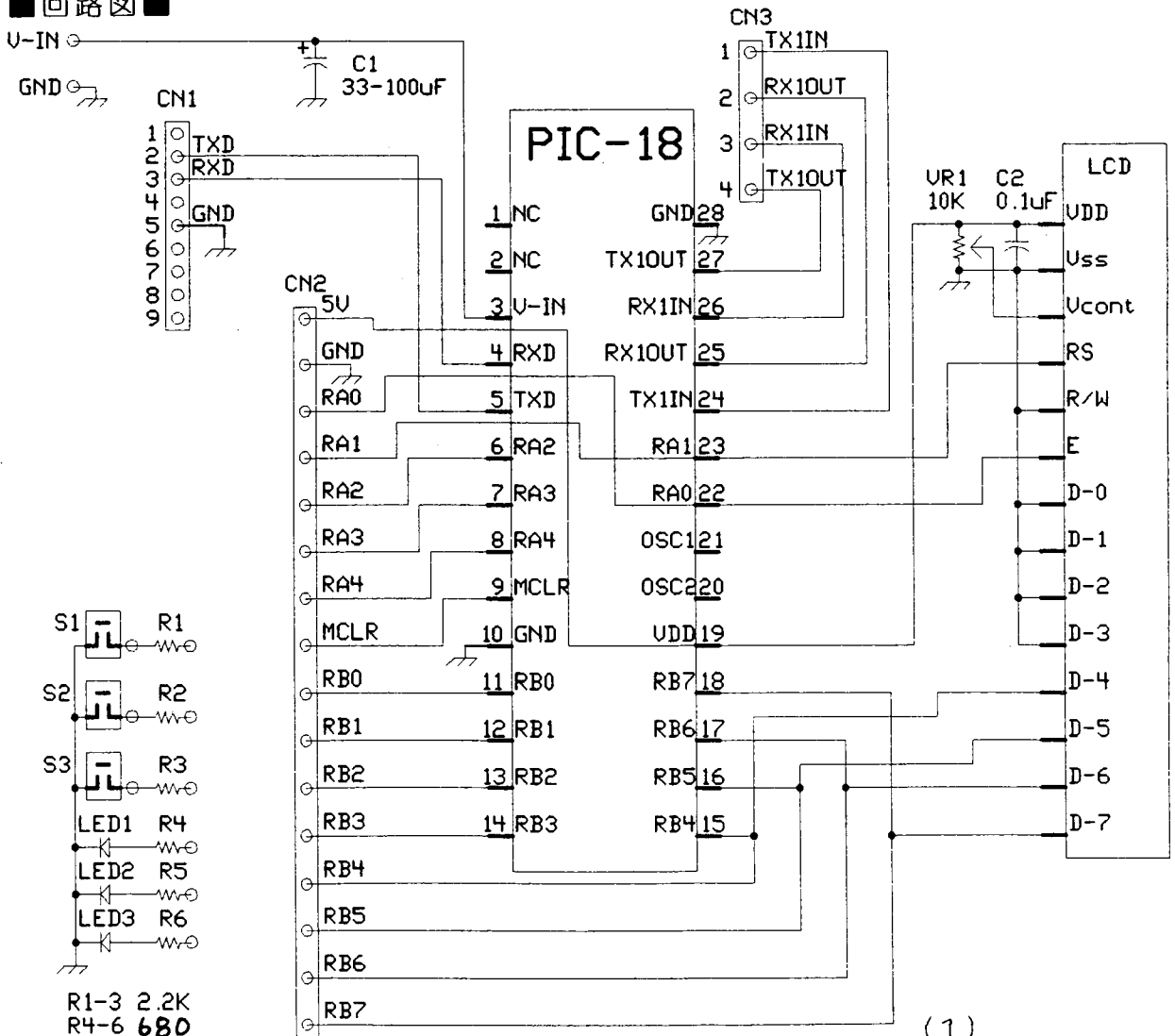
# AKI-PIC18 キャリーボードキット

## 16文字×2行液晶表示器付

### PIC18 マイコンモジュールキット専用

- ご好評のPIC18 マイコンモジュールキット専用のマザーボードです。
- 16文字×2行液晶表示器が付いていますので、文字数字などの表示が簡単に出来るようになりました。
- パソコンとのRS232通信用にDサブ9Pコネクタがついています。
- タクトスイッチ、LEDが基板に実装してあります。PICマイコンの学習や各種製作に便利です。

#### ■ 回路図 ■



## ■ 部品表 ■

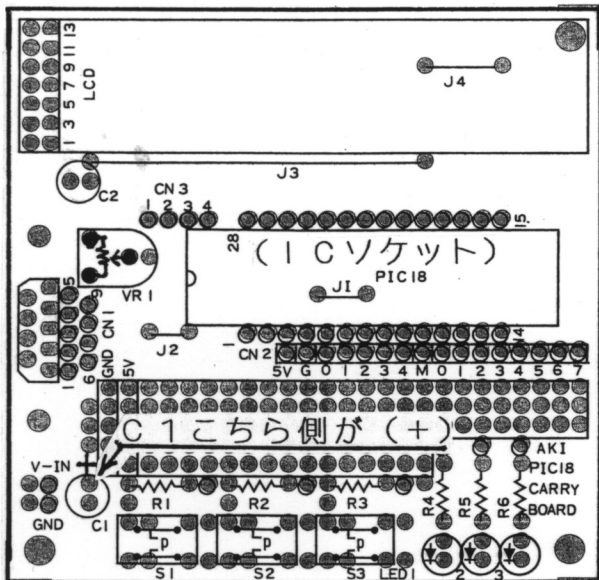
番号	名称	数	備考
C 1	コンデンサ 100 $\mu$ F	1	電解コンデンサ 33~100 $\mu$ F 16V以上
C 2	コンデンサ 0.1 $\mu$ F	1	積層セラミックコンデンサ (104)
R 1~3	抵抗 2.2K $\Omega$	3	1/4W 赤赤赤金
R 4~6	抵抗 680 $\Omega$	3	1/4W 青灰茶金
LED 1~3	LED	3	発光ダイオード
S 1~3	スイッチ	3	押しボタンスイッチ
LCD	液晶表示器	1	16文字×2行タイプ
VR 1	半固定抵抗 10K $\Omega$	1	半固定抵抗 (103)
CN 1	コネクタ	1	Dサブ9Pコネクタ基板用
基板	専用基板	1	AKI PIC18 CARRY BOARD
	ICソケット	1	28P

