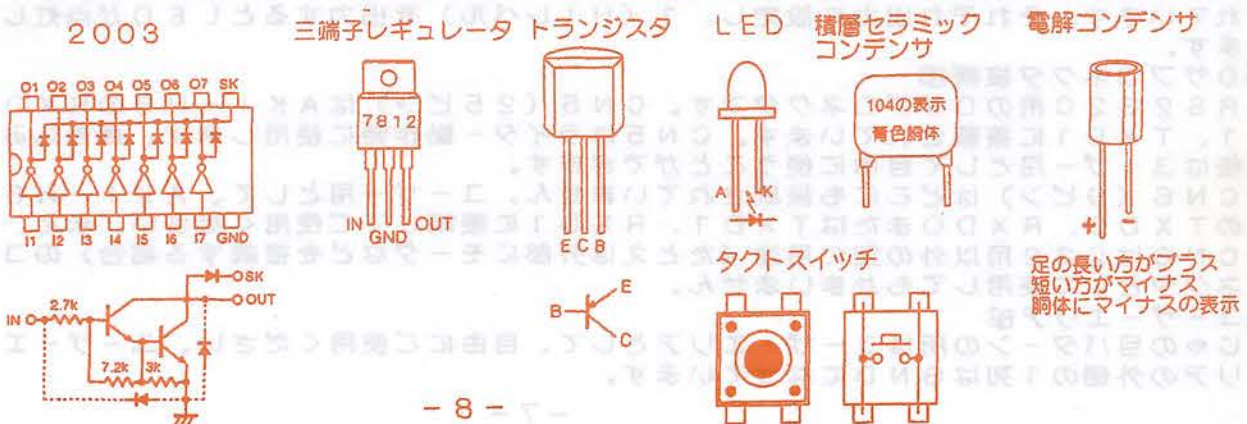


◆各I/O部品とAKI-H8I/Oとの接続表

	I/O部品	コネクタピン	I/O名称	備考
スイッチ	S1	CN3-3	P4-4	押すと0、離すと1
	S2	CN3-4	P4-5	押すと0、離すと1
	S3	CN3-5	P4-6	押すと0、離すと1
	S4	CN3-6	P4-7	押すと0、離すと1
	S5-1	CN3-23	P2-0	ONで0、OFFで1
	S5-2	CN3-24	P2-1	ONで0、OFFで1
	S5-3	CN3-25	P2-2	ONで0、OFFで1
	S5-4	CN3-26	P2-3	ONで0、OFFで1
	S5-5	CN3-27	P2-4	ONで0、OFFで1
液晶表示部	D4	CN3-7	P3-0	データ信号D4
	D5	CN3-8	P3-1	データ信号D5
	D6	CN3-9	P3-2	データ信号D6
	D7	CN3-10	P3-3	データ信号D7
	RS	CN3-11	P3-4	コマンド・データ選択信号
	E	CN3-12	P3-5	イネーブル信号
LED	LED1	CN3-31	P5-0	1で点灯、0で消灯
	LED2	CN3-32	P5-1	1で点灯、0で消灯
	LED3			電源SW(S6)オンで点灯
Dサブコネクタ	CN5-2	CN4-4	RXD1	パソコンのTXDに接続
	CN5-3	CN4-6	TXD1	パソコンのRXDに接続
	CN5-4	接続		パソコンのCTSに接続
	CN5-5	接続		パソコンのRTSに接続
	CN5-7	GND		パソコンのSGに接続
	CN6-1	無接続(ユーザーに開放)		CD(参考)
	CN6-2	無接続(ユーザーに開放)		RXD(参考)
	CN6-3	無接続(ユーザーに開放)		TXD(参考)
	CN6-4	無接続(ユーザーに開放)		DTR(参考)
CN6-5	無接続(ユーザーに開放)		SG(参考)	
CN6-6	無接続(ユーザーに開放)		DSR(参考)	
CN6-7	無接続(ユーザーに開放)		RTS(参考)	
CN6-8	無接続(ユーザーに開放)		CTS(参考)	
CN6-9	無接続(ユーザーに開放)		RI(参考)	



AKI-H8マイコン専用マザーボード

★各種応用製作のマザーボードに またAKI-H8マイコンの学習に最適です
★便利な内蔵フラッシュROMオンボード書き込みライター機能付き

- ◆ご好評をいただいておりますAKI-H8マイコンキットにライター機能付属のマザーボードができました。
 - ◆ライター基板のめんどろなパターン配線が不要です。
 - ◆オンボード書き込みのできるマザーボードとして書き込み後、スイッチ切り替えで即プログラムが動作します。
 - ◆液晶表示機、タクトスイッチ、ディップスイッチ、LED等がはじめから付けられる様にパターンが出来ていますので、応用機器の製作やH8マイコンの学習に便利です。ユーザーエリアとして、じゃの目パターン部が用意されています。
 - ◆キットの種類により液晶モジュールが付いていない場合があります。(液晶は取り付けなくても、付属の部品のみでライター付きマザーボードとして機能します。) 液晶モジュールが必要な場合は、別途ご購入ください。
 - はじめにお読みください■
- AKI-H8マイコンボード(専用フラッシュROMライター付き)で購入の方は、ユニバーサル基板でのライター製作は行わず、このマザーボードを製作してください。AKI-H8取説の■部品表(ライター)の部品はこのマザーボードに使用します。(Cタイプユニバーサル基板等部品が余る場合があります。) また、AKI-H8の取説にしたがってアセンブラソフト、ライターソフトを用意してください。

■部品表■

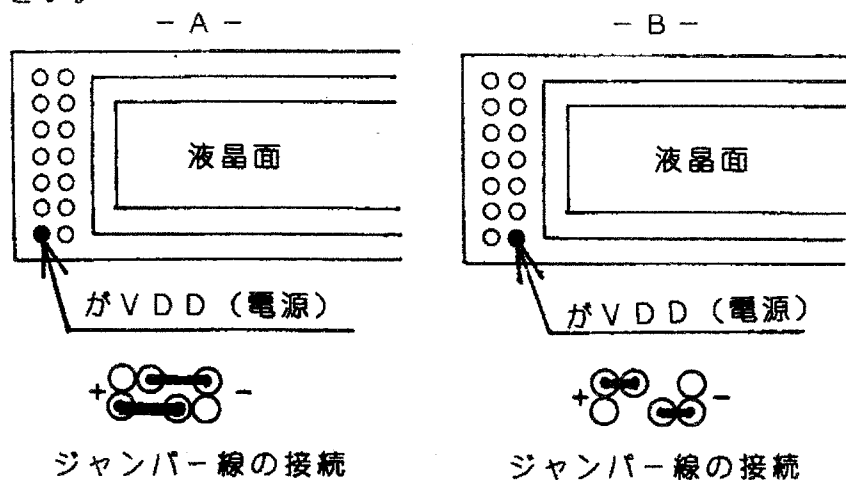
種類	番号	品名	数	備考(表示)
IC	IC1	7812	1	12V 3端子(7812, 2940-CT 12等)
	IC2	2003(MCT1413P)	1	7回路入りトランジスタアレ-(2003, MCT1413)
コンデンサ	C1~C5	0.1~0.22μF	5	積層セラミックコンデンサ(104~224)
	C6~C8	10μF	3	電解コンデンサ(10μF)
	LED	LED1~3	3	発光ダイオード
抵抗	R1~R5	4.7KΩ	5	1/4Wカーボン抵抗(黄紫赤金)
	R6	47KΩ	1	1/4Wカーボン抵抗(黄紫橙金)
	R7	10KΩ	1	1/4Wカーボン抵抗(茶黒橙金)
	R8, R9	1.5KΩ	2	1/4Wカーボン抵抗(茶緑赤金)
	トランジスタ	Q1	PNPトランジスタ	1
半固定抵抗	VR1	10KΩ	1	半固定抵抗(103または10K)
コネクタ	CN1, CN3	ピンフレーム	2	40P用
	CN2	ピンフレーム	1	20P用
	CN4	ピンフレーム	1	10P用
	CN5	DSUB25メス	1	25P Dサブ メス
	CN6	DSUB9メス	1	9P Dサブ メス
	スイッチ	S1~S4	タクトスイッチ	4
S5		ディップスイッチ	1	8Pディップスイッチ
S6, S7		トグルスイッチ	2	2回路2接点
基板			AE-H8MB	1
液晶表示機		16文字2行又は20文字4行	1	キットの種類により、付いていないキットがあります。

■製作■

①基板製作

(部品表、回路図、部品実装図、接続図、各部品の説明をあわせてごらんください。)

- 1、製作前に部品表と照らし合わせて各部品、数を確認してください。
- 2、抵抗、半固定抵抗を取り付けます。抵抗は「」形に足を加工し、ねかせて取り付けます。抵抗の切り取ったリードはあとでジャンパー線として使用します。半固定抵抗VR1は1本足側が基板の外側になるように取り付けてください。
- 3、コンデンサを取り付けます。青いコンデンサが積層セラミックコンデンサC1~C5です。円筒の2本足の物が電解コンデンサC6~8です。電解コンデンサには、極性が有り、足の長いほうが+ですので基板印刷の+マークにあわせてください。
- 4、LEDを取り付けます。LEDは足の長いほうがアノードです。基板印刷のアノード→←カソードにあわせて取り付けてください。
- 5、コネクタを取り付けます。CN1~CN4はAKI-H8ボードのCN1~CN4と対になります。CN5、6の固定用ネジは付属していませんので、お手持ちに無い場合はピンを半田付けするだけでかまいません。液晶表示機のピンヘッダ、ピンフレームは液晶表示機の袋に入っていますのでピンフレームをマザーボードに取り付けてください。ピンヘッダは液晶表示機の液晶面の反対側にとりつけ、液晶面側から半田付けしてください。
- 6、スイッチを取り付けます。タクトスイッチS1~S4の足が5本の物が入っていた場合は余分な1本はあらかじめ切ってください。ディップスイッチS5は「ON」の表示がLED1、2側になるように取り付けてください。トグルスイッチS6、S7は基板にはつきません。②スイッチの接続で使用します。
- 7、トランジスタ、ICを取り付けます。トランジスタQ1は表示面がCN3側になる様に取り付けます。AKI-H8ボードの下になりますので当たらないように低く取り付けてください。IC1 7812は表示面がLED3側になるように取り付けてください。IC2 2003は基板印刷の切り欠きマークに向きをあわせて取り付けてください。
- 8、ジャンパー線J1~8を取り付けます。メッキ線または、抵抗の切り取ったリードをご使用ください。液晶表示機用の電源・GND接続ジャンパーを取り付けます。液晶表示機は機種により、電源ピン、GNDピンの位置が逆の物があります。このマザーボードでは、電源・GNDの接続ジャンパーによりどちらにも対応出来るようになっていています。液晶表示機の取説をごらんになり、下図A、Bどちらかで配線してください。



- ②スイッチ、電源の接続 (回路図、接続図をあわせてごらんください)
スイッチS6、S7は基板印刷のS6、S7から、ビニール線等で配線します。電源は基板印刷の+、-の所にビニール線等で配線してください。

扱すると、MBTEST、MOTが転送されます。
7、100%転送後電源SW(S6)をオフにしてください。これで書き込みは終了です。

- ⑤プログラム「MBTEST、MOT」を動作させる。
S7をオフにし、その後電源SW(S6)をオンするとMBTEST、MOTが動作します。液晶表示機を付けている場合は1行目にタイトルが表示され、2行目にスイッチS1~S5の状態が表示されます。
S1~S4を押してください。表示が1→0に変わります。
S3、S4を押すと液晶表示と同時にLED1、2が点灯します。
S5の各スイッチを操作すると表示1→0または、0→1が変化します。
液晶表示機が無い場合は確認できる動作はS3、S4、LED1、2のみです。
以上が「MBTEST、MOT」の動作です。
ユーザープログラムで各種文字を表示させるには、■液晶表示機の使い方■および、各液晶表示機の取説をごらんください。

■マザーボード各部の説明■ (回路図をあわせてごらんください。)

- ①電源部
外部電源14Vから、AKI-H8およびライター部用の12Vを作る回路です。S6をオンにすることで、AKI-H8、ライター部に12Vが供給されます。また回路全体のインジケータとして、LED3が点灯します
- ②ライター部
ライター用のタイミング生成、VPP供給回路です。S7はライター書き込み時のみあらかじめ、オンさせます。通常動作時はオフのまま使用します。
- ③AKI-H8部
AKI-H8を差し込む所です。CN1~3の所にユーザー回路接続用パターンが来ています。AKI-H8は各コネクタの外側が1番ピン(奇数番ピン)内側が2番ピン(偶数番ピン)ですが、ユーザー回路接続用コネクタは内側が1番ピン(奇数番ピン)外側が2番ピン(偶数番ピン)と逆になっています。また、5V、12Vも出ています。
- ④スイッチ部
タクトスイッチS1~S4がI/OポートP4-4~P4-7に接続されています。ディップスイッチS5がI/OポートP2-0~P2-7に接続されています。これらのポートは入力に設定し、内部プルアップをオンにしてご使用ください。(詳しくはサンプルソフトをごらんください。)
タクトスイッチは押すとオンで0(Lowレベル)、離すとオフで1(Highレベル)になります。
ディップスイッチは表示のON側にすると0(Lowレベル)、オフ側(逆側)で1(Highレベル)になります。
- ⑤液晶表示部
16文字×2行、40文字×4行等の液晶表示機が接続されます。液晶のデータ線D4~D7がI/OポートP3-0~P3-3に接続されています。またRS信号(データ・コマンド選択信号)がP3-4に、E信号(イネーブル)がP3-5に接続されています。この6本の信号でAKI-H8が液晶表示機をコントロールし文字を出すことができます。詳しくはサンプルソフトをごらんください。
VR1は、液晶表示機のコントラスト調整用です。
- ⑥LED部
LED1がI/OポートP5-0に接続、LED2がI/OポートP5-1に接続されています。それぞれ出力に設定し、1(Highレベル)を出力するとLEDが点灯します。
- ⑦Dサブコネクタ接続部
RS232C用のDサブコネクタです。CN5(25ピン)はAKI-H8のRXD1、TXD1に接続されています。CN5はライター動作時に使用します。書き込み後はユーザー用として自由に使うことができます。
CN6(9ピン)はどこにも接続されていません。ユーザー用として、AKI-H8のTXD0、RXD0またはTXD1、RXD1に接続してご使用ください。また、CN6は232用以外の別の用途(たとえば外部にモータなどを接続する場合)のコネクタとして使用してもかまいません。
- ⑧ユーザーエリア部
ジャの目パターンの所はユーザーエリアとして、自由にご使用ください。ユーザーエリアの外側の1列はGNDになっています。

■マザーボード使い方■

このボードはユーザーが自由に使えるようになっています。ユーザーソフトは、各自が製作しなければいけません。まず、サンプルソフトMBTESTを動作させてみましょう

①電源について

電源は14V以上 200mA以上取れるものがが必要です。接続は接続図に従い基板印刷の+、-に接続してください。

電源にスイッチング電源 0KB-1705-28を使用する場合は、R204 (500ΩVR) を左いっぱいにまわし、電圧を上げてください。また、電源基板は、必ずケースに入れてご使用ください。

②S6、S7について

S6は電源スイッチ、S7はライターモード・通常動作モードの切替えスイッチです。

★注意★ S7がオンになると書き込み電圧12VがH8-3048Fに供給されますので、S6オン(電源が入っている状態)の時に絶対にS7を切り替えてはいけません。かならずS6オフ(電源が入っていない状態)の時に切り替えてください。(間違ってもすぐに壊れることはありませんが、書き込みプログラムが不安定になる可能性があります。)ライターモードでプログラムを書き込み後、いったんS6をオフにし、その後S7を通常モードに切り替え、電源SWS6をオンにしてプログラムを動作させる という手順で行なってください。また、S6オン(電源が入っている状態)時にAKI-H8ボードを抜き差ししてはいけません。

スイッチ		状態		状態
S6	OFF	S7	-	電源OFF (この時だけS7を切替えても良い)
	ON	S7	OFF	通常動作 (ユーザープログラムが動作する。)
			ON	ライターの書き込みモードになる。