## ■ 製作 ■

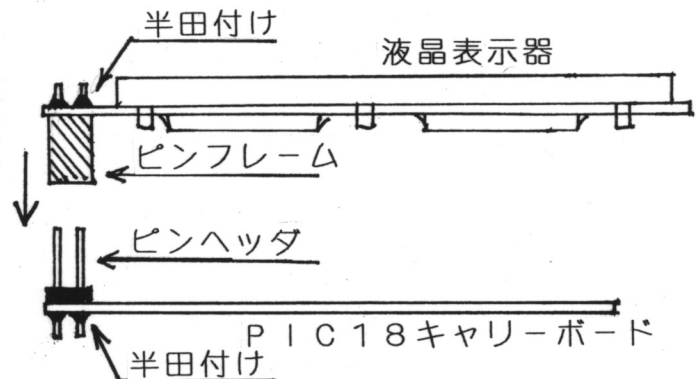
あらかじめ部品表と部品をてらしあわせ、数等をチェックしてから製作してください。

- ① コンデンサ、抵抗、VR 1、LEDの順で半田付けしていきます。  
電解コンデンサC 1には、極性があります。基板には極性マークがありませんので、部品配置図の指示にしたがい取り付けてください。  
LEDには極性がありますので基板印刷のマークにあわせてください。
- ② J 1~J 4を取り付けます。メッキ線、ビニール線などで配線してください。  
(メッキ線、ビニール線はキットに入っていないので別にご用意ください)
- ③ ICソケット、Dサブ9Pコネクタ、スイッチをとりつけます。
- ④ LCD用ピンヘッダ(オス)を基板に半田付けします。ピンフレーム(メス)はあらかじめ、LCDに半田付けしてください。  
(ピンヘッダ、ピンフレームは、LCDの袋に入っています。)

## ■ 部品配置図 ■



## ■ 液晶部取り付け図 ■



■ CN1、2、3接続表 ■

CN1	機能	CN2	機能
1	未設続 (CD)	5V	5V
2	TXD (232-RA3)	G	GND
3	RXD (232-RA4)	0	RA0 (LCD E)
4	未設続 (DTR)	1	RA1 (LCD RS)
5	GND	2	RA2
6	未設続 (DSR)	3	RA3 (232 TXD)
7	未設続 (RTS)	4	RA4 (232 RXD)
8	未設続 (CTS)	M	MCLR
9	未設続 (RI)	0	RBO
未設続の ( ) 内はパソコンCOM1の信号名です。		1	RB1
		2	RB2
		3	RB3
		4	RB4 (LCD D4)
CN3	機能	5	RB5 (LCD D5)
1	TXIN (ADM232 11PIN)	6	RB6 (LCD D6)
2	RXOUT (ADM232 12PIN)	7	RB7 (LCD D7)
3	RXIN (ADM232 13PIN)		
4	TXOUT (ADM232 14PIN)		

■回路の説明■ 回路図を参考にお読みください。

①電源

電源は5.5V~12V 100mA程度の電源をご用意ください。

基板の電源入力はV-IN (+) GND (-) です。

AKI-PI C18マイコンモジュールのU1 (S81350) が回路全体の5Vをつくります。

②PIC18マイコンモジュールキット (PIC18)

プログラムを書き込んだPIC18マイコンモジュールをICソケットに挿入して使用します。PIC18マイコンモジュールキットにあらかじめ書き込まれているサンプルプログラムはLED1~3, CN1 (7-8) を接続すれば、そのまま動作します。

★接続 CN1 (7) - CN1 (8)

LED1 (R4) - CN2 (0 RBO)

LED2 (R5) - CN2 (1 RB1)

LED3 (R6) - CN2 (2 RB1)

★動作 CN1をパソコンRS232Cに接続し、パソコンから、1 (31H) を送信すると、LED1が点灯します。また、2 (32H) でLED2 3 (33H) でLED3が点灯します。

詳しくはPIC18マイコンモジュールキットのサンプルソフトをご覧ください。

③液晶表示器 (LCD)

16文字×2行の液晶表示器が4ビットモードでPICマイコンに接続されています。

LCD	PICマイコン	LCD	PICマイコン
D4	———— RB4	E	———— RA0
D5	———— RB5	RS	———— RA1
D6	———— RB6		
D7	———— RB7		

LCD DO~D3、R/WはGNDに接続

VR1はLCDのコントラスト調整用です。時計方向に回すと表示が濃くなります。あらかじめ、時計方向いっぱい回しておいてください。

### ③ つづき

★液晶の使い方 (サンプルソフトもごらんください。)

必要なサブルーチンとレジスタ

init_lcd (LCD初期化)	
write_lcd8 (8ビットモード液晶書き込み)	d8
write_lcd4 (4ビットモード液晶書き込み)	d4
wait_ms (msのウェイト)	wait_cn
wait_us (μsのウェイト)	wait_cn2

使い方

#### ◆初期設定

- ①液晶に接続しているピン (ra0, ra1 rb4~rb7) を出力ピンに設定する
- ②液晶初期設定 (init\_lcd) をコールする。  
4bitモード・2行・カーソル=ON・インクリメント=ONに設定されています。

#### ◆コマンド (カーソルを指定位置に移動する等) を送信する

- ①RS (ra1) を0にする (clrb RS)
- ②レジスタ d4にコマンドを入れる
- ③write\_lcd4をコールする。

#### ◆文字データ (#'0'等) を送信する

- ①RS (ra1) を1にする (setb RS)
- ②レジスタ d4に文字データを入れる
- ③write\_lcd4をコールする。

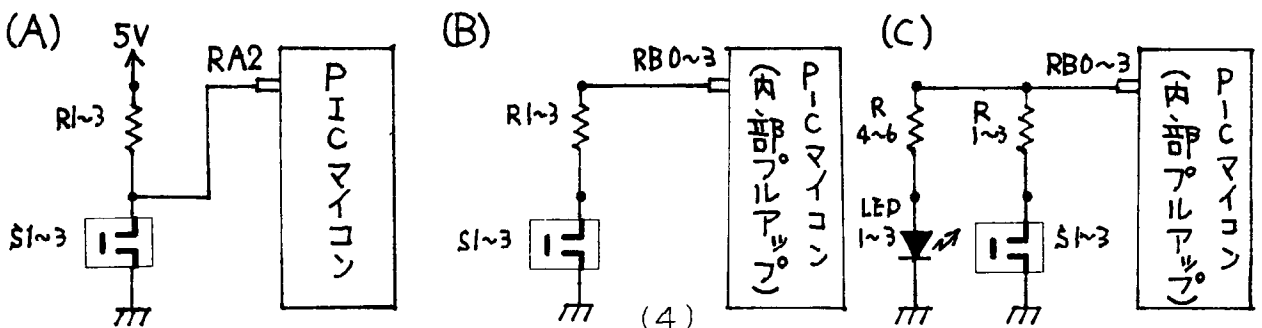
### ④ RS 2 3 2 C (CN 1)

パソコンなどとの通信用です。PICマイコンモジュールに送信2CH、受信2CH分あります。そのうち1CH分が、Dサブコネクタ (CN 1) に接続されています。他の1CH分はCN 3に出ていますので必要にあわせて、接続してください。

CN 1		PICマイコン
2 (TXD)	——	5 (TXD-ADM232-RA3)
3 (RXD)	——	4 (RXD-ADM232-RA4)
7	□	制御無し (PIC18マイコンモジュールキットの
8	□	サンプルソフトなどの場合) CN 1の7・8を接続する。

### ⑤ スイッチ (S1~3)

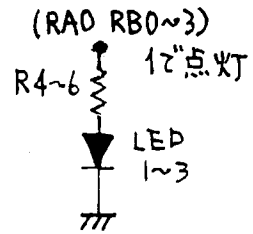
スイッチは自由に使えるようにどこにも接続されていません。通常は抵抗 (R1~3) を5Vに接続し、スイッチをRA2, RB0~3が未使用ですのでそのどれかに接続してください。(図A) RBは入力時内部プルアップの設定ができます。その場合は抵抗 (R1~3) をRB0~3に接続してください。(図B) この場合サンプルソフトで行なっているように、1つのピンに同時にLEDを接続することが、できます。入力を見るときだけI/O設定を入力にし、その他の時間はI/O設定を出力にすることで、I/Oピンを節約できます。(図C)





⑥ LED (LED 1~3)

LEDは自由に使える様にどこにも接続されていません。I/Oピン (RA2, RB0~3) で点灯させる場合は、そのピンを出力に設定し '1' (Hi) を出力すると点灯します。



⑦ ユニバーサルエリア

ユニバーサルエリアは簡単なユーザー回路を付けられる様になっています。5V、GNDも用意してありますので、各種実験等に活用してください。

■ サンプルプログラム ■

液晶・スイッチ・LEDを動作させてみましょう。

あらかじめ、このサンプルプログラムをパソコンで打ち込み、PICプログラマキットでPICマイコンモジュールキットに書き込んでください。

① キットの接続

I/Oピンを節約するために1つのピンに 入力 (スイッチ) 出力 (LED) を接続しています。

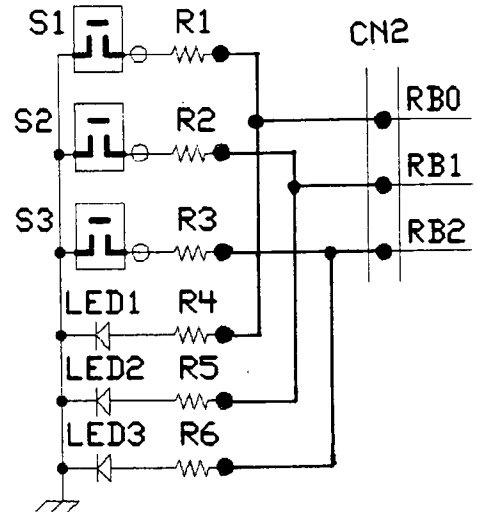
② サンプルプログラム動作概要

S1~S3を押すとそれぞれLED1~3が点灯し、はなすと消灯します。

液晶にS1~3の状態を表示します。

(スイッチを押す (Low) と0, はなすと (Hi) 1を表示します。)

■ キットの接続



1つのピンに出力、入力を接続しているため、入力を見るときだけ、プログラムでピンを入力に指定し、内部プルアップさせています。その他の時間は出力に指定し、LEDを点灯または消灯させています。

下図(a)の時間は、入力でLEDは消灯していますが、内部プルアップさせているためLEDにわずかですが電流が流れます。実際は入力に設定している時間が短いため、ほとんど見えませんが暗い所では薄く光って見えます。

■ I/Oピンの設定とLEDの点灯 ■

I/Oピンの設定	入力	出力	入力	出力	入力
スイッチの状態		O F F		O N	
LED	消 灯	消 灯	点 灯	消 灯	消 灯

(a)

# ■ サンプルプログラムソースリスト ■

PIC プログラマキット用ですので、キット付属の P A . E X E 用でアセンブルしてください。

## P I C - 1 8 キャリーボードサンプルプログラム

```

include 16184.h
.osc      hs
.wdt      off
.pwr1     off
.protect  off

led       equ      rb
E         equ      ra, 0           ;LCD E 信号
RS        equ      ra, 1           ;LCD RS 信号

org       0ch
def       ds        1             ;スイッチの状態を入れるレジスタ
sw1       equ      def, 0         ;S1 BIT
sw2       equ      def, 1         ;S2 BIT
sw3       equ      def, 2         ;S3 BIT

wait_cn   ds        1             ;ウェイトカウンタ用
wait_cn2  ds        1             ;ウェイトカウンタ用
d4        ds        1             ;液晶へのコマンドデータを格納するレジスタ
d8        ds        1             ;サブ関数 write_lcd8 で使用
cn        ds        1             ;サブ関数 title で使用 (文字数)
poi       ds        1             ;サブ関数 title で使用 (ポインタ)

org
goto     start
org       4