③プログラム「MBTEST. MOT」を用意する。

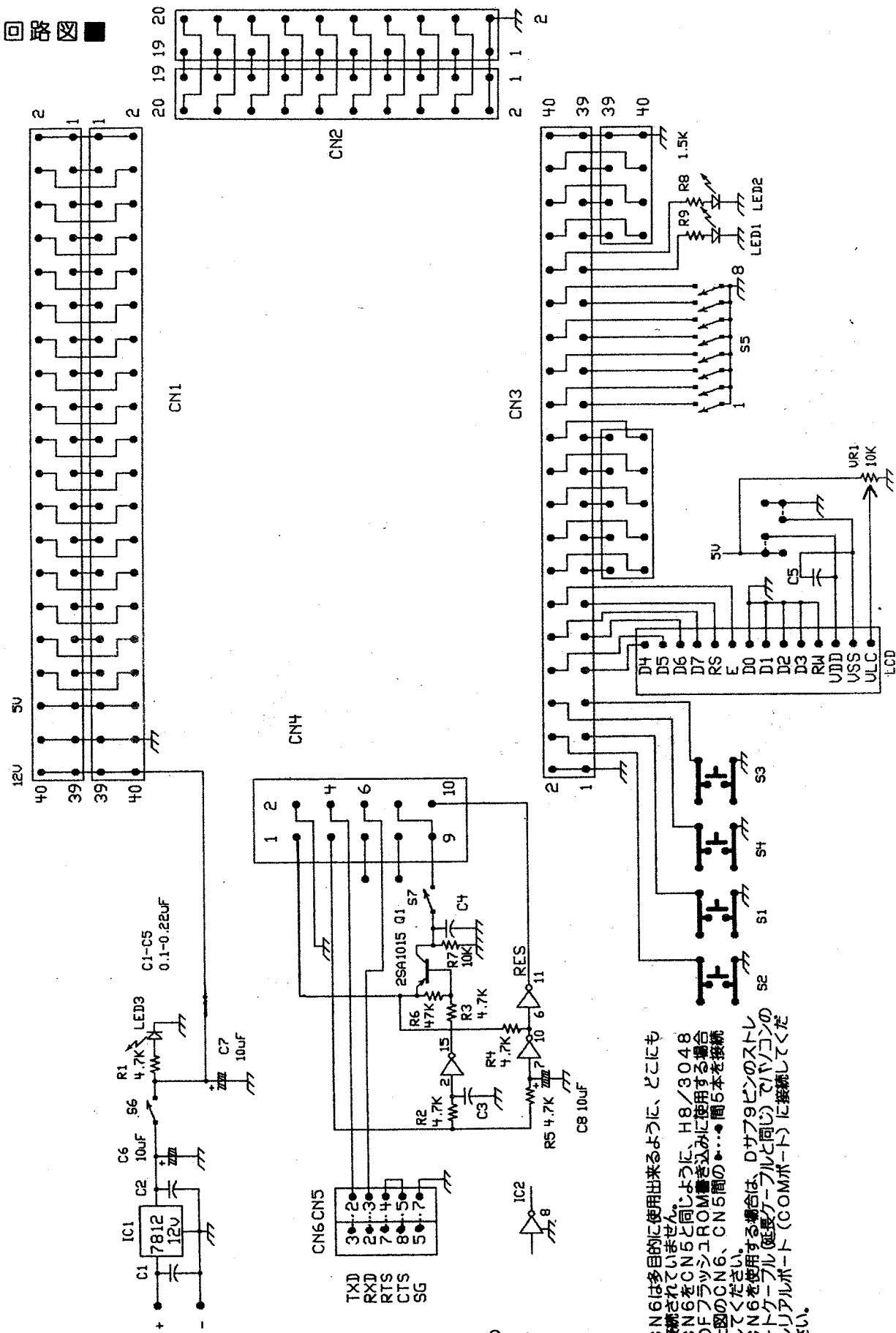
1. テストプログラムソースファイル MBTEST. MARを用意する。本取説巻末のMBTEST. MARをパソコンのエディターなどで打ち込むか、または当社インターネットホームページ等で入手してください。
2. アセンブラA38H. EXE、リンカーL38H. EXE、コンバータC38H. EXEを用意してください。AKI-H8マイコンボード(ライター、コントロールソフト、アセンブラ付きのもの)に付属しています。これらのソフトを使用してライター用HEXファイルMBTEST. MOTをつくってください。

A38H. EXE MBTEST. MAR → MBTEST. OBJができる。
L38H. EXE MBTEST. OBJ → MBTEST. ABSができる。
C38H. EXE MBTEST. ABS → MBTEST. MOTができる。

④プログラム「MBTEST. MOT」をAKI-H8に書き込む。

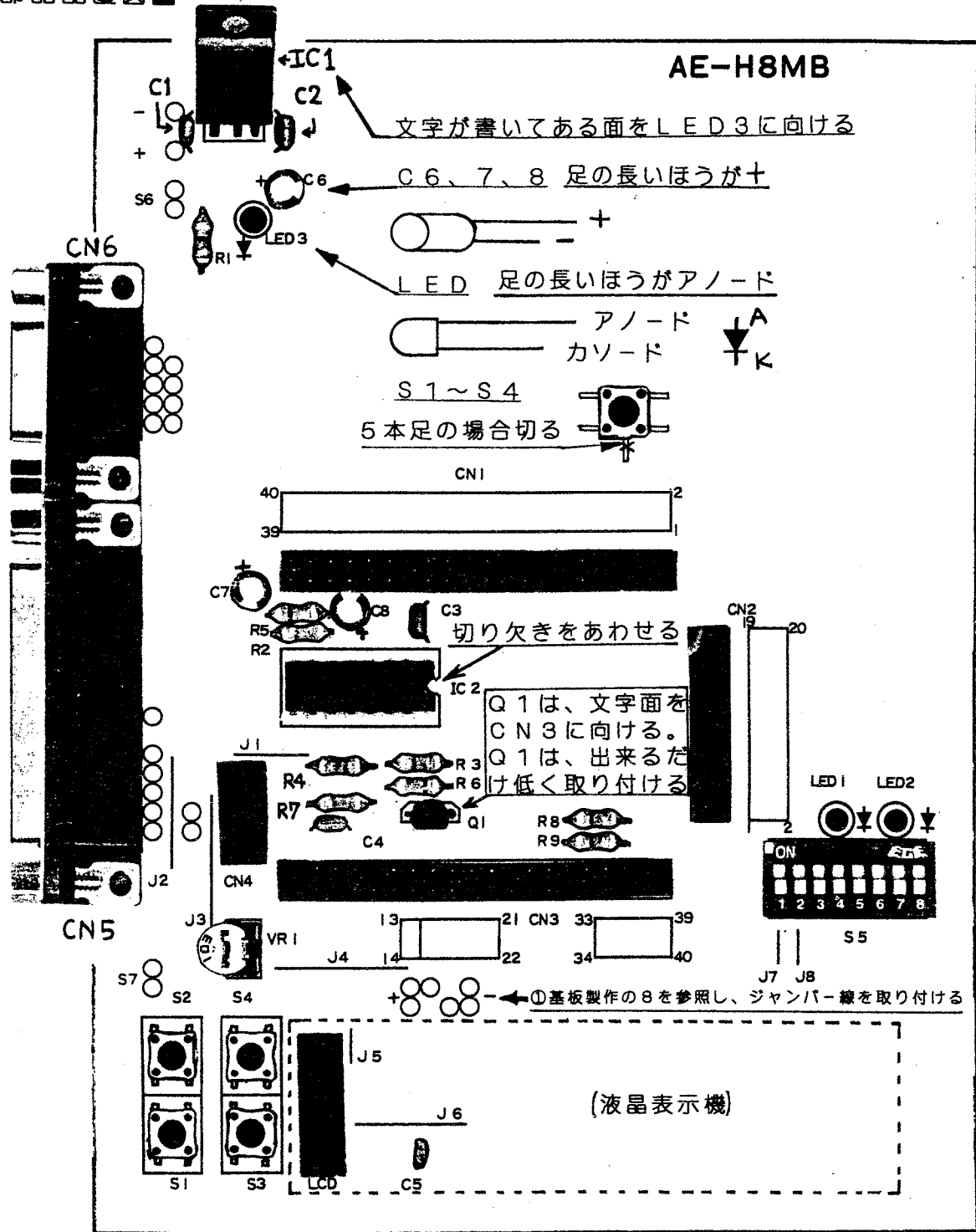
1. ROMライターソフトFLASH. EXEをAKI-H8マイコンボード(ライター、コントロールソフト、アセンブラ付きのもの)付属のフロッピーディスクからWINDOWS上でインストール(セットアップ)し用意してください。
2. マザーボードにAKI-H8マイコンボードを差し込み、CN5をパソコンのCOMポート(シリアルポート)にDサブストレートケーブル(外付けモデムなどの接続につかわれる物)で接続してください。
3. 電源SW(S6)がオフになっている事を確認し、電源を接続してください。
4. WINDOWSからF-ZTAT(FLASH. EXE)を起動してください。「起動」ウィンドウが開いたら、「フラッシュメモリロック情報ファイル」を3048. infと指定し「設定」を選択してください。3048. infの指定は2回目以降は、必要ありません。
5. 「ブートモード設定」ウィンドウが開いたら、S7をオンにし、その後電源SW(S6)をオンにしてください。マザーボードのLED3点灯確認後「ブートモード設定」ウィンドウの「OK」を選択してください。書き込み制御プログラムが転送され、100%まで転送後「ブートモード設定」ウィンドウが閉じます。
6. メニューバーの「WRITE」を選択し、「WRITEコマンド」ウィンドウを開きます。「参照」でファイル名「MBTEST. MOT」を指定し「OK」を選択してください。ファイル名の所がMBTEST. MOTとなっていることを確認し「OK」を選

■回路図■

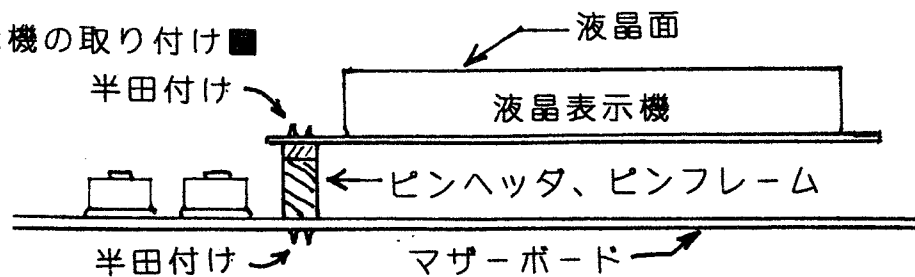


CN6は多目的に使用出来るように、どこにも接続されていません。H8/8048のCN6をCN5と同じように、H8/8048のCN6をCN5と同じように使用する場合、上のCN6、CN5間の... 同5本を接続してください。Dサブストレートの場合は、Dサブ9ピンのストリートをケーブル(延長ケーブルと同じ)でパソコンのシリアルポート(COMポート)に接続してください。

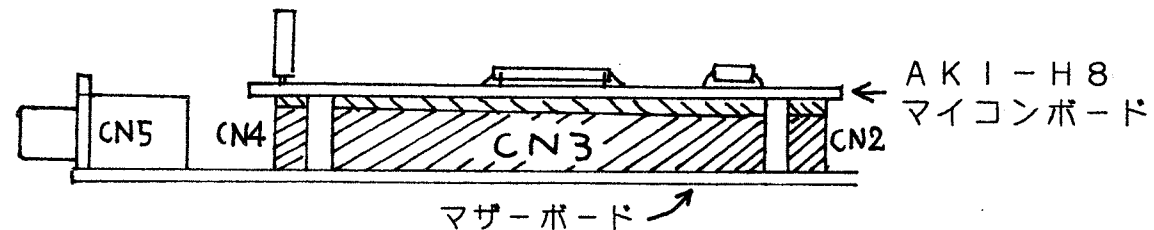
■ 部品配置図 ■



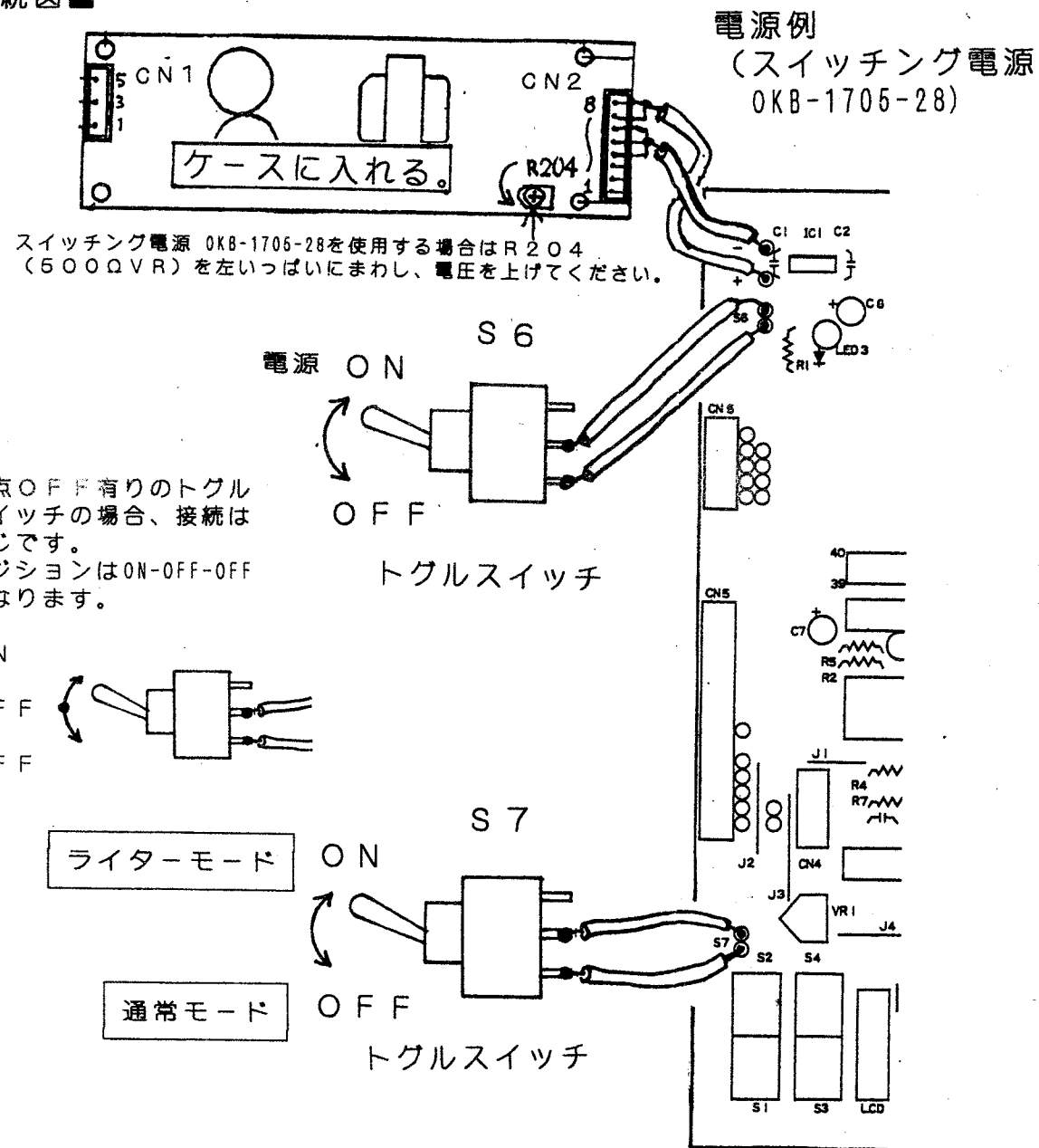
■ 液晶表示機の取り付け ■



■ AKI-H8の取り付け ■



■ 接続図 ■



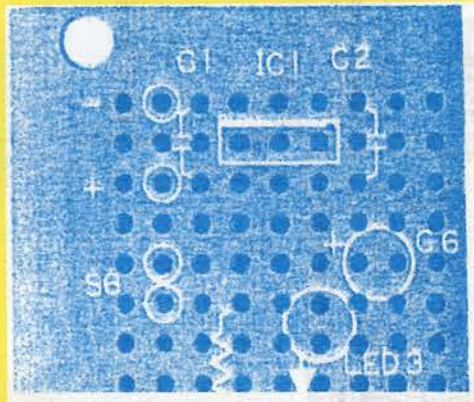
AKI-H8/3052F 用マザーボード

- ★このキットはマイコンボードが H8/3052F (25MHz) になっています。(セットによっては、H8/3052Fマイコンボード無し場合があります。)
- ★3052は、3048に比べ ROM, RAM 容量が増えています。また書き込み電圧が 5V です。
- ★3052基板のピン配置は3048基盤とおなじです。
- ★RS232C通信など速度に依存するソフトを除き、ほとんどの3048用ソフト動作します。