```

### 液晶初期表示データ

```

title_tbl jmp      pctw           ; テーブル参照
title_mes  retw     'S1=0 S2=0 S3=0'
           retw     'PIC-18Pキャリーボード'

```

### 初期設定

```

start     mov      |ra, #00011100b ; E, RSピンをOUTに
           mov      |rb, #00001111b ; DB7-DB4ピンをOUTに
           clr      ra
           clr      rb

           call     init_lcd       ; LCD初期化
           call     title          ; サンプル文字列表示

```

### メインルーチン

```

start1   mov      |rb, #00000111b ; RB0-3を入力にする。

```

```

setb     rp0           ; RAMバンク1を指定する
clrb     rbpu          ; RBをプルアップする
clrb     rp0           ; RAMバンク0にもどす

mov      def, rb       ; スwitchの値を読む
-----LED点灯-----
not      def           ; データを反転する
           ; スwitchを押すと点灯
           ; 離すと消灯
mov      |rb, #00000000b ; RB0-7を出力にする
mov      led, def
not      def           ; データを元に戻す
-----S1液晶表示-----
clrb     RS           ; コマンドなのでRSを0にする
mov      d4, #10000011b ; カーソルを4文字目に移動
call     write_lcd4

setb     RS           ; データなのでRSを1にする
mov      d4, #'0'     ; S1の値が0 (LOW)ならば
snb     sw1           ; 0を表示し、1ならば
inc      d4           ; 1を表示する
call     write_lcd4

-----S2液晶表示-----
clrb     RS
mov      d4, #10001000b ; カーソルを9文字目に移動
call     write_lcd4

setb     RS
mov      d4, #'0'
snb     sw2
inc      d4
call     write_lcd4

-----S3液晶表示-----
clrb     RS
mov      d4, #10001101b ; カーソルを14文字目に移動
call     write_lcd4

setb     RS
mov      d4, #'0'
snb     sw3
inc      d4
call     write_lcd4

goto     start1

```

### 液晶初期化ルーチン

```

init_lcd  mov      wait_cn, #15    ; wait 15ms
           call     wait_ms

           clr     RS             ; RS='L'
           mov     d8, #00110000b
           call     write_lcd8
           mov     wait_cn, #5     ; wait 4.1ms
           call     wait_ms

```

```

mov     d8, #00110000b
call   write_lcd8
mov     wait_cn, #100      ; wait 100us
call   wait_us

mov     d8, #00110000b
call   write_lcd8      ; 0 0 0011 (3)

mov     d8, #00100000b
call   write_lcd8      ; 0 0 0010 (4bit)

mov     d4, #00101000b    ; duty, font set 9
call   write_lcd4

mov     d4, #00000001b    ; クリアコマンド
call   write_lcd4
mov     wait_cn, #2      ; クリアが終わるまで待つ
call   wait_ms

mov     d4, #00000110b    ; entry mode set
call   write_lcd4

mov     d4, #00001110b    ; display on, cursor on
call   write_lcd4
ret

```

### 液晶タイトル(初期表示)ルーチン

```

title   clr     RS
        mov     d4, #10000000b    ; カーソルを1行目に移動
        call   write_lcd4

        clr     poi
        call   putchar

        clr     RS
        mov     d4, #10000000b+64 ; カーソルを2行目に移動
        call   write_lcd4

        call   putchar
        ret

putchar mov     cn, #16          ; 16文字
        setb    RS              ; 以後のコマンドは文字表示

putchar0 mov     w, poi
        call   title_tbl
        mov     d4, w
        call   write_lcd4      ; 文字表示(RS='H'にしてください)

        inc     poi
        djnz   cn, putchar0    ; 16文字分くり返し
        ret

```

### 8ビットモード専用液晶ライトルーチン

```

write_lcd8 and     rb, #00001111b    ; RBの上位を0に
           and     d8, #11110000b    ; d8の下位を0に
           or      rb, d8            ; RB=RB or d8
           nop
           setb    E                  ; Eピンを'H'
           nop

           clr     E
           mov     wait_cn, #40
           call   wait_us

           ret

```

### 4ビットモード専用液晶ライトルーチン

```

write_lcd4 mov     d8, d4
           call   write_lcd8

           mov     d8, d4
           swap    d8
           call   write_lcd8

           ret

```

### msオーダーのウェイト

```

wait_ms
wait_ms0
wait_ms1  mov     wait_cn2, #0
           nop
           nop
           nop
           nop
           nop
           nop
           djnz   wait_cn2, wait_ms1
           djnz   wait_cn, wait_ms0
           ret

```

### μsオーダーのウェイト

```

wait_us
wait_us0  djnz   wait_cn, wait_us0
           ret

```

PIC16F84マイコンモジュール (PIC-18) のソフト開発ツール

■PICプログラマキット日本語対応 DOS/V・PC98シリーズ用

◎秋月オリジナルアセンブラ・16F84用シミュレータ付き

◎DOS/V・PC98シリーズ用コントロールソフト付き

◎高信頼28Pゼロプレッシャーソケット使用

(PIC16F84マイコンモジュール (PIC-18) は基板のまま28Pゼロプレッシャーソケットに差してプログラムの書き込みができます。)

◎実用になるサンプルプログラム多数収録

製作マニュアル・基板・パーツ式付

1台 ¥5,700.

■PicCコンパイラ WINDOWS3.1/95対応

◎ANSI C準拠のCコンパイラです。

◎当社プログラマキットに対応したHEXファイルを出力します。

◎対応CPU 16C554・16C556・16C558・16C61・16C62・16C620・16C621・16C622・16C63

・16C64・16C641・16C642・16C65・16C66・16C661・16C662・16C67・16C71・16C710・16C711

・16C715・16C72・16C73・16C74・16C76・16C77・16C9XX・16F83・16F84等多数対応


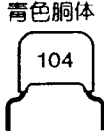
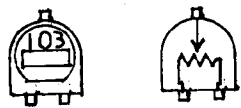
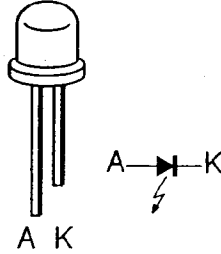

日本語解説書付

プログラムメモリ1K限定版

1本 ¥6,800.