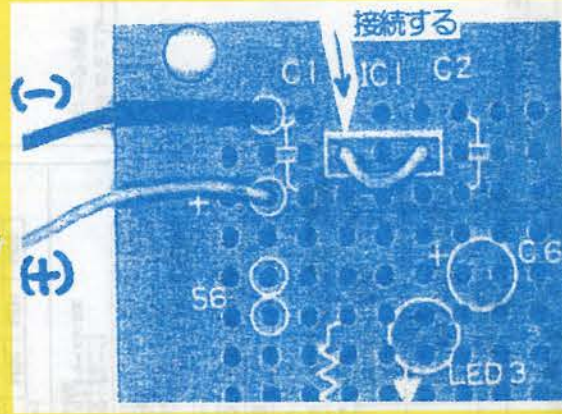
■マザーボード基板の製作■

H8/3052Fマイコンボードは、書き込み電圧が5Vですので、H8/3052Fマイコンボード上の5V3、端子レギュレータ (IC4) の5Vを書き込み電圧5Vとして使用します。そのため、マザーボードの12V3端子レギュレータ (IC1) は使用せず、外部電源として、5. 5V~9Vを使用します。その他の部品はH8-3048用と同じです。ページ1の部品表をごらんください。回路図はH8-3048用と一部違います。この用紙の『3052マザーボード回路図』をごらんください。J1の変更とCN4の8番ピン、9番ピンのパターン変更があります。

- マザーボードの12V3端子レギュレータ (IC1) は取り付けません。IC1の入力と出力をメッキ線などで接続 (ショート) し、外部から5. 5V~9Vを入力してください。

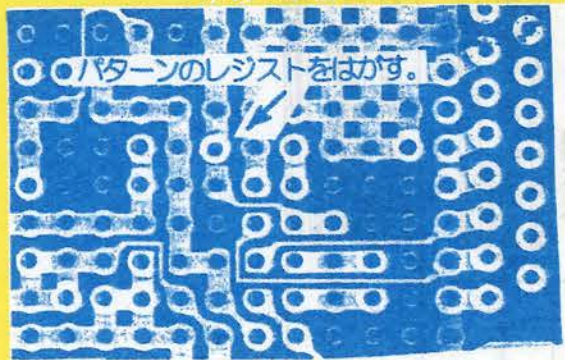


外部電源
5. 5V~9V

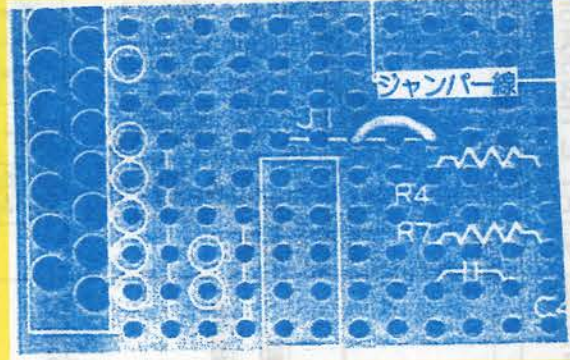


- ページ2『①基板製作』にしたがって、マザーボード基板を製作してください。8のジャンパー線J1は下図の様にパターン面のレジストをはがし、+5Vラインに接続してください。

パターン面

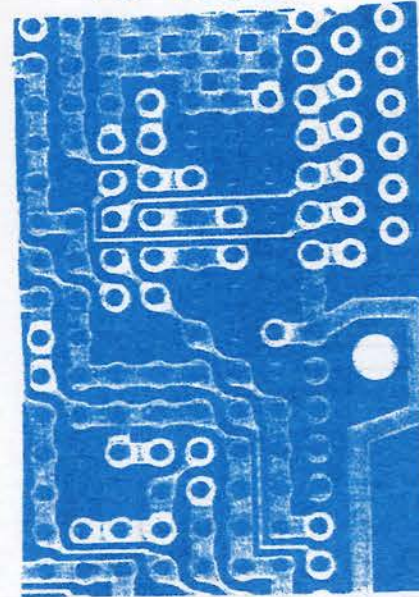


表面

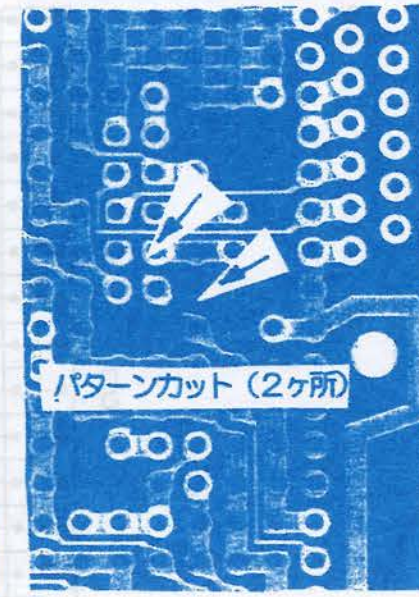


- CN4、S7を取り付け後、CN4の8番ピン、9番ピンのパターンをカットし、S7と、ビニール線等で接続してください。

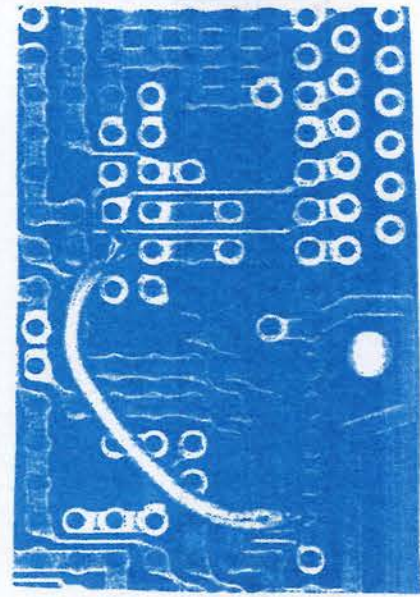
パターンカット前



パターンカット後



ビニール線で接続

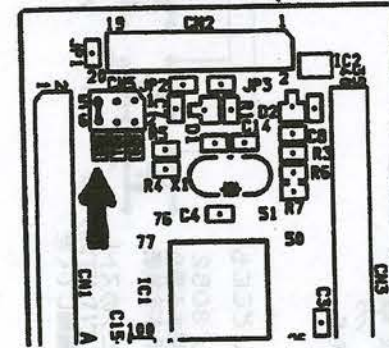


■書き込みと動作■

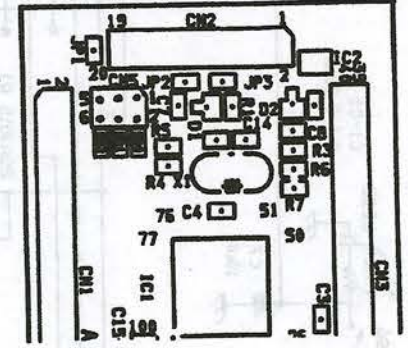
H8/3052Fは書き込み時、動作時に、マザーボードS6、S7とマイコンボードCN5を設定する必要があります。

	S6	S7	CN5 MD2
書き込み	ON	ON	ショート
通常動作	ON	OFF	オープン

書き込み時 (ブートモード7)



動作時 (モード7)