1本 ¥2,000.

■部品外観図■

<p>タクトスイッチ</p>  <p>内部接続図</p>	<p>積層セラミックコンデンサ</p> <p>青色胴体</p> 	<p>半固定抵抗</p> 
<p>発光ダイオード(LED)</p> <p>足の長い方がアノード</p> <p>A:アノード K:カソード</p>  <p>A K</p>	<p>電解コンデンサ</p>  <p>足の長い方が+側で胴体には-側の表示</p> <p>+</p>	

AKI-PIC18キャリーボードキット 秋月電子 KAKE 1998/9  
 お問い合わせは往復葉書、または返信用切手同封の封書にてお願いいたします。  
 電話、ファックス、Eメールでのお問い合わせは受け付けておりません。  
 〒158-0095 東京都世田谷区瀬田5-35-6 秋月電子通商 質問係宛

インターネットでホームページを開設しております。そのダウンロードコーナーで本取説上のサンプルソフトソースファイルを公開しております。また、新製品情報やバージョンアップ情報など出ております。ぜひご利用ください。

(URL) [www.tomakomai.or.jp/akizuki/](http://www.tomakomai.or.jp/akizuki/)

# 16文字×2行 超ハイコントラスト大文字LCDモジュール

SC1602BSLB(バックライト付) SC1602BS\*B(無) 共通資料

## ■ FEATURE:

1. 5×7ドット+カーソル表示
2. 液晶コントローラ内蔵
3. 5V単一電源(低消費電流)
4. 1/16デューティサイクル
5. Vf=4.2V LED(バックライト内蔵品のみ)
6. M1632, L1672ピンコンパチブル

## ■ MECHANICAL DATA

ITEM	DIMENSIONS	UNIT
Module Size (W × H × T)	85.0×30.0×8.8(12.7LED)	mm
Viewing Area (W × H)	66.0 × 16.0	mm
Character Size (W × H)	2.96 × 5.56	mm
Character Pitch(W × H)	3.55 × 5.94	mm
Dot Size(W × H)	0.56 × 0.66	mm
Dot Pitch(W × H)	0.60 × 0.70	mm

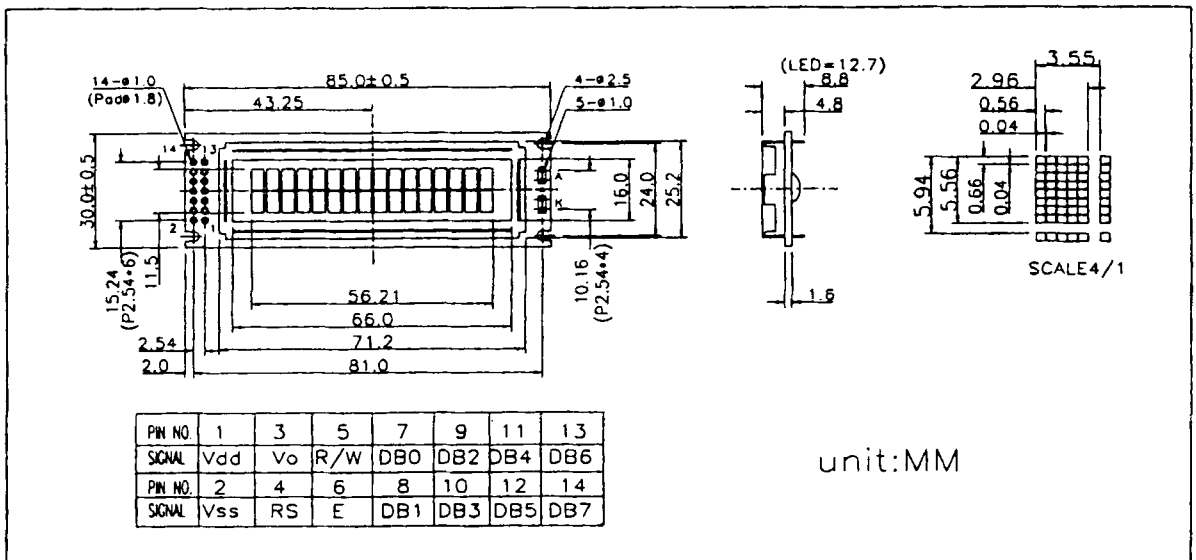
## ■ INTERFACE PIN CONNECTIONS:

No	Symbol	Function	No	Symbol	Function
1	V <sub>DD</sub>	5V	9	DB2	DATA BIT2
2	V <sub>SS</sub>	0V	10	DB3	DATA BIT3
3	V <sub>0</sub>	CONTRAST ADJ.	11	DB4	DATA BIT4
4	RS	REGISTER SELECT	12	DB5	DATA BIT5
5	R/W	READ/WRITE	13	DB6	DATA BIT6
6	E	ENABLE SIGNAL	14	DB7	DATA BIT7
7	DB0	DATA BIT0			
8	DB1	DATA BIT1			

## ■ ELECTRICAL CHARACTERISTICS:

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
LCD Operating Voltage	V <sub>DD-V<sub>0</sub></sub>	Ta=0 °C	—	4.8	—	V
		Ta=25 °C	—	4.5	—	V
		Ta=50 °C	—	4.2	—	V
Supply Voltage	V <sub>DD-V<sub>SS</sub></sub>	—	4.7	5	5.3	V
Input Voltage	"High" Level	V <sub>IH</sub>	—	2.2	—	V <sub>DD</sub> V
	"Low" Level	V <sub>IL</sub>	—	0	—	0.6 V
Output Voltage	"High" Level	V <sub>OH</sub>	—	2.4	—	V
	"Low" Level	V <sub>OL</sub>	—	—	—	0.4 V

## ■ EXTERNAL DIMENSION



■ M1632とピン・インストラクションは同じです。(差し替え可能)  
但し、メーカーのピン番号の振り方が逆になっています。(1~14, 14~1)

■ LEDバックライトについて

- ・ LEDの順方向電圧が4.2V(typ)ですので、電源とは抵抗を介して接続してください。抵抗値は周囲の明るさで多少変わります(約10Ω~100Ω)
- ・ 液晶基板上のJ3部分をショートし、LEDのカソード側のパターンに抵抗(チップ抵抗)を半田付けしてもバックライトが点灯します。(液晶側電源と共用できます)