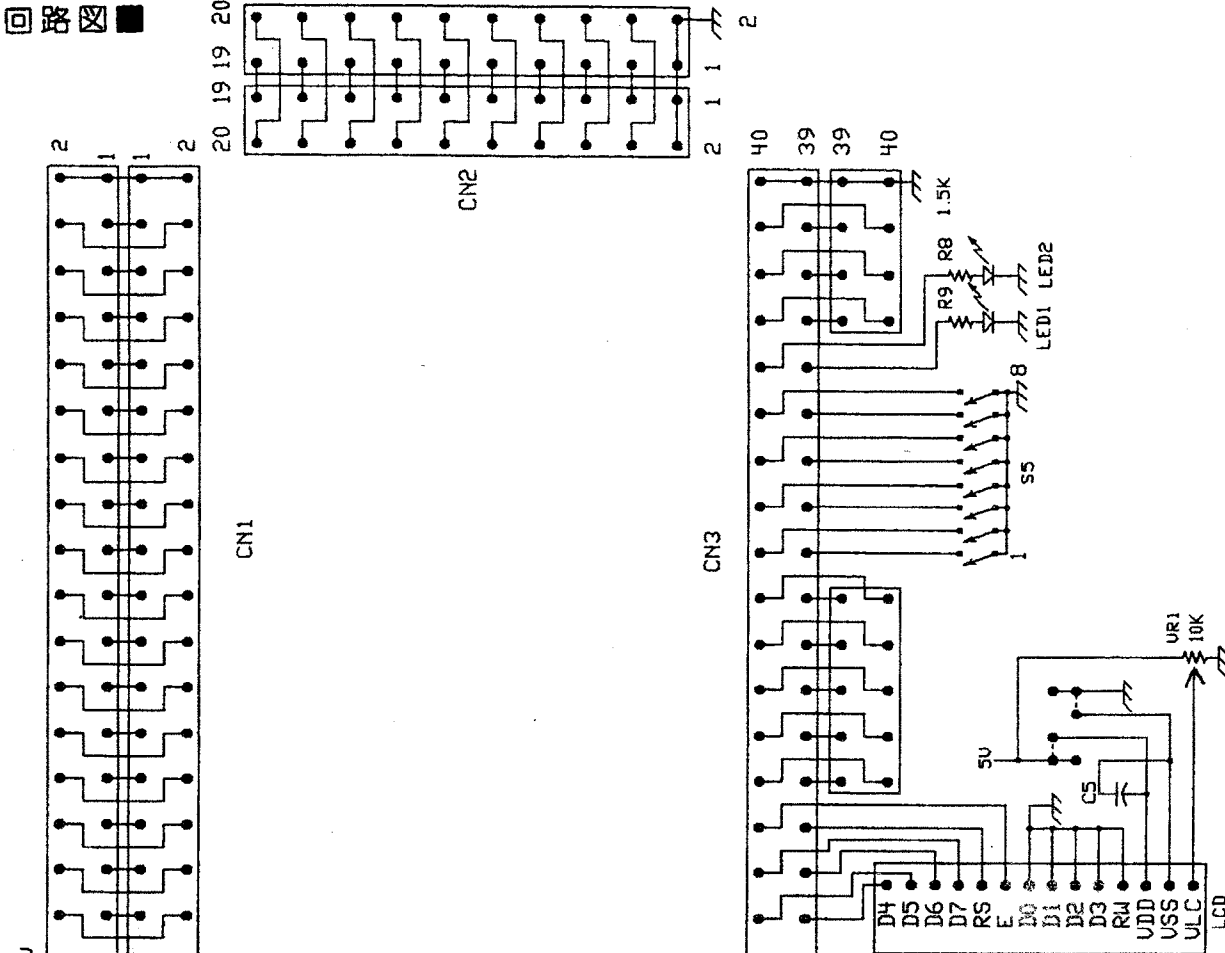


■ライターソフト■

H8/3052Fは書き込みアルゴリズムが3048と違う為、3048用のFLASH.EXEでは書き込みができません。新しい『ターポライター』になります。この『ターポライター』は、ブートモード7専用です。『ターポライター』はH8-3052専用CDのh8wtフォルダのSETUPを実行するとインストールされます。使い方は、インストール後にヘルプをごらんください。

■サンプルソフト■

書き込みチェックとマザーボード動作チェックを兼ねて、H8-3052専用CDのASMフォルダのMBTEST.MOTを書き込み、チェックしてください。S1~S5の状態がLCDに表示されます。



IC1 (12V3端子レギュレーター) は、
取り付けず、INとOUTを接続する。

CN6は多目的に使用出来るように、どこにも
接続されていません。
CN6をCN5と同じように、H8/3052
のFフラッシュROM書き込みを使用する場合
上図のCN6、CN5間の...間5本を接続
してください。
CN6を使用する場合は、Dサブ9ピンのストレ
ートケーブル (延長ケーブルと同じ) でVCONの
シリアルポート (COMポート) に接続してくだ
さい。

H8/3052F-ZTAT と H8/3048F-ZTAT との相違点

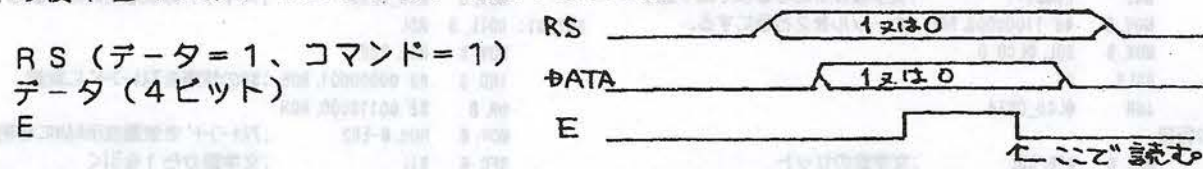
端子仕様	H8/3048F-ZTAT	H8/3052F-ZTAT	5V動作品 1ピン → V _{CC} 外付け容量 0.1µF を付けて V _{CC} に接続
ROM/RAM			3V動作品 1ピン → V _{CC} 電源電圧に接続
書き込み消去電圧	10ピン → V _{RESO}	10ピン → V _{RESO}	10ピン → FWE
V _{CC} 端子機能	FLASHメモリ 128kバイト RAM4kバイト	FLASHメモリ 128kバイト RAM8kバイト	
ブートモード設定方法	RESO端子とマルチプレクス RESO=12V	RESO端子のみ (RESO機能削除)	
ユーザプログラムモ ード設定方法			
書き込み処理	書き込み前に書き込み対象アドレスに相 当するブロックをEBR1/EBR2に設定		
FLMCR			
EBR			
RAMCR			

項目	H8/3048F-ZTAT	H8/3052F-ZTAT
FLASHメモリブロック 分割	18ブロック分割 12kバイト x 4 64kバイト x 8 512kバイト x 8 S87	18ブロック分割 12kバイト x 4 64kバイト x 7 E87 64kバイト x 7 E88-EB15
RAMエミュレーション ブロック分割	HFE00 HFE0F HFF00 HFF0F	H00000 H0000F H0F000 H0F00F H10000 H1000F H1F000 H1F00F H20000 H2000F H2F000 H2F00F H30000 H3000F H3F000 H3F00F H40000 H4000F H4F000 H4F00F H50000 H5000F H5F000 H5F00F H60000 H6000F H6F000 H6F00F H70000 H7000F H7F000 H7F00F
リフレッシュ コントロール	モード1, 2, 3, 4, 5, 6時、 エリア3にDRAMもしくはPSRAMを 直接接続可。	モード1, 2, 3, 4, 6時、 エリア3にDRAMもしくはPSRAMを 直接接続可。
DMACの MAR0AR, MAR0BR, MAR1AR, MAR1BR	MAR0AR(HFF20), MAR0BR(HFF28), MAR1AR(HFF30), MAR1BR(HFF38) 全ビットリザーブビットでリードすると 常に1が読める。ライトは無効。	MAR0AR(HFF20), MAR0BR(HFF28), MAR1AR(HFF30), MAR1BR(HFF38) 全ビットリザーブビットでリードすると 常に1が読める。ライトは無効。
AD/DのADCR	ADCR(HFFE9) 初期値 H7F ビット7のみリードライト可。 その他はリザーブビット。リードすると 常に1が読める。ライトは無効。	ADCR(HFFE8) 初期値 H7E ビット7のみリードライト可。 その他はリザーブビット。リードすると 常に1が読める。ライトは無効。
WDTのRSTCSR	RSTCSR(HFFAB) 初期値 3F ビット7, 6のみリードライト可。 その他はリザーブビット。リードすると 常に1が読める。ライトは無効。	RSTCSR(HFFAB) 初期値 3F ビット7, 6のみリードライト可。 その他はリザーブビット。リードすると 常に1が読める。ライトは無効。

【注】 本LSIの書き込み/消去フローは H8/3048F-ZTAT とは異なります。

■液晶表示機の使い方■

1、マザーボードでは、液晶表示機は4ビットモード（データ線が四本）で接続されています。1回のデータまたはコマンドを送る場合、上位4ビット（D7~D4）を送りその後下位4ビットを送ります。実際の送出波形は次のようになります。