## ■ Absolute maximum ratings

Item	Symbol	Test condition	Standard value		Unit
			min.	max.	
Supply voltage for logic	Vdd-Vss	Ta=25 °C	-0.3	7	V
Supply voltage for LCD drive	Vlcd	Ta=25 °C	Vdd-13.5	Vdd+0.3	V
Input voltage	Vin	Ta=25 °C	-0.3	Vdd+0.3	V
Operating temperature	Topr	....	0	50	°C
Storage temperature	Tstg.	....	-20	70	°C

## ■ Reliability conditions

LCD Module (Consumer Type)		
High temperature operation	Operation 96~100Hrs at 50 ± 2 °C surrounding temp.	No visible inferiority in appearance no function.
Low temperature operation	Operation 96~100Hrs at 0 ± 2 °C surrounding temp.No dew to be found.	
High temperature storage	Storage 96~100Hrs at 60 ± 2 °C surrounding temp.then storage 4Hrs at normal condition (Power Off).	
Low temperature storage	Storage 96~100Hrs at -20 ± 2 °C surrounding temp.then storage 4Hrs at normal condition (Power Off). No dew to be found.	
Damp proof	Storage 96~100Hrs at 40 ± 2 °C and 90~95% RH surrounding condition then storage 4Hrs at normal condition(Power Off).No dew to be found.	

Note:The above mentioned conditions are nominal ones.which may differ in special specifications.

## ■ Optical characteristics

### 1.STN Type

Item	Symbol	Condition	min.	typ.	max.	Unit	Note
Viewing angle	$\psi$ 2- $\psi$ 1	K=2.0	60	....	....	deg.	A
Contrast ratio	K	$\psi = 10, \theta = 0$	5	....	....	....	B
Response time (Rise)	tr	$\psi = 10, \theta = 0$	....	150	250	ms	C
Response time (Fall)	tf	$\psi = 10, \theta = 0$	....	200	300	ms	C

### 2.TN Type

Item	Symbol	Condition	min.	typ.	max.	Unit	Note
Viewing angle	$\psi$ 2- $\psi$ 1	K=2.0	....	40	....	deg.	A
Contrast ratio	K	$\psi = 25, \theta = 0$	....	5	....	....	B
Response time (Rise)	tr	$\psi = 25, \theta = 0$	....	80	120	ms	C
Response time (Fall)	tf	$\psi = 25, \theta = 0$	....	60	90	ms	C

## ■ ELECTRICAL CHARACTERISTICS

### 1.DC Characteristics (Vdd=5V+10%,Vss=0V,Ta=25C)

Parameter	Symbol	Condition	Applicable PIN	Min.	Typ.	Max.	Unit
H level input voltage(1)	Vih1	——	DB0-DB7	2.2	——	Vdd	V
L level input voltage(1)	Vil1	——	RS,R/W,E	-0.3	——	0.6	V
H level input voltage(2)	Vih2	——	OSC1	Vdd-1.0	——	Vdd	V
L level input voltage(2)	Vil2	——		-0.2	——	1	V
H level output voltage(1)	Voh1	Ioh=-0.205mA	DB0-DB7	2.4	——	——	V
L level output voltage(1)	Vol1	Iol=1.2mA		——	——	0.4	V
H level output voltage(2)	Voh2	Ioh=-40uA	XSC LP DO	0.9Vdd	——	——	V
L level output voltage(2)	Vol2	Iol=40uA		——	——	0.1 Vdd	V
I/O leakage current	Iil	Vin=0 to Vdd		-1	——	1	uA
Pull-UP Mos Current	-Ip	Vdd=5V		50	125	250	uA
Supply current	Iop	Rf oscillation from external clock Vdd=5V fosc=270kHz	Vdd	——	0.35	0.6	uA

#### Internal clock operation (RF oscillation)

Oscillation frequency	fosc	Rf=91k+2%	OSC1-OSC2	190	270	350	kHz
Oscillation frequency	fosc	Ceramic filter	OSC1-OSC2	245	250	255	kHz
LCD driving voltage	Vicd	Vdd-V5	V1-V5	3.0	——	11.0	V

### 2.AC Characteristics (Vdd=5V+10%,Vss=0V,Ta=25C)

#### 1).Read Cycle

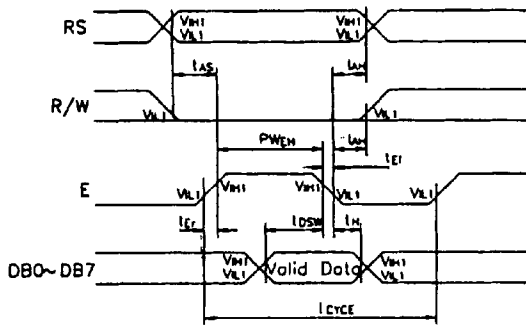
Parameter	Symbol	Min.	Typ.	Max.	Unit.	TEST PIN
Enable cycle time	tc	500	——	——	ns	E
Enable "H" level pulse width	tw	220	——	——	ns	E
Enable rise / fall time	tr,tf	——	——	25	ns	E
RS, R/W setup time	tsu	40	——	——	ns	R/W,RS
RS,R/W address hold time	th	10	——	——	ns	R/W,RS
Read data output delay	td	60	——	120	ns	DB0-DB7
Read data hold time	tdh	20	——	——	ns	DB0-DB7

#### 2).Write Cycle

Parameter	Symbol	Min.	Typ.	Max.	Unit.	TEST PIN
Enable cycle time	tc	500	——	——	ns	E
Enable "H" level pulse width	tw	220	——	——	ns	E
Enable rise / fall time	tr,tf	——	——	25	ns	E
RS, R/W setup time	tsu1	40	——	——	ns	R/W,RS
RS,R/W address hold time	th1	10	——	——	ns	R/W,RS
Data setup time	tsu2	60	——	——	ns	DB0-DB7
Write data hold time	th2	10	——	——	ns	DB0-DB7