MBTEST, MARでは、RAM上に1バイト分のデータエリア（@LCD_D）を確保しておき、『そこに送出したいデータまたはコマンドを書き込み、RSを1または0にセットし、LCD_OUT4をサブルーチンコールする』という事をしています

文字データの場合 RS = 1
コマンドの場合 RS = 0

```

;-----1文字送出-----
MOV.B #A', ROH ;文字
MOV.B ROH, @LCD_D ;文字データをRAMに入れる。
BSET RS ;文字データなのでRSを1にする。
JSR @LCD_OUT4 ;サブルーチンコール
BCLR RS ;RSを0にもどす。

```

```

;-----LCDへのデータ、コマンド出力 4bit -----
LCD_OUT4: PUSH.L ERO ;レジスタ待避

```

```

;--上位4ビット送出
BSET E_SIG ;液晶のE信号を1にする。
MOV.B @LCD_D, ROL ;データ(コマンド)をレジスタに入れる
SHLR.B ROL ;4ビットモードなので上位4ビットを
SHLR.B ROL ;下位に移動
SHLR.B ROL
SHLR.B ROL
AND.B #B'00001111, ROL ;データ線以外をマスクする。
MOV.B @P3_D, ROH ;RS信号の待避
AND.B #B'11110000, ROH ;RS信号、E信号以外をマスク
OR.B ROH, ROL ;RS信号、E信号、データ(4ビット)
;を合成する。
MOV.B ROL, @P3_D ;合成したすべての信号を液晶に出力
JSR @TIME10 ;WAIT
BCLR E_SIG ;液晶のE信号を0にする
JSR @TIME10 ;WAIT

```

```

;--下位4ビット送出
BSET E_SIG ;液晶のE信号を1にする
MOV.B @LCD_D, ROL ;データ(コマンド)をレジスタに入れる
AND.B #B'00001111, ROL ;データ線下位4ビット以外をマスクする
MOV.B @P3_D, ROH ;RS信号の待避
AND.B #B'11110000, ROH ;RS信号、E信号以外をマスク
OR.B ROH, ROL ;RS信号、E信号、データ(4ビット)を合成する
MOV.B ROL, @P3_D ;合成したすべての信号を液晶に出力
JSR @TIME10 ;WAIT
BCLR E_SIG ;液晶のE信号を0にする
JSR @TIME10 ;WAIT
POP.L ERO ;レジスタの復帰
RTS

```

2、この液晶表示機は電源オン後15ms待ち、ソフトウェアリセット、初期設定を行なう必要があります。その後文字を表示出来るようになります。

ソフトウェアリセット ファンクションセット1
ファンクションセット2
ファンクションセット3
初期設定 ファンクションセット
表示on
エントリーモード

3、液晶に対するコマンドは、表示クリア、カーソル位置指定などがあります。

```

●表示クリア
MOV.B #B'00000001, ROL
MOV.B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT4

●カーソルを2行目にする。
MOV.B #B'11000000, ROL
MOV.B ROL, @LCD_D
BCLR RS ←RS=0にしている
JSR @LCD_OUT4

```

4、スイッチなどの状態を表示する場合などは、RAMに16文字×二行分のテーブルを用意しておき、そこに表示したい文字をアスキーコードで書き、LCDDSPをサブルーチンコールし、1画面分表示しています。

```

--例(液晶初期画面の表示)--
LCD162 EQU H'FFEF13 ;16文字2行分のデータをいれるRAMアドレス

MOV.B #32, ROL ;文字数のセット
MOV.L #LCD162, ER1 ;液晶表示RAMのアドレスをセット
MOV.L #MOJI, ER2 ;初期文字データのアドレスをセット
SYOKIO: MOV.B @ER2+, ROH ;初期文字データをレジスタに入れる。
MOV.B ROH, @ER1 ;レジスタ値を液晶表示RAMに入れる
INC.L #1, ER1 ;アドレスに1を足す。
DEC.B ROL ;文字数から1を引く
BNE SYOKIO ;文字数が0になるまでくりかえす。
JSR @LCDDSP ;サブルーチンコール
(LCDDSPの内容は、MBTEST.MARをごらんください。)

```

```

;-----文字データ-----
ALIGN 2
SECTION LCDDATA, DATA, LOCATE=H'01000
MOJI: SDATA "AKI-H8"
DATA.B H'0CF, H'0BB, H'0DE, H'0B0, H'0CE, H'0DE, H'0B0, H'0C4, H'0DE
SDATA "1111 11111111"
END

```

●画面表示とRAMアドレス

画面位置(1行目)	1	2	...	15	16
RAMアドレス	@LCD162+0	@LCD162+1	...	@LCD162+14	@LCD162+15
画面位置(2行目)	1	2	...	15	16
RAMアドレス	@LCD162+16	@LCD162+17	...	@LCD162+30	@LCD162+31

```

サンプルソフト MBTEST, MAR
-----CPUの指定-----
CPU 300HA
-----シンボル-----
SW_D EQU H' FFEF10 ; SW1-4の状態を入れるRAM
SW_D5 EQU H' FFEF11 ; SW5の状態を入れるRAM
LCD_D EQU H' FFEF12 ; LCDへのデータ転送RAM
; (1/1バイト分)
LCD162 EQU H' FFEF13 ; 16文字2行分のデータ転送RAM
P2DR EQU H' FFFFC3 ; ポート2データレジスタ (S5)
P3_D EQU H' FFFFC6 ; ポート3データレジスタ
; LCDのデータ線
E_SIG BEQU 5, P3_D ; イネーブル信号
RS BEQU 4, P3_D ; RS信号
P4DR EQU H' FFFFC7 ; ポート4データレジスタ
P5DR EQU H' FFFFC8 ; ポート5データレジスタ
LED1 BEQU 0, P5DR ; LED1
LED2 BEQU 1, P5DR ; LED2
-----リセットベクトル-----
SECTION RESET0, DATA, LOCATE=H' 00000
DATA, L INIT ; リセットベクトル
-----1/0の初期設定-----
SECTION ROM, CODE, LOCATE=H' 00100
INIT: MOV, L #H' FFF10, ER7 ; スタックポインタ設定
MOV, B #H' 00, ROL ; ポート2を入力に設定 (S5)
MOV, B ROL, @H' FFFFC1 ; ポート2
MOV, B #H' FF, ROL ; ポート2をプルアップに設定
MOV, B ROL, @H' FFFFCB ; ポート2
MOV, B #H' FF, ROL ; ポート3を出力に設定
MOV, B ROL, @H' FFFFC4 ; ポート3
MOV, B #H' 00, ROL ; ポート4を入力に設定
MOV, B ROL, @H' FFFFC5 ; ポート4
MOV, B #H' FF, ROL ; ポート4をプルアップに設定
MOV, B ROL, @H' FFFCDA ; ポート4
MOV, B #H' FF, ROL ; ポート5を出力に設定
MOV, B ROL, @H' FFFFC8 ; ポート5
-----LCDの初期設定-----
ソフトウェアリセット-----
JSR @TIME00 ; 15msのWAIT (4ms*4)
JSR @TIME00
JSR @TIME00
JSR @TIME00
MOV, B #B' 00100011, ROL ; LCDファンクションセット1
MOV, B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT8
JSR @TIME00 ; 4msのWAIT
MOV, B #B' 00100011, ROL ; LCDファンクションセット2
MOV, B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT8
JSR @TIME00 ; 4msのWAIT
MOV, B #B' 00100010, ROL ; ファンクションセット
MOV, B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT8

```

```

JSR @TIME00 ; 4msのWAIT
-----初期設定-----
MOV, B #B' 00101000, ROL ; ファンクションセット
MOV, B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT4
JSR @TIME00 ; 4msのWAIT
MOV, B #B' 00001110, ROL ; 表示ON
MOV, B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT4
JSR @TIME00 ; 4msのWAIT
MOV, B #B' 00000110, ROL ; エントリーモード
MOV, B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT4
JSR @TIME00 ; 4msのWAIT
-----LCDの初期設定終了-----
液晶初期画面表示-----
MOV, B #B' 00000001, ROL ; 表示クリア
MOV, B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT4
JSR @TIME00 ; 4msのWAIT
MOV, B #B2, ROL
MOV, L #LCD162, ER1 ; 液晶表示RAMのアドレスをセット
MOV, L #MOJ1, ER2 ; 初期文字データのアドレスをセット
SYOKIO: MOV, B @ER2+, ROH ; 初期文字データをレジスタに入れる
MOV, B ROH, @ER1 ; レジスタを液晶表示RAMに入れる
INC, L #1, ER1
DEC, B ROL ; 文字数から1を引く
BNE SYOKIO ; 文字数が0になるまで繰り返す
JSR @LCDOSP
-----MAIN-----
MOV, B #0, ROL ; RAMのクリア
MOV, B ROL, @SW_D
MOV, B ROL, @SW_D5
BOTAN: MOV, B @P4DR, ROL ; SW1-4の状態の取得
MOV, B @SW_D, ROH ; RAMの内容をレジスタに読む
CMP, B ROH, ROL ; SWの状態をチェック
BEQ S5CHK ; 変化が無ければS5CHKにジャンプ
JSR @S1_4 ; 変化が有るので処理ルーチンへ
S5CHK: MOV, B @P2DR, ROL ; S5の状態を取得
MOV, B @SW_D5, ROH ; RAMの内容をレジスタに読む
CMP, B ROH, ROL ; SWの状態をチェック
BEQ BOTAN ; 変化が無ければBOTANにジャンプ
JSR @S5 ; 変化が有るので処理ルーチンへ
JMP @BOTAN
-----MAIN END-----
-----サブルーチン-----
液晶文字出力16文字*2行
LCDOSP: PUSH, L ERO
PUSH, L ER1
MOV, B #B' 00000010, ROL ; カーソルホーム
MOV, B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT4
JSR @TIME00 ; 4msのWAIT
-----1行目-----
MOV, B #16, ROL ; 文字数のセット
MOV, L #LCD162, ER1 ; 液晶表示RAMのアドレスをセット
LCDOSP1: MOV, B @ER1+, ROH ; 文字データをレジスタに入れる
MOV, B ROH, @LCD_D ; 文字データをRAMに入れる。

```

```

BSET RS ; データなのでRSを1にする。
JSR @LCD_OUT4 ; サブルーチンコール
BCLR RS ; RSを0にもどす。
DEC, B ROL ; 文字数から1を引く
BNE LCDOSP1 ; 文字数が0になるまで繰り返す
MOV, B #B' 11000000, ROL ; カーソルを2行目にする。
MOV, B ROL, @LCD_D
BCLR RS
JSR @LCD_OUT4
-----2行目-----
MOV, B #16, ROL ; 文字数のセット
MOV, L #LCD162+16, ER1 ; 液晶表示RAMのアドレスをセット
LCDOSP2: MOV, B @ER1+, ROH ; 文字データをレジスタに入れる
MOV, B ROH, @LCD_D ; 文字データをRAMに入れる。
BSET RS ; データなのでRSを1にする。
JSR @LCD_OUT4 ; サブルーチンコール
BCLR RS ; RSを0にもどす。
DEC, B ROL ; 文字数から1を引く
BNE LCDOSP2 ; 文字数が0になるまで繰り返す
POP, L ER1
POP, L ERO
RTS
-----LCDへのデータ、コマンド出力-----
LCD_OUTB: PUSH, L ERO ; レジスタ待避
BSET E_SIG ; 液晶のE信号をセットする。
MOV, B @LCD_D, ROL ; データ(コマンド)をレジスタに入れる
MOV, B ROL, @P3_D ; 液晶にデータ(コマンド)を出力
JSR @TIME10 ; WAIT
BCLR E_SIG ; 液晶のE信号を0にする。
JSR @TIME10 ; WAIT
POP, L ERO ; レジスタの復帰
RTS
-----LCDへのデータ、コマンド出力 4bit -----
LCD_OUT4: PUSH, L ERO ; レジスタ待避
-----上位4ビット送出-----
BSET E_SIG ; 液晶のE信号を1にする。
MOV, B @LCD_D, ROL ; データ(コマンド)をレジスタに入れる
SHL, B ROL ; 4ビットモードなので
SHL, B ROL ; 上位4ビットを下位に移動
SHL, B ROL
AND, B #B' 00001111, ROL ; データ線以外をマスクする。
MOV, B @P3_D, ROH ; RS信号の待避
AND, B #B' 11110000, ROH ; RS信号、E信号以外をマスク
OR, B ROH, ROL ; RS信号、E信号、データ(4ビット)を合成する。
MOV, B ROL, @P3_D ; 合成した信号を液晶に出力
JSR @TIME10 ; WAIT
BCLR E_SIG ; 液晶のE信号を0にする
JSR @TIME10 ; WAIT
-----下位4ビット送出-----
BSET E_SIG ; 液晶のE信号を1にする
MOV, B @LCD_D, ROL ; データ(コマンド)をレジスタに入れる
AND, B #B' 00001111, ROL ; データ線下位4ビット以外をマスク
MOV, B @P3_D, ROH ; RS信号の待避
AND, B #B' 11110000, ROH ; RS信号、E信号以外をマスク
OR, B ROH, ROL ; RS信号、E信号、データ(4ビット)を合成する。
MOV, B ROL, @P3_D ; 合成した信号を液晶に出力
JSR @TIME10 ; WAIT
BCLR E_SIG ; 液晶のE信号を0にする
JSR @TIME10 ; WAIT
POP, L ERO ; レジスタの復帰
RTS

```

```

-----液晶表示S1-4-----
S1_4: MOV, B ROL, @SW_D ; 現在の状態をRAMに格納
MOV, B #4, R1L ; 文字数のセット
MOV, L #LCD162 + 20, ER2 ; 液晶表示アドレスをセット
MOV, B @SW_D, ROL ; スイッチの状態をレジスタに読む
BOTAN1: ROTL, B ROL
MOV, B ROL, ROH
AND, B #B' 00000001, ROH ; SWの状態をアスキーコードに変換
OR, B #B' 00110000, ROH
MOV, B ROH, @-ER2 ; アスキーコードを液晶表示RAMに格納
DEC, B R1L ; 文字数から1を引く
BNE BOTAN1 ; 文字数が0になるまで繰り返す
JSR @LCDOSP
-----LED-----
SW3LED: MOV, B @SW_D, ROL ; 在の状態をRAMに格納
BTST #6, ROL ; S3の状態をチェック
BEQ BOTAN2 ; 押されていればBOTAN2にジャンプ
BCLR LED1 ; 押されていないので消灯
JMP @SW4LED
BOTAN2: BSET LED1 ; 押されているので点灯
SW4LED: BTST #7, ROL ; S4の状態をチェック
BEQ BOTAN3 ; 押されていればBOTAN3にジャンプ
BCLR LED2 ; 押されていないので消灯
RTS
BOTAN3: BSET LED2 ; 押されているので点灯
RTS
-----液晶表示S5-----
S5: MOV, B ROL, @SW_D5 ; 現在の状態をRAMに格納
MOV, B #8, R1L ; 文字数のセット
MOV, L #LCD162 + 29, ER2 ; 液晶表示アドレスを読む
MOV, B @SW_D5, ROL ; RAMの内容をレジスタに読む
BOTAN4: ROTL, B ROL
MOV, B ROL, ROH
AND, B #B' 00000001, ROH ; SWの状態をアスキーコードに変換
OR, B #B' 00110000, ROH
MOV, B ROH, @-ER2 ; アスキーコードを液晶表示RAMに格納
DEC, B R1L ; 文字数から1を引く
BNE BOTAN4 ; 文字数が0になるまで繰り返す
JSR @LCDOSP
RTS
-----4msタイマ-----
TIME00: PUSH, L ERO
MOV, L #H' 2000, ERO
TIME01: SUB, L #1, ERO
BNE TIME01
POP, L ERO
RTS
TIME10: PUSH, L ERO
MOV, L #H' AA, ERO ; 450ns TIMER
TIME11: SUB, L #1, ERO
BNE TIME11
POP, L ERO
RTS
-----サブルーチン終了-----
-----文字データ-----
ALIGN 2
SECTION LCDDATA, DATA, LOCATE=H' 01000
SDATA ' AKI-H8 '
DATA, B #H' 0CF, #H' 0BB, #H' 0DE, #H' 0B0, #H' 0CE
DATA, B #H' 0DE, #H' 0B0, #H' 0C4, #H' 0DE
SDATA ' 1111 11111111 '
END

```