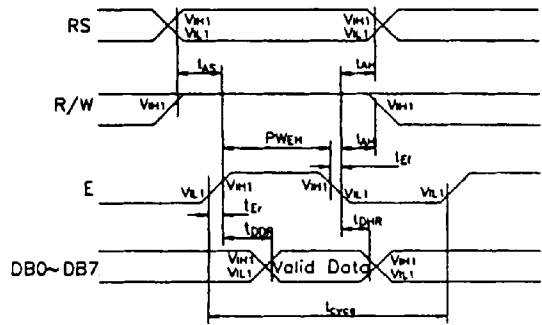
## Pin assignment

FIG.1 WRITE OPERATION



(Write Data from MPU to MODULE)

FIG.2 READ OPERATION



(Read Data from MODULE to MPU)

Pin No	Symbol	Level	Function	
1	V <sub>dd</sub>	...	Power Supply 5V	
2	V <sub>ss</sub>	...		0V(GND)
3	V <sub>o</sub>	...		for LCD Drive
4	RS	H/L	Register select signal Register H:Data input Select L:Instruction Input	
5	R/W	H/L	H:Data read(Module → MPU) L:Data Write(Module ← MPU)	
6	E	H,H → L	Enable signal(no pull-up resistor)	
7	DB0	H/L	Data bus line	
8	DB1	H/L		
9	DB2	H/L		
10	DB3	H/L		
11	DB4	H/L		
12	DB5	H/L		
13	DB6	H/L		
14	DB7	H/L		

※ Interface between data bus line and 4-bit or 8-bit MPU is available. Data transfer are made in twice in case of 4-bit MPU, and once in case of 8-bit MPU.

■ If interface data is 4-bit long

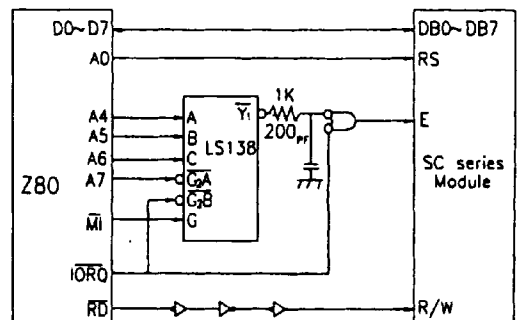
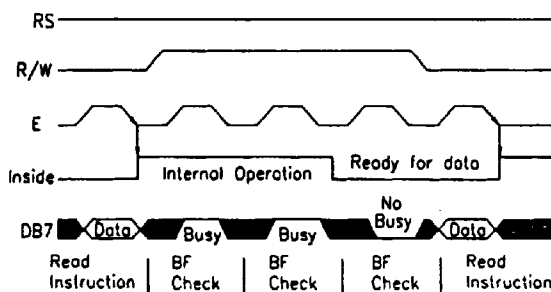
Data transfer are made through 4 bus line from DB4 to DB7. While the rest of 4 bus line from DB0 to DB3 are not used. Data transfer with MPU are completed when 4-bit data are transferred in twice. First upper 4-bit data, then lower 4-bit data.

■ If interface data is 8-bit long

Data transfer are made through all of 8 bus line from DB0 to DB7.

## Interface with MPU

Example of Intreface with 8-bit MPU (Z80)





● Standard Character Pattern(S0)

		Higher 4-bit (D4 to D7) of Character Code (Hexadecimal)																			
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F				
Lower 4-bit (D0 to D3) of Character Code (Hexadecimal)	0	CG RAM (1)			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
	1	CG RAM (2)	.	:	;	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	2	CG RAM (3)		;	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	3	CG RAM (4)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	4	CG RAM (5)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	5	CG RAM (6)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	6	CG RAM (7)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	7	CG RAM (8)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	8	CG RAM (1)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	9	CG RAM (2)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	A	CG RAM (3)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	B	CG RAM (4)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	C	CG RAM (5)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	D	CG RAM (6)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	E	CG RAM (7)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^
	F	CG RAM (8)		.	:	<	=	>	?	@	+	~	^	~	^	~	^	~	^	~	^

## ■ Instructions

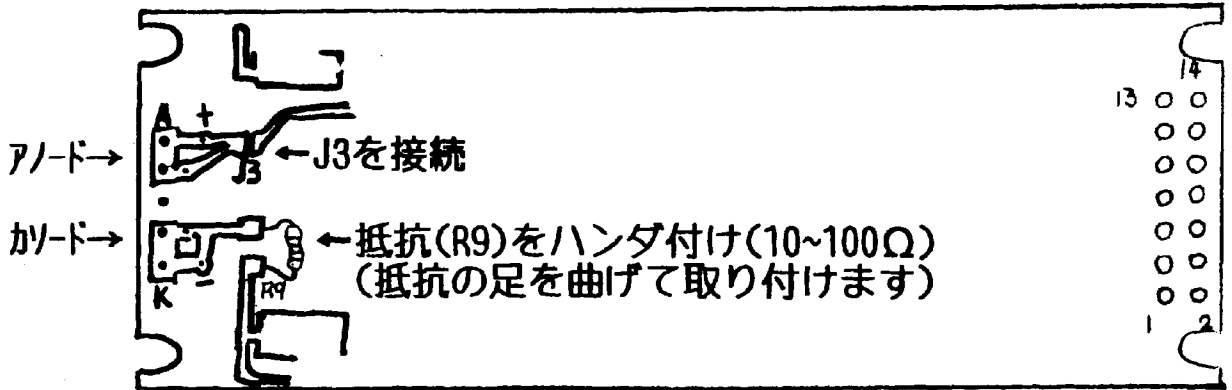
Instruction	Code										Description	Executed Time(max.)
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear Display	0	0	0	0	0	0	0	0	0	1	Clears all display and returns the cursor to home position (Address 0)	1.64ms
Cursor At Home	0	0	0	0	0	0	0	0	1	*	Returns the cursor to the home position(Address 0).Also returns the display being shifted to the original position DDAM contents remain unchanged. (during data write and read.)	1.64ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S	Sets the cursor move direction and specifies or not shift the display.These operations are performed during data write and read	40 $\mu$ s
Display On/Off Control	0	0	0	0	0	0	1	D	C	B	Sets ON/OFF of all display(D) cursor ON/OFF(C),and blink of cursor position character(B).	40 $\mu$ s
Courosr/Display Shift	0	0	0	0	0	1	S/C	R/L	*	*	Moves the cursor and shifts the display without changing DDRAM contents.	40 $\mu$ s
Function Set	0	0	0	0	1	DL	N	F	*	*	Sets interface data length(DL) number of display lines (N)and character font(F).	40 $\mu$ s
CGRAM Address Set	0	0	0	1	ACG					Sets the CGRAM address.CGRAM data is sent and received after this setting.		40 $\mu$ s
DDRAM Address Set	0	0	1	ADD					Sets the DDRAM address. DDRAM data is sent and received after this setting.		40 $\mu$ s	
Busy Flag/ Address Read	0	1	BF	AC					Reads Busy flag (BF)indicating internal operation is being performed and read address counter contents.		40 $\mu$ s	
CGRAM/DDRAM Data Write	1	0	WRITE DATA					Writes data into DDRAM or CGRAM		40 $\mu$ s		
CGRAM/DDRAM Data Read	1	1	READ DATA					Reads data into DDRAM or CGRAM		40 $\mu$ s		

Code	Description	Executed Time(max)
<p>I/D=1:Increment I/D=0:Decrement S=1:With display shift S/C=1:Display shift S/C=0:Cursor movement R/L=1:Shift to the right R/L=0:Shift to the left DL=1:8-bit</p> <p>DL=0:4bit N=1:2 lines N=0:1 lines F=1:5 <math>\times</math> 10dots F=0:5 <math>\times</math> 7dots BF=1:Internal operation is being performed BF=0:Instruction acceptable</p>	<p>DDRAM:Display Data RAM CGRAM:Character Generator RAM ACG:CGRAM Address ADD:DDRAM Address Corresponds to cursor address. AC:Address Counter,used for both DDRAM and CGRAM *:Invalid</p>	<p>fcp or fosc=250kHz However,when frequency changes,ecution time also changes EX if fcp or fosc is 270kHz, 40<math>\mu</math>s <math>\times</math> 250/270=37<math>\mu</math>s</p>

## ■ LEDバックライト接続方法

- ・直列に入れる抵抗で明るさ・消費電流が変化します。
- 抵抗は100Ω1/6Wを1本又は2本並列にしてご利用下さい。

### 液晶基板ウラ面



## ■ Electrical characteristics

$VDD=5V \pm 5\%$   
 $VSS=0V, Top=0-50^\circ C$

Item	Symbol	Condition	Standard Value			Unit	Applicable terminal
			min.	typ.	max.		
Power Voltage	$V_{DD}$	....	4.75	5	5.25	V	$V_{DD}$
Input H-level voltage	$V_{IH}$	....	2.2	...	$V_{DD}$	V	RS,R/W,E
Input L-level voltage	$V_{IL}$	....	-0.3	...	0.6	V	DB0~DB7
Output H-level voltage	$V_{OH}$	$I_{OH}=0.205mA$	2.4	...	...	V	DB0~DB7
Output L-level voltage	$V_{OL}$	$I_{OL}=1.2mA$	...	...	0.4	V	RS,R/W,E
I/O leakage current	$I_{IL}$	$V_{IN}=0-V_{DD}$	-1	...	1	$\mu A$	DB0~DB7
Supply current	$I_{DD}$	$V_{DD}=5V$	...	0.35	0.6	mV	$V_{DD}$
LCD operation voltage	$V_{LDC}$	$V_{DD}-V_0$	3	...	11	V	$V_0$

## ■ Timing chart

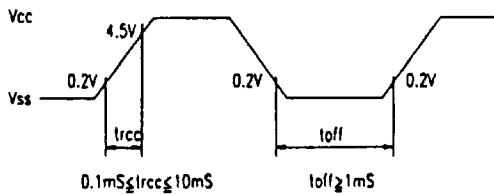
$VDD=5V \pm 5\%$   
 $VSS=0V, Top=0-50^\circ C$

Item	Symbol	Min.	Max.	Unit
Enable cycle time	$t_{CYCE}$	500	...	ns
Enable pulse width "High" Level	$P_{WEH}$	220	...	ns
Enable rise/fall time	$t_{ER}, t_{EF}$	...	25	ns
Set-up time RS,R/W-E	$t_{AS}$	40	...	ns
Address hold time	$t_{AH}$	10	...	ns
Data set-up time	$t_{DSW}$	60	...	ns
Data delay time	$t_{DDR}$	60	120	ns
Data hold time (writing)	$t_H$	10	...	ns
Data hold time (reading)	$t_{DHR}$	20	...	ns
Colck oscillating frequency	$t_{OSC}$	270 (TYP)		KHZ

## Power supply reset

The internal reset circle will be operated properly when the following power supply conditions are satisfied. If it is not operated properly, please perform initial setting along with the instruction.

Item	Symbol	Measuring Condition	Standard Value min.   typ.   max.	Unit
Power Supply Rise Time	trcc	—	0.1 — 10	mS
Power Supply OFF Time	toff	—	1 — —	mS



Note: toff defines period that power supply is off when power supply shut down momentarily or repeats on/off state.

## Reset function

### Initialization mode by Internal Reset Circuit

HD44780 automatically initializes (resets) when power is supplied (built-in internal reset circuit). The following instructions are executed in initialization. The busy flag (BF) is kept in busy state until initialization ends. (BF=1)

The busy state is 10mS after Vdd reach to 4.5V.

1. Display clear

2. Function set

DL=1: 8bit long interface data

DL=0: 4bit F=0: 5 × dot character font

N=1: 2 lines

N=0: 1 lines

3. Display ON/OFF control

D=0: Display OFF C=0: Cursor OFF B=0: Blink OFF

4. Entry mode set

I/D=1: +1 (increment) S=0: No shift

Note: When conditions stated in "Power Supply Conditions Using Internal Reset Circuit" are not satisfied, the internal reset circuit will not operate properly and initialization will not be performed. Please make initialization using MPU along with Initialization (along with) instructions.

### Initialization along with instruction

If power supply conditions are not satisfied, which for proper operation of internal reset circuit, it is required to make initialization along with instruction. Please make following procedures:

