

お客様各位

カタログ等資料中の旧社名の扱いについて

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願ひ申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

ご注意書き

1. 本資料に記載されている内容は本資料発行時点のものであり、予告なく変更することがあります。当社製品のご購入およびご使用にあたりましては、事前に当社営業窓口で最新の情報をご確認いただきますとともに、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
3. 当社製品を改造、改変、複製等しないでください。
4. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器の設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りが無いことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。
標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めておりますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害などを生じさせないようお客様の責任において冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、詳細につきましては製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

注 1. 本資料において使用されている「当社」とは、ルネサスエレクトロニクス株式会社およびルネサスエレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

お客様各位

資料中の「日立製作所」、「日立XX」等名称の株式会社ルネサス テクノロジへの変更について

2003年4月1日を以って三菱電機株式会社及び株式会社日立製作所のマイコン、ロジック、アナログ、ディスクリート半導体、及びDRAMを除くメモリ(フラッシュメモリ・SRAM等)を含む半導体事業は株式会社ルネサス テクノロジに承継されました。従いまして、本資料中には「日立製作所」、「株式会社日立製作所」、「日立半導体」、「日立XX」といった表記が残っておりますが、これらの表記は全て「株式会社ルネサス テクノロジ」に変更されておりますのでご理解の程お願い致します。尚、会社商標・ロゴ・コーポレートステートメント以外の内容については一切変更しておりませんので資料としての内容更新ではありません。

ルネサステクノロジ ホームページ (<http://www.renesas.com>)

2003年4月1日
株式会社ルネサス テクノロジ
カスタマサポート部

ご注意

安全設計に関するお願い

1. 弊社は品質、信頼性の向上に努めておりますが、半導体製品は故障が発生したり、誤動作する場合があります。弊社の半導体製品の故障又は誤動作によって結果として、人身事故、火災事故、社会的損害などを生じさせないような安全性を考慮した冗長設計、延焼対策設計、誤動作防止設計などの安全設計に十分ご注意ください。

本資料ご利用に際しての留意事項

1. 本資料は、お客様が用途に応じた適切なルネサス テクノロジ製品をご購入いただくための参考資料であり、本資料中に記載の技術情報についてルネサス テクノロジが所有する知的財産権その他の権利の実施、使用を許諾するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例の使用に起因する損害、第三者所有の権利に対する侵害に関し、ルネサス テクノロジは責任を負いません。
3. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他全ての情報は本資料発行時点のものであり、ルネサス テクノロジは、予告なしに、本資料に記載した製品または仕様を変更することがあります。ルネサス テクノロジ半導体製品のご購入に当たりましては、事前にルネサス テクノロジ、ルネサス販売または特約店へ最新の情報をご確認頂きますとともに、ルネサス テクノロジホームページ (<http://www.renesas.com>)などを通じて公開される情報に常にご注意ください。
4. 本資料に記載した情報は、正確を期すため、慎重に制作したのですが万一本資料の記述誤りに起因する損害がお客様に生じた場合には、ルネサス テクノロジはその責任を負いません。
5. 本資料に記載の製品データ、図、表に示す技術的な内容、プログラム及びアルゴリズムを流用する場合は、技術内容、プログラム、アルゴリズム単位で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断してください。ルネサス テクノロジは、適用可否に対する責任を負いません。
6. 本資料に記載された製品は、人命にかかわるような状況の下で使用される機器あるいはシステムに用いられることを目的として設計、製造されたものではありません。本資料に記載の製品を運輸、移動体用、医療用、航空宇宙用、原子力制御用、海中継用機器あるいはシステムなど、特殊用途へのご利用をご検討の際には、ルネサス テクノロジ、ルネサス販売または特約店へご照会ください。
7. 本資料の転載、複製については、文書によるルネサス テクノロジの事前の承諾が必要です。
8. 本資料に関し詳細についてのお問い合わせ、その他お気付きの点がございましたらルネサス テクノロジ、ルネサス販売または特約店までご照会ください。

H8/3857 シリーズ

アプリケーションノート

ご注意

1. 本書に記載の製品及び技術のうち「外国為替及び外国貿易法」に基づき安全保障貿易管理関連貨物・技術に該当するものを輸出する場合、または国外に持ち出す場合は日本国政府の許可が必要です。
2. 本書に記載された情報の使用に際して、弊社もしくは第三者の特許権、著作権、商標権、その他の知的所有権等の権利に対する保証または実施権の許諾を行うものではありません。また本書に記載された情報を使用した事により第三者の知的所有権等の権利に関わる問題が生じた場合、弊社はその責を負いませんので予めご了承ください。
3. 製品及び製品仕様は予告無く変更する場合がありますので、最終的な設計、ご購入、ご使用に際しましては、事前に最新の製品規格または仕様書をお求めになりご確認ください。
4. 弊社は品質・信頼性の向上に努めておりますが、宇宙、航空、原子力、燃焼制御、運輸、交通、各種安全装置、ライフサポート関連の医療機器等のように、特別な品質・信頼性が要求され、その故障や誤動作が直接人命を脅かしたり、人体に危害を及ぼす恐れのある用途にご使用をお考えのお客様は、事前に弊社営業担当迄ご相談をお願い致します。
5. 設計に際しては、特に最大定格、動作電源電圧範囲、放熱特性、実装条件及びその他諸条件につきましては、弊社保証範囲内でご使用いただきますようお願い致します。
保証値を越えてご使用された場合の故障及び事故につきましては、弊社はその責を負いません。
また保証値内のご使用であっても半導体製品について通常予測される故障発生率、故障モードをご考慮の上、弊社製品の動作が原因でご使用機器が人身事故、火災事故、その他の拡大損害を生じないようにフェールセーフ等のシステム上の対策を講じて頂きますようお願い致します。
6. 本製品は耐放射線設計をしておりません。
7. 本書の一部または全部を弊社の文書による承認なしに転載または複製することを堅くお断り致します。
8. 本書をはじめ弊社半導体についてのお問い合わせ、ご相談は弊社営業担当迄お願い致します。

はじめに

H8/300Lシリーズは、高速H8/300L CPUを核にして、システム構成に必要な周辺機能を集積したシングルチップマイクロコンピュータです。

H8/300L CPUは、H8/300 CPUと互換性のある命令体系を備えています。

H8/3857シリーズは、システム構成に必要な周辺機能として、LCDコントローラ/ドライバ、4種類のタイマ、14ビットPWM、2チャンネルのシリアルコミュニケーションインタフェース、A/D変換器を内蔵しています。LCD表示を必要とするシステムの組込み用マイコンとして活用できます。

H8/3857シリーズ アプリケーションノートは、H8/3857シリーズの内蔵周辺機能を単独で使った場合の動作例を示した“基礎編”により構成されており、ユーザーにてソフトウェア設計およびハードウェア設計の際、ご参考として役立てていただけるようにまとめたものです。

なお、本アプリケーションノートに掲載されているプログラム、回路等の動作は確認しておりますが、実際にご使用になる場合は、必ず動作確認の上ご使用くださいますようお願い致します。

目次

1. H8/3857シリーズ アプリケーションノート使用手引	1
1.1 基礎編構成	2
2. 基礎編	4
2.1 ドットマトリックスLCD制御 - キャラクタ表示モード	5
2.2 ドットマトリックスLCD制御 - グラフィック表示モード	62

1. H8/3857シリーズ アプリケーションノート使用手引

目次

1.1 基礎編構成

2

本アプリケーションノートは、図1に示すように2部構成になっています。

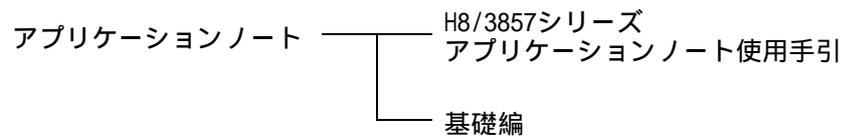


図1 アプリケーションノート構成

- (1)H8/3857シリーズ アプリケーションノート使用手引
H8/3857シリーズ アプリケーションノートの使用法について説明しています。
- (2)基礎編
H8/3857シリーズの内蔵周辺機能の使用法を簡単なタスク例をもとに説明しています。

1.1 基礎編構成

基礎編は図2に示す構成で内蔵周辺機能の使用法について説明しています。

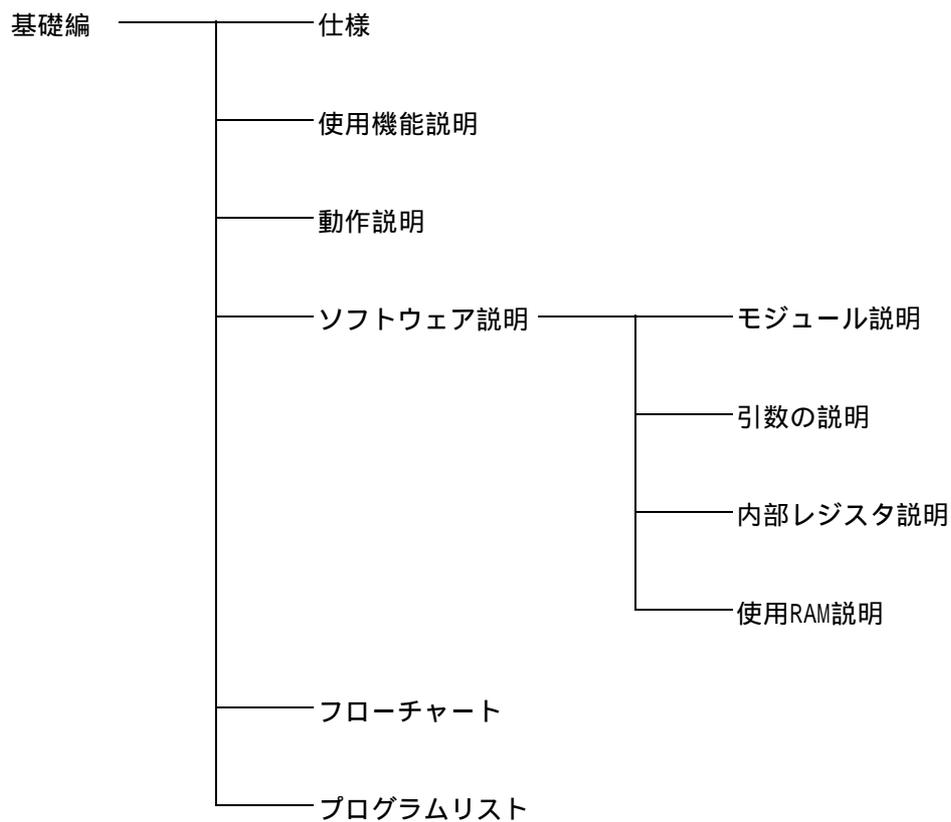


図2 基礎編構成

- (1) 仕様
タスク例のシステム仕様について説明しています。
- (2) 使用機能説明
タスク例で使用する周辺機能の特徴および周辺機能の割付けについて説明しています。
- (3) 動作説明
タスク例の動作をタイミングチャートを使用し説明しています。
- (4) ソフトウェア説明
 - (a) モジュール説明
タスク例を動作させるソフトウェアのモジュールについて説明しています。
 - (b) 引数の説明
モジュールを実行する際に必要な入力引数と、実行後の出力引数について説明しています。
 - (c) 内部レジスタ説明
モジュールで設定する周辺機能の内部レジスタ（タイマコントロールレジスタ、シリアルモードレジスタ等）について説明しています。
 - (d) 使用RAM説明
モジュールで使用するRAMのラベル名及び機能について説明しています。
- (5) フローチャート
タスク例を実行するソフトウェアについてフローチャートを使用し説明しています。
- (6) プログラムリスト
タスク例を実行するソフトウェアのプログラムリストを示しています。

2. 基礎編

目次

2.1	ドットマトリックスLCD制御 - キャラクタ表示モード	5
2.2	ドットマトリックスLCD制御 - グラフィック表示モード	62

2.1 ドットマトリックスLCD制御-キャラクタ表示モード

ドットマトリックスLCD制御 -キャラクタ表示モード	MCU	H8/3857シリーズ	使用機能	ドットマトリックス LCDコントローラ
-------------------------------	-----	-------------	------	------------------------

仕様

- (1) ドットマトリックスLCDコントローラ機能を使用して、液晶表示を行いません。
- (2) 本タスク例では40×32のドットマトリックスLCDを使用します。
- (3) 表示モードはキャラクタ表示モードで、1/5バイアス、1/32デューティによる表示を行いません。
- (4) 5×8ドットフォントで8桁4行の文字の表示、輝度調整、スクロール、およびプリンキングの制御を行います。
- (5) 本タスク例における液晶表示例を図1に示します。

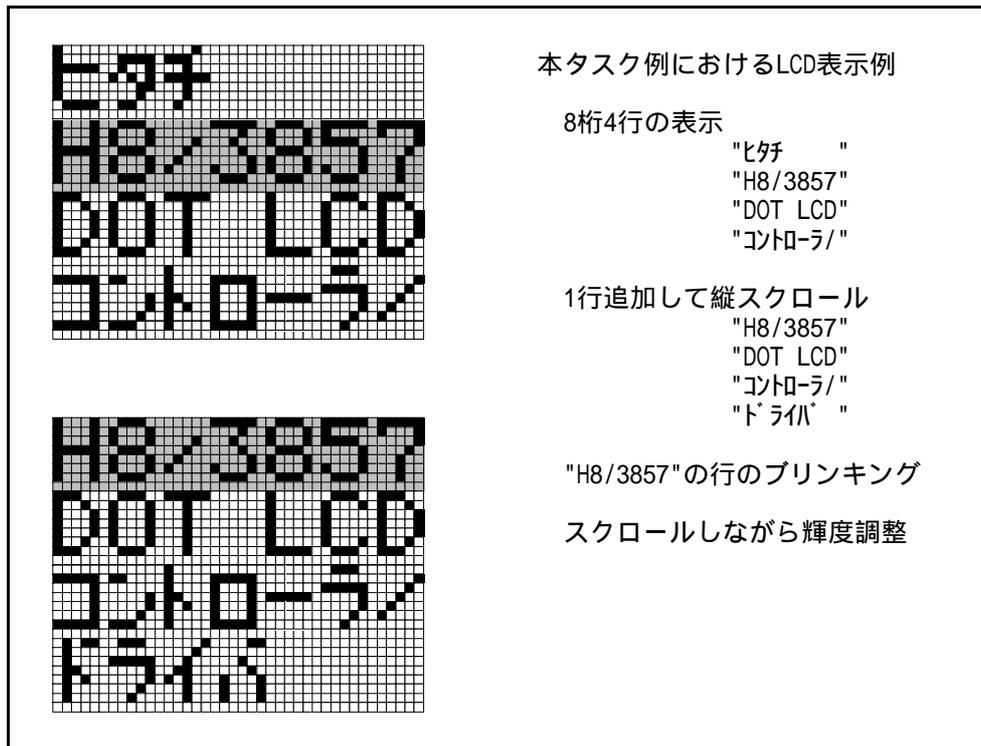


図1 本タスク例におけるLCD表示例

使用機能説明

- (1) 本タスク例では、ドットマトリックスLCDコントローラを使用して、LCD表示を行いません。以下にドットマトリックスLCDコントローラの特徴を示します。
 - ・ビットマップ方式の表示RAM (2048ビット) を内蔵し、最大表示ビット数は1280ビット (40×32ビット、56×16ビット、64×8ビット、40×16ビット、40×8ビットから選択可能)
 - ・1/8、1/16、または1/32デューティを選択可能
 - ・低消費電力機能 (サブクロック動作、モジュールスタンバイ) による電池での長時間駆動が可能
 - ・2倍または3倍の液晶電源昇圧回路内蔵
 - ・豊富な表示制御機能 (表示データリード/ライト、表示オン/オフ、表示縦方向スクロール、任意のエリアのプリンク、リードモディファイライト)
 - ・I/OポートインタフェースによるCPUインタフェース
 - ・輝度調整回路内蔵

使用機能説明

(2) 図2に本タスク例で使用するドットマトリクスLCDコントローラのブロック図を示します。

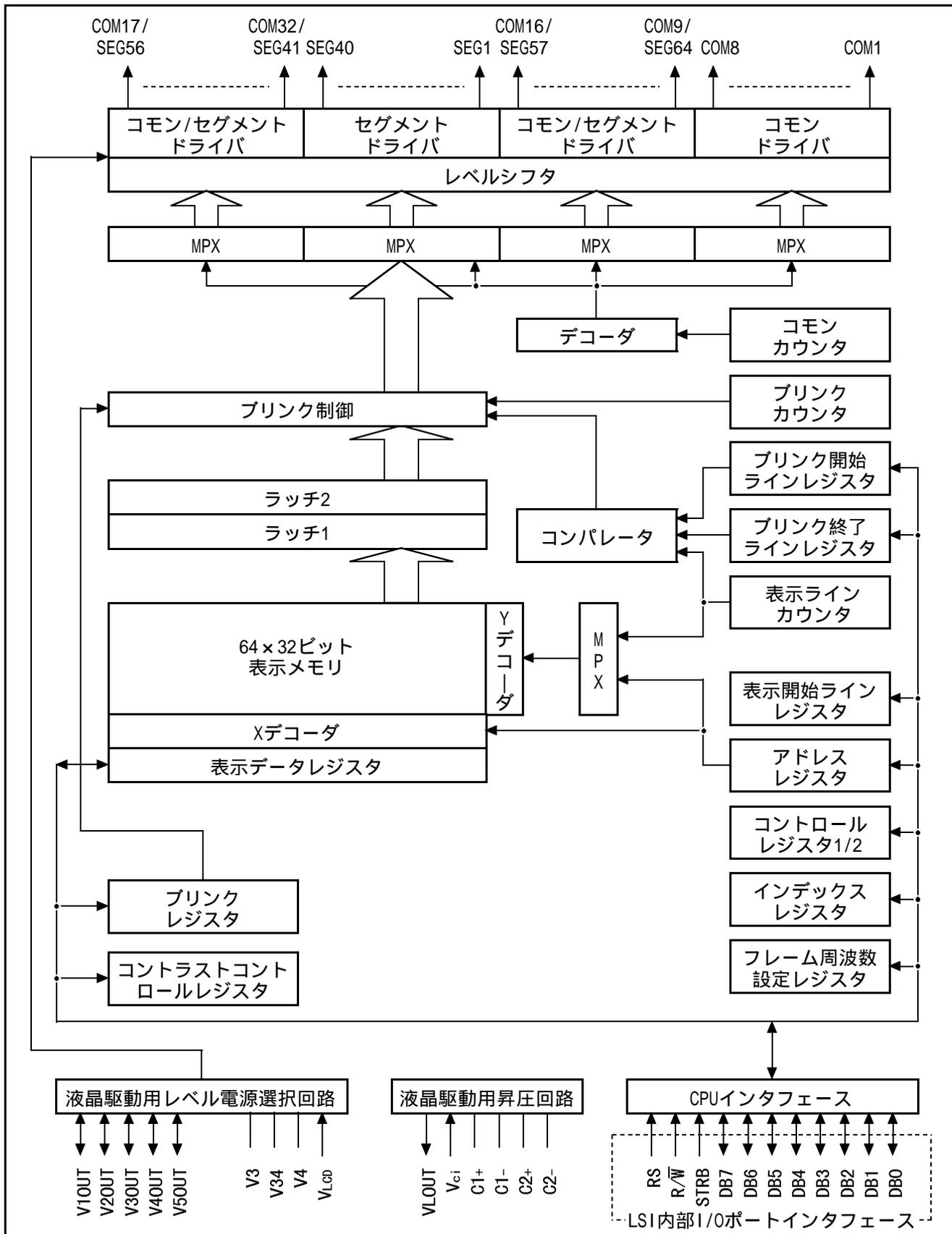


図2 ドットマトリクスLCDコントローラのブロック図

ドットマトリックスLCD制御 -キャラクタ表示モード	MCU	H8/3857シリーズ	使用機能	ドットマトリックス LCDコントローラ
-------------------------------	-----	-------------	------	------------------------

使用機能説明

(3) 表1にドットマトリックスLCDコントローラの各機能について説明します。

表1 ドットマトリックスLCDコントローラ機能

インデックスレジスタ (IR)	
機能	IRは、8ビットのライト専用のレジスタでLCDコントローラの10本の制御レジスタのうち一つを選択します。IRはRSが"0"のとき選択されます。リセット時、IRはH'00にイニシャライズされます。
コントロールレジスタ1 (R0)	
機能	R0は、8ビットのライト専用のレジスタでLCDモジュールスタンバイモードの設定、昇圧回路の制御、キャラクタ表示/グラフィック表示の切り換え、駆動デューティの選択を行ないます。リセット時、R0はH'00にイニシャライズされます。
コントロールレジスタ2 (R1)	
機能	R1は、8ビットのライト専用のレジスタで液晶表示の動作/停止の設定、オペアンプ回路のパワーセーブの設定、オペアンプ回路の動作/停止の設定、リード・モディファイ・ライトモードの設定、表示メモリのインクリメントするアドレスの設定を行います。リセット時、R1はH'00にイニシャライズされます。
アドレスレジスタ (R2)	
機能	R2は、8ビットのライト専用のレジスタでCPUからアクセスする表示メモリのX、Y方向それぞれのアドレスの設定を行ないます。リセット時、R2はH'00にイニシャライズされます。
フレーム周波数設定レジスタ (R3)	
機能	R3は、8ビットのライト専用のレジスタでフレーム周波数の設定を行ないます。リセット時、R3はH'00にイニシャライズされます。
表示データレジスタ (R4)	
機能	R4は、8ビットのライト専用のレジスタでR2のXA2~XA0とYA4~YA0で指定される表示メモリにリード/ライトを行ないます。表示メモリへのライトは、本レジスタを通して表示メモリに直接書込みます。リードは、いったんこのレジスタにデータが取り込まれてからバスに出力されます。リセット時、表示メモリおよびR4は不定です。
表示開始ラインレジスタ (R5)	
機能	R5は、8ビットのライト専用のレジスタで表示を開始するラインを設定します。リセット時、表示メモリおよびR4は不定です。
ブリンクレジスタ (R6)	
機能	R6は、8ビットのライト専用のレジスタで、ブリンクさせたい領域に対応するビットに"1"を書込みます。同時にブリンクさせる範囲には制限がなく、全ビットに"1"を書込むことによって全画面をブリンクさせることも可能です。このレジスタの設定値は、BLKが"1"に設定されているときにのみ有効です。リセット時、R6はH'00にイニシャライズされます。
ブリンク開始ラインレジスタ (R8)	
機能	R8は、8ビットのライト専用のレジスタでブリンクさせる領域の始まりのラインを設定します。リセット時、R8はH'00にイニシャライズされます。
ブリンク終了ラインレジスタ (R9)	
機能	R9は、8ビットのライト専用のレジスタでブリンクさせる領域の終わりのラインを設定します。リセット時、R9はH'00にイニシャライズされます。
コントラストコントロールレジスタ (RA)	
機能	RAは、8ビットのライト専用のレジスタで輝度調整抵抗値を設定します。リセット時、RAはH'00にイニシャライズされます。

使用機能説明

表1 ドットマトリックスLCDコントローラ機能(つづき)

コモン出力端子 (COM1 ~ COM32)	
機能	液晶のコモン駆動用の出力端子です。
セグメント出力端子 (SEG1 ~ SEG64)	
機能	液晶のセグメント駆動用の出力端子です。
LCDバイアス設定端子 (V3、V4)	
機能	LCDのバイアスを設定する入力端子です。
LCDテスト端子 (V34)	
機能	内部抵抗テスト用の入力端子で、V3とショートさせます。
LCD昇圧用容量接続端子 (C1+、C1-、C2+、C2-)	
機能	LCD昇圧用の外部容量を接続するための端子です。
LCD駆動電源レベル (V10UT ~ V50UT)	
機能	LCD駆動電源レベル用の入出力端子です。
LCD昇圧回路基準電源 (V _{ci})	
機能	LCDの昇圧回路への基準入力電圧、兼昇圧回路電源用の入力端子です。
LCD昇圧電源出力端子 (VLOUT)	
機能	LCDの昇圧電圧の出力端子です。
LCD駆動電源 (V _{LCD})	
機能	LCDの駆動電源の入力端子です。

(4) ドットマトリックスLCDコントローラシステム概要

LCDコントローラは、1/32、1/16、または1/8デューティで動作します。表示サイズは最大40×32ドット(5×8ドットフォント時8桁4行)です。サブクロックで動作し、表示制御を行ないますので、時計などを常時表示することができます。2倍または3倍のLCD電源昇圧回路を内蔵していますので、いくつかの外付けの部品(抵抗、容量)のみで、LCD表示システムを構成することができます。また、モジュールスタンバイモード時でもRAMデータは保持され、昇圧動作は行ないませんので、表示に影響をおよぼすことなく低消費電力化を行なうことができます。

図3にLCDコントローラのシステムブロックダイアグラムを示します。

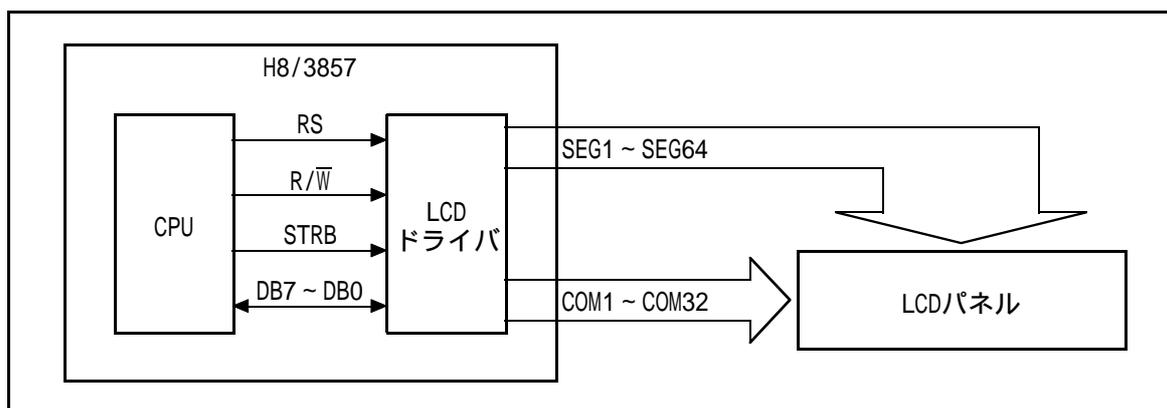


図3 LCDコントローラのシステムブロックダイアグラム

使用機能説明

(5) CPUとのインタフェース

LCDコントローラのレジスタはH8/3857のメモリマップ上にはありません。CPUからLSI内部I/Oポートのポート9およびポートAを介して、LSI内部LCD端子であるDB7～DB0、RS、R/W、STRBにより制御されます。各端子の構成を表2に示します。LCDコントローラの内部にあるレジスタへのアクセスタイミング例を図4に示します。

表2 LSI内部LCD端子構成

端子名	記号	入出力	機能
データバス端子	DB7～DB0	入出力	R/W="0"のときレジスタにライトするデータを入力、R/W="1"のとき、レジスタからリードしたデータを出力します。
レジスタセレクト端子	RS	入力	RS="0"のときインデックスレジスタが、RS="1"のとき制御レジスタが選択されます。
リード/ライト選択端子	R/W	入力	R/W="0"のときライトが、R/W="1"のときリードが選択されます。
ストロブ端子	STRB	入力	STRBの立ち下がりで、RSで選択されたレジスタにR/Wで選択されたリード/ライトのアクセスを行ないます。

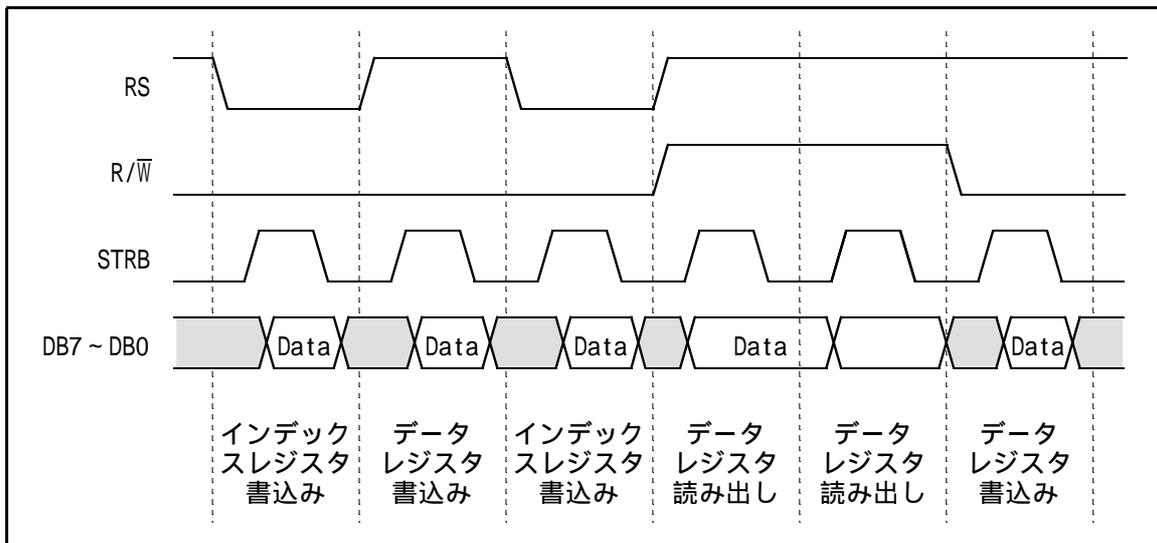


図4 8ビットデータ転送のタイミングシーケンス例

(a) インデックスレジスタへのライト

RSに"0"を、R/Wに"0"をセットした状態のとき、STRBの立ち下がりエッジによりDB7～DB0のデータがインデックスレジスタ(IR)にライトされます。STRBの立ち下がりではRS、R/Wを変化させないで下さい。

使用機能説明

(b) 制御レジスタへのリード/ライト

制御レジスタへアクセスする場合は、インデックスレジスタ（IR）にアクセスしたいレジスタ番号のデータをライトしてからアクセスする必要があります。インデックスレジスタ（IR）へライトするレジスタ番号のデータを表3に示します。インデックスレジスタ（IR）にライトされたレジスタ番号は再度インデックスレジスタ（IR）へのライトが行われない限り保存されますので、同じ制御レジスタにアクセスする場合は、一回ごとにインデックスレジスタにライトする必要はありません。

表3 レジスタ構成

名称	略称	R/W	RS	インデックスレジスタ			
				IR3	IR2	IR1	IR0
インデックスレジスタ	IR	W	0	-	-	-	-
コントロールレジスタ1	R0	W	1	0	0	0	0
コントロールレジスタ2	R1	W		0	0	0	1
アドレスレジスタ	R2	W		0	0	1	0
フレーム周波数設定レジスタ	R3	W		0	0	1	1
表示データレジスタ	R4	R/W		0	1	0	0
表示開始ラインレジスタ	R5	W		0	1	0	1
ブリンクレジスタ	R6	W		0	1	1	0
ブリンク開始ラインレジスタ	R8	W		1	0	0	0
ブリンク終了ラインレジスタ	R9	W		1	0	0	1
コントラストコントロールレジスタ	RA	W		1	0	1	0

制御レジスタへのライトはRSに"1"を、R/Wに"0"をセットした状態のとき、STRBの立ち下がりエッジによりDB7～DB0のデータがインデックスレジスタ（IR）で指定された制御レジスタにライトされます。

制御レジスタは表示データレジスタ（R4）以外はリードできません。表示データレジスタ（R4）のリードはインデックスレジスタ（IR）に表示データレジスタ（R4）のレジスタ番号がライトされ、RSに"1"を、R/Wに"1"をセットすると、DB7～DB0は出力に設定され、STRBの立ち上がりエッジによりアドレスレジスタ（R2）で指定されるアドレスの表示メモリのデータをDB7～DB0から出力されます。次のサイクルもリードを行なう場合は次のSTRBの立ち上がりまでデータ出力は保持されますが、次のサイクルでライトを行なう場合は、R/Wが"0"にセットされた時点からDB7～DB0は入力に設定され、出力は解除されます。

いずれの場合もSTRBの立ち下がりではRS、R/Wを変化させないで下さい。

使用機能説明

(6) 液晶駆動端子機能 (キャラクタ表示モード)

LCDコントローラは、液晶駆動出力のうち、COM9～COM32とSEG64～SEG41が表示デューティと表示モードにより切り替わります。

キャラクタ表示モード (SOB="0") の場合のコモン/セグメント出力切り替えについて表4に示します。

表4 キャラクタ表示モードにおける
コモン/セグメント出力切り替え

端子名	SOB="0" (キャラクタ表示モード)		
	1/8 Duty	1/16 Duty	1/32 Duty
COM1～COM8	COM1～COM8	COM1～COM16	COM1～COM16
COM9/SEG64～ COM16/SEG57	コモン信号の 非選択波形		
SEG1～SEG40	SEG1～SEG40	SEG1～SEG40	SEG1～SEG40
COM32/SEG41～ COM25/SEG48	コモン信号の 非選択波形	COM16～COM1	COM32～COM17
COM24/SEG49～ COM17/SEG56	COM8～COM1		

使用機能説明

(7) 表示メモリ構成と表示 (キャラクタ表示モード)

LCDコントローラは、容量64×32ビットのビットマップ方式の表示メモリを内蔵しています。表示メモリの構成はX方向が5ビット×8または8ビット×n (n=5,7,8) ビットの組み合わせが選択可能で、Y方向が32ビットになっています。CPUからライトする表示データは、図5に示すようにMSBが左、LSBが右に横に格納されます。表示データと表示の関係は、データ"1"が点灯(黒)、データ"0"が非点灯(無色)に対応しています。

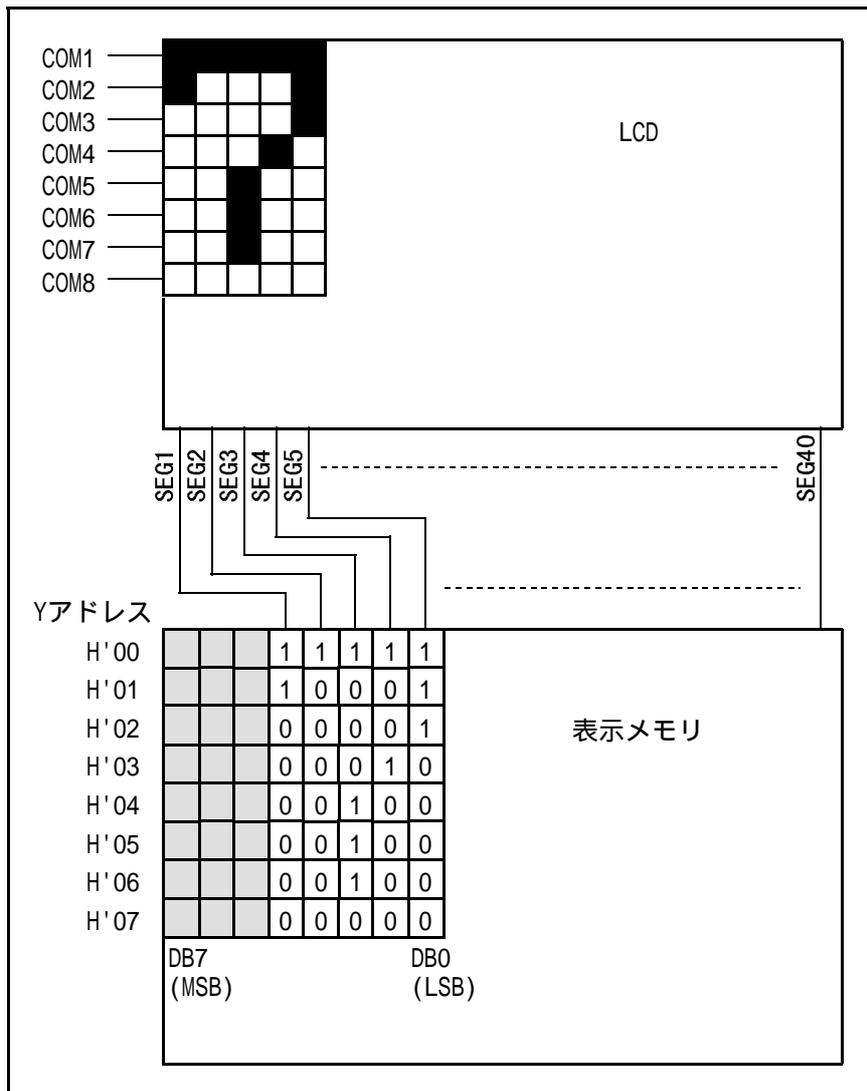


図5 キャラクタ表示モードにおけるメモリデータと表示

使用機能説明

(8) 表示データ出力 (キャラクタ表示モード)

LCDコントローラは、5ドット×8ドットのキャラクタ出力を効率的に行なうために表示データの1バイトのうち5ビットのみ出力可能なキャラクタ表示モード (SOB="0") と、フルドットのグラフィック表示を効率的に行なうために、1バイトの全データを出力可能なグラフィック表示モード (SOB="1") があります。

キャラクタ表示モード (SOB="0") における表示デューティと出力端子の関係を図6に示します。

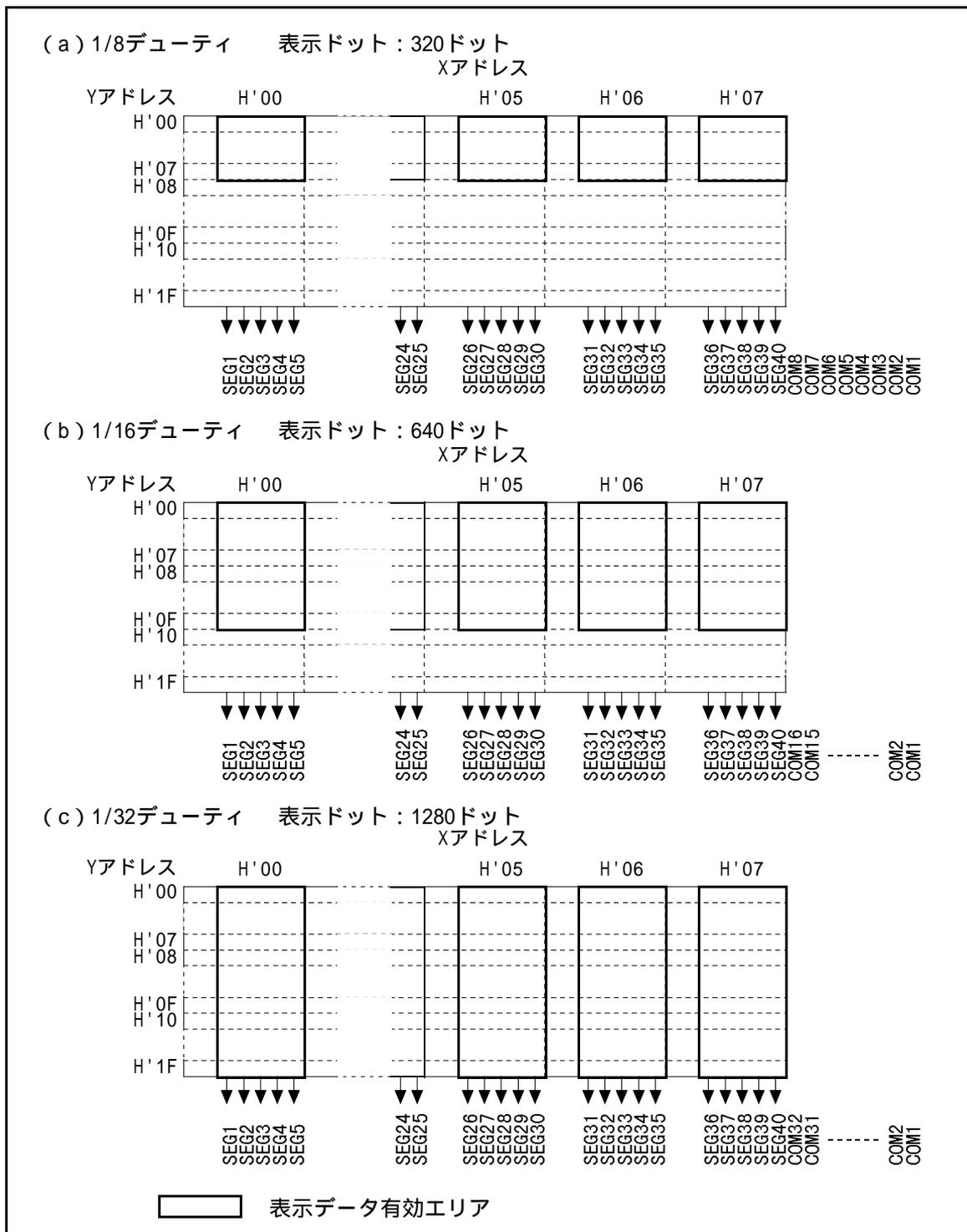


図6 キャラクタ表示モードにおける表示デューティと表示有効エリア

使用機能説明

(9) レジスタおよび表示メモリへのアクセス

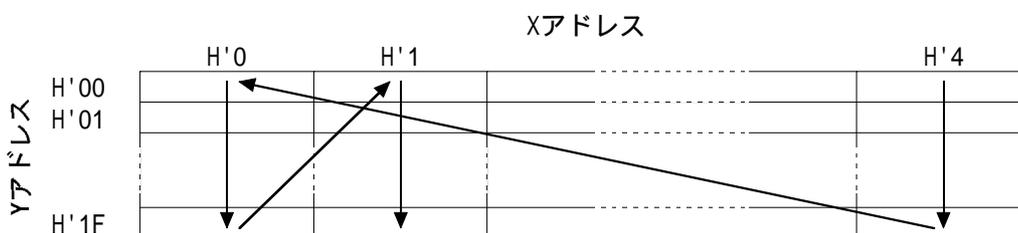
(a) レジスタへのアクセス

レジスタへアクセスするには、まずRSを"0"にしてインデックスレジスタにアクセスしたいレジスタのレジスタ番号を設定します。その後RSを"1"にして指定したレジスタにアクセスすることができます。一部の内部レジスタには存在しないビットがありますが、これらのビットには"0"を書込むようにしてください。表示データレジスタ (R4) 以外のリードはできません。

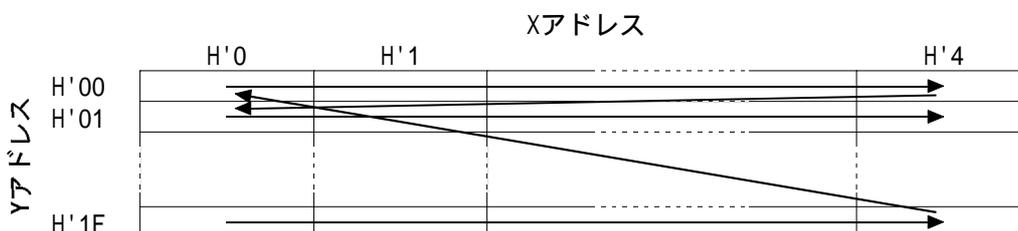
(b) 表示メモリへのアクセス

表示メモリへのアクセスを行なうには、アドレスレジスタ (R2) にアクセスしたいアドレスを設定します。その後表示データレジスタ (R4) を通してメモリをアクセスします。このアクセスは表示側の読み出しを意識せずに行なうことができます。1/32デューティの場合のこれらの手順を図7に示します。

() Y方向のデータアクセスを優先する場合 (INC="0")



() X方向のデータアクセスを優先する場合 (INC="1")



- 【注】
1. アドレスレジスタ (R2) のXA2 ~ XA0がXアドレス、YA4 ~ YA0がYアドレスを示します。
 2. Xアドレスの動作
 - (1) SOB="0"のとき、表示デューティにかかわらず、H'7の次にH'0になります。
 - (2) SOB="1"のとき、1/16デューティの場合H'6の次にH'0、1/8デューティの場合、H'7の次にH'0になります。
 3. Yアドレスの動作
 - 1/16デューティの場合H'0Fの次にH'00、1/8デューティの場合H'07の次にH'00になります。

図7 表示メモリへのアクセス方法 (1/32デューティの場合)

X、Yアドレスは各々表示データレジスタ (R4) へのアクセス後、コントロールレジスタ2 (R1) のINCの設定値に基づき自動的にインクリメントされるため、アドレスの設定を毎回する必要はありません。

使用機能説明

1/32デューティ (DDTY1="0"、DDTY0="0") において、コントロールレジスタ1 (R0) のSOBが"1"でINCが"0"の場合、表示データレジスタ (R4) のリード/ライトのアクセスごとにXアドレスは同じままYアドレスが自動的にH'1Fまでインクリメントされます。H'1Fの次はYアドレスはH'00に戻りますが、同時にXアドレスもインクリメントされます。また、INCが"1"の場合、表示データレジスタ (R4) のリード/ライトのアクセスごとにYアドレスは同じままXアドレスが自動的にH'4までインクリメントされます。H'4の次はXアドレスはH'0に戻りますが、同時にYアドレスもインクリメントされますので、表示メモリ全域への連続リード/ライトが可能です。

(c) 表示のための読み出し

LCDコントローラの表示RAMは、CPUからのアクセスと液晶表示のための読み出しが独立したデュアルポート方式であるため、フレキシブルなインタフェースが可能です。

図8にメモリ読み出しの手順を示します。

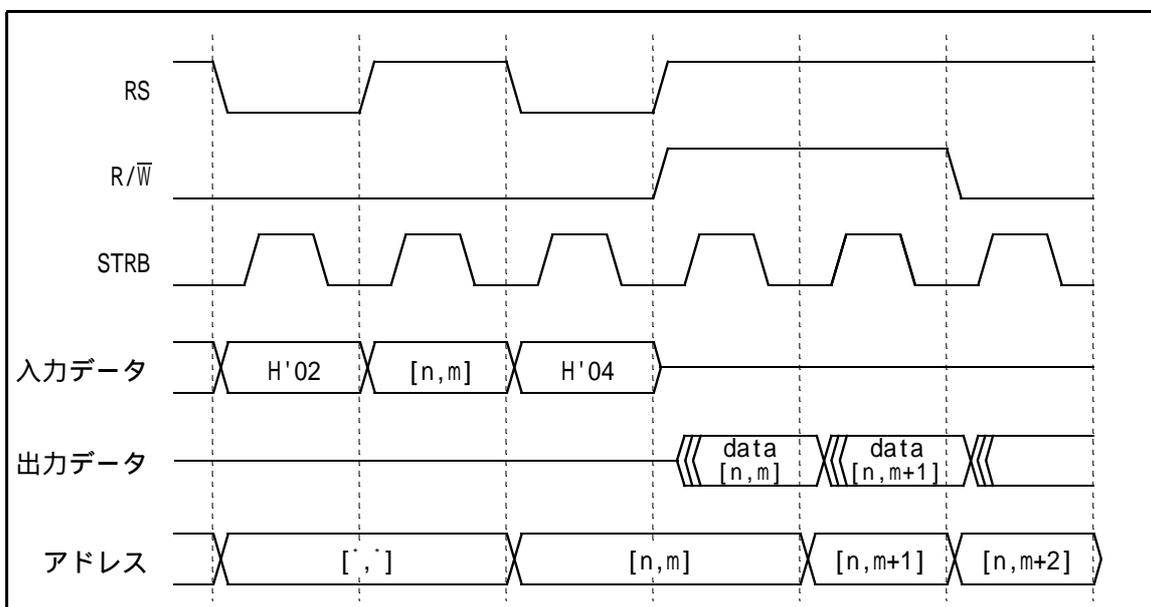


図8 メモリ読み出し手順

(d) リード・モディファイ・ライトモード

通常状態では、表示メモリへのリード/ライト双方のアクセス後にXまたはYアドレスがインクリメントされます。これがリード・モディファイ・ライトモードでは、ライト後のみにインクリメントされ、リード後のアドレスは同じままです。このモードを使用すると、すでに書き込まれているデータの読み出し、そのデータを加工して再び同じアドレスに書き込むといったことができます。

図9にリード・モディファイ・ライトモードのフローを示します。

使用機能説明

1/32デューティ (DDTY1="0"、DDTY0="0") において、コントロールレジスタ1 (R0) のSOBが"1"でINCが"0"の場合、表示データレジスタ (R4) のリード/ライトのアクセスごとにXアドレスは同じままYアドレスが自動的にH'1Fまでインクリメントされます。H'1Fの次はYアドレスはH'00に戻りますが、同時にXアドレスもインクリメントされます。また、INCが"1"の場合、表示データレジスタ (R4) のリード/ライトのアクセスごとにYアドレスは同じままXアドレスが自動的にH'4までインクリメントされます。H'4の次はXアドレスはH'0に戻りますが、同時にYアドレスもインクリメントされますので、表示メモリ全域への連続リード/ライトが可能です。

(c) 表示のための読み出し

LCDコントローラの表示RAMは、CPUからのアクセスと液晶表示のための読み出しが独立したデュアルポート方式であるため、フレキシブルなインタフェースが可能です。

図9にメモリ読み出しの手順を示します。

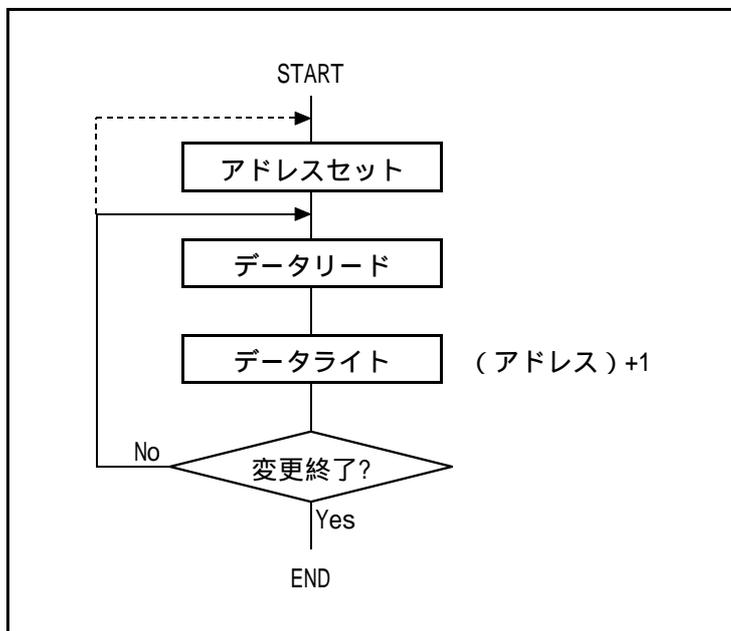


図9 リード・モディファイ・ライトモードのフロー

使用機能説明

(10) スクロール機能

LCDコントローラは、表示の開始ラインを指定することによって任意のライン数の縦方向のスクロールが実現できます。図10に1/16デューティおよび1/8デューティの場合の表示メモリとYアドレス、スクロール後の表示メモリと液晶表示の関係を示します。表示開始アドレスを1に設定した場合、16ライン目にはYアドレス0のデータが表示されます。そのため、次の画面と表示させるためにスクロールさせたときには、Yアドレス0のデータを次の表示データに書き換える必要があります。その際はスクロールさせた後にデータの書き換えを行なってください。

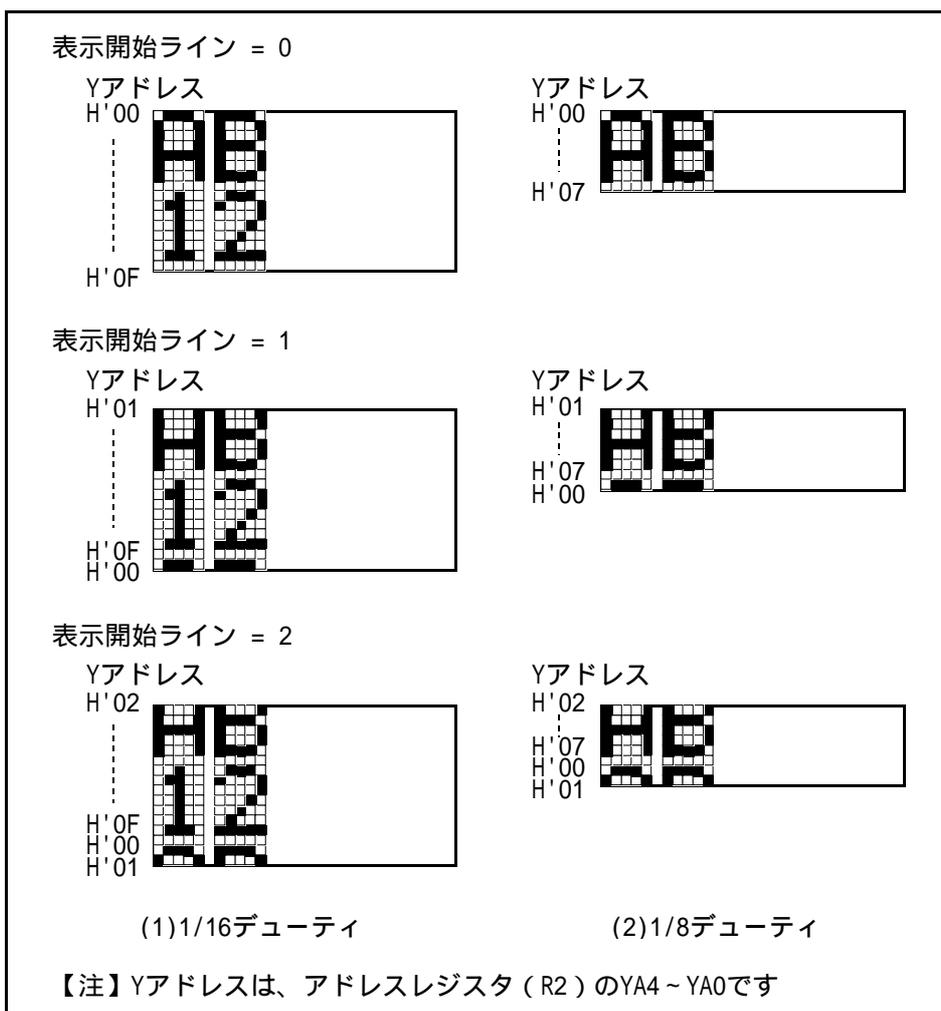


図10 縦方向スクロール

使用機能説明

(11) ブリンク機能

LCDコントローラは、任意の領域に対してブリンク表示をすることができます。80Hzのフレーム周波数では約1.6秒の周期で表示と非表示を繰り返します。

ブリンク領域の設定は、水平方向がブリンク開始ラインレジスタ (R8) とブリンク終了ラインレジスタ (R9) によりライン単位の指示を行ない、垂直方向にはブリンクレジスタ (R6) により5ビット単位 (SOB="0") または8ビット単位 (SOB="1") の設定を行ないます。ブリンクレジスタ (R6) に"1"を設定すると、対応するドットのブリンクをコントロールします。これらのレジスタを設定した後に、コントロールレジスタ2 (R1) のBLKを"1"に設定することによりブリンクが開始します。ブリンク領域は表示メモリに対する絶対指定のため、表示のスクロールを行ってもブリンク領域はそれにともない移動しません。

図11にブリンクレジスタ (R6) とブリンク位置を、図12に表示スクロール時のブリンクについて示します。

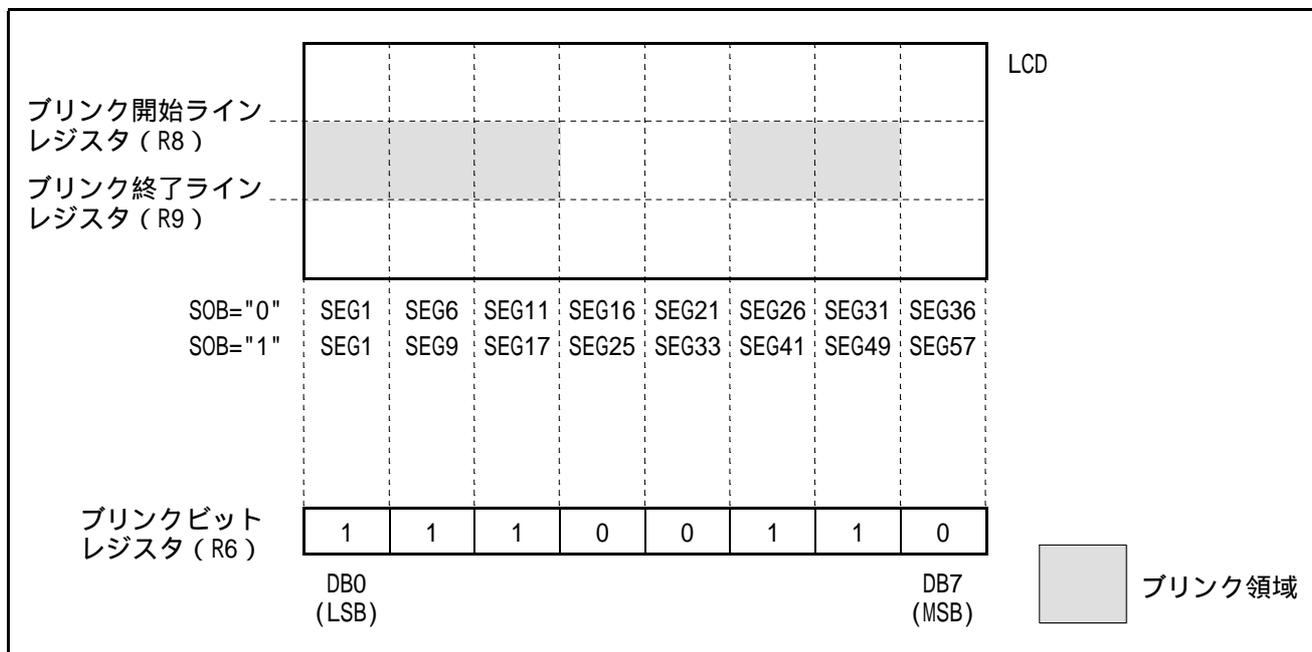


図11 ブリンクレジスタ (R6) とブリンク位置

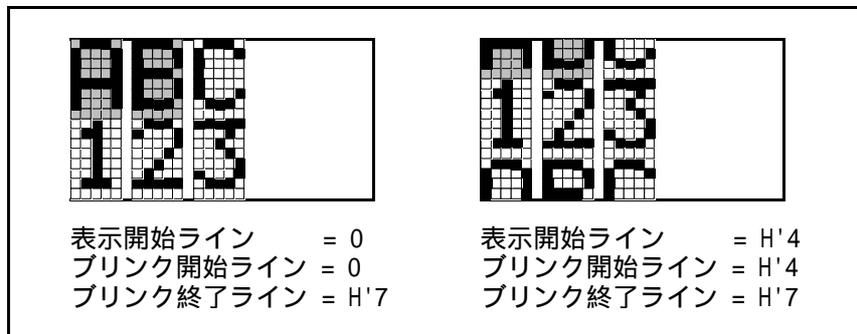


図12 表示スクロール時のブリンク (SOB="0"、1/16デューティの場合)

使用機能説明

(12) モジュールスタンバイモード

LCDコントローラは、モジュールスタンバイ機能を持っており、低消費電力化を実現することができます。モジュールスタンバイモードでは内部昇圧回路およびオペアンプは停止し、セグメントおよびコモン出力は V_{SS} （表示オフ状態）レベルとなります。また、表示RAMおよび内部レジスタのデータはコントロールレジスタ2（R1）のDISPおよびOPONを除いて保持されます。ただし、モジュールスタンバイ状態でもコントロールレジスタはアクセス可能です。図13にモジュールスタンバイモードの起動、解除手順を示します。表示メモリの内容を保護するために、起動解除手順は必ず守ってください。

CPUをスタンバイモードにする場合は、スタンバイ命令実行前にコントロールレジスタ1（R0）のLSBYを"1"にセットしてください。またスタンバイモード解除後は、モジュールスタンバイ解除の手順に従って表示を開始してください。

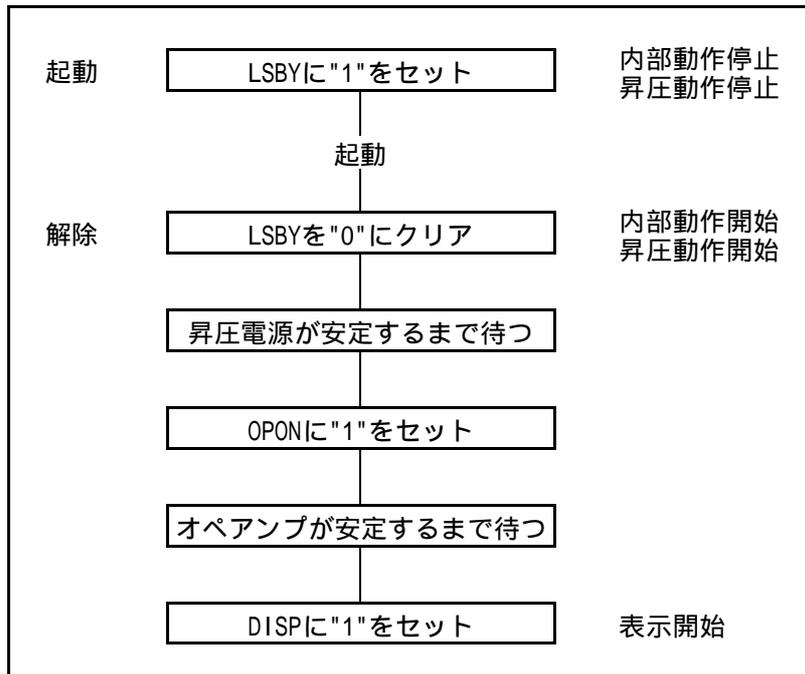


図13 モジュールスタンバイモード起動、解除手順

(13) 電源投入、切断手順

LCDコントローラは、電源回路全体を内蔵していますので、電源投入、切断の手順は図14のように行ってください。この手順を守りませんと異常な表示をすることがあります。

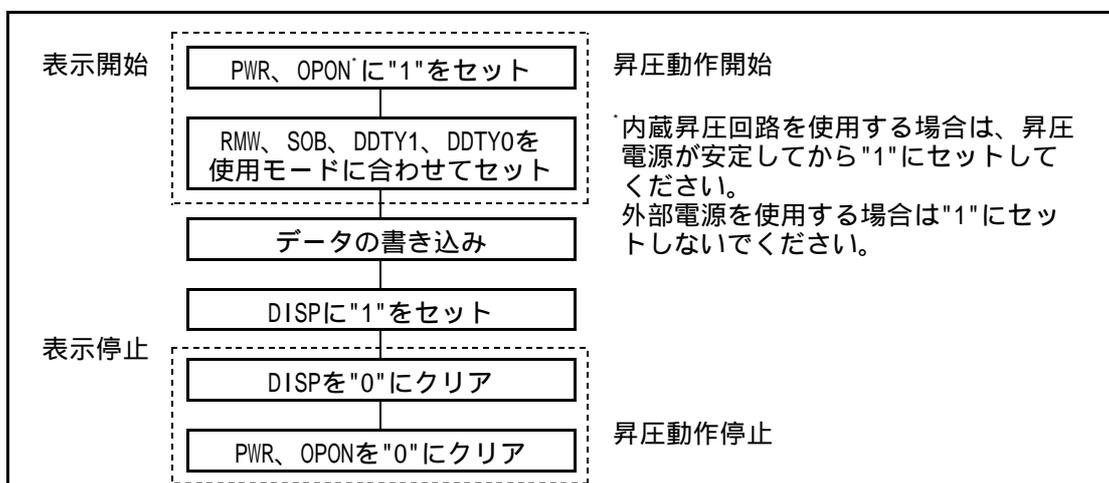


図14 電源投入、切断手順

使用機能説明

(13) 電源回路

LCDコントローラは、液晶駆動するための2倍または3倍の昇圧回路を内蔵しています。スタンバイモード時には最大2サブクロック後に自動的に電源回路がオフされますので、昇圧回路の消費電力は0となります。また、コマンドで電源回路をオン/オフすることが可能ですので、内蔵昇圧回路の電流容量が不足するような場合は外部電源回路を使用してください。

(a) 昇圧回路

V_{ci} に基準電圧 ($V_{ci} = V_{cc}$) を入力し、 $V_{ss} \sim V_{LOUT}$ 、 $C1+ \sim C1-$ 、および $C2+ \sim C2-$ 間にコンデンサを接続してコントロールレジスタ1 (R0) のPWRに"1"をセットすることにより、 V_{ci} と V_{ss} 間の電位が2倍または3倍に昇圧されます。昇圧にはサブクロックを使用していますので、サブクロックが供給されていないと昇圧されません。 V_{ci} は昇圧回路の電源を兼ねるため、十分電流を確保できるようにして下さい。

昇圧回路を使用しない場合、 V_{ci} は V_{cc} に接続してください。

(b) 液晶駆動レベル電源

液晶駆動には、V1、V2、V3、V4、V5、 V_{ss} の6レベルの電源が必要です。これらのレベルは通常V1～ V_{ss} の電源を抵抗分割で発生させます。本電源回路は、抵抗分割で発生した各電圧レベルごとにボルテージフォロアオペアンプを内蔵しています。オペアンプは、OPSビットにより駆動電流を制限することによりパワーセーブが可能です。この場合、V1OUTからV5OUT端子には、 V_{ss} との間に0.1～0.5 μ F程度のコンデンサを挿入してください。

液晶表示に1/4バイアスを使用する場合はV3とV4端子を短絡、1/5バイアスを使用する場合は、V3とV4端子を解放してください。V34端子は内部抵抗テスト用端子で、常に外部でV3端子と短絡してください。

(c) 輝度調節

LCDコントローラは、次の2通りの方法で輝度調整が可能です。

() 内蔵輝度調整回路による方法

LCDコントローラは、プログラマブル輝度調整回路を内蔵しています。コントラストコントロールレジスタ (RA) の選択により、一定の昇圧電圧のもとで液晶電源電圧の調整が可能です。

() 昇圧基準電圧 V_{ci} を変更することにより、昇圧電圧レベルの変更が可能です。

(d) 外部電源

() V_{LCD} に外部電源を入力する場合

V_{LCD} に外部電源を入力し、コントロールレジスタ2 (R1) のOPONを"1"にして内蔵オペアンプを使用することにより、V1～V5を生成することができます。 V_{LCD} の入力レベルは必ず V_{cc} 以上7.0V以下としてください。

() V1～V5端子に直接外部電源を入力する場合

コントロールレジスタ1 (R0) PWRとコントロールレジスタ2 (R1) のOPONを"0"として、内蔵昇圧回路を停止し内蔵オペアンプの電源を切断することにより、外部からV1、V2、V3、V4、V5に直接電源を印可することができます。 V_{LCD} にはV1と同電位を入力してください。 V_{LCD} およびV1の入力レベルは V_{cc} 以上7.0V以下としてください。

いずれの場合も最大定格電圧を超える電圧を入力すると、LSIの信頼性に悪影響をおよぼすことがあります。

使用機能説明

(14) 液晶駆動電源電圧

液晶駆動電源電圧は、V1～V5、Vssの6値あります。V1が最も高い電圧で、Vssが最も低い電圧です。図15のように、コモンは計はV1、V2、V5、Vssの組み合わせからなり、セグメント波形はV1、V3、V4、Vssの組み合わせからなっています。V1とVssは共通ですが、中間電圧がセグメントとコモンで異なっています。

図15において、SEG1～SEG40の出力は表示データによって異なる波形になります。この例では、液晶パネルのCOM1が接続されているラインが点灯し、他のドットはすべて非点灯になります。

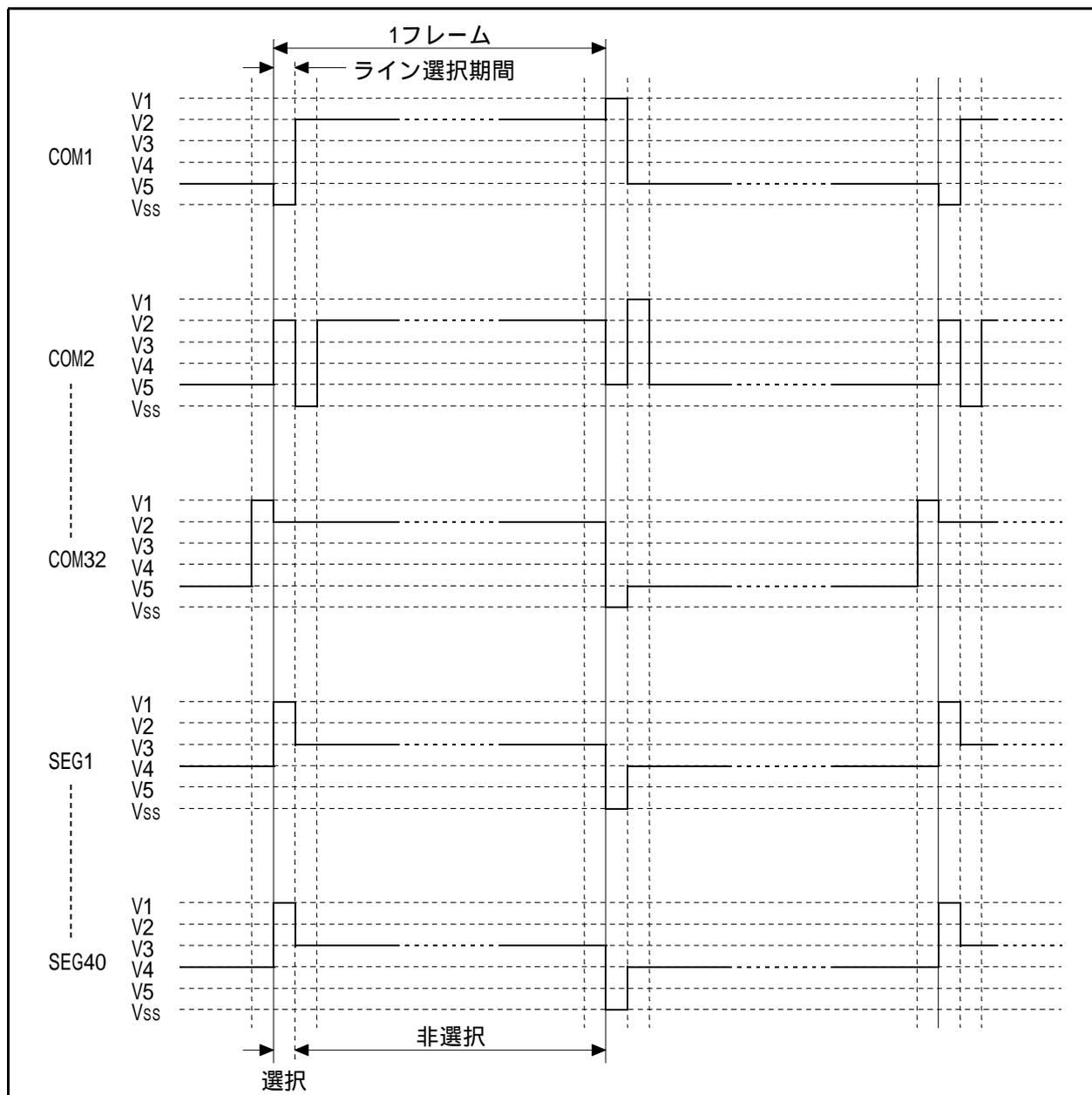
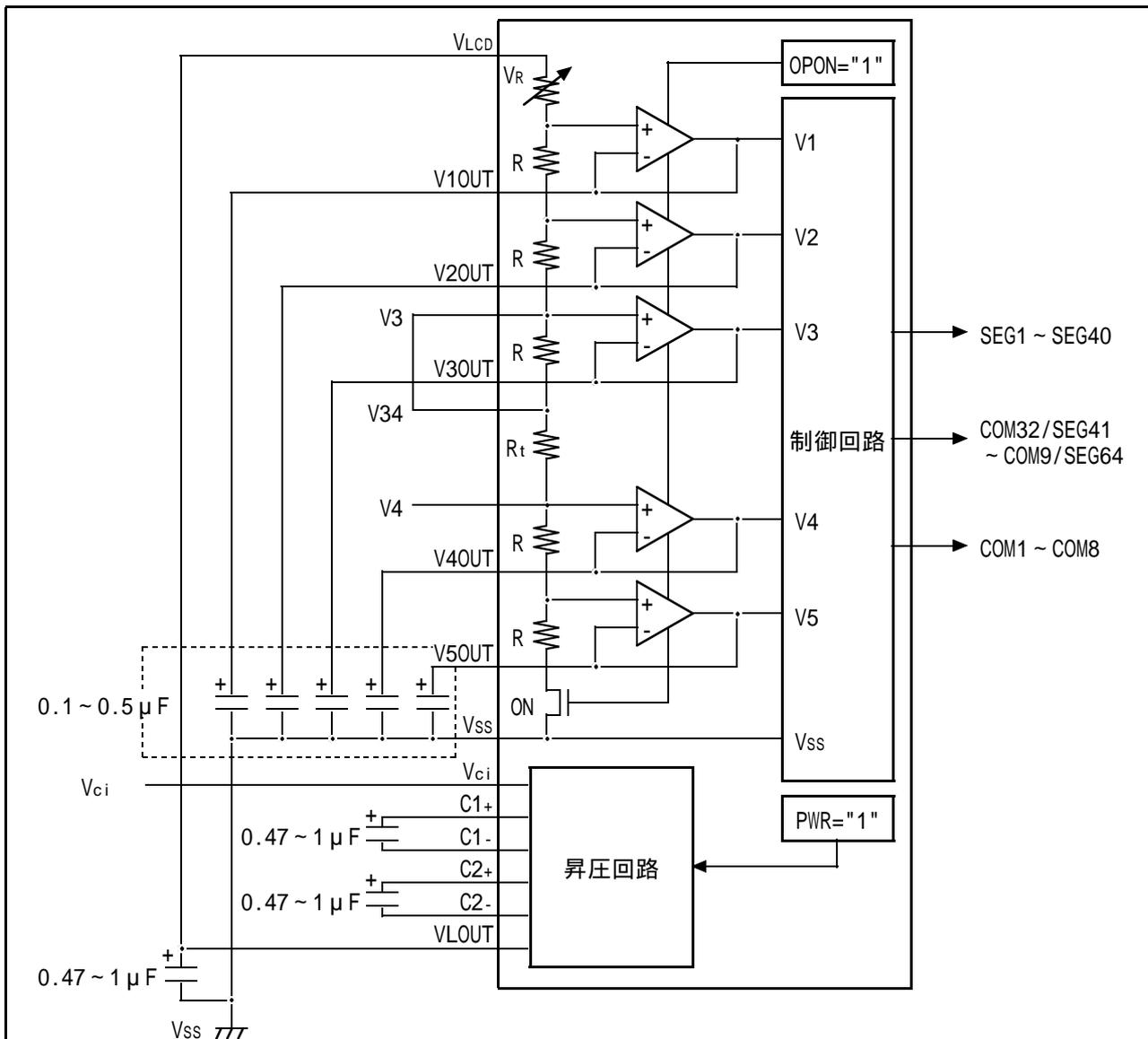


図15 液晶駆動電圧波形 (1/32デューティ)

使用機能説明

(15) 液晶電圧発生回路

内蔵の昇圧回路（3倍昇圧、1/5バイアス）を使用する場合の接続例を図16に示します。LCDコントローラは、V1レベルからV5レベルを発生するブリーダ抵抗、およびボルテージフォロア型オペアンプを内蔵しています。内蔵オペアンプを使用する場合は、コントロールレジスタ2 (R1) のOPONを"1"にしてください。輝度の調整は、コントラストコントロールレジスタ (RA) を用いてソフトウェアで制御できます。V1OUT ~ V50UTの内蔵オペアンプの出力とVss間には、0.1 ~ 0.5 μ F程度のコンデンサを挿入して安定化してください。OPSビットによりパワーセーブ機能を設定した場合は、必ずコンデンサを挿入してください。またオペアンプを正常に動作するために、V_{LCD} ~ V1間とV5 ~ V_{ss}間の電位差が0.4V以上になるようにコントラストコントロールレジスタ (RA) を設定してください。V_{ci}端子は昇圧回路の電源も兼用していますので、基準電圧調整時には十分電流を供給できるようにしてください。V_{ci}の入力レベルは必ずV_{cc}以下としてください。



- 【注】
1. R_tはテスト用抵抗です。V3端子とV34端子は短絡して使用してください。
 2. 昇圧後の出力電圧は、7.0Vを超えないようにしてください。特に3倍昇圧時には基準電圧 (V_{ci}) に2.3V以上の電圧を入力しないで下さい。
 3. V_{ci}は昇圧回路の電源も兼用しています。液晶駆動電流を十分に確保できるように、ドランジスタ等で電流増幅してください。
 4. V_{ci}の入力レベルは必ずV_{cc}以下としてください。
 5. 極性のあるコンデンサを使用する場合、接続に注意して下さい。

図16 昇圧回路（3倍昇圧、1/5バイアス）の使用例

使用機能説明

(16) 輝度調整回路

液晶駆動電圧（ $V_{LCD} \sim V1$ 間の電位差）をコントラストコントロールレジスタ（RA）で制御して、ソフトウェアによる輝度調整が行なえます（電子ボリューム機能）。 $V_{LCD} \sim V1$ 間の分割基本ブリーダ抵抗値Rに対し、可変抵抗値 V_R は0.1R～1.6Rまでの範囲で調整できます。コントラストコントロールレジスタ（RA）のCCR3～CCR0による輝度調整の設定値を表5に示します。

$V1$ レベルから $V5$ レベルを出力するボルテージフォロア型オペアンプ回路を安定して動作させるために、 $V_{LCD} \sim V1$ 間と $V5 \sim V_{SS}$ 間の電位差が0.4V以上になるように、コントラストコントロールレジスタ（RA）を設定してください。コントラスト調整範囲を表6に示します。

輝度調整が内蔵抵抗 V_R で行なうことができない場合、 $V_{LOUT} \sim V_{LCD}$ 間に抵抗を入れて調整することが可能です。

表5 輝度調整の設定値

コントラストコントロールレジスタ（RA）				可変抵抗値 （ V_R ）	$V1 \sim V_{SS}$ 間 電位差	表示色
CCR3	CCR2	CCR1	CCR0			
0	0	0	0	1.6R	↑ ↓	薄 濃
0	0	0	1	1.5R		
0	0	1	0	1.4R		
0	0	1	1	1.3R		
0	1	0	0	1.2R		
0	1	0	1	1.1R		
0	1	1	0	1.0R		
0	1	1	1	0.9R		
1	0	0	0	0.8R		
1	0	0	1	0.7R		
1	0	1	0	0.6R		
1	0	1	1	0.5R		
1	1	0	0	0.4R		
1	1	0	1	0.3R		
1	1	1	0	0.2R		
1	1	1	1	0.1R		

表6 コントラスト調整範囲

バイアス	液晶電源電圧（ V_{DR} ）	コントラスト調整範囲
1/5 バイアス 駆動	$\frac{5 \times R}{5 \times R + V_R} \times (V_{LCD} - V_{SS})$	<ul style="list-style-type: none"> ・液晶駆動電圧調整：$0.758 \times (V_{LCD} - V_{SS}) \quad V_{DR} \quad 0.980 \times (V_{LCD} - V_{SS})$ ・$V5 - V_{SS}$間電位差制約：$\frac{R}{5 \times R + V_R} \times (V_{LCD} - V_{SS}) \quad 0.4[V]$ ・$V_{LCD} - V1$間電位差制約：$\frac{V_R}{5 \times R + V_R} \times (V_{LCD} - V_{SS}) \quad 0.4[V]$
1/4 バイアス 駆動	$\frac{4 \times R}{4 \times R + V_R} \times (V_{LCD} - V_{SS})$	<ul style="list-style-type: none"> ・液晶駆動電圧調整：$0.714 \times (V_{LCD} - V_{SS}) \quad V_{DR} \quad 0.976 \times (V_{LCD} - V_{SS})$ ・$V5 - V_{SS}$間電位差制約：$\frac{R}{4 \times R + V_R} \times (V_{LCD} - V_{SS}) \quad 0.4[V]$ ・$V_{LCD} - V1$間電位差制約：$\frac{V_R}{4 \times R + V_R} \times (V_{LCD} - V_{SS}) \quad 0.4[V]$

使用機能説明

(17) 液晶駆動バイアス選択回路

コントラストが最良となる理想的なバイアス値は以下の式で算出されます。最適なバイアス値より低いバイアス値で駆動するとコントラストは低下しますが、液晶駆動電源 (V1 ~ Vss間の電位差) を低く抑えることができます。Vci電圧が低く3倍昇圧回路使用しても液晶駆動電圧が不足する場合や、バッテリー寿命により出力電圧が低下し液晶表示薄くなるなど、液晶駆動バイアスを低くすると液晶表示が見やすくなります。

$$1/N \text{ デューティ駆動時の最適バイアス値} = \frac{1}{(N)^{(1/2)+1}}$$

- 【注】
1. 1/5バイアスを使用する場合は、V3端子とV4端子を解放してください。
 2. 1/4バイアスで使用する場合は、V3端子とV4端子を短絡してください。
 3. V3端子とV34端子は必ず短絡してください。

(18) フレーム周波数

LCDコントローラはフレーム周波数設定レジスタ (R3) のFS5 ~ FS0を設定することによりサブクロックの分周比を制御し、液晶表示のFフレーム周波数を設定します。液晶表示フレーム周波数f_F(Hz)、サブクロック周波数f_w(Hz)、分周比r、液晶表示デューティ1/Nには、

$$f_F = \frac{f_w}{r \times N}$$

の関係がありますので、使用する液晶パネルの特性に合わせて最適な分周比を設定してください。レジスタ設定値と分周比の関係を表7に、サブクロック周波数、液晶表示デューティ、分周比の設定とフレーム周波数の例を表8に示します。

表7 レジスタ設定値と分周比

FS					分周比	FS					分周比	FS					分周比	FS					分周比				
5	4	3	2	1	0	r	5	4	3	2	1	0	r	5	4	3	2	1	0	r	5	4	3	2	1	0	r
0	0	0	0	0	0	2	0	1	0	0	0	0	34	1	0	0	0	0	0	66	1	1	0	0	0	0	98
0	0	0	0	0	1	4	0	1	0	0	0	1	36	1	0	0	0	0	1	68	1	1	0	0	0	1	100
0	0	0	0	1	0	6	0	1	0	0	1	1	38	1	0	0	0	1	0	70	1	1	0	0	1	0	102
0	0	0	0	1	1	8	0	1	0	0	1	1	40	1	0	0	0	1	1	72	1	1	0	0	1	1	104
0	0	0	1	0	0	10	0	1	0	1	0	0	42	1	0	0	1	0	0	74	1	1	0	1	0	0	106
0	0	0	1	0	1	12	0	1	0	1	0	1	44	1	0	0	1	0	1	76	1	1	0	1	0	1	108
0	0	0	1	1	0	14	0	1	0	1	1	0	46	1	0	0	1	1	0	78	1	1	0	1	1	0	110
0	0	0	1	1	1	16	0	1	0	1	1	1	48	1	0	0	1	1	1	80	1	1	0	1	1	1	112
0	0	1	0	0	0	18	0	1	1	0	0	0	50	1	0	1	0	0	0	82	1	1	1	0	0	0	114
0	0	1	0	0	1	20	0	1	1	0	0	1	52	1	0	1	0	0	1	84	1	1	1	0	0	1	116
0	0	1	0	1	0	22	0	1	1	0	1	0	54	1	0	1	0	1	0	86	1	1	1	0	1	0	118
0	0	1	0	1	1	24	0	1	1	0	1	1	56	1	0	1	0	1	1	88	1	1	1	0	1	1	120
0	0	1	1	0	0	26	0	1	1	1	0	0	58	1	0	1	1	0	0	90	1	1	1	1	0	0	122
0	0	1	1	0	1	28	0	1	1	1	0	1	60	1	0	1	1	0	1	92	1	1	1	1	0	1	124
0	0	1	1	1	0	30	0	1	1	1	1	0	62	1	0	1	1	1	0	94	1	1	1	1	1	0	126
0	0	1	1	1	1	32	0	1	1	1	1	1	64	1	0	1	1	1	1	96	1	1	1	1	1	1	128

使用機能説明

表8 フレーム周波数の設定例

表示デューティ 1/N		サブクロック周波数 (kHz)		
		32.768	38.4	76.8
1/8	分周比 r	48	56	112
	フレーム周波数 f_F (Hz)	85.3	85.7	85.7
1/16	分周比 r	24	28	56
	フレーム周波数 f_F (Hz)	85.3	85.7	85.7
1/32	分周比 r	12	14	28
	フレーム周波数 f_F (Hz)	85.3	85.7	85.7

動作説明

(1) 本タスク例の動作原理について説明します。

(a) 表示メモリと表示データの設定について

本タスク例における表示メモリと表示データについて表9に示します。表9に示すように、アドレスレジスタ (R2) にX方向、Y方向のアドレスを指定し、表示データレジスタ (R4) に表示データを書込むことによりLCD表示を行ないます。

表9 表示データと表示メモリ

B'	A'	X=H'00	X=H'01	X=H'02	X=H'03	X=H'04	X=H'05	X=H'06	X=H'07
-	Y=H'00	1 0 0 0 0	0 0 0 0 0	0 0 0 0 0	1 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
-	Y=H'01	1 0 0 0 0	0 0 1 1 1	1 0 1 1 1	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
-	Y=H'02	1 1 1 1 1	0 0 1 0 0	1 0 0 0 1	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
-	Y=H'03	1 0 0 0 0	0 1 0 1 0	1 0 1 1 1	1 1 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
-	Y=H'04	1 0 0 0 0	0 0 0 0 1	1 0 0 0 1	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
-	Y=H'05	1 0 0 0 0	0 0 0 0 1	0 0 0 0 1	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
-	Y=H'06	0 1 1 1 1	0 0 1 1 0	0 0 0 1 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
-	Y=H'07	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'00	Y=H'08	1 0 0 0 1	0 0 1 1 1	0 0 0 0 0	0 0 1 1 1	1 1 0 0 1	1 1 0 0 1	1 1 1 1 0	1 1 1 1 1
Y=H'01	Y=H'09	1 0 0 0 1	0 1 0 0 0	1 0 0 0 0	0 0 0 0 0	1 0 0 1 0	0 0 1 0 1	1 0 0 0 0	1 0 0 0 1
Y=H'02	Y=H'0A	1 0 0 0 1	0 1 0 0 0	1 0 0 0 0	1 0 0 0 1	0 0 0 1 0	0 0 1 0 1	1 1 1 1 0	0 0 0 0 1
Y=H'03	Y=H'0B	1 1 1 1 1	0 0 1 1 1	0 0 0 0 1	0 0 0 0 0	1 0 0 0 1	1 1 0 0 0	0 0 0 1 0	0 0 0 1 0
Y=H'04	Y=H'0C	1 0 0 0 1	0 1 0 0 0	1 0 0 1 0	0 0 0 0 0	0 1 0 1 0	0 0 1 0 0	0 0 0 1 0	0 0 1 0 0
Y=H'05	Y=H'0D	1 0 0 0 1	0 1 0 0 0	1 0 1 0 0	0 0 1 0 0	0 1 0 1 0	0 0 1 0 1	1 0 0 0 1	0 0 0 1 0 0
Y=H'06	Y=H'0E	1 0 0 0 1	0 0 1 1 1	0 0 0 0 0	0 0 0 1 1	1 0 0 0 1	1 1 0 0 0	1 1 1 0 0	0 0 1 0 0
Y=H'07	Y=H'0F	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'08	Y=H'10	1 1 1 0 0	0 0 1 1 1	0 0 1 1 1	1 1 0 0 0	0 0 0 1 0	0 0 0 0 0	1 1 1 0 0	1 1 1 0 0
Y=H'09	Y=H'11	1 0 0 1 0	0 1 0 0 0	1 0 0 0 1	0 0 0 0 0	0 0 0 1 0	0 0 0 0 1	0 0 0 1 0	1 0 0 1 0
Y=H'0A	Y=H'12	1 0 0 0 1	0 1 0 0 0	1 0 0 0 1	0 0 0 0 0	0 0 0 1 0	0 0 0 0 1	0 0 0 0 0	1 0 0 0 1
Y=H'0B	Y=H'13	1 0 0 0 1	0 1 0 0 0	1 0 0 0 1	0 0 0 0 0	0 0 0 1 0	0 0 0 0 1	0 0 0 0 0	1 0 0 0 1
Y=H'0C	Y=H'14	1 0 0 0 1	0 1 0 0 0	1 0 0 0 1	0 0 0 0 0	0 0 0 1 0	0 0 0 0 1	0 0 0 0 0	1 0 0 0 1
Y=H'0D	Y=H'15	1 0 0 1 0	0 1 0 0 0	1 0 0 0 1	0 0 0 0 0	0 0 0 1 0	0 0 0 0 1	0 0 0 1 0	1 0 0 1 0
Y=H'0E	Y=H'16	1 1 1 0 0	0 0 1 1 1	0 0 0 0 1	0 0 0 0 0	0 0 0 1 1	1 1 1 0 0	1 1 1 0 0	1 1 1 0 0
Y=H'0F	Y=H'17	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'10	Y=H'18	0 0 0 0 0	0 0 0 0 0	0 0 0 1 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	1 1 1 0 0	0 0 0 0 0
Y=H'11	Y=H'19	1 1 1 1 1	0 1 1 0 0	0 0 0 1 0	0 0 1 1 1	1 1 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 1
Y=H'12	Y=H'1A	0 0 0 0 1	0 0 0 0 0	0 0 0 1 0	0 0 1 0 0	0 1 0 0 0	0 0 0 0 1	1 1 1 1 0	0 0 0 1 0
Y=H'13	Y=H'1B	0 0 0 0 1	0 0 0 0 0	1 0 0 1 1	0 0 1 0 0	0 1 0 1 1	1 1 1 0 0	0 0 0 1 0	0 0 1 0 0
Y=H'14	Y=H'1C	0 0 0 0 1	0 0 0 0 0	1 0 0 1 0	1 0 1 0 0	0 1 0 0 0	0 0 0 0 0	0 0 0 1 0	0 1 0 0 0
Y=H'15	Y=H'1D	0 0 0 0 1	0 0 0 0 1	0 0 0 1 0	0 0 1 0 0	0 1 0 0 0	0 0 0 0 0	0 0 1 0 0	1 0 0 0 0
Y=H'16	Y=H'1E	1 1 1 1 1	0 1 1 1 0	0 0 0 1 0	0 0 1 1 1	1 1 0 0 0	0 0 0 0 0	0 1 0 0 0	0 0 0 0 0
Y=H'17	Y=H'1F	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'18	-	0 1 0 1 1	0 0 1 1 1	0 0 0 0 0	0 1 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'19	-	0 1 0 0 0	0 0 0 0 0	0 0 0 0 0	1 0 0 0 0	0 1 1 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'1A	-	0 1 0 0 0	0 1 1 1 1	1 0 0 0 1	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'1B	-	0 1 1 0 0	0 0 0 0 0	1 0 0 1 1	0 0 0 0 0	1 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'1C	-	0 1 0 1 0	0 0 0 0 0	1 0 1 0 1	0 0 0 1 0	0 1 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'1D	-	0 1 0 0 0	0 0 0 0 1	0 0 0 0 1	0 0 0 1 0	0 0 1 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'1E	-	0 1 0 0 0	0 0 0 1 0	0 0 0 0 1	0 0 0 1 0	0 0 1 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0
Y=H'1F	-	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0 0

【注】 A:スクロール前
B:スクロール後

動作説明

(b)動作原理

図17に本タスク例における動作原理を示します。図17に示すようなソフトウェア処理、およびハードウェア処理によりキャラクター表示モードによるLCD表示を行ないます。

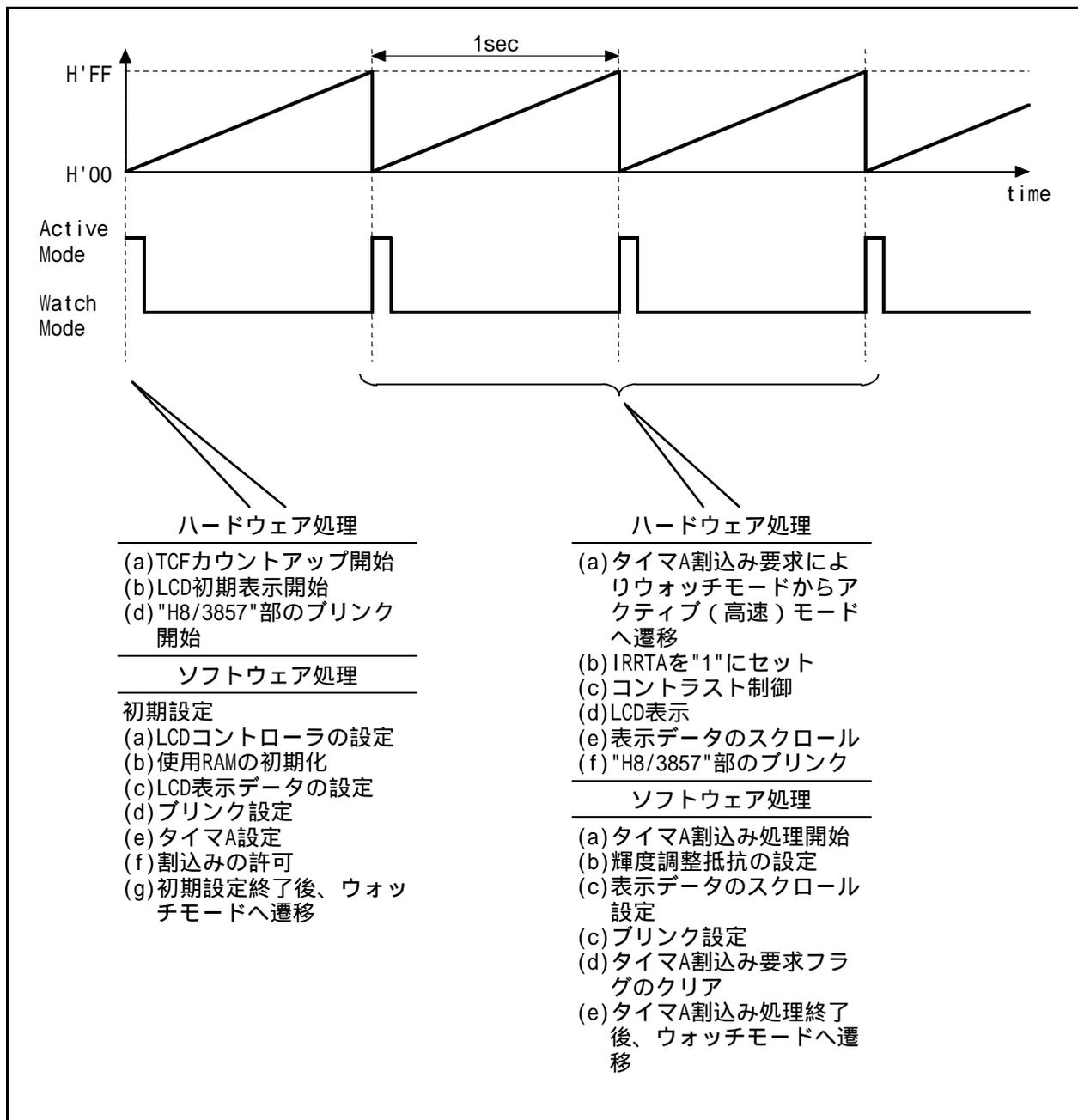


図17 動作原理

ソフトウェア説明

(1) モジュール説明

表10に本タスク例におけるモジュール説明を示します。

表10 モジュール説明

モジュール名	ラベル名	機能
メインルーチン	INIT	スタックポインタ、使用RAM、ドットマトリックスLCDコントローラ、タイマAのイニシャライズ、表示データの表示メモリへの書き込み、割り込み許可、ウォッチモードへの遷移を行う。
インデックスレジスタ選択ルーチン	INDEX	インデックスレジスタ (IR) の選択を行う。
LCDコントロールレジスタライトルーチン	WRDATA	IRにより選択した制御レジスタにデータを書き込む。
タイマA割り込み処理ルーチン	TMRA	表示データのスクロール、およびブリンクの制御を行う。

(2) 引数の説明

本タスク例では、引数は使用していません。

(3) 使用内部レジスタ説明

表11に本タスク例における使用内部レジスタ説明を示します。

表11 使用内部レジスタ説明

レジスタ名	説明	IR 設定値	設定値
IR IR3~ IRO	<p>インデックスレジスタ (インデックスレジスタ3~0) ~LCDコントローラの10本の制御レジスタのうち1つを選択するビットです。</p> <p>: IR3="0"、IR2="0"、IR1="0"、IRO="0"のとき、R0を選択 : IR3="0"、IR2="0"、IR1="0"、IRO="1"のとき、R1を選択 : IR3="0"、IR2="0"、IR1="1"、IRO="0"のとき、R2を選択 : IR3="0"、IR2="0"、IR1="1"、IRO="1"のとき、R3を選択 : IR3="0"、IR2="1"、IR1="0"、IRO="0"のとき、R4を選択 : IR3="0"、IR2="1"、IR1="0"、IRO="1"のとき、R5を選択 : IR3="0"、IR2="1"、IR1="1"、IRO="0"のとき、R6を選択 : IR3="1"、IR2="0"、IR1="0"、IRO="0"のとき、R8を選択 : IR3="1"、IR2="0"、IR1="0"、IRO="1"のとき、R9を選択 : IR3="1"、IR2="0"、IR1="1"、IRO="0"のとき、RAを選択</p> <p>【注】1. レジスタへアクセスするには、RSを"0"にしてIRにアクセスしたいレジスタのレジスタ番号を設定します。その後、RSを"1"にして指定したレジスタにアクセスすることができます。 2. 一部の内部レジスタには存在しないビットがありますが、これらのビットには"0"を書込むようにしてください。</p>	- bit 3~ bit 0	-
R0 LSBY	<p>コントロールレジスタ (モジュールスタンバイ) ~LSBYはモジュールスタンバイ設定ビットです。LSBYを"1"に設定するとLCDコントローラはスタンバイモードになります。その時、PSRビットの状態には影響を与えませんが、DISPビットおよびOPONビットはリセットされます。</p> <p>: LSBY="0"のとき、LCDは通常動作します。 : LSBY="1"のとき、昇圧、内部動作を停止し、表示をオフし、LCDはスタンバイモードとなります。</p>	IR3=0 IR2=0 IR1=0 IRO=0 bit 5	"0"

ソフトウェア説明

表11 使用内部レジスタ説明(つづき)

レジスタ名	説明	IR 設定値	設定値
R0	PWR (昇圧回路動作設定) ~ PWRは昇圧回路の動作/停止を設定するビットです。 : PWR="0"のとき、昇圧回路は停止します。 : PWR="1"のとき、昇圧回路は動作します。	bit 4	"1"
	SOB (表示モード選択) ~ SOBは表示モードをキャラクタ表示モードか、グラフィック表示モードか選択するビットです。 : SOB="0"のとき、キャラクタ表示モードに設定します。表示メモリの1バイトのうち、ビット4~0をセグメント端子に出力します。 : SOB="1"のとき、グラフィック表示モードに設定します。表J9位メモリの1バイトのデータのすべてをセグメント端子に出力します。出力できるXアドレスは、1/32デューティの場合H'0~H'4、1/16デューティの場合H'0~H'6、1/8デューティの場合H'0~H'7の範囲です。	bit 2	"0"
	DDTY1, DDTY0 (表示デューティ選択) ~ DDTY1、DDTY0は表示デューティを1/32、1/16、または1/8から選択するビットです。 : DDTY1="0"、DDTY0="0"のとき、1/32デューティを選択します。 : DDTY1="0"、DDTY0="1"のとき、1/16デューティを選択します。YアドレスのH'10~H'1Fの表示データは無効です。 : DDTY1="1"、DDTY0="*"のとき、1/8デューティを選択します。YアドレスのH'08~H'1Fの表示データは無効です。	bit1, bit0	DDTY1 =0 DDTY0 =0
R1	DISP コントロールレジスタ2(液晶表示動作設定) ~ DISPは液晶表示の動作/停止を設定するビットです。R0のLSBYを"1"に設定するとDISPはクリアされます。 : DISP="0"のとき、液晶表示をオフします。全液晶表示出力はVssレベルとなります。 : DISP="1"のとき、液晶表示をオンします。	IR3=0 IR2=0 IR1=0 IR0=1 bit 6	"1"
	OPS (オペアンプ回路のパワーセーブ設定) ~ OPSはオペアンプ回路のパワーセーブ機能の有効/向こうを設定するビットです。OPONが"1"で内蔵のオペアンプ回路が動作しているとき、OPSを"1"に設定するとパワーセーブ機能が有効になります。液晶表示出力が安定しないばあいはOPSを"0"に設定してください。 : OPS="0"のとき、パワーセーブ機能は無効です。 : OPS="1"のとき、パワーセーブ機能が有効になり、ボルテージフォロア型オペアンプの駆動電流を低減します。この場合、V1OUT~V5OUT端子とVss間に0.1~0.5μFのコンデンサを挿入してください。	bit 5	"0"
	OPON (オペアンプ回路動作設定) ~ OPONはオペアンプ回路を動作/停止を設定するビットです。OPONを"1"に設定するとオペアンプ回路が動作します。外部からV1OUT~V5OUTにLCD駆動電源レベルを印加する場合は必ずOPONを"0"に設定してください。 : OPON="0"のとき、内蔵オペアンプは停止し、出力はハイインピーダンスとなります。外部から液晶駆動電圧を入力できます。 : OPON="1"のとき、内蔵オペアンプは動作します。	bit 4	"1"
	RMW (リード・モディファイ・ライトモード設定) ~ RMWは表示メモリのXまたはYアドレスのインクリメントをライト/リードのアクセス後に行なうか、ライト後のみ(リード・モディファイ・ライトモード)で行なうかを選択するビットです。 : RMW="0"のとき、表示メモリへのライト/リードアクセス後にアドレスをインクリメントします。 : RMW="1"のとき、リード・モディファイ・ライトモードに設定します。このモードでは表示メモリへのライトアクセス後にのみアドレスをインクリメントします。	bit 3	"0"

【注】*:Don't Care

ソフトウェア説明

表11 使用内部レジスタ説明(つづき)

レジスタ名	説明	IR 設定値	設定値
R1	INC (インクリメントアドレス選択) ~ INCは表示データレジスタ(R4)へのRMWで設定したアクセス後にインクリメントするアドレスをXアドレスかYアドレスか選択するビットです。選択されたアドレスは表示データの有効エリアの最大値で表示データレジスタ(R4)へのアクセス後にクリアされ、そのとき、他方のアドレスをインクリメントします。 : INC="0"のとき、表示メモリのYアドレスのインクリメントが優先され、Yアドレスのオーバフロー後にXアドレスをインクリメントします。 : INC="1"のとき、邦字メモリのXアドレスのインクリメントが優先され、Xアドレスのオーバフロー後にYアドレスをインクリメントします。	bit 1	"0"
	BLK (ブリンク動作設定) ~ BLKはブリンク機能の有効/向こうを設定するビットです。DISPが"1"で液晶表示しているとき、BLKを"1"に設定するとブリンク機能が有効になり、R6のBK7~BK0、R8のBSL4~BSL0、R9のBEL4~BELOで設定した範囲をブリンク表示します。 : BLK="0"のとき、ブリンクは無効です。 : BLK="1"のとき、ブリンクが有効となります。	bit 0	"1"
R2	XA2~XA0 アドレスレジスタ(Xアドレス設定) ~ XA2~XA0は表示メモリのX方向のアドレスを設定するビットです。H'0~H'7の範囲が設定可能ですが、SOB="1"の場合は1/16デューティ時H'7、1/32デューティ時H'5~H'7の範囲の表示データは無効となります。INCが"1"のとき、RMWで設定したアクセス後に自動的にインクリメントされ、表示データの有効エリアの最大値のアクセス後にクリアされます。また、INCが"0"でYA4~YA0が表示データの有効のエリアの最大値のとき、RMWで設定したアクセス後にインクリメントされます。	IR3=0 IR2=0 IR1=1 IR0=0 bit 7~ bit 5	XA2=0 XA1=0 XA0=0
	YA4~YA0 (Yアドレス設定) ~ YA4~YA0は表示メモリのY方向のアドレスを設定するビットです。H'00~H'1Fの範囲が設定可能ですが、1/16デューティの場合はH'10~H'1Fの範囲、また1/8デューティの場合はH'08~H'1Fの範囲の表示データは向こうとなります。INCが"0"のとき、RMWで設定したアクセス後に自動的にインクリメントされ、表示データの有効エリアの最大値のアクセス後にクリアされます。また、INCが"1"でXA2~XA0が表示データの有効のエリアの最大値のとき、RMWで設定したアクセス後にインクリメントされます。	bit 4~ bit 0	YA4=0 YA3=0 YA2=0 YA1=0 YA0=0
R3	FS5~FS0 フレーム周波数設定レジスタ(フレーム周波数設定5~0) ~ FS5~FS0は使用する液晶パネルの特性に合わせて最適な分周比を設定します。レジスタ設定値と分周比の関係を表8に示します。 : FS5="0"、FS4="0"、FS3="0"、FS2="1"、FS1="0"、FS0="1"のとき、サブクロックの分周比を12に設定します。これにより、フレーム周波数は85.3Hzに設定されます。	IR3=0 IR2=0 IR1=1 IR0=1 bit 5~ bit 0	FS5=0 FS4=0 FS3=0 FS2=1 FS1=0 FS0=1
R4	表示データレジスタ ~ R2のXA2~XA0とYA4~YA0で指定される表示メモリにリード/ライトを行ないます。	IR3=0 IR2=1 IR1=0 IR0=0	-

ソフトウェア説明

表11 使用内部レジスタ説明(つづき)

レジスタ名		説明	IR 設定値	設定値
R5	ST4 ~ ST0	表示開始ラインレジスタ(表示開始ライン設定4~0) ~ST4~ST0は表示を開始するラインを指定するビットです。設定値は、「表示開始ライン - 1」を指定してください。本レジスタの設定値を換えることにより、縦方向のスクロールが実現できます。設定範囲は、1/32デューティの場合0~31、1/16デューティの場合0~15、1/8デューティの場合0~7です。これらの制限値を超えた値を設定しますと、正常な表示は行なえません。	IR3=0 IR2=1 IR1=0 IR0=1 bit 4~ bit 0	ST4=0 ST3=0 ST2=0 ST1=0 ST0=0
R6		ブリンクレジスタ ~R6はブリンクさせたい領域に対応するビットに"1"を書込みます。同時にブリンクさせる範囲には制限がなく、全ビットに"1"を書込むことによって全画面をブリンクさせることも可能です。このレジスタの設定値は、BLKが"1"に設定されているときにのみ有効です。各ビットに対応するブリンクの範囲は、SOBの値により次のようになります。 : SOB="0"のとき、BK7 : SEG36 ~ SEG40, BK6 : SEG31 ~ SEG35 BK5 : SEG26 ~ SEG30, BK4 : SEG21 ~ SEG25 BK3 : SEG16 ~ SEG20, BK2 : SEG11 ~ SEG15 BK1 : SEG6 ~ SEG10, BK0 : SEG1 ~ SEG5 : SOB="1"のとき、BK7 : SEG57 ~ SEG64, BK6 : SEG49 ~ SEG56 BK5 : SEG41 ~ SEG48, BK4 : SEG33 ~ SEG40 BK3 : SEG25 ~ SEG32, BK2 : SEG17 ~ SEG24 BK1 : SEG9 ~ SEG16, BK0 : SEG1 ~ SEG8	IR3=0 IR2=1 IR1=1 IR0=0	H'FF
R8	BSL4 ~ BSL0	ブリンク開始ラインレジスタ(ブリンク開始ライン設定4~0) ~BSL4~BSL0はブリンクさせる領域の先頭のラインを指定するビットです。設定値は、「ブリンク開始ライン - 1」をセットしてください。設定範囲は、1/32デューティの場合0~31、1/16デューティの場合0~15、1/8デューティの場合0~7です。これらの制限値を超えた値を設定しますと、正常な動作は保証されません。	IR3=0 IR2=1 IR1=1 IR0=0 bit 4~ bit 0	BSL4=0 BSL4=1 BSL4=0 BSL4=0 BSL4=0
R9	BEL4 ~ BELO	ブリンク終了ラインレジスタ(ブリンク終了ライン設定4~0) ~BEL4~BELOはブリンクさせる領域の終わりのラインを指定するビットです。設定値は、「ブリンク終了ライン - 1」をセットしてください。設定範囲は、1/32デューティの場合0~31、1/16デューティの場合0~15、1/8デューティの場合0~7です。これらの制限値を超えた値を設定しますと、正常な動作は保証されません。	IR3=0 IR2=1 IR1=1 IR0=0 bit 4~ bit 0	BEL4=0 BEL4=1 BEL4=1 BEL4=1 BEL4=1

ソフトウェア説明

表11 使用内部レジスタ説明(つづき)

レジスタ名	説明	IR/ アドレス	設定値
RA CCR3 ~ CCR0	<p>コントラストコントロールレジスタ(輝度調整設定3~0) ~CCR3~BELOはV_{LCD}とV₁レベルの間の輝度調整抵抗値を設定するビットです。V_{LCD}とV₁レベルの間の輝度調整抵抗と調節することにより、液晶パネルの輝度調整が可能です。輝度調整抵抗は、液晶プリアダ抵抗R_iに対し0.1R~1.6Rの範囲で設定可能です。</p> <p>: CCR3="0", CCR2="0", CCR1="0", CCR0="0"のとき、輝度調整抵抗は1.6R : CCR3="0", CCR2="0", CCR1="0", CCR0="1"のとき、輝度調整抵抗は1.5R : CCR3="0", CCR2="0", CCR1="1", CCR0="0"のとき、輝度調整抵抗は1.4R : CCR3="0", CCR2="0", CCR1="1", CCR0="1"のとき、輝度調整抵抗は1.3R : CCR3="0", CCR2="1", CCR1="0", CCR0="0"のとき、輝度調整抵抗は1.2R : CCR3="0", CCR2="1", CCR1="0", CCR0="1"のとき、輝度調整抵抗は1.1R : CCR3="0", CCR2="1", CCR1="1", CCR0="0"のとき、輝度調整抵抗は1.0R : CCR3="0", CCR2="1", CCR1="1", CCR0="1"のとき、輝度調整抵抗は0.9R : CCR3="1", CCR2="0", CCR1="0", CCR0="0"のとき、輝度調整抵抗は0.8R : CCR3="1", CCR2="0", CCR1="0", CCR0="1"のとき、輝度調整抵抗は0.7R : CCR3="1", CCR2="0", CCR1="1", CCR0="0"のとき、輝度調整抵抗は0.6R : CCR3="1", CCR2="0", CCR1="1", CCR0="1"のとき、輝度調整抵抗は0.5R : CCR3="1", CCR2="1", CCR1="0", CCR0="0"のとき、輝度調整抵抗は0.4R : CCR3="1", CCR2="1", CCR1="0", CCR0="1"のとき、輝度調整抵抗は0.3R : CCR3="1", CCR2="1", CCR1="1", CCR0="0"のとき、輝度調整抵抗は0.2R : CCR3="1", CCR2="1", CCR1="1", CCR0="1"のとき、輝度調整抵抗は0.1R</p>	IR3=0 IR2=1 IR1=1 IR0=0 bit 4~ bit 0	CCR3=0 CCR2=0 CCR1=0 CCR0=0
PDR9	<p>ポートデータレジスタ9 ~ポート9の各端子P9₇~P9₀のデータを格納します。 : P9_n="0"のとき、P9_n端子のデータは0 : P9_n="1"のとき、P9_n端子のデータは1</p>	H'FFDC	H'00
PDRA	<p>ポートデータレジスタA ~ポートAの各端子PA₃~PA₀のデータを格納します。 : PA_n="0"のとき、PA_n端子のデータは0 : PA_n="1"のとき、PA_n端子のデータは1</p>	H'FFDD	H'00
PCR9	<p>ポートコントロールレジスタ9 ~ポート9の各端子P9₇~P9₀のデータを格納します。 : PCR9_n="0"のとき、P9_nは入力端子機能 : PCR9_n="1"のとき、P9_nは出力端子機能</p>	H'FFEC	H'FF
PCRA	<p>ポートコントロールレジスタA ~ポートAの各端子PA₃~PA₀のデータを格納します。 : PCRA_n="0"のとき、PA_nは入力端子機能 : PCRA_n="1"のとき、PA_nは出力端子機能</p>	H'FFED	H'FF
TMA TMA3	<p>タイマモードレジスタA(タイマモードレジスタA3) ~TCAの入カクロックソースを選択します。 : TMA3="0"のとき、TCAの入カクロックソースはPSS : TMA3="1"のとき、TCAの入カクロックソースはPSW</p>	H'FFB0 ビット3	"1"
IENR1 IENTA	<p>割込み許可レジスタ1(タイマ割込みイネーブル) ~タイマ割込み要求の許可/禁止を制御します。 : IENTA="0"のとき、タイマ割込み要求を禁止 : IENTA="1"のとき、タイマ割込み要求を許可</p>	H'FFF3 ビット7	"1"
IRR1 IRRTA	<p>割込み要求レジスタ1(タイマ割込み要求フラグ) ~タイマ割込み要求の有無を反映します。 : IRRTA="0"のとき、タイマ割込みが要求されていない : IRRTA="1"のとき、タイマ割込みが要求されている</p>	H'FFF6 ビット7	"0"

ソフトウェア説明

表11 使用内部レジスタ説明(つづき)

レジスタ名	説明	RAM アドレス	設定値
SYSCR1 SSBY	システムコントロールレジスタ1(ソフトウェアスタンバイ) ~スタンバイモード、ウォッチモードへの遷移を指定します。 : SSBY="0"のとき、アクティブモードでSLEEP命令を実行後、スリープ モードに遷移 : SSBY="1"のとき、アクティブモードでSLEEP命令を実行後、ウォッチ モードあるいはスタンバイモードに遷移	H'FFF0 ビット7	"1"
STS2 STS1 STS0	(スタンバイタイムセレクト2~0) ~特定の割込みにより、スタンバイモード、ウォッチモードを解除 し、アクティブモードに遷移する場合、クロックが安定するまでCPU と周辺機能が待機する時間を指定します。 : STS2="0", STS1="0", STS0="0"のとき、待機時間は 8,192ステート : STS2="0", STS1="0", STS0="1"のとき、待機時間は 16,384ステート : STS2="0", STS1="1", STS0="0"のとき、待機時間は 32,768ステート : STS2="0", STS1="1", STS0="1"のとき、待機時間は 65,536ステート : STS2="1", STS1="*", STS0="*"のとき、待機時間は131,972ステート ただし、*:Don't Care	H'FFF0 ビット6 ビット5 ビット4	STS2=0 STS1=0 STS0=0
LSON	(ロースピードオンフラグ) ~ウォッチモード解除時に、CPUの動作クロックを指定します。 : LSON="0"のとき、CPUの動作クロックはシステムクロック : LSON="1"のとき、CPUの動作クロックはサブクロック	H'FFF0 ビット3	"0"
SYSCR2 MSON	(ミドルスピードオンフラグ) ~ウスタンバイモード、ウォッチモード、スリープモード解除後、ア クティブ(高速)モードで動作させるか、アクティブ(中速)モー ドで動作させるかを選択します。 : MSON="0"のとき、アクティブ(高速)モードで動作 : MSON="1"のとき、アクティブ(中速)モードで動作	H'FFF1 ビット2	"0"

(4) 使用RAM説明

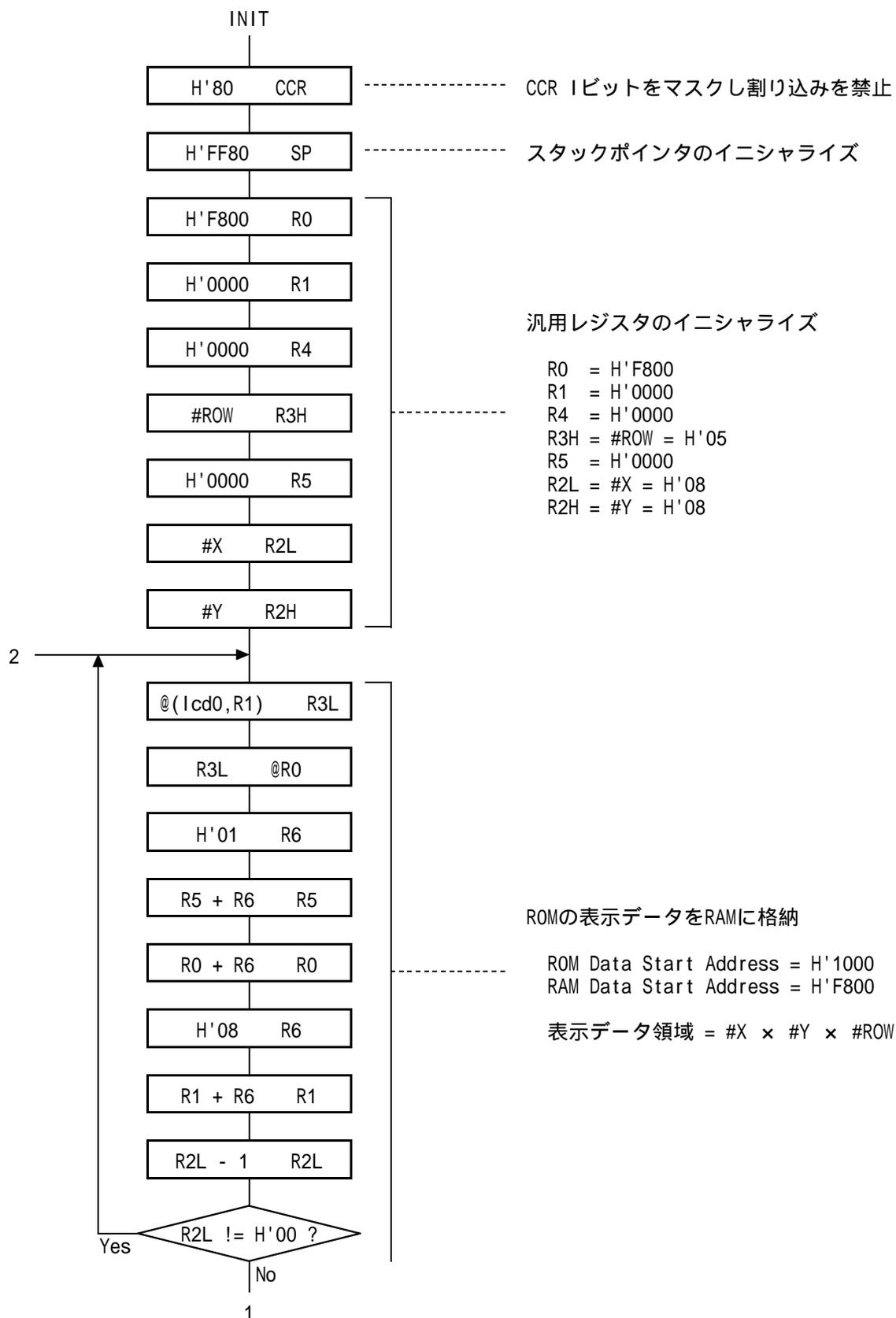
表12に本タスク例における使用RAM説明を示します。

表12 使用RAM説明

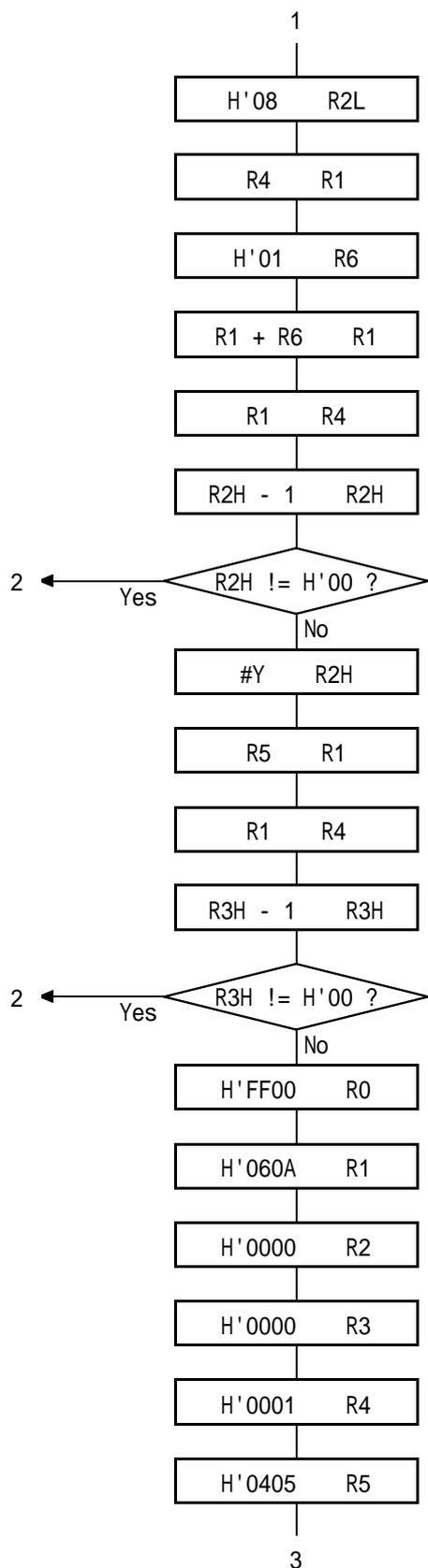
ラベル名	機能	RAM アドレス	使用モジュール
POINTER	現在書き込んでいる表示データを格納しているRAMのアドレスを 示すポインタ	H'FA00	INIT, TMRA
ADDRESS	現在書き込んでいる表示メモリのアドレスを格納	H'FA04	INIT, TMRA
POS	現在書き込んでいる表示メモリのアドレスを示すポインタ	H'FA06	INIT, TMRA
COLOR	コントラストコントロールレジスタ(RA)に設定する輝度調整抵 抗の抵抗値を設定するデータを格納します	H'FA08	INIT, TMRA

フローチャート

(1) メインルーチン



フローチャート



ROMの表示データをRAMに格納

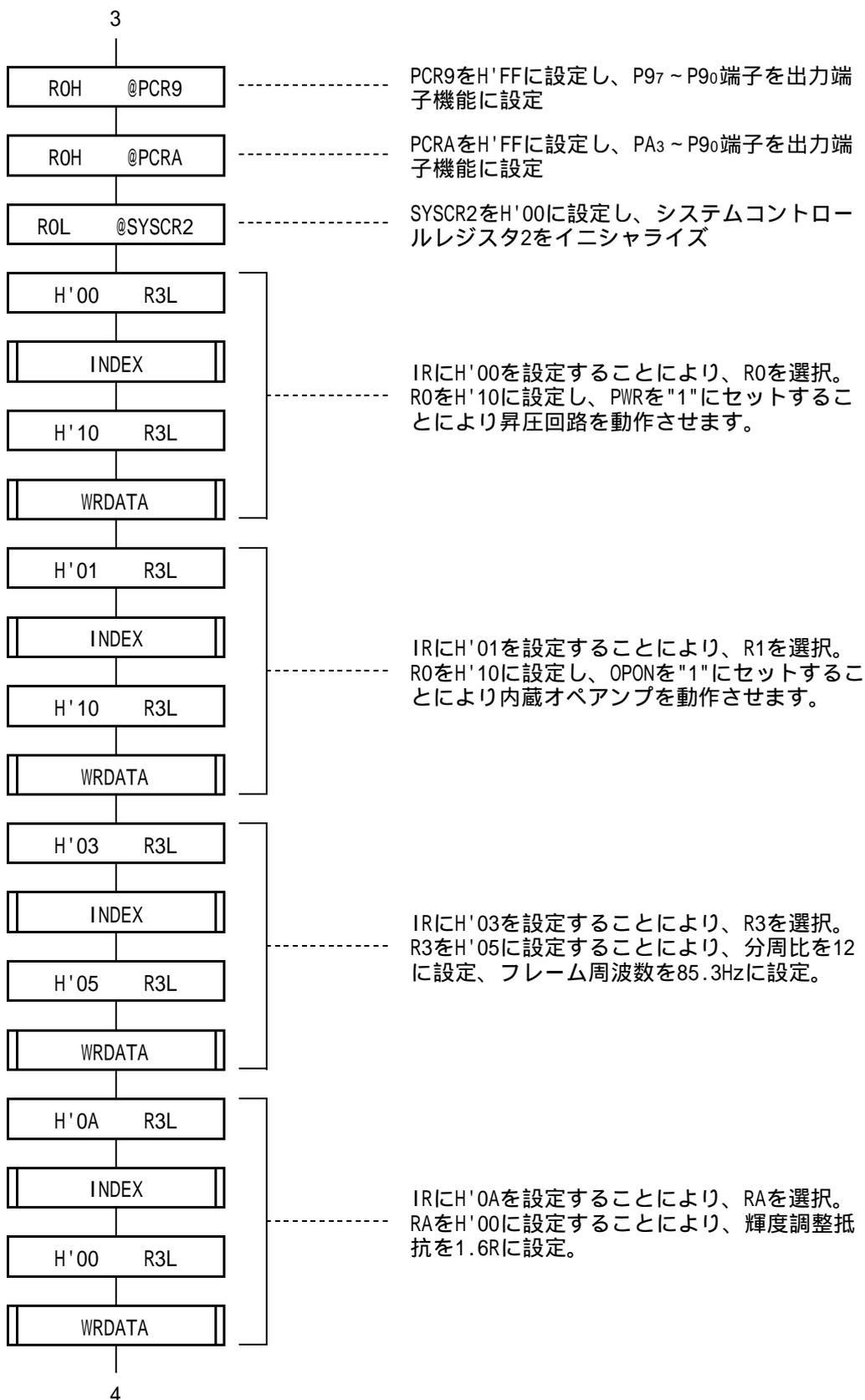
ROM Data Start Address = H'1000
RAM Data Start Address = H'F800

表示データ領域 = X × Y × ROW Byte

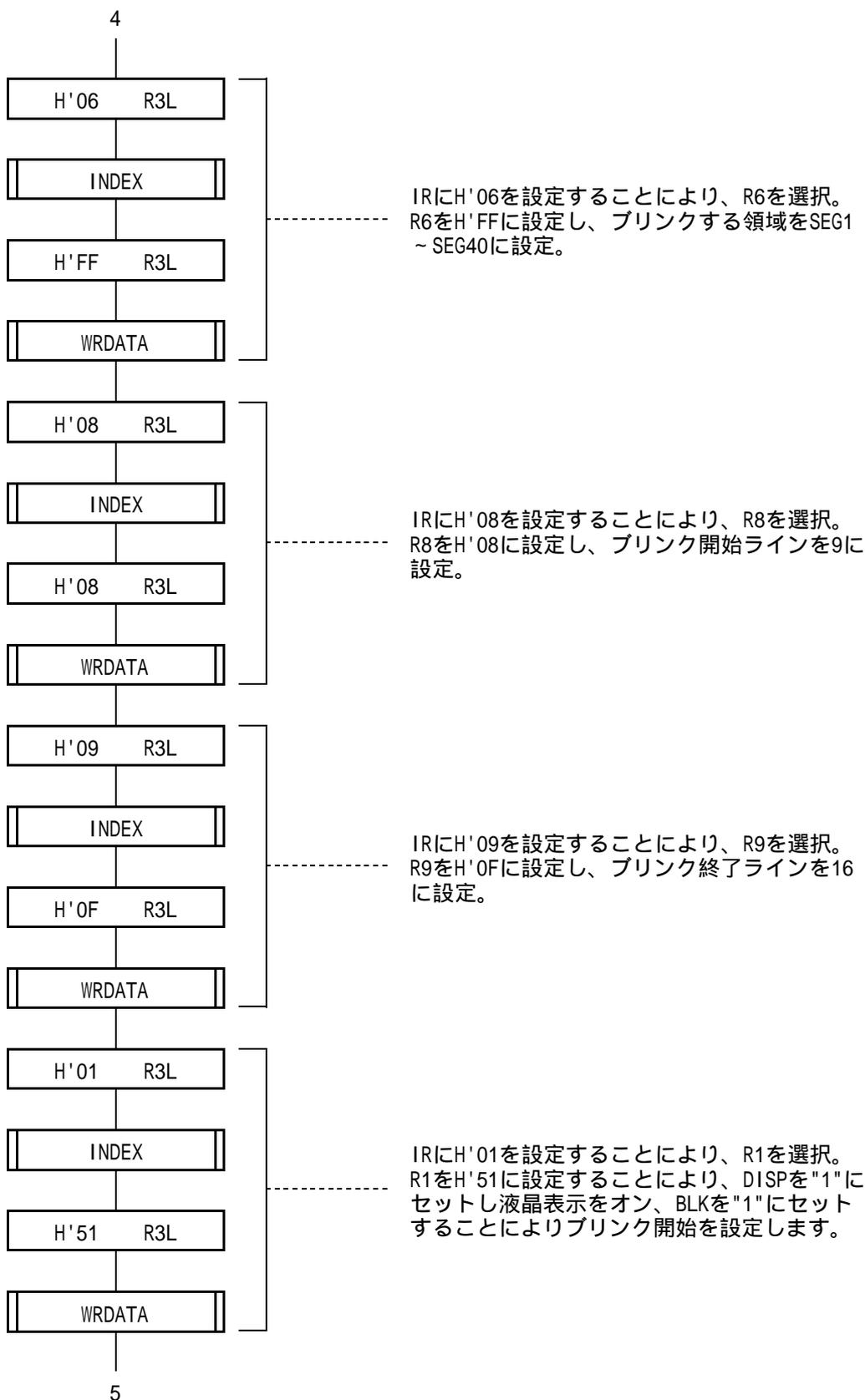
汎用レジスタのイニシャライズ

R0 = H'FF00
R1 = H'060A
R2 = H'0000
R3 = H'0000
R4 = H'0001
R5 = H'0405

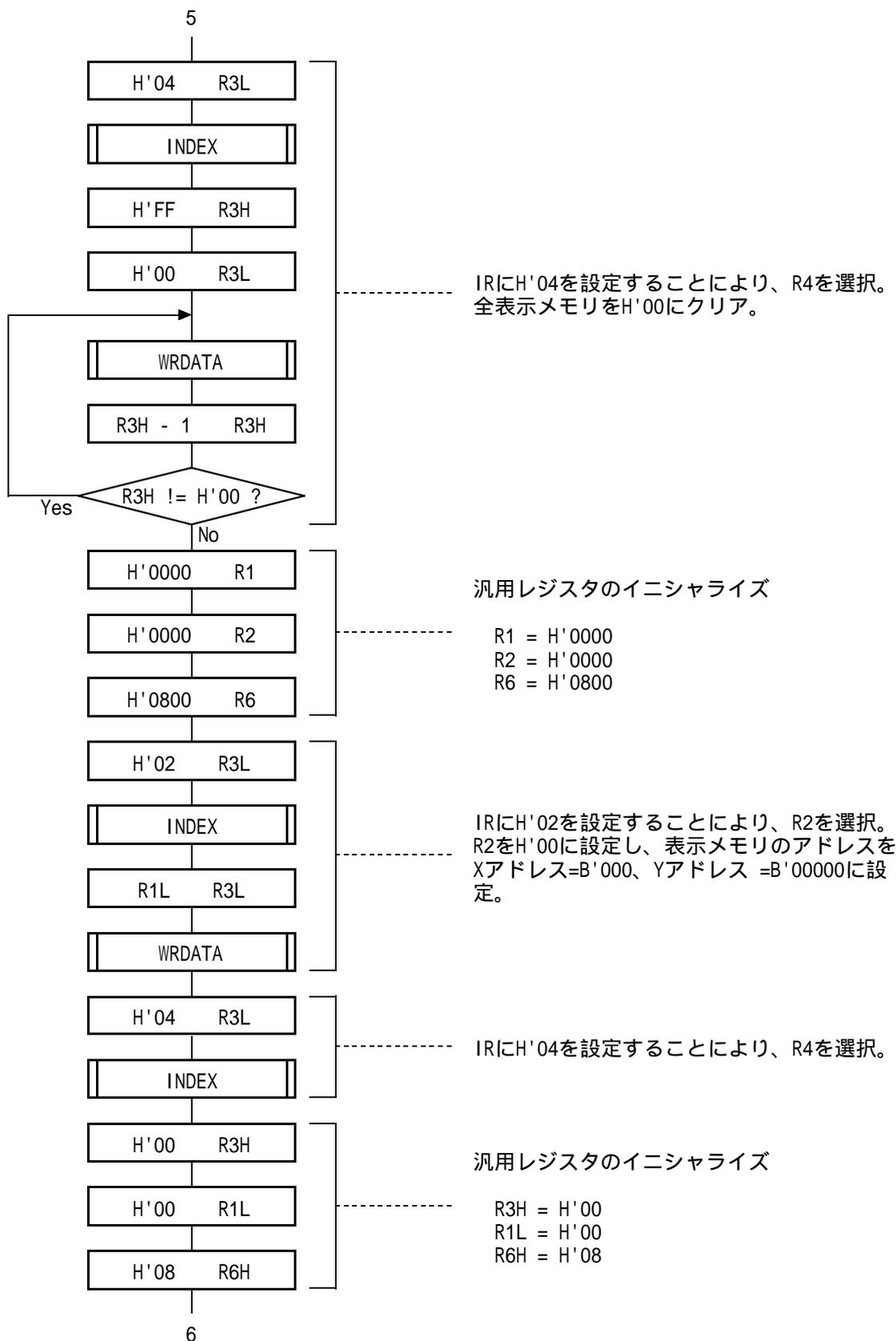
フローチャート



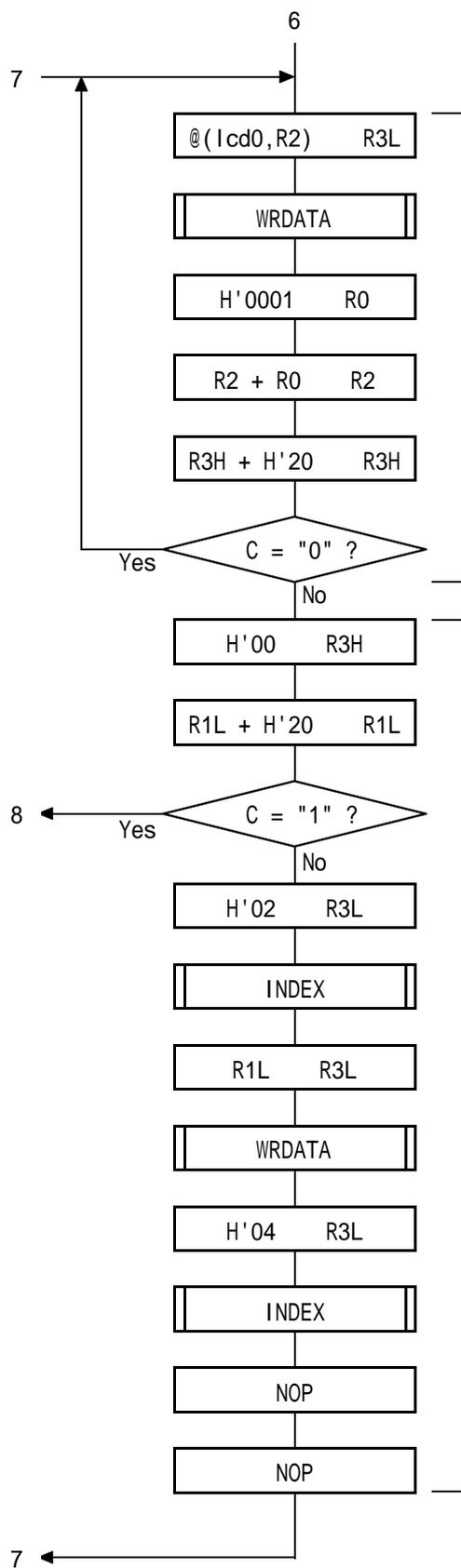
フローチャート



フローチャート



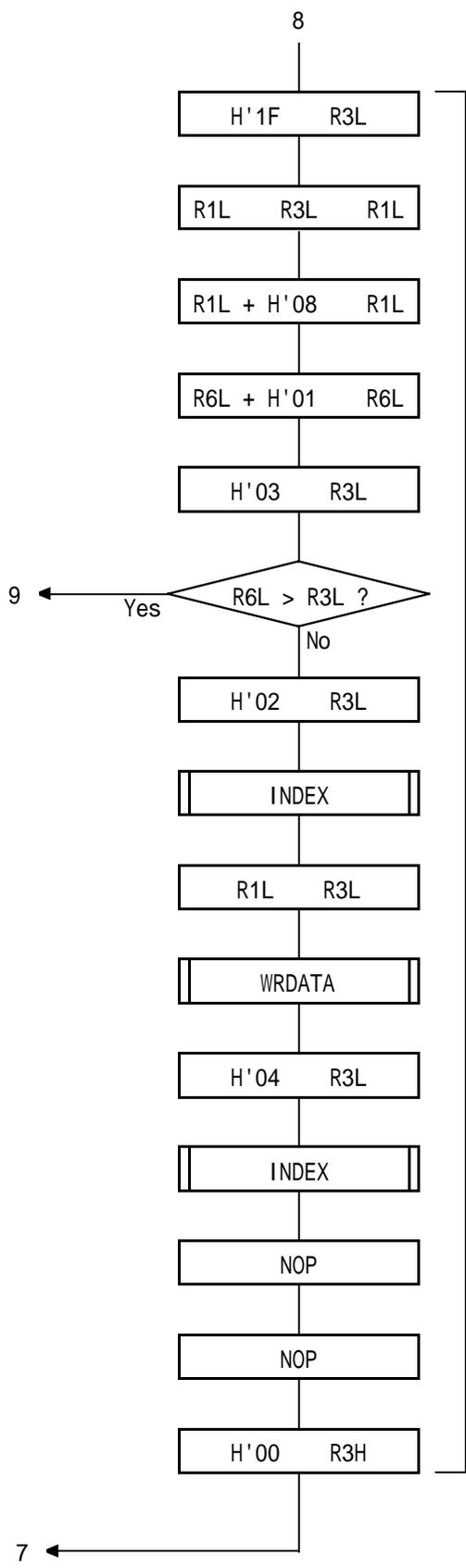
フローチャート



1キャラクタ分の表示データ (1Byte×8) を表示RAMに設定
Yアドレス : H'00 ~ H'07
Xアドレス : H'00

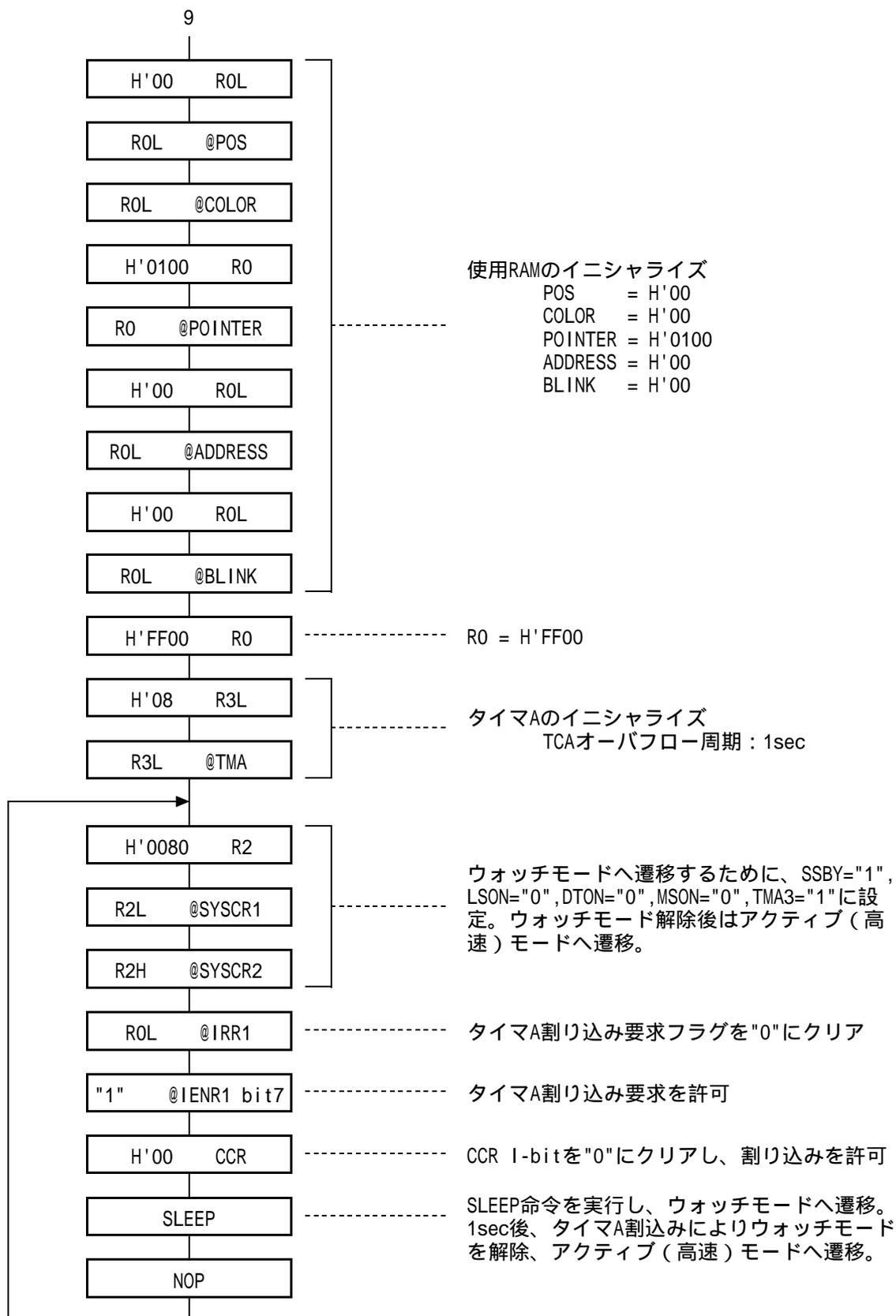
1行分の表示データ (1キャラクタ×8) を表示RAMに設定
Yアドレス : H'00からH'07
Xアドレス : H'00からH'07

フローチャート



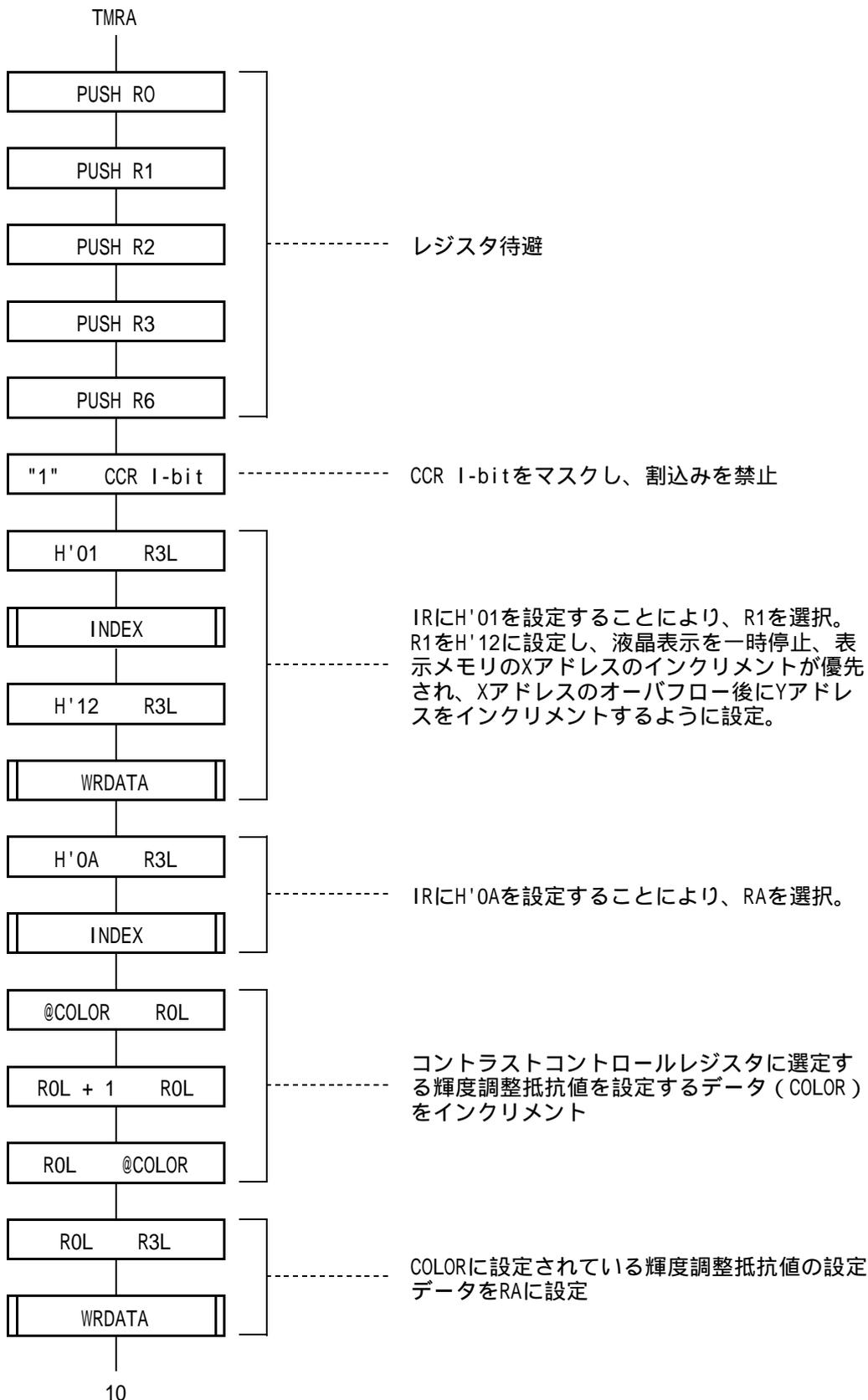
4行分の表示データ（8キャラクタ×4）を表示
RAMに設定
Yアドレス：H'00～H'1F
Xアドレス：H'00～H'07

フローチャート

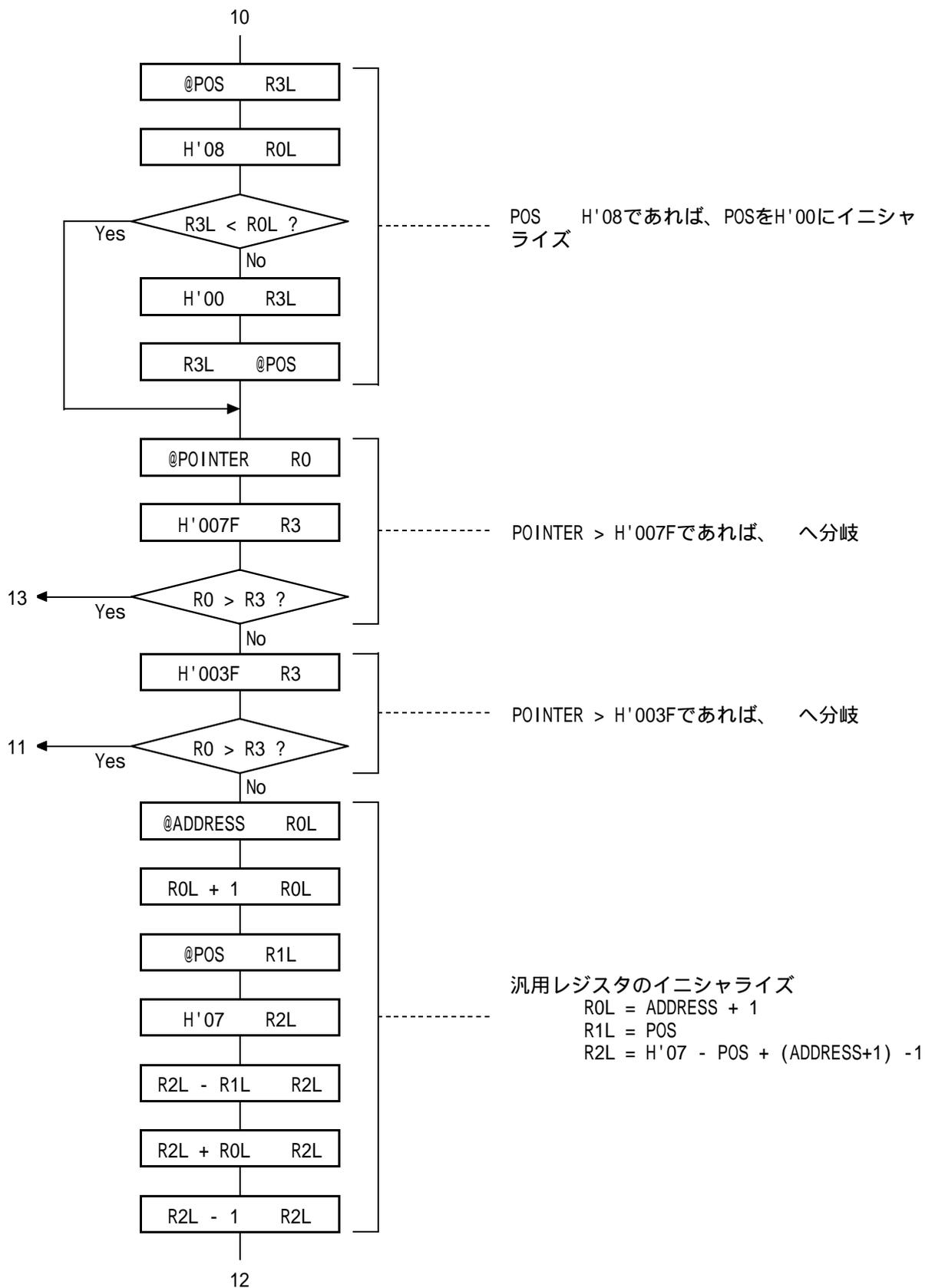


フローチャート

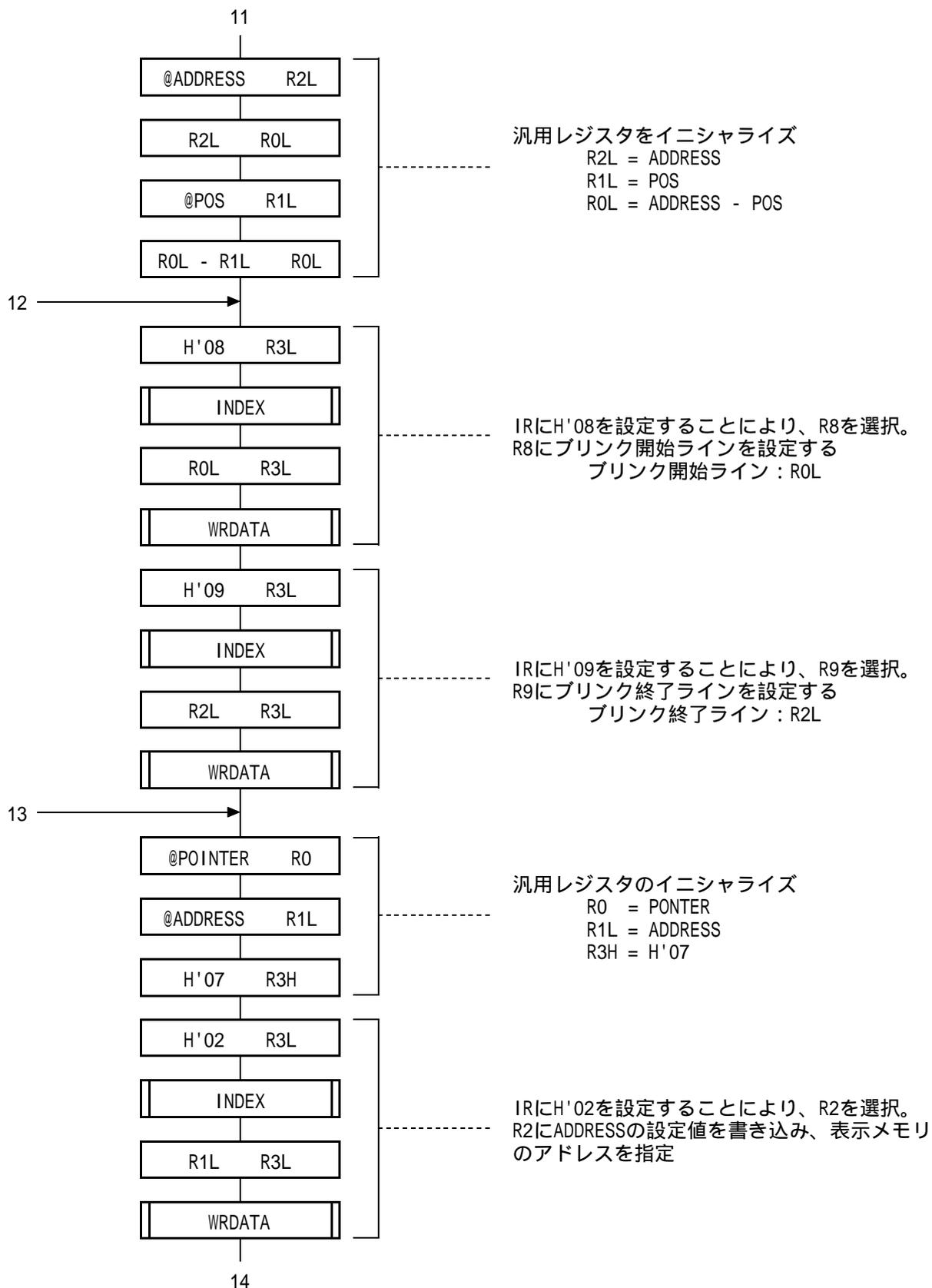
(2) タイマA割込み処理ルーチン



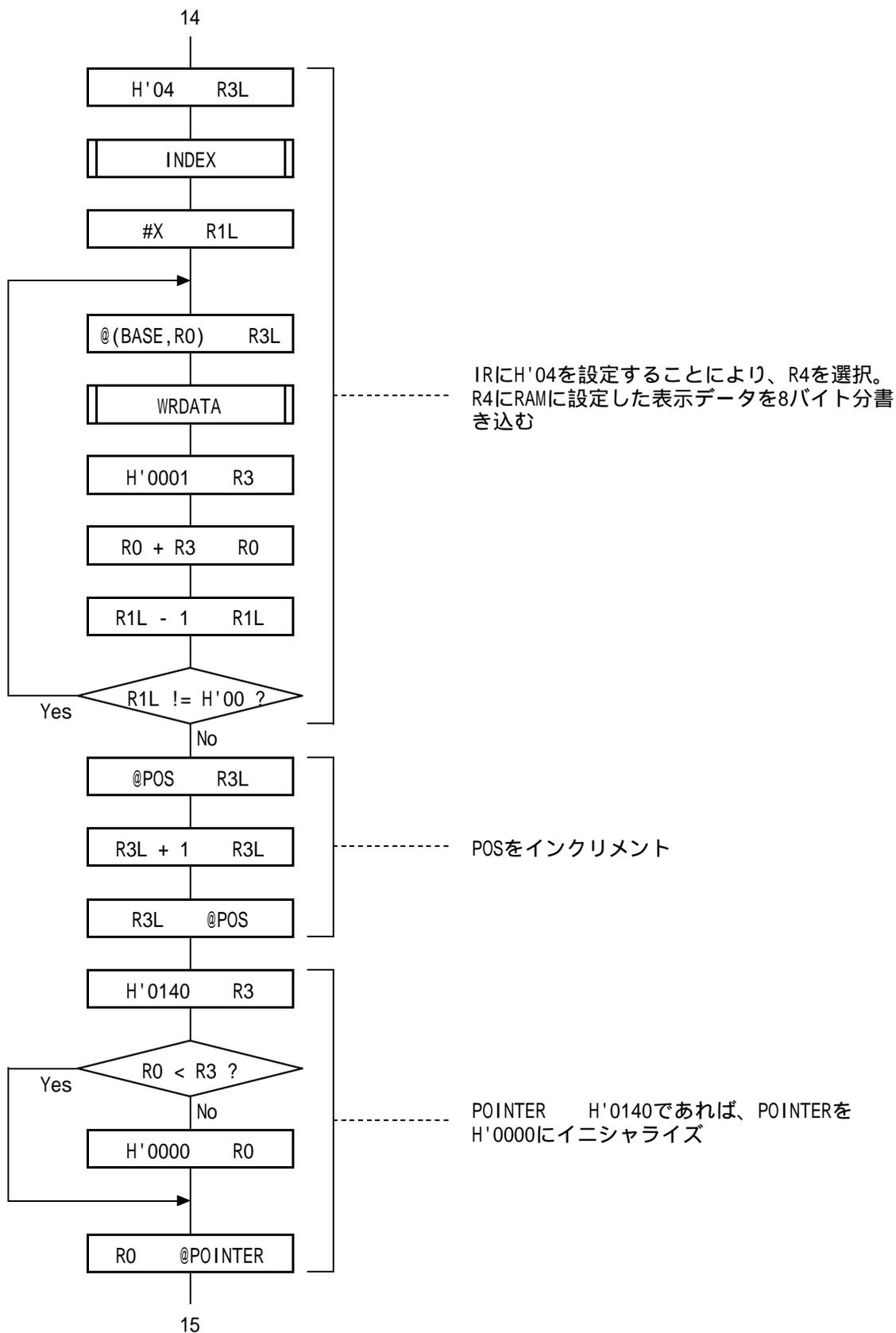
フローチャート



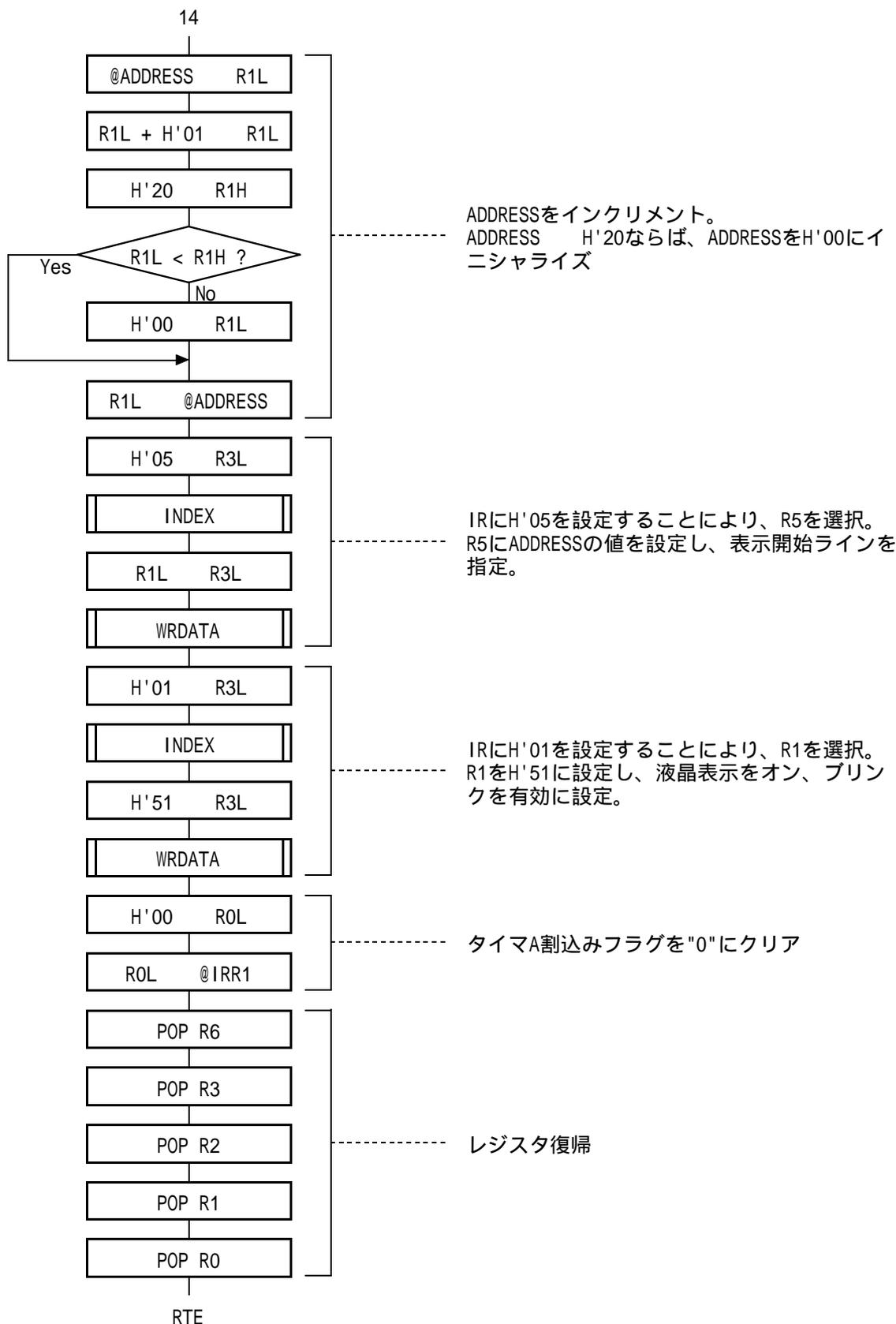
フローチャート



フローチャート

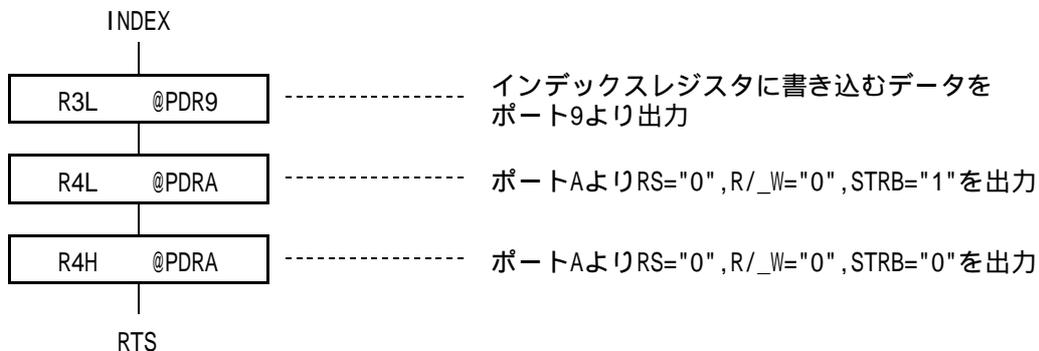


フローチャート

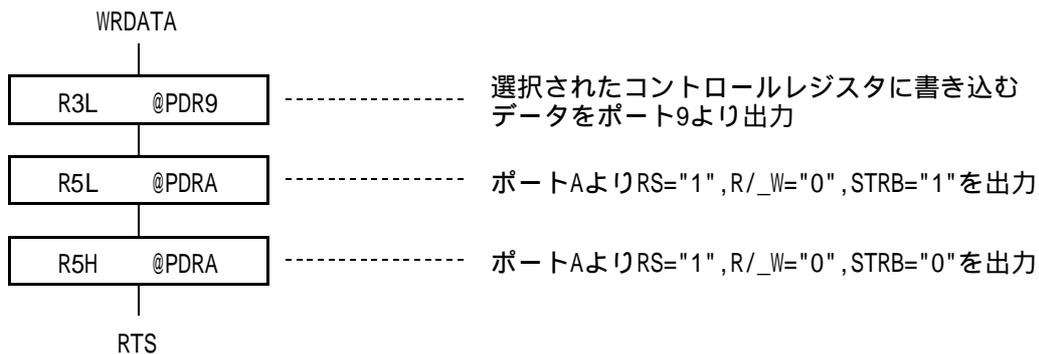


フローチャート

(3) インデックスレジスタ選択ルーチン



(4) LCDコントロールレジスタデータライトルーチン



プログラムリスト

```

;*****
;
; H8/3857F Application Note
;
; "Liquid Crystal Display -Charactor Display Mode-"
;
;           Function : Dot Matrix LCD Controller
;
;           External Clock : 10MHz
;           Internal Clock : 5MHz
;           Sub Clock      : 32.768kHz
;*****
;
;           .CPU          300L
;
;*****
;SYMBOL DEFINITION
;*****
;
TMA:          .EQU          H'FFB0          ;Timer Mode Register A
PDR9:         .EQU          H'FFDC          ;Port Data Register 9
PDRA:         .EQU          H'FFDD          ;Port Data Register A
PCR9:         .EQU          H'FFEC          ;Port Control register 9
PCRA:         .EQU          H'FFED          ;Port Control Register A
SYSCR1:       .EQU          H'FFF0          ;System Control Register 1
SYSCR2:       .EQU          H'FFF1          ;System Control Register 2
IENR1:        .EQU          H'FFF3          ;Interrupt Enable Register 1
IRR1:         .EQU          H'FFF6          ;Interrupt Request Register
;
;*****
;RAM ALLOCATION
;*****
;
BASE          .EQU          H'F800          ; START ADDRESS OF THE DATA TABLE IN THE RAM
X             .EQU          H'8            ; DISPLAY CHARACTER NUMBER PER LINE
Y             .EQU          H'8            ; DISPLAY CHARACTER LENGTH IN VERTICAL DIRECTION
ROW           .EQU          H'5            ; THE LENGTH OF ALL THE DISPLAY DATAS
POINTER       .EQU          H'FA00
COLOR         .EQU          H'FA08
ADDRESS       .EQU          H'FA04          ; ADDRESS OF THE CURRENT WRITEING LCD RAM POSITION
POS           .EQU          H'FA06          ; POINTER OF THE CURRENT WRITEING LCD RAM LINE
;
;*****
;VECTOR ADDRESS
;*****
;
;           .ORG          H'0000
;           .DATA.W      INIT              ;Reset Vector
;
;           .ORG          H'0008
;           .DATA.W      INIT              ;IRQ0 Vector
;           .DATA.W      INIT              ;IRQ1 Vector
;           .DATA.W      INIT              ;IRQ2 Vector
;           .DATA.W      INIT              ;IRQ3 Vector
;           .DATA.W      INIT              ;IRQ4 Vectoe
;           .DATA.W      INIT              ;WKPO Vector
;           .DATA.W      INIT              ;SC11 Vector
;           .DATA.W      TMRA              ;Timer A Vector
;           .DATA.W      INIT              ;Timer B Vector
;           .DATA.W      INIT              ;Timer C Vector
;           .DATA.W      INIT              ;Timer FH Vector

```

プログラムリスト

```

;
;          .DATA.W      INIT          ;Timer FL Vector
;
;          .ORG         H'0024
;          .DATA.W      INIT          ;SCI3 Vector
;          .DATA.W      INIT          ;A/D Vector
;          .DATA.W      INIT          ;DT Vector
;
;*****
;INIT : MAIN ROUTINE
;*****
;
;          .ORG         H'0100
;
;INIT:     .EQU         $
;          LDC          #H'80,CCR      ;INT DISABLE
;          MOV.W        #H'FF80,SP    ;SP INIT
;
;*****
;MOVE THE DISPLAY DATA FROM THE ROM TO THE RAM
; THIS PROGRAM WILL CHANGE THE DISPLAY DATA FORMAT
; DISPLAY DATA STORING IN THE ROM IS CHARACTER BY CHARACTER
; DISPLAY DATA STORING IN THE RAM IS LINE BY LINE
; USING THIS DATA FORMAT WILL MAKE IT EASY TO SCROLL
; THE LCD DISPLAY
;*****
;
;PTOP1:
;          MOV.W        #H'0F80,R0    ;STORE THE ROM DATA IN THE RAM STARTING
;                                     ;FROM #0F80
;
;          MOV.W        #0,R1
;          MOV.W        #0,R4
;          MOV.B        #ROW,R3H
;          MOV.W        #0,R5
;          MOV.B        #X,R2L
;          MOV.B        #Y,R2H      ;THE DISPLAY DATA AREA SIZE IS #X * #Y * ROW
;                                     ;BYTES
;
;JJ1:     MOV.B        @(1cd0,R1),R3L ;THE DISPALY DATA IN THE ROM
;                                     ; STARTING FROM #LCDO
;
;          MOV.B        R3L,@R0
;          MOV.W        #1,R6
;          ADD.W        R6,R5
;          ADD.W        R6,R0
;          MOV.W        #Y,R6
;          ADD.W        R6,R1
;          DEC          R2L
;          BNE          JJ1
;          MOV.B        #8,R2L
;          MOV.W        R4,R1
;          MOV.W        #1,R6
;          ADD.W        R6,R1
;          MOV.W        R1,R4
;          DEC          R2H
;          BNE          JJ1
;          MOV.B        #Y,R2H
;          MOV.W        R5,R1
;          MOV.W        R1,R4
;          DEC          R3H
;          BNE          JJ1
;
;*****
;

```

プログラムリスト

```

;SET THE GENERAL REGISTER
;*****
;
;
PTOP:      MOV.W      #H'OFF00,R0      ;GENERAL
           MOV.W      #H'060A,R1      ;WATCH
           MOV.W      #H'0000,R2      ;MODE TRANSITION
           MOV.W      #H'0000,R3      ;TEMP
           MOV.W      #H'0001,R4      ;LCD I/F INDEX
           MOV.W      #H'0405,R5      ; LCD I/F DATA
           MOV.B      ROH,@PCR9:8
           MOV.B      ROH,@PCRA:8
           MOV.B      ROL,@SYSCR2:8   ;NC ON
;
;*****
;INITIALIZE THE LCD CONTROL REGISTER
;*****
;
;
           MOV.B      #H'00,R3L        ;SELECT CTL1
           JSR        @INDEX
           MOV.B      #H'10,R3L        ;PWR = 1, SOB = 0
           JSR        @WRDATA
;
;
           MOV.B      #H'01,R3L        ;SELECT CTL2
           JSR        @INDEX
           MOV.B      #H'10,R3L        ;OPON = 1,OPS = 0
           JSR        @WRDATA
;
;
           MOV.B      #H'03,R3L        ;SELECT FS
           JSR        @INDEX
           MOV.B      #H'05,R3L        ;FS = 05
           JSR        @WRDATA
;
;
           MOV.B      #H'0A,R3L        ;SELECT CCR
           JSR        @INDEX
           MOV.B      #H'0,R3L         ;CCR = 8
           JSR        @WRDATA
;
;
           MOV.B      #H'06,R3L
           JSR        @INDEX
           MOV.B      #H'OFF,R3L       ;SET THE BLINK COM
           JSR        @WRDATA
;
;
           MOV.B      #H'08,R3L        ;SET THE BLINK START LINE AS 8
           JSR        @INDEX
           MOV.B      #H'08,R3L
           JSR        @WRDATA
;
;
           MOV.B      #H'09,R3L
           JSR        @INDEX           ;SET THE BLINK END LINE AS 16
           MOV.B      #H'0f,R3L
           JSR        @WRDATA
;
;
           MOV.B      #H'01,R3L        ;SELECT CTL2
           JSR        @INDEX
           MOV.B      #H'51,R3L        ; DISP = 1,OPON = 1,OPS = 0
           JSR        @WRDATA
;
;
           MOV.B      #H'04,R3L        ;SELECT RAM
           JSR        @INDEX
           MOV.B      #H'OFF,R3H       ;BYTE NUMBER

```

プログラムリスト

```

LRAMI:      MOV.B      #H'00,R3L      ;WRITE DATA
            JSR        @WRDATA      ;CLEAR ALL THE LCD RAM
            DEC        R3H
            BNE        LRAMI
;
;*****
;WRITE THE DISPLAY DATA INTO THE LCD RAM
; IN THIS PART , THE LCD RAM INCREASEMENT IS Y ADDRESS FIRST
;*****
;
            mov.w      #'0000,r1
            mov.w      #'0000,r2
            mov.w      #'0800,r6
            mov.b      #'02,r3l
            jsr        @INDEX
            mov.b      r1l,r3l
            jsr        @WRDATA
            mov.b      #'4,r3l
            jsr        @INDEX
            mov.b      #'0,r3h
            mov.b      #'00,r1l
            mov.b      #8,r6h
;
gg1:        mov.b      @(lcd0,r2),r3l
            jsr        @WRDATA      ;STORE ONE CHARACTER
            mov.w      #'1,r0
            add.w      r0,r2
            add.b      #'20,r3h
            bcc        gg1
;
gg2:        mov.b      #'0,r3h
            add.b      #'20,r1l      ;STORE 8 CHARACTER
            bcs        gg3
gg23:      mov.b      #'02,r3l
            jsr        @INDEX
            mov.b      r1l,r3l
            jsr        @WRDATA
            mov.b      #'4,r3l
            jsr        @INDEX
            nop
            nop
            jmp        @gg1
;
gg3:        mov.b      #'1f,r3l
            and.b      r3l,r1l
            add.b      #'08,r1l
            add.b      #'1,r6l
            mov.b      #'3,r3l      ;STORE 32 CHARACTER
            cmp.b      r3l,r6l
            bhi        TIMA
            mov.b      #'02,r3l
            jsr        @INDEX
            mov.b      r1l,r3l
            jsr        @WRDATA
            mov.b      #'4,r3l
            jsr        @INDEX

```

プログラムリスト

```

nop
nop
mov.b      #0, r3h
jmp        @gg1
;
;*****
;SET THE VARIABLE
;*****
;
;
TIMA:
MOV.B      #H'0, ROL
MOV.B      ROL, @POS
MOV.B      ROL, @COLOR
MOV.W      #H'100, R0
MOV.W      R0, @POINTER
MOV.B      #H'00, ROL
MOV.B      ROL, @ADDRESS
mov.w      #h'0ff00, r0
MOV.B      #H'08, R3L      ;SET THE TIMERA
MOV.B      R3L, @TMA:8
;
;*****
;SLEEP
;*****
;
SLEEP:
MOV.W      #H'0080, R2
MOV.B      R2L, @SYSCR1:8 ;7:SSBY 3:LSON 3:DTON 2:MSON TMA3
MOV.B      R2H, @SYSCR2:8 ; 1 0 0 0 1
MOV.B      ROL, @IRR1:8
BSET      #H'7, @IENR1:8 ;IENTA=1
LDC       #H'00, CCR
SLEEP                    ;TO WATCH MODE (CKW/2)
NOP
BRA       SLEEP
;
;*****
;TMRA : TIMERA INTERRUPT ROUTINE
;*****
;
TMRA:
PUSH.W     R0
PUSH.W     R1
PUSH.W     R2
PUSH.W     R3
PUSH.W     R6
ORC        #H'80, CCR      ;INT DISABLE
MOV.B      #H'01, R3L
JSR        @INDEX          ;STOP THE LCD DISPLAY
MOV.B      #H'12, R3L      ;CHANGE THE INC MODE TO SET THE X ADDRESS
;FIRST
JSR        @WRDATA
;
;*****
;CHANGE THE COLOR
;*****
;
MOV.B      #H'0A, R3L      ; SELECT CCR
JSR        @INDEX
MOV.B      @COLOR, ROL
INC        ROL

```

プログラムリスト

```

MOV.B      ROL,@COLOR
MOV.B      ROL,R3L      ;CCR = 8
JSR        @WRDATA
;
;*****
; CONTROL THE BLINK AREA POSITION
;*****
;
MOV.B      @POS,R3L
MOV.B      #H'8,R0L
CMP.B      ROL,R3L
BLO        NOPOS
MOV.B      #H'0,R3L
MOV.B      R3L,@POS
NOPOS:
MOV.W      @POINTER,R0
MOV.W      #H'7F,R3      ;(H'7F=127=8*8*2-1)
CMP.W      R3,R0
BHI        NOTHING
MOV.W      #H'3f,R3      ;(H'3F=63=8*8-1)
CMP.W      R3,R0
BHI        CHAN1
MOV.B      @ADDRESS,R0L ;HI TA CHI LINE WRITE IN THE RAM
INC        ROL
MOV.B      @POS,R1L
MOV.B      #H'7,R2L
SUB.B      R1L,R2L
ADD.B      ROL,R2L
dec        r2l
JMP        @SETB
;
CHAN1:
MOV.B      @ADDRESS,R2L ;H8 LINE WRITE IN THE RAM
MOV.B      R2L,R0L
MOV.B      @POS,R1L
SUB.B      R1L,R0L
;
SETB:
MOV.B      #H'8,R3L      ;SET THE BLINK CONTROL REGISTER
JSR        @INDEX
MOV.B      ROL,R3L
JSR        @WRDATA
MOV.B      #H'9,R3L
JSR        @INDEX
MOV.B      R2L,R3L
JSR        @WRDATA
;
NOTHING:
MOV.W      @POINTER,R0 ;POINTER IS THE REPLACE DATA
;FROM 0 TO 320 (8*8*5) (INITIALIZED TO 256)
MOV.B      @ADDRESS,R1L ;ADDRESS IS THE REPLACE LINE
;NUMBER FROM 0 TO 31(0 TO 1F)
MOV.B      #7,R3H
mov.b      #h'02,r3l
jsr        @INDEX      ;SET THE REPLACE DATA POSITION
mov.b      r1l,r3l
jsr        @WRDATA
mov.b      #h'4,r3l
jsr        @INDEX
MOV.B      #X,R1L

```

プログラムリスト

```

;
TMRA2:
    MOV.B    @(BASE,R0),R3L
    JSR     @WRDATA
    MOV.W    #1,R3          ;CHANGE THE DATA IN ONE LINE
    ADD.W    R3,R0
    DEC     R1L
    BNE     TMRA2
    MOV.B    @POS,R3L      ; CHANGE THE LCD BLINK POSITION
    INC     R3L
    MOV.B    R3L,@POS
    MOV.W    #H'140,R3     ;(H'140=320=8*8*5)
    CMP.W    R3,R0
    BLO     TMRA3
    MOV.W    #0,R0
;
TMRA3:
    MOV.W    R0,@POINTER
    MOV.B    @ADDRESS,R1L
    ADD.B    #1,R1L
    MOV.B    #H'20,R1H
    CMP.B    R1H,R1L
    BLO     TMRA1
    MOV.B    #H'0,R1L
TMRA1:
    MOV.B    R1L,@ADDRESS
    MOV.B    #5,R3L
    JSR     @INDEX
    MOV.B    R1L,R3L
    JSR     @WRDATA      ;SET THE SCROLL LINE
;
;*****
;
;
BLEND1:
    MOV.B    #1,R3L
    JSR     @INDEX
    MOV.B    #H'51,R3L    ;START DISPLAY WITH BLINK
    JSR     @WRDATA
    mov.b    #0,r0I      ;CLEAR THE TIMERA INTERRUPT FLAG
    mov.b    r0I,@IRR1
;
BTMRA:
    POP.W    R6
    POP.W    R3
    POP.W    R2
    POP.W    R1
    POP.W    R0
    RTE
;
;*****
;
;INDEX , WRDATA : PROGRAM SUBROUTINE
;*****
;
;
INDEX:
;
    MOV.B    R3L,@PDR9:8
    MOV.B    R4L,@PDRA:8 ;RS = RW = 0 , STRB = 1
    MOV.B    R4H,@PDRA:8 ;RS = RW = 0 , STRB = 0
    RTS
;
;

```

プログラムリスト

```

WRDATA:
;
;          MOV.B          R3L,@PDR9:8
;          MOV.B          R5L,@PDRA:8      ;RS = 1 , RW = 0 , STRB = 1
;          MOV.B          R5H,@PDRA:8      ;RS = 1 , RW = 0 , STRB = 0
;          RTS
;
;*****
;LCD DATA AREA
; DATA FORMAT IS CHARACTER BY CHARACTER
; ONE CHARACTER CONSISTS OF 8 BYTES
;*****
;
;          .ORG          H'1000
lcd0:
;1
;          .DATA.B        H'10          ;- - - 1 0 0 0 0
;          .DATA.B        H'10          ;- - - 1 0 0 0 0
;          .DATA.B        H'1F          ;- - - 1 1 1 1 1
;          .DATA.B        H'10          ;- - - 1 0 0 0 0
;          .DATA.B        H'10          ;- - - 1 0 0 0 0
;          .DATA.B        H'10          ;- - - 1 0 0 0 0
;          .DATA.B        H'0F          ;- - - 0 1 1 1 1
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;2
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'07          ;- - - 0 0 1 1 1
;          .DATA.B        H'04          ;- - - 0 0 1 0 0
;          .DATA.B        H'0A          ;- - - 0 1 0 1 0
;          .DATA.B        H'01          ;- - - 0 0 0 0 1
;          .DATA.B        H'01          ;- - - 0 0 0 0 1
;          .DATA.B        H'06          ;- - - 0 0 1 1 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;3
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'17          ;- - - 1 0 1 1 1
;          .DATA.B        H'11          ;- - - 1 0 0 0 1
;          .DATA.B        H'17          ;- - - 1 0 1 1 1
;          .DATA.B        H'11          ;- - - 1 0 0 0 1
;          .DATA.B        H'01          ;- - - 0 0 0 0 1
;          .DATA.B        H'02          ;- - - 0 0 0 1 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;4
;          .DATA.B        H'10          ;- - - 1 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'18          ;- - - 1 1 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;5
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0
;          .DATA.B        H'00          ;- - - 0 0 0 0 0

```

プログラムリスト

```

;6
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;7
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;8
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0

lcd1:
;1
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'1F      ; - - - 1 1 1 1 1
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;2
    .DATA.B      H'07      ; - - - 0 0 1 1 1
    .DATA.B      H'08      ; - - - 0 1 0 0 0
    .DATA.B      H'08      ; - - - 0 1 0 0 0
    .DATA.B      H'07      ; - - - 0 0 1 1 1
    .DATA.B      H'08      ; - - - 0 1 0 0 0
    .DATA.B      H'08      ; - - - 0 1 0 0 0
    .DATA.B      H'07      ; - - - 0 0 1 1 1
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;3
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'10      ; - - - 1 0 0 0 0
    .DATA.B      H'10      ; - - - 1 0 0 0 0
    .DATA.B      H'01      ; - - - 0 0 0 0 1
    .DATA.B      H'12      ; - - - 1 0 0 1 0
    .DATA.B      H'14      ; - - - 1 0 1 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;4
    .DATA.B      H'07      ; - - - 0 0 1 1 1
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'00      ; - - - 0 0 0 0 0

```

プログラムリスト

```

;5
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'04      ; - - - 0 0 1 0 0
    .DATA.B      H'03      ; - - - 0 0 0 1 1
    .DATA.B      H'00      ; - - - 0 0 0 0 0

    .DATA.B      H'19      ; - - - 1 1 0 0 1
    .DATA.B      H'12      ; - - - 1 0 0 1 0
    .DATA.B      H'02      ; - - - 0 0 0 1 0
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'0A      ; - - - 0 1 0 1 0
    .DATA.B      H'0A      ; - - - 0 1 0 1 0
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;6
    .DATA.B      H'19      ; - - - 1 1 0 0 1
    .DATA.B      H'05      ; - - - 0 0 1 0 1
    .DATA.B      H'05      ; - - - 0 0 1 0 1
    .DATA.B      H'18      ; - - - 1 1 0 0 0
    .DATA.B      H'04      ; - - - 0 0 1 0 0
    .DATA.B      H'05      ; - - - 0 0 1 0 1
    .DATA.B      H'18      ; - - - 1 1 0 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;7
    .DATA.B      H'1E      ; - - - 1 1 1 1 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0
    .DATA.B      H'1C      ; - - - 1 1 1 0 0
    .DATA.B      H'02      ; - - - 0 0 0 1 0
    .DATA.B      H'02      ; - - - 0 0 0 1 0
    .DATA.B      H'02      ; - - - 0 0 0 1 0
    .DATA.B      H'1C      ; - - - 1 1 1 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;8
    .DATA.B      H'1F      ; - - - 1 1 1 1 1
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'01      ; - - - 0 0 0 0 1
    .DATA.B      H'02      ; - - - 0 0 0 1 0
    .DATA.B      H'04      ; - - - 0 0 1 0 0
    .DATA.B      H'04      ; - - - 0 0 1 0 0
    .DATA.B      H'04      ; - - - 0 0 1 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0

lcd2:
;1
    .DATA.B      H'1C      ; - - - 1 1 1 0 0
    .DATA.B      H'12      ; - - - 1 0 0 1 0
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'11      ; - - - 1 0 0 0 1
    .DATA.B      H'12      ; - - - 1 0 0 1 0
    .DATA.B      H'1C      ; - - - 1 1 1 0 0
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;2
    .DATA.B      H'07      ; - - - 0 0 1 1 1
    .DATA.B      H'08      ; - - - 0 1 0 0 0
    .DATA.B      H'08      ; - - - 0 1 0 0 0
    .DATA.B      H'08      ; - - - 0 1 0 0 0
    .DATA.B      H'08      ; - - - 0 1 0 0 0
    .DATA.B      H'08      ; - - - 0 1 0 0 0
    .DATA.B      H'07      ; - - - 0 0 1 1 1
    .DATA.B      H'00      ; - - - 0 0 0 0 0

;3

```

プログラムリスト

```

        .DATA.B      H'07      ; - - 0 0 1 1 1
        .DATA.B      H'11      ; - - 1 0 0 0 1
        .DATA.B      H'11      ; - - 1 0 0 0 1
        .DATA.B      H'11      ; - - 1 0 0 0 1
        .DATA.B      H'11      ; - - 1 0 0 0 1
        .DATA.B      H'11      ; - - 1 0 0 0 1
        .DATA.B      H'01      ; - - 0 0 0 0 1
        .DATA.B      H'00      ; - - 0 0 0 0 0
;4
        .DATA.B      H'18      ; - - 1 1 0 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
;5
        .DATA.B      H'02      ; - - 0 0 0 1 0
        .DATA.B      H'02      ; - - 0 0 0 1 0
        .DATA.B      H'02      ; - - 0 0 0 1 0
        .DATA.B      H'02      ; - - 0 0 0 1 0
        .DATA.B      H'02      ; - - 0 0 0 1 0
        .DATA.B      H'02      ; - - 0 0 0 1 0
        .DATA.B      H'03      ; - - 0 0 0 1 1
        .DATA.B      H'00      ; - - 0 0 0 0 0
;6
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'01      ; - - 0 0 0 0 1
        .DATA.B      H'01      ; - - 0 0 0 0 1
        .DATA.B      H'01      ; - - 0 0 0 0 1
        .DATA.B      H'01      ; - - 0 0 0 0 1
        .DATA.B      H'01      ; - - 0 0 0 0 1
        .DATA.B      H'1C      ; - - 1 1 1 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
;7
        .DATA.B      H'1C      ; - - 1 1 1 0 0
        .DATA.B      H'02      ; - - 0 0 0 1 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'02      ; - - 0 0 0 1 0
        .DATA.B      H'1C      ; - - 1 1 1 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
;8
        .DATA.B      H'1C      ; - - 1 1 1 0 0
        .DATA.B      H'12      ; - - 1 0 0 1 0
        .DATA.B      H'11      ; - - 1 0 0 0 1
        .DATA.B      H'11      ; - - 1 0 0 0 1
        .DATA.B      H'11      ; - - 1 0 0 0 1
        .DATA.B      H'12      ; - - 1 0 0 1 0
        .DATA.B      H'1C      ; - - 1 1 1 0 0
        .DATA.B      H'00      ; - - 0 0 0 0 0
lcd3:
;1
        .DATA.B      H'00      ; - - 0 0 0 0 0
        .DATA.B      H'1F      ; - - 1 1 1 1 1
        .DATA.B      H'01      ; - - 0 0 0 0 1
        .DATA.B      H'01      ; - - 0 0 0 0 1
        .DATA.B      H'01      ; - - 0 0 0 0 1
    
```

プログラムリスト

```

;2
.DATA.B H'01 ; - - 0 0 0 0 1
.DATA.B H'1F ; - - 1 1 1 1 1
.DATA.B H'00 ; - - 0 0 0 0 0

.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'0C ; - - 0 1 1 0 0
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'01 ; - - 0 0 0 0 1
.DATA.B H'0E ; - - 0 1 1 1 0
.DATA.B H'00 ; - - 0 0 0 0 0

;3
.DATA.B H'02 ; - - 0 0 0 1 0
.DATA.B H'02 ; - - 0 0 0 1 0
.DATA.B H'02 ; - - 0 0 0 1 0
.DATA.B H'13 ; - - 1 0 0 1 1
.DATA.B H'12 ; - - 1 0 0 1 0
.DATA.B H'02 ; - - 0 0 0 1 0
.DATA.B H'02 ; - - 0 0 0 1 0
.DATA.B H'00 ; - - 0 0 0 0 0

;4
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'07 ; - - 0 0 1 1 1
.DATA.B H'04 ; - - 0 0 1 0 0
.DATA.B H'04 ; - - 0 0 1 0 0
.DATA.B H'14 ; - - 1 0 1 0 0
.DATA.B H'04 ; - - 0 0 1 0 0
.DATA.B H'07 ; - - 0 0 1 1 1
.DATA.B H'00 ; - - 0 0 0 0 0

;5
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'18 ; - - 1 1 0 0 0
.DATA.B H'08 ; - - 0 1 0 0 0
.DATA.B H'0B ; - - 0 1 0 1 1
.DATA.B H'08 ; - - 0 1 0 0 0
.DATA.B H'08 ; - - 0 1 0 0 0
.DATA.B H'18 ; - - 1 1 0 0 0
.DATA.B H'00 ; - - 0 0 0 0 0

;6
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'01 ; - - 0 0 0 0 1
.DATA.B H'1C ; - - 1 1 1 0 0
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'00 ; - - 0 0 0 0 0

;7
.DATA.B H'1C ; - - 1 1 1 0 0
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'1E ; - - 1 1 1 1 0
.DATA.B H'02 ; - - 0 0 0 1 0
.DATA.B H'02 ; - - 0 0 0 1 0
.DATA.B H'04 ; - - 0 0 1 0 0
.DATA.B H'08 ; - - 0 1 0 0 0
.DATA.B H'00 ; - - 0 0 0 0 0

;8
.DATA.B H'00 ; - - 0 0 0 0 0
.DATA.B H'01 ; - - 0 0 0 0 1

```

プログラムリスト

	.DATA.B	H'02	;- - - 0 0 0 1 0
	.DATA.B	H'04	;- - - 0 0 1 0 0
	.DATA.B	H'08	;- - - 0 1 0 0 0
	.DATA.B	H'10	;- - - 1 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
lcd4:			
;1	.DATA.B	H'0B	;- - - 0 1 0 1 1
	.DATA.B	H'08	;- - - 0 1 0 0 0
	.DATA.B	H'08	;- - - 0 1 0 0 0
	.DATA.B	H'0C	;- - - 0 1 1 0 0
	.DATA.B	H'0A	;- - - 0 1 0 1 0
	.DATA.B	H'08	;- - - 0 1 0 0 0
	.DATA.B	H'08	;- - - 0 1 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
;2	.DATA.B	H'07	;- - - 0 0 1 1 1
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'0F	;- - - 0 1 1 1 1
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'01	;- - - 0 0 0 0 1
	.DATA.B	H'02	;- - - 0 0 0 1 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
;3	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'11	;- - - 1 0 0 0 1
	.DATA.B	H'13	;- - - 1 0 0 1 1
	.DATA.B	H'15	;- - - 1 0 1 0 1
	.DATA.B	H'01	;- - - 0 0 0 0 1
	.DATA.B	H'01	;- - - 0 0 0 0 1
	.DATA.B	H'00	;- - - 0 0 0 0 0
;4	.DATA.B	H'08	;- - - 0 1 0 0 0
	.DATA.B	H'10	;- - - 1 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'02	;- - - 0 0 0 1 0
	.DATA.B	H'02	;- - - 0 0 0 1 0
	.DATA.B	H'02	;- - - 0 0 0 1 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
;5	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'0C	;- - - 0 1 1 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'10	;- - - 1 0 0 0 0
	.DATA.B	H'08	;- - - 0 1 0 0 0
	.DATA.B	H'04	;- - - 0 0 1 0 0
	.DATA.B	H'04	;- - - 0 0 1 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
;6	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0
	.DATA.B	H'00	;- - - 0 0 0 0 0

プログラムリスト

```

;7      .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
;8      .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
        .DATA.B      H'00      ;- - - 0 0 0 0 0
;
        .END

```

2.2 ドットマトリックスLCD制御-グラフィック表示モード

ドットマトリックスLCD制御 -グラフィック表示モード	MCU	H8/3857シリーズ	使用機能	ドットマトリックス LCDコントローラ
--------------------------------	-----	-------------	------	------------------------

仕様

- (1) ドットマトリックスLCDコントローラ機能を使用して、液晶表示を行いません。
- (2) 本タスク例では40×32のドットマトリックスLCDを使用します。
- (3) 表示モードはグラフィック表示モードで、1/5パイアス、1/32デューティによる表示を行います。
- (4) スクロール、およびプリンキングの制御を行います。
- (5) 本タスク例における液晶表示例を図1に示します。

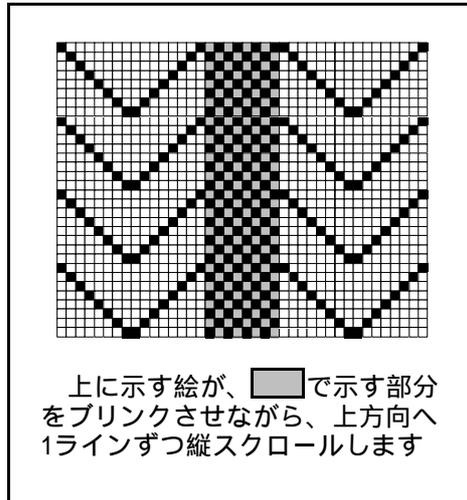


図1 本タスク例におけるLCD表示例

使用機能説明

- (1) ドットマトリックスLCDコントローラの使用機能でキャラクタ表示モードと共通の機能については「2.1 ドットマトリックスLCD制御 - キャラクタ表示モード」で説明しているため、本項目では省略致します。

(2) 液晶駆動端子機能 (グラフィック表示モード)

LCDコントローラは、液晶駆動出力のうち、COM9～COM32とSEG64～SEG41が表示デューティと表示モードにより切り替わります。

グラフィック表示モード (SOB="1") の場合のコモン/セグメント出力切り替えについて表1に示します。

表1 グラフィック表示モードにおけるコモン/セグメント出力切り替え

端子名	SOB="1" (グラフィック表示モード)		
	1/8 Duty	1/16 Duty	1/32 Duty
COM1～COM8	COM1～COM8	COM1～COM16	COM1～COM16
COM9/SEG64～COM16/SEG57	SEG64～SEG57		
SEG1～SEG40	SEG1～SEG56	SEG1～SEG56	SEG1～SEG40
COM32/SEG41～COM25/SEG48			COM32～COM17
COM24/SEG49～COM17/SEG56			

使用機能説明

(3) 表示メモリ構成と表示 (グラフィック表示モード)

LCDコントローラは、容量64×32ビットのビットマップ方式の表示メモリを内蔵しています。表示メモリの構成はX方向が5ビット×8または8ビット×n (n=5,7,8) ビットの組み合わせが選択可能で、Y方向が32ビットになっています。CPUからライトする表示データは、図2に示すようにMSBが左、LSBが右に横に格納されます。表示データと表示の関係は、データ"1"が点灯(黒)、データ"0"が非点灯(無色)に対応しています。

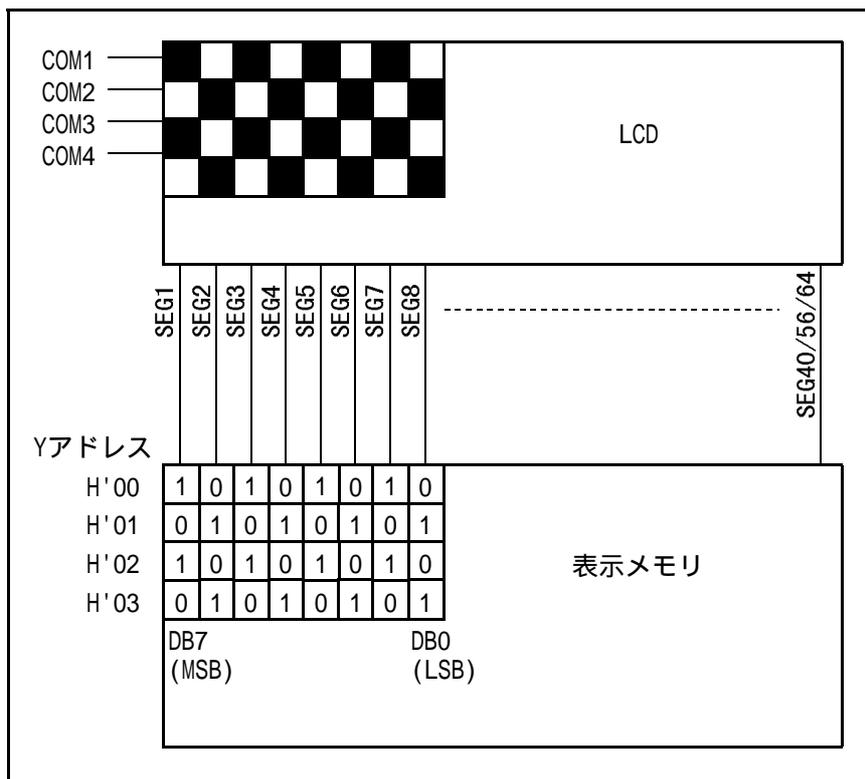


図2 グラフィック表示モードにおけるメモリデータと表示

使用機能説明

(4) 表示データ出力 (グラフィック表示モード)

LCDコントローラは、5ドット×8ドットのキャラクタ出力を効率的に行なうために表示データの1バイトのうち5ビットのみ出力可能なキャラクタ表示モード (SOB="0") と、フルドットのグラフィック表示を効率的に行なうために、1バイトの全データを出力可能なグラフィック表示モード (SOB="1") があります。

グラフィック表示モード (SOB="1") における表示デューティと出力端子の関係を図3に示します。

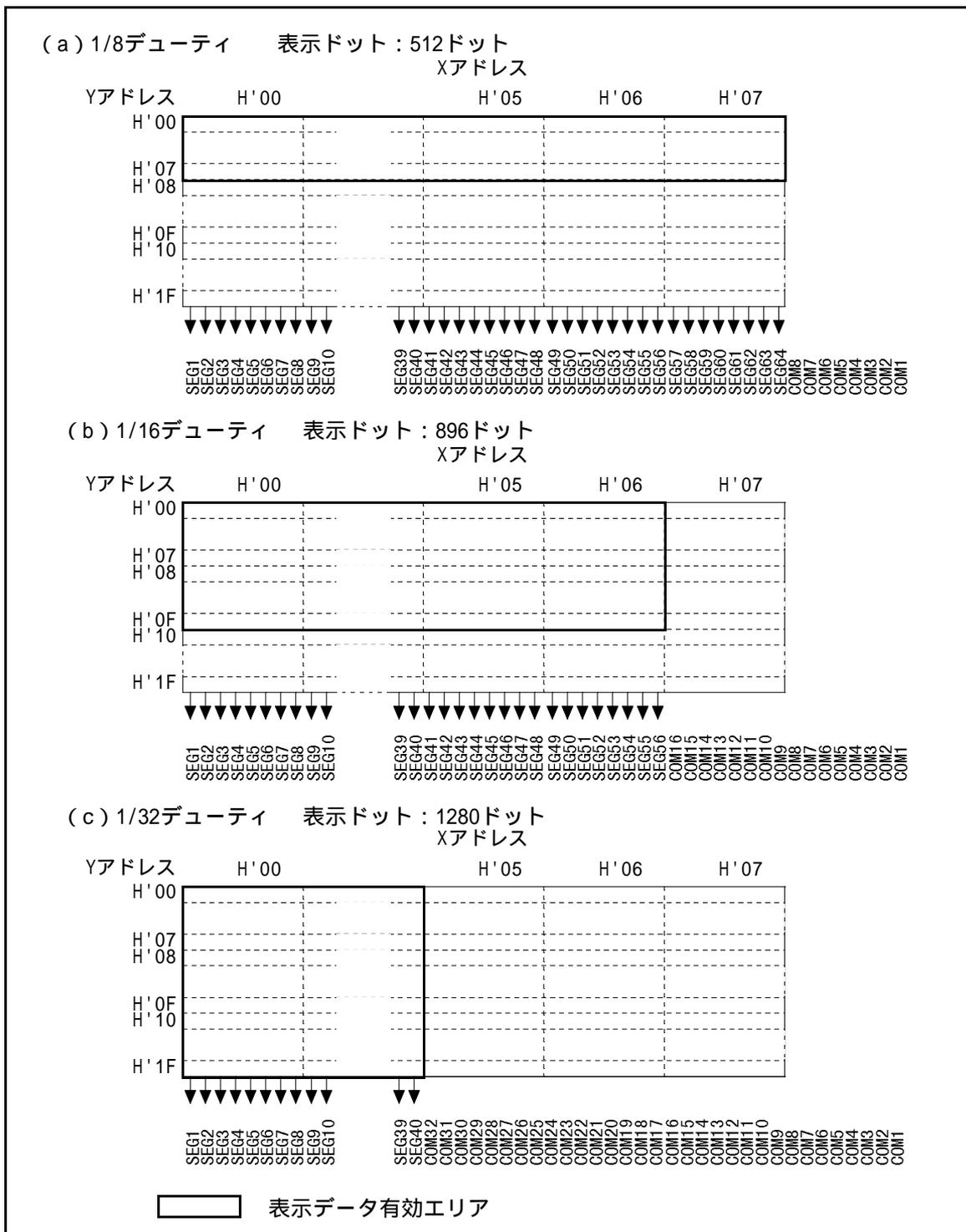


図3 グラフィック表示モードにおける表示デューティと表示有功エリア

使用機能説明

(5) 表2に本タスク例における機能割付けを示します。

表3 機能割付け

機能	機能割付け
IR	LCDコントローラの10本の制御レジスタのうち1つを選択します。
R0	LCDモジュールスタンバイモードの設定、昇圧回路の制御、キャラクタ/グラフィック表示の切り換え、駆動デューティの選択を行ないます。
R1	液晶表示の動作/停止の設定、オペアンプ回路のパワーセーブの設定、オペアンプ回路の動作/停止の設定、リード・モディファイ・ライトモードの設定、表示メモリのインクリメントするアドレスの設定を行ないます。
R2	CPUからアクセスする表示メモリのX、Y方向それぞれのアドレスの設定を行ないます。
R3	フレーム周波数の設定を行ないます。
R4	R2で指定されたアドレスの表示メモリにリード/ライトを行ないます。
R5	表示を開始するラインを設定します。
R6	ブリンクさせたい領域を指定します。
R8	ブリンクさせる領域の始まりのラインを設定します。
R9	ブリンクさせる領域の終わりのラインを設定します。
RA	輝度調整抵抗値を設定します。
COM1 ~ COM32	液晶のコモン駆動用の出力端子です。
SEG1 ~ SEG40	液晶のセグメント駆動用の出力端子です。
V3、V4	LCDのバイアスを設定します。
V34	内部抵抗テスト用の端子で、V3とショートさせます。
C1+, C1-, C2+, C2-	LCD昇圧用の外部容量接続端子です。
V10OUT ~ V50OUT	LCD駆動電源レベルの入力端子です。
V _{ci}	LCDの昇圧回路への基準入力電圧、兼昇圧回路電源です。
V _L OUT	LCDの昇圧電圧の出力端子です。
V _L CD	LCDの駆動電源入力端子です。
ポート9	LSIに内蔵したLCDコントローラとインタフェースする8ビットの入出力ポートです。
ポートA	LSIに内蔵したLCDコントローラとインタフェースする4ビットの入出力ポートです。

動作説明

(1) 本タスク例の動作原理について説明します。

(a) 表示メモリと表示データの設定について

本タスク例における表示メモリと表示データについて表3に示します。表3に示すように、アドレスレジスタ (R2) にX方向、Y方向のアドレスを指定し、表示データレジスタ (R4) に表示データを書込むことによりLCD表示を行ないます。

表3 表示データと表示メモリ

	X=H'00	X=H'01	X=H'02	X=H'03	X=H'04
Y=H'00	10000000	00000001	01010101	10000000	00000001
Y=H'01	01000000	00000010	10101010	01000000	00000010
Y=H'02	00100000	00000100	01010101	00100000	00000100
Y=H'03	00010000	00001000	10101010	00010000	00001000
Y=H'04	00001000	00010000	01010101	00001000	00010000
Y=H'05	00000100	00100000	10101010	00000100	00100000
Y=H'06	00000010	01000000	01010101	00000010	01000000
Y=H'07	00000001	10000000	10101010	00000001	10000000
Y=H'08	10000000	00000001	01010101	10000000	00000001
Y=H'09	01000000	00000010	10101010	01000000	00000010
Y=H'0A	00100000	00000010	01010101	00100000	00000010
Y=H'0B	00010000	00000100	10101010	00010000	00000100
Y=H'0C	00001000	00001000	01010101	00001000	00001000
Y=H'0D	00000100	00010000	10101010	00000100	00010000
Y=H'0E	00000010	01000000	01010101	00000010	01000000
Y=H'0F	00000001	10000000	10101010	00000001	10000000
Y=H'10	10000000	00000001	01010101	10000000	00000001
Y=H'11	01000000	00000010	10101010	01000000	00000010
Y=H'12	00100000	00000100	01010101	00100000	00000100
Y=H'13	00010000	00001000	10101010	00010000	00001000
Y=H'14	00001000	00010000	01010101	00001000	00010000
Y=H'15	00000100	00100000	10101010	00000100	00100000
Y=H'16	00000010	01000000	01010101	00000010	01000000
Y=H'17	00000001	10000000	10101010	00000001	10000000
Y=H'18	10000000	00000001	01010101	10000000	00000001
Y=H'19	01000000	00000010	10101010	01000000	00000010
Y=H'1A	00100000	00000010	01010101	00100000	00000010
Y=H'1B	00010000	00000100	10101010	00010000	00000100
Y=H'1C	00001000	00001000	01010101	00001000	00001000
Y=H'1D	00000100	00010000	10101010	00000100	00010000
Y=H'1E	00000010	01000000	01010101	00000010	01000000
Y=H'1F	00000001	10000000	10101010	00000001	10000000

動作説明

(b)動作原理

図4に本タスク例における動作原理を示します。図4に示すようなソフトウェア処理、およびハードウェア処理によりグラフィック表示モードによるLCD表示を行ないます。

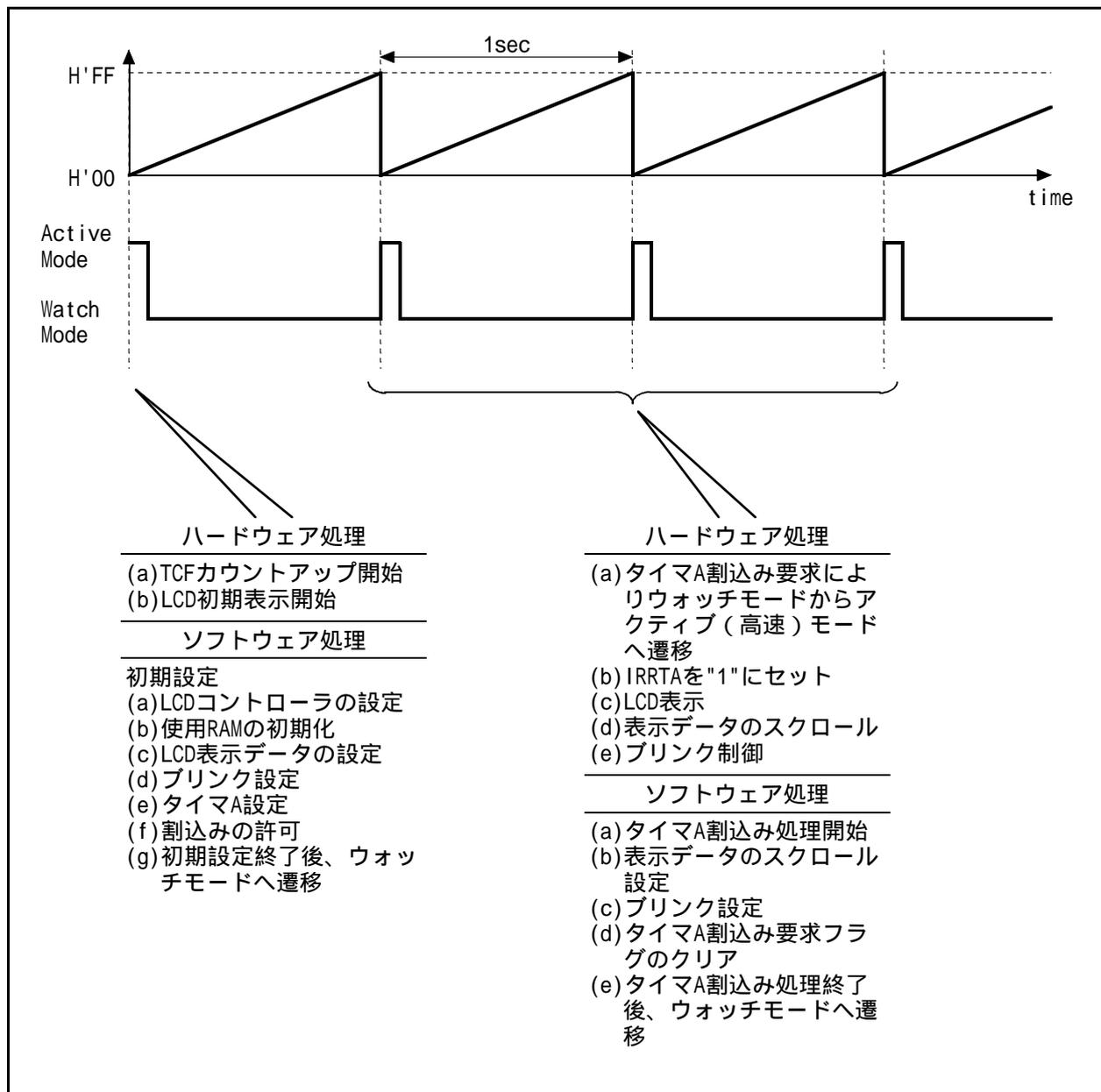


図4 動作原理

ソフトウェア説明

(1) モジュール説明

表4に本タスク例におけるモジュール説明を示します。

表4 モジュール説明

モジュール名	ラベル名	機能
メインルーチン	INIT	スタックポインタ、使用RAM、ドットマトリックスLCDコントローラ、タイマAのイニシャライズ、表示データの表示メモリへの書き込み、割り込み許可、ウォッチモードへの遷移を行う。
インデックスレジスタ選択ルーチン	INDEX	インデックスレジスタ (IR) の選択を行う。
LCDコントロールレジスタライトルーチン	WRDATA	IRにより選択した制御レジスタにデータを書き込む。
タイマA割り込み処理ルーチン	TMRA	表示データのスクロール、およびブリンクの制御を行う。

(2) 引数の説明

本タスク例では、引数は使用していません。

(3) 使用内部レジスタ説明

表5に本タスク例における使用内部レジスタ説明を示します。

表5 使用内部レジスタ説明

レジスタ名	説明	IR 設定値	設定値
IR IR3~ IRO	<p>インデックスレジスタ (インデックスレジスタ3~0) ~LCDコントローラの10本の制御レジスタのうち1つを選択するビットです。</p> <p>: IR3="0"、IR2="0"、IR1="0"、IRO="0"のとき、R0を選択 : IR3="0"、IR2="0"、IR1="0"、IRO="1"のとき、R1を選択 : IR3="0"、IR2="0"、IR1="1"、IRO="0"のとき、R2を選択 : IR3="0"、IR2="0"、IR1="1"、IRO="1"のとき、R3を選択 : IR3="0"、IR2="1"、IR1="0"、IRO="0"のとき、R4を選択 : IR3="0"、IR2="1"、IR1="0"、IRO="1"のとき、R5を選択 : IR3="0"、IR2="1"、IR1="1"、IRO="0"のとき、R6を選択 : IR3="1"、IR2="0"、IR1="0"、IRO="0"のとき、R8を選択 : IR3="1"、IR2="0"、IR1="0"、IRO="1"のとき、R9を選択 : IR3="1"、IR2="0"、IR1="1"、IRO="0"のとき、RAを選択</p> <p>【注】1. レジスタへアクセスするには、RSを"0"にしてIRにアクセスしたいレジスタのレジスタ番号を設定します。その後、RSを"1"にして指定したレジスタにアクセスすることができます。 2. 一部の内部レジスタには存在しないビットがありますが、これらのビットには"0"を書込むようにしてください。</p>	- bit 3~ bit 0	-
RO LSBY	<p>コントロールレジスタ (モジュールスタンバイ) ~LSBYはモジュールスタンバイ設定ビットです。LSBYを"1"に設定するとLCDコントローラはスタンバイモードになります。その時、PSRビットの状態には影響を与えませんが、DISPビットおよびOPONビットはリセットされます。</p> <p>: LSBY="0"のとき、LCDは通常動作します。 : LSBY="1"のとき、昇圧、内部動作を停止し、表示をオフし、LCDはスタンバイモードとなります。</p>	IR3=0 IR2=0 IR1=0 IRO=0 bit 5	"0"

ソフトウェア説明

表5 使用内部レジスタ説明(つづき)

レジスタ名	説明	IR 設定値	設定値
R0	PWR (昇圧回路動作設定) ~ PWRは昇圧回路の動作/停止を設定するビットです。 : PWR="0"のとき、昇圧回路は停止します。 : PWR="1"のとき、昇圧回路は動作します。	bit 4	"1"
	SOB (表示モード選択) ~ SOBは表示モードをキャラクタ表示モードか、グラフィック表示モードか選択するビットです。 : SOB="0"のとき、キャラクタ表示モードに設定します。表示メモリの1バイトのうち、ビット4~0をセグメント端子に出力します。 : SOB="1"のとき、グラフィック表示モードに設定します。表J9位メモリの1バイトのデータのすべてをセグメント端子に出力します。出力できるXアドレスは、1/32デューティの場合H'0~H'4、1/16デューティの場合H'0~H'6、1/8デューティの場合H'0~H'7の範囲です。	bit 2	"1"
	DDTY1, DDTY0 (表示デューティ選択) ~ DDTY1、DDTY0は表示デューティを1/32、1/16、または1/8から選択するビットです。 : DDTY1="0"、DDTY0="0"のとき、1/32デューティを選択します。 : DDTY1="0"、DDTY0="1"のとき、1/16デューティを選択します。YアドレスのH'10~H'1Fの表示データは無効です。 : DDTY1="1"、DDTY0="*"のとき、1/8デューティを選択します。YアドレスのH'08~H'1Fの表示データは無効です。	bit1, bit0	DDTY1 =0 DDTY0 =0
R1	DISP コントロールレジスタ2(液晶表示動作設定) ~ DISPは液晶表示の動作/停止を設定するビットです。R0のLSBYを"1"に設定するとDISPはクリアされます。 : DISP="0"のとき、液晶表示をオフします。全液晶表示出力はVssレベルとなります。 : DISP="1"のとき、液晶表示をオンします。	IR3=0 IR2=0 IR1=0 IR0=1 bit 6	"1"
	OPS (オペアンプ回路のパワーセーブ設定) ~ OPSはオペアンプ回路のパワーセーブ機能の有効/向こうを設定するビットです。OPONが"1"で内蔵のオペアンプ回路が動作しているとき、OPSを"1"に設定するとパワーセーブ機能が有効になります。液晶表示出力が安定しないばあいはOPSを"0"に設定してください。 : OPS="0"のとき、パワーセーブ機能は無効です。 : OPS="1"のとき、パワーセーブ機能が有効になり、ボルテージフォロア型オペアンプの駆動電流を低減します。この場合、V1OUT~V5OUT端子とVss間に0.1~0.5μFのコンデンサを挿入してください。	bit 5	"0"
	OPON (オペアンプ回路動作設定) ~ OPONはオペアンプ回路を動作/停止を設定するビットです。OPONを"1"に設定するとオペアンプ回路が動作します。外部からV1OUT~V5OUTにLCD駆動電源レベルを印加する場合は必ずOPONを"0"に設定してください。 : OPON="0"のとき、内蔵オペアンプは停止し、出力はハイインピーダンスとなります。外部から液晶駆動電圧を入力できます。 : OPON="1"のとき、内蔵オペアンプは動作します。	bit 4	"1"
	RMW (リード・モディファイ・ライトモード設定) ~ RMWは表示メモリのXまたはYアドレスのインクリメントをライト/リードのアクセス後に行なうか、ライト後のみ(リード・モディファイ・ライトモード)で行なうかを選択するビットです。 : RMW="0"のとき、表示メモリへのライト/リードアクセス後にアドレスをインクリメントします。 : RMW="1"のとき、リード・モディファイ・ライトモードに設定します。このモードでは表示メモリへのライトアクセス後にのみアドレスをインクリメントします。	bit 3	"0"

ソフトウェア説明

表5 使用内部レジスタ説明(つづき)

レジスタ名	説明	IR 設定値	設定値
R1	INC (インクリメントアドレス選択) ~ INCは表示データレジスタ(R4)へのRMWで設定したアクセス後にインクリメントするアドレスをXアドレスかYアドレスか選択するビットです。選択されたアドレスは表示データの有効エリアの最大値で表示データレジスタ(R4)へのアクセス後にクリアされ、そのとき、他方のアドレスをインクリメントします。 : INC="0"のとき、表示メモリのYアドレスのインクリメントが優先され、Yアドレスのオーバフロー後にXアドレスをインクリメントします。 : INC="1"のとき、邦字メモリのXアドレスのインクリメントが優先され、Xアドレスのオーバフロー後にYアドレスをインクリメントします。	bit 1	"0"
	BLK (ブリンク動作設定) ~ BLKはブリンク機能の有効/無効を設定するビットです。DISPが"1"で液晶表示しているとき、BLKを"1"に設定するとブリンク機能が有効になり、R6のBK7~BK0、R8のBSL4~BSL0、R9のBEL4~BELOで設定した範囲をブリンク表示します。 : BLK="0"のとき、ブリンクは無効です。 : BLK="1"のとき、ブリンクが有効となります。	bit 0	"1"
R2	XA2~XA0 (アドレスレジスタ(Xアドレス設定)) ~ XA2~XA0は表示メモリのX方向のアドレスを設定するビットです。H'0~H'7の範囲が設定可能ですが、SOB="1"の場合は1/16デューティ時H'7、1/32デューティ時H'5~H'7の範囲の表示データは無効となります。INCが"1"のとき、RMWで設定したアクセス後に自動的にインクリメントされ、表示データの有効エリアの最大値のアクセス後にクリアされます。また、INCが"0"でYA4~YA0が表示データの有効のエリアの最大値のとき、RMWで設定したアクセス後にインクリメントされます。	IR3=0 IR2=0 IR1=1 IR0=0 bit 7~ bit 5	XA2=0 XA1=0 XA0=0
	YA4~YA0 (Yアドレス設定) ~ YA4~YA0は表示メモリのY方向のアドレスを設定するビットです。H'00~H'1Fの範囲が設定可能ですが、1/16デューティの場合はH'10~H'1Fの範囲、また1/8デューティの場合はH'08~H'1Fの範囲の表示データは向こうとなります。INCが"0"のとき、RMWで設定したアクセス後に自動的にインクリメントされ、表示データの有効エリアの最大値のアクセス後にクリアされます。また、INCが"1"でXA2~XA0が表示データの有効のエリアの最大値のとき、RMWで設定したアクセス後にインクリメントされます。	bit 4~ bit 0	YA4=0 YA3=0 YA2=0 YA1=0 YA0=0
R3	FS5~FS0 (フレーム周波数設定レジスタ(フレーム周波数設定5~0)) ~ FS5~FS0は使用する液晶パネルの特性に合わせて最適な分周比を設定します。レジスタ設定値と分周比の関係を表8に示します。 : FS5="0"、FS4="0"、FS3="0"、FS2="1"、FS1="0"、FS0="1"のとき、サブクロックの分周比を12に設定します。これにより、フレーム周波数は85.3Hzに設定されます。	IR3=0 IR2=0 IR1=1 IR0=1 bit 5~ bit 0	FS5=0 FS4=0 FS3=0 FS2=1 FS1=0 FS0=1
R4	表示データレジスタ ~ R2のXA2~XA0とYA4~YA0で指定される表示メモリにリード/ライトを行いません。	IR3=0 IR2=1 IR1=0 IR0=0	-

ソフトウェア説明

表5 使用内部レジスタ説明(つづき)

レジスタ名		説明	IR 設定値	設定値
R5	ST4 ~ ST0	表示開始ラインレジスタ(表示開始ライン設定4~0) ~ST4~ST0は表示を開始するラインを指定するビットです。設定値は、「表示開始ライン - 1」を指定してください。本レジスタの設定値を換えることにより、縦方向のスクロールが実現できます。設定範囲は、1/32デューティの場合0~31、1/16デューティの場合0~15、1/8デューティの場合0~7です。これらの制限値を超えた値を設定しますと、正常な表示は行なえません。	IR3=0 IR2=1 IR1=0 IR0=1 bit 4~ bit 0	ST4=0 ST3=0 ST2=0 ST1=0 ST0=0
R6		ブリンクレジスタ ~R6はブリンクさせたい領域に対応するビットに"1"を書込みます。同時にブリンクさせる範囲には制限がなく、全ビットに"1"を書込むことによって全画面をブリンクさせることも可能です。このレジスタの設定値は、BLKが"1"に設定されているときにのみ有効です。各ビットに対応するブリンクの範囲は、SOBの値により次のようになります。 : SOB="0"のとき、BK7 : SEG36 ~ SEG40, BK6 : SEG31 ~ SEG35 BK5 : SEG26 ~ SEG30, BK4 : SEG21 ~ SEG25 BK3 : SEG16 ~ SEG20, BK2 : SEG11 ~ SEG15 BK1 : SEG6 ~ SEG10, BK0 : SEG1 ~ SEG5 : SOB="1"のとき、BK7 : SEG57 ~ SEG64, BK6 : SEG49 ~ SEG56 BK5 : SEG41 ~ SEG48, BK4 : SEG33 ~ SEG40 BK3 : SEG25 ~ SEG32, BK2 : SEG17 ~ SEG24 BK1 : SEG9 ~ SEG16, BK0 : SEG1 ~ SEG8	IR3=0 IR2=1 IR1=1 IR0=0	H'FF
R8	BSL4 ~ BSL0	ブリンク開始ラインレジスタ(ブリンク開始ライン設定4~0) ~BSL4~BSL0はブリンクさせる領域の先頭のラインを指定するビットです。設定値は、「ブリンク開始ライン - 1」をセットしてください。設定範囲は、1/32デューティの場合0~31、1/16デューティの場合0~15、1/8デューティの場合0~7です。これらの制限値を超えた値を設定しますと、正常な動作は保証されません。	IR3=0 IR2=1 IR1=1 IR0=0 bit 4~ bit 0	BSL4=0 BSL3=0 BSL2=0 BSL1=0 BSL0=0
R9	BEL4 ~ BELO	ブリンク終了ラインレジスタ(ブリンク終了ライン設定4~0) ~BEL4~BELOはブリンクさせる領域の終わりのラインを指定するビットです。設定値は、「ブリンク終了ライン - 1」をセットしてください。設定範囲は、1/32デューティの場合0~31、1/16デューティの場合0~15、1/8デューティの場合0~7です。これらの制限値を超えた値を設定しますと、正常な動作は保証されません。	IR3=0 IR2=1 IR1=1 IR0=0 bit 4~ bit 0	BEL4=1 BEL3=1 BEL2=1 BEL1=1 BELO=1

ソフトウェア説明

表5 使用内部レジスタ説明(つづき)

レジスタ名	説明	IR/ アドレス	設定値
RA CCR3~ CCRO	<p>コントラストコントロールレジスタ(輝度調整設定3~0) ~ CCR3~ BEL0はV_{LCD}とV₁レベルの間の輝度調整抵抗値を設定するビットです。V_{LCD}とV₁レベルの間の輝度調整抵抗と調節することにより、液晶パネルの輝度調整が可能です。輝度調整抵抗は、液晶ブリーダ抵抗Rに対し0.1R~1.6Rの範囲で設定可能です。</p> <p>: CCR3="0", CCR2="0", CCR1="0", CCR0="0"のとき、輝度調整抵抗は1.6R : CCR3="0", CCR2="0", CCR1="0", CCR0="1"のとき、輝度調整抵抗は1.5R : CCR3="0", CCR2="0", CCR1="1", CCR0="0"のとき、輝度調整抵抗は1.4R : CCR3="0", CCR2="0", CCR1="1", CCR0="1"のとき、輝度調整抵抗は1.3R : CCR3="0", CCR2="1", CCR1="0", CCR0="0"のとき、輝度調整抵抗は1.2R : CCR3="0", CCR2="1", CCR1="0", CCR0="1"のとき、輝度調整抵抗は1.1R : CCR3="0", CCR2="1", CCR1="1", CCR0="0"のとき、輝度調整抵抗は1.0R : CCR3="0", CCR2="1", CCR1="1", CCR0="1"のとき、輝度調整抵抗は0.9R : CCR3="1", CCR2="0", CCR1="0", CCR0="0"のとき、輝度調整抵抗は0.8R : CCR3="1", CCR2="0", CCR1="0", CCR0="1"のとき、輝度調整抵抗は0.7R : CCR3="1", CCR2="0", CCR1="1", CCR0="0"のとき、輝度調整抵抗は0.6R : CCR3="1", CCR2="0", CCR1="1", CCR0="1"のとき、輝度調整抵抗は0.5R : CCR3="1", CCR2="1", CCR1="0", CCR0="0"のとき、輝度調整抵抗は0.4R : CCR3="1", CCR2="1", CCR1="0", CCR0="1"のとき、輝度調整抵抗は0.3R : CCR3="1", CCR2="1", CCR1="1", CCR0="0"のとき、輝度調整抵抗は0.2R : CCR3="1", CCR2="1", CCR1="1", CCR0="1"のとき、輝度調整抵抗は0.1R</p>	IR3=0 IR2=1 IR1=1 IR0=0 bit 4~ bit 0	CCR3=1 CCR2=0 CCR1=0 CCR0=0
PDR9	<p>ポートデータレジスタ9 ~ ポート9の各端子P9₇~P9₀のデータを格納します。 : P9_n="0"のとき、P9_n端子のデータは0 : P9_n="1"のとき、P9_n端子のデータは1</p>	H'FFDC	H'00
PDRA	<p>ポートデータレジスタA ~ ポートAの各端子PA₃~PA₀のデータを格納します。 : PA_n="0"のとき、PA_n端子のデータは0 : PA_n="1"のとき、PA_n端子のデータは1</p>	H'FFDD	H'00
PCR9	<p>ポートコントロールレジスタ9 ~ ポート9の各端子P9₇~P9₀のデータを格納します。 : PCR9_n="0"のとき、P9_nは入力端子機能 : PCR9_n="1"のとき、P9_nは出力端子機能</p>	H'FFEC	H'FF
PCRA	<p>ポートコントロールレジスタA ~ ポートAの各端子PA₃~PA₀のデータを格納します。 : PCRA_n="0"のとき、PA_nは入力端子機能 : PCRA_n="1"のとき、PA_nは出力端子機能</p>	H'FFED	H'FF
TMA TMA3	<p>タイマモードレジスタA(タイマモードレジスタA3) ~ TCAの入力クロックソースを選択します。 : TMA3="0"のとき、TCAの入力クロックソースはPSS : TMA3="1"のとき、TCAの入力クロックソースはPSW</p>	H'FFB0 ビット3	"1"
IENR1 IENTA	<p>割込み許可レジスタ1(タイマA割込みイネーブル) ~ タイマA割込み要求の許可/禁止を制御します。 : IENTA="0"のとき、タイマA割込み要求を禁止 : IENTA="1"のとき、タイマA割込み要求を許可</p>	H'FFF3 ビット7	"1"
IRR1 IRRTA	<p>割込み要求レジスタ1(タイマA割込み要求フラグ) ~ タイマA割込み要求の有無を反映します。 : IRRTA="0"のとき、タイマA割込みが要求されていない : IRRTA="1"のとき、タイマA割込みが要求されている</p>	H'FFF6 ビット7	"0"

表5 使用内部レジスタ説明(つづき)

レジスタ名		説明	RAM アドレス	設定値
SYSCR1	SSBY	システムコントロールレジスタ1(ソフトウェアスタンバイ) ~スタンバイモード、ウォッチモードへの遷移を指定します。 : SSBY="0"のとき、アクティブモードでSLEEP命令を実行後、スリープ モードに遷移 : SSBY="1"のとき、アクティブモードでSLEEP命令を実行後、ウォッチ モードあるいはスタンバイモードに遷移	H'FFF0 ビット7	"1"
	STS2 STS1 STS0	(スタンバイタイムセレクト2~0) ~特定の割込みにより、スタンバイモード、ウォッチモードを解除 し、アクティブモードに遷移する場合、クロックが安定するまでCPU と周辺機能が待機する時間を指定します。 : STS2="0", STS1="0", STS0="0"のとき、待機時間は 8,192ステート : STS2="0", STS1="0", STS0="1"のとき、待機時間は 16,384ステート : STS2="0", STS1="1", STS0="0"のとき、待機時間は 32,768ステート : STS2="0", STS1="1", STS0="1"のとき、待機時間は 65,536ステート : STS2="1", STS1="*", STS0="*"のとき、待機時間は131,972ステート ただし、*:Don't Care	H'FFF0 ビット6 ビット5 ビット4	STS2=0 STS1=0 STS0=0
	LSON	(ロースピードオンフラグ) ~ウォッチモード解除時に、CPUの動作クロックを指定します。 : LSON="0"のとき、CPUの動作クロックはシステムクロック : LSON="1"のとき、CPUの動作クロックはサブクロック	H'FFF0 ビット3	"0"
SYSCR2	MSON	(ミドルスピードオンフラグ) ~ウスタンバイモード、ウォッチモード、スリープモード解除後、ア クティブ(高速)モードで動作させるか、アクティブ(中速)モー ドで動作させるかを選択します。 : MSON="0"のとき、アクティブ(高速)モードで動作 : MSON="1"のとき、アクティブ(中速)モードで動作	H'FFF1 ビット2	"0"

(4) 使用RAM説明

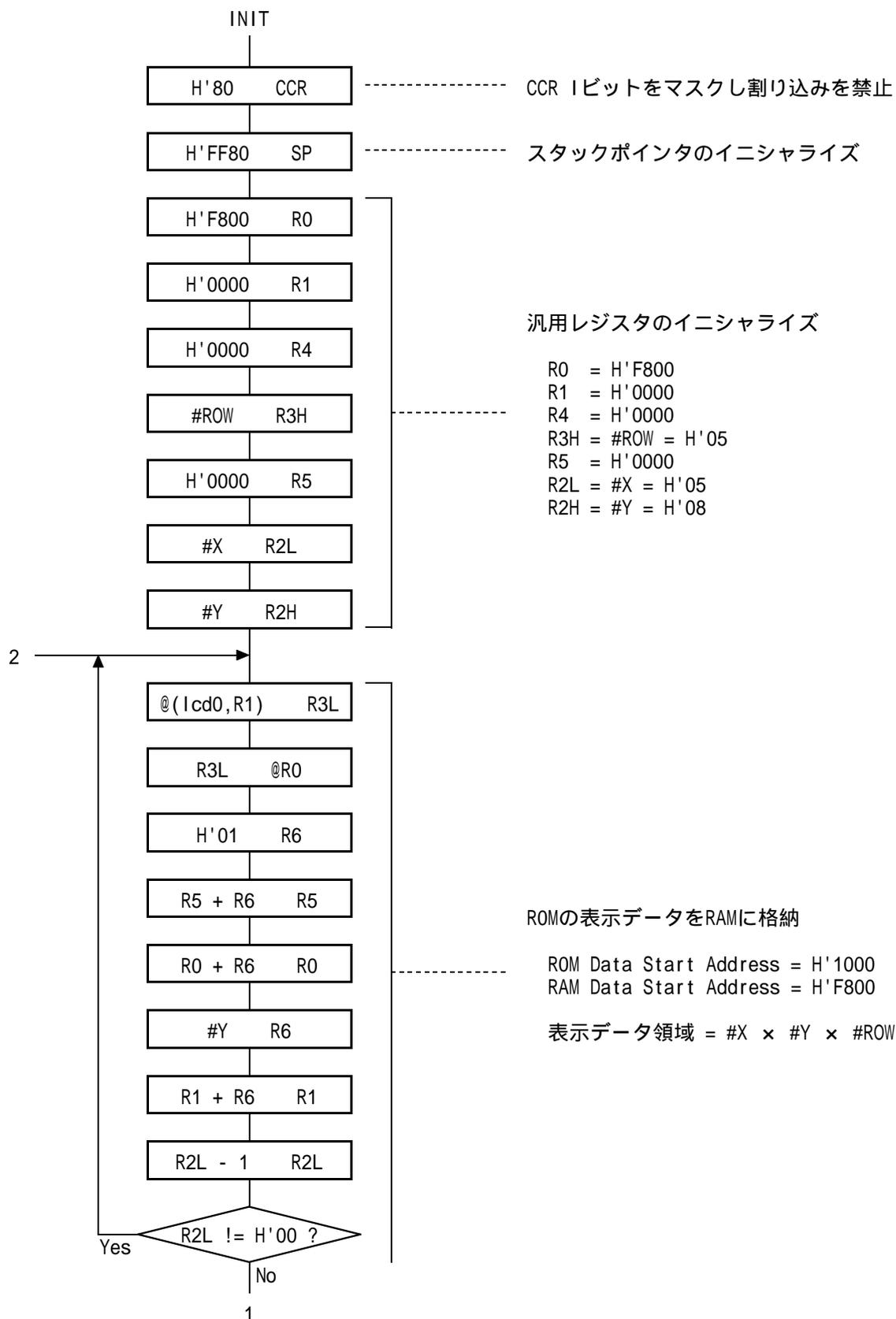
表6に本タスク例における使用RAM説明を示します。

表6 使用RAM説明

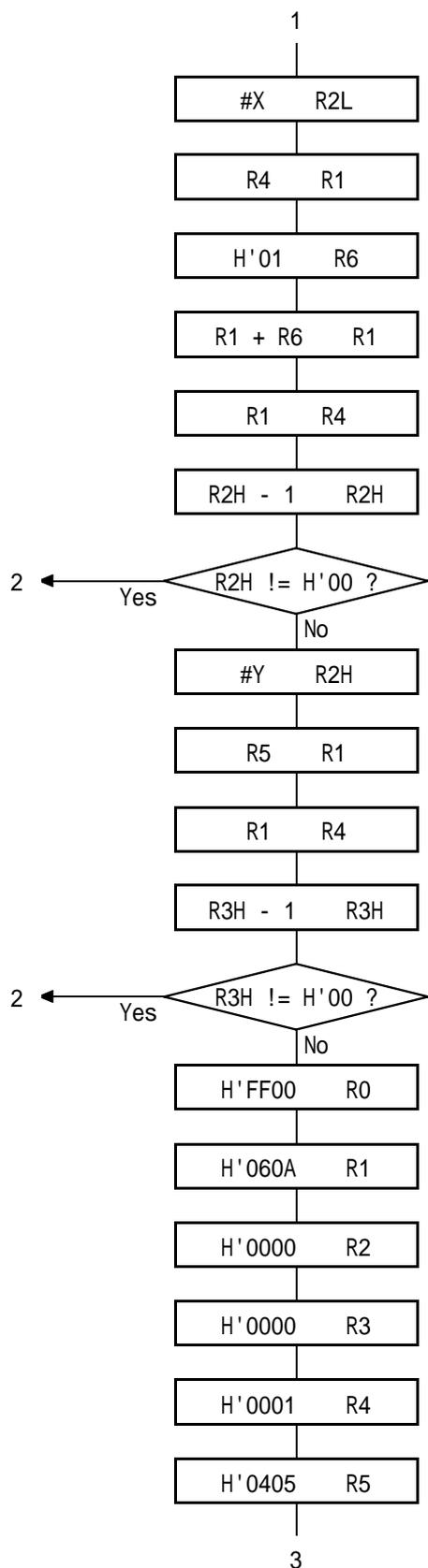
ラベル名	機能	RAM アドレス	使用モジュール
POINTER	現在書き込んでいる表示データを格納しているRAMのアドレスを 示すポインタ	H'FA00	INIT, TMRA
ADDRESS	現在書き込んでいる表示メモリのアドレスを格納	H'FA04	INIT, TMRA
POS	現在書き込んでいる表示メモリのアドレスを示すポインタ	H'FA06	INIT, TMRA

フローチャート

(1) メインルーチン



フローチャート



ROMの表示データをRAMに格納

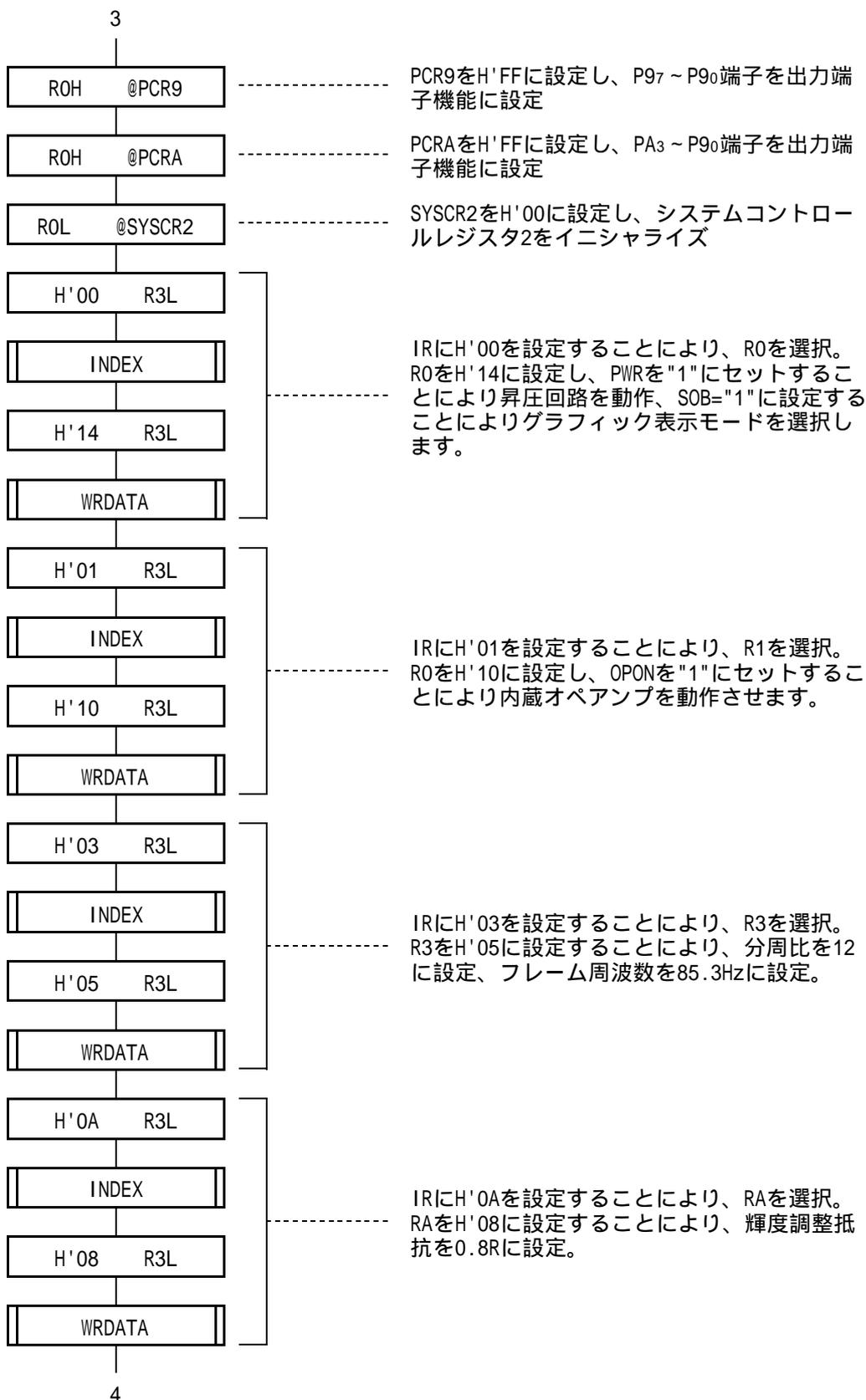
ROM Data Start Address = H'1000
RAM Data Start Address = H'F800

表示データ領域 = X × Y × ROW Byte

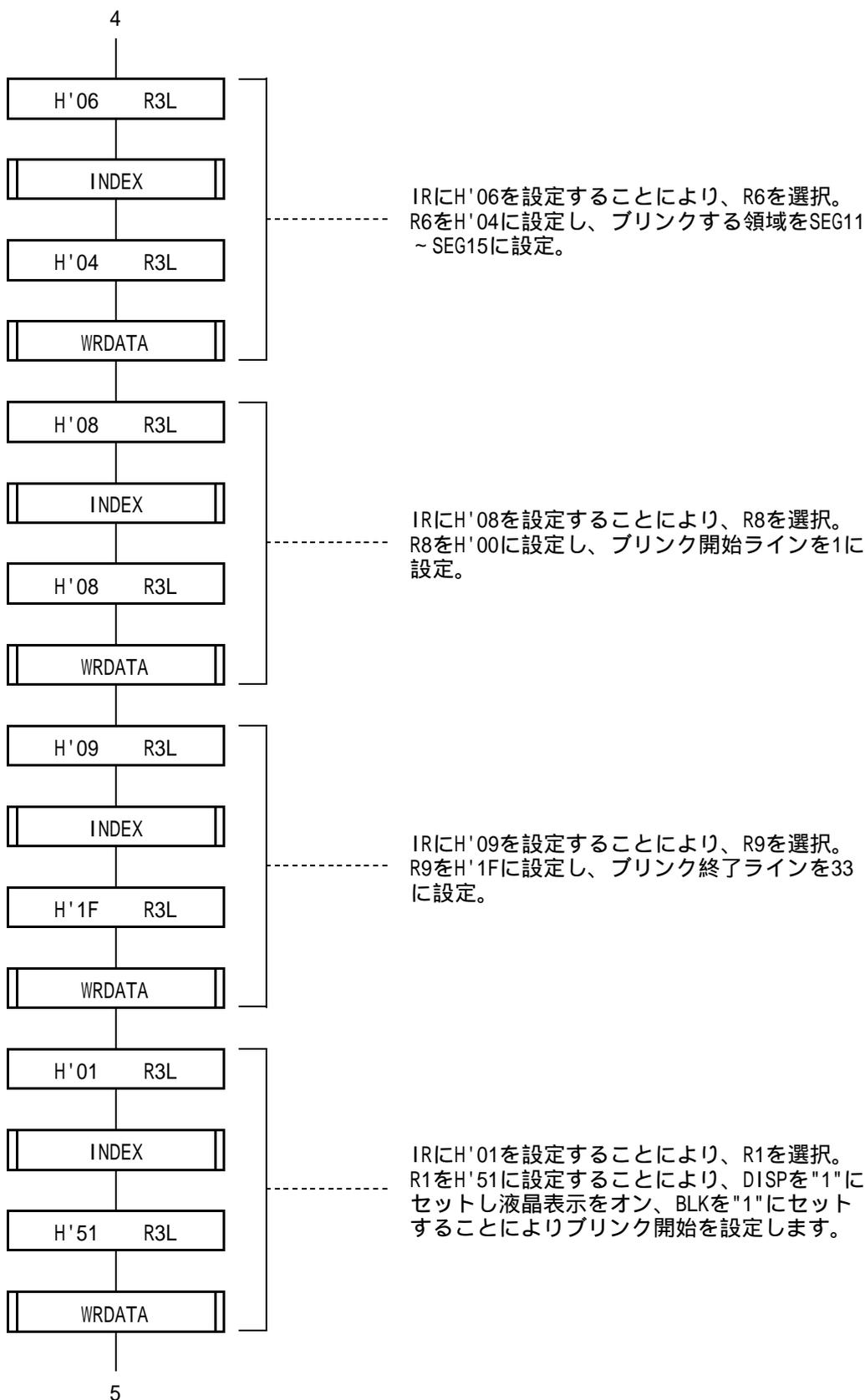
汎用レジスタのイニシャライズ

R0 = H'FF00
R1 = H'060A
R2 = H'0000
R3 = H'0000
R4 = H'0001
R5 = H'0405

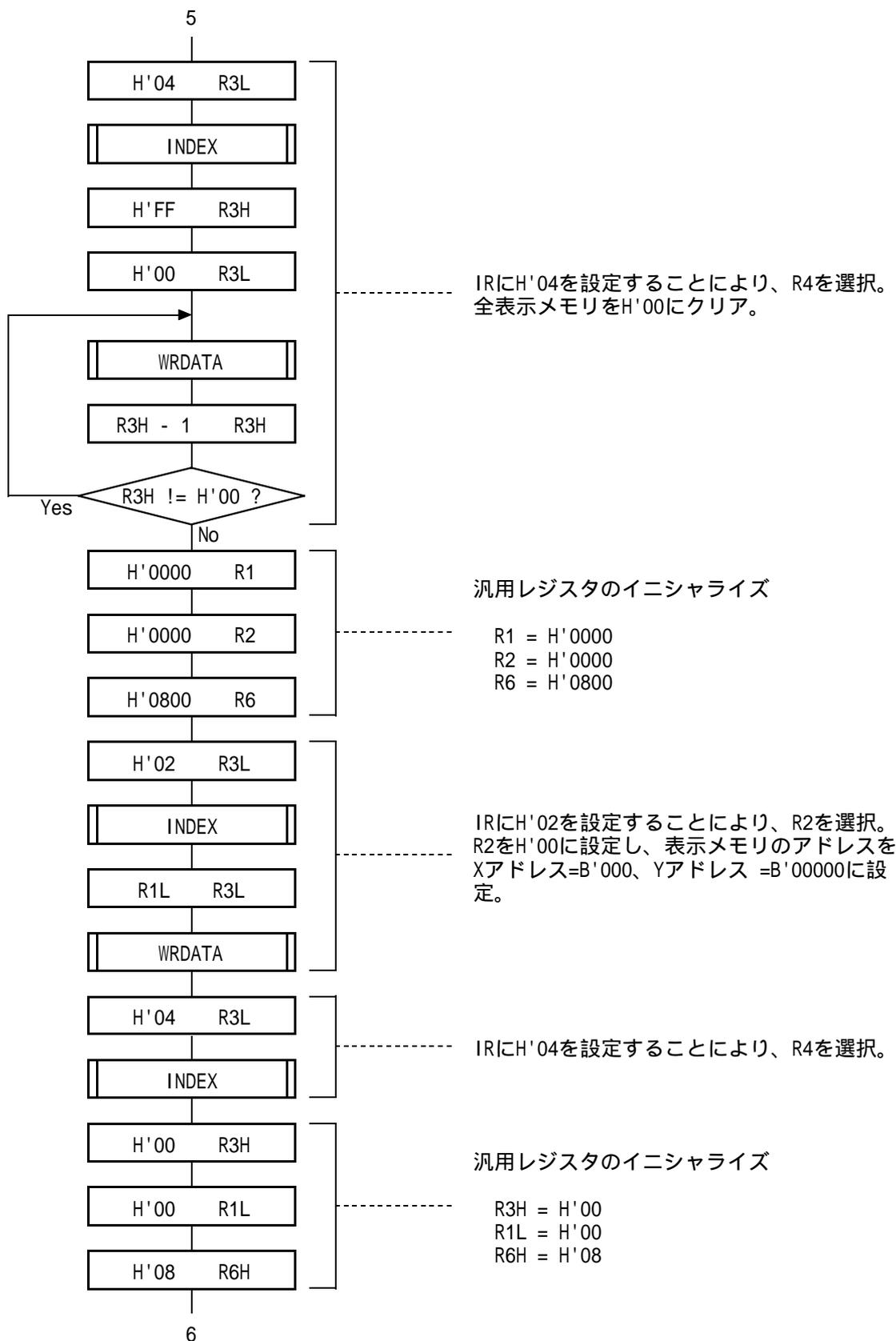
フローチャート



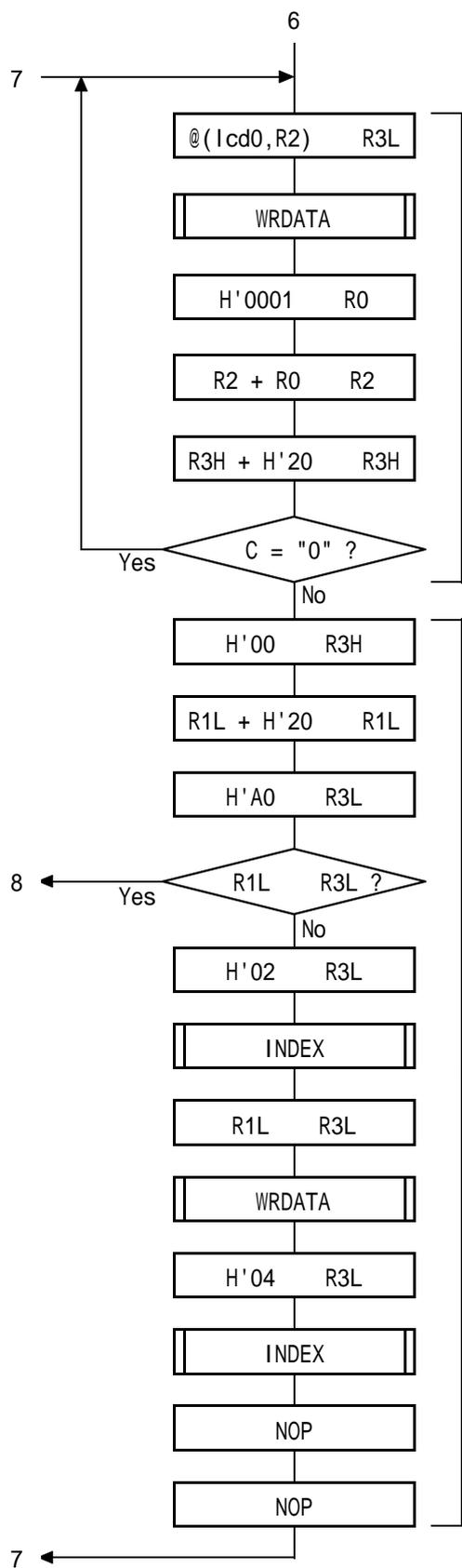
フローチャート



フローチャート



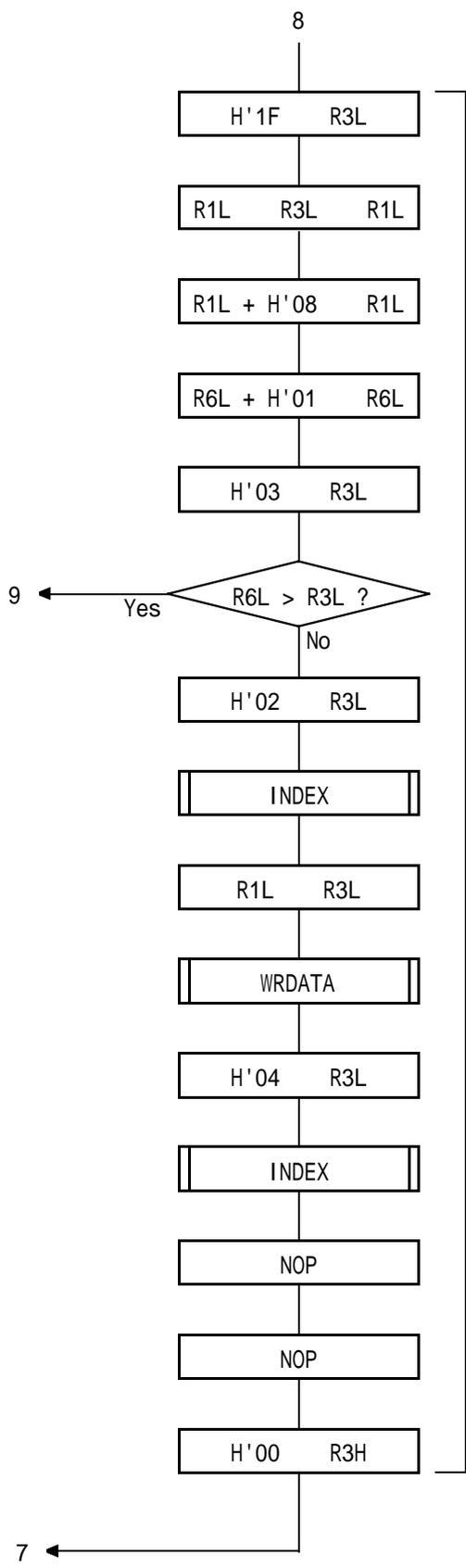
フローチャート



表示データ (1Byte×8) を表示RAMに設定
Yアドレス : H'00 ~ H'07
Xアドレス : H'00

表示データ (1Byte×8×5) を表示RAMに設定
Yアドレス : H'00からH'07
Xアドレス : H'00からH'04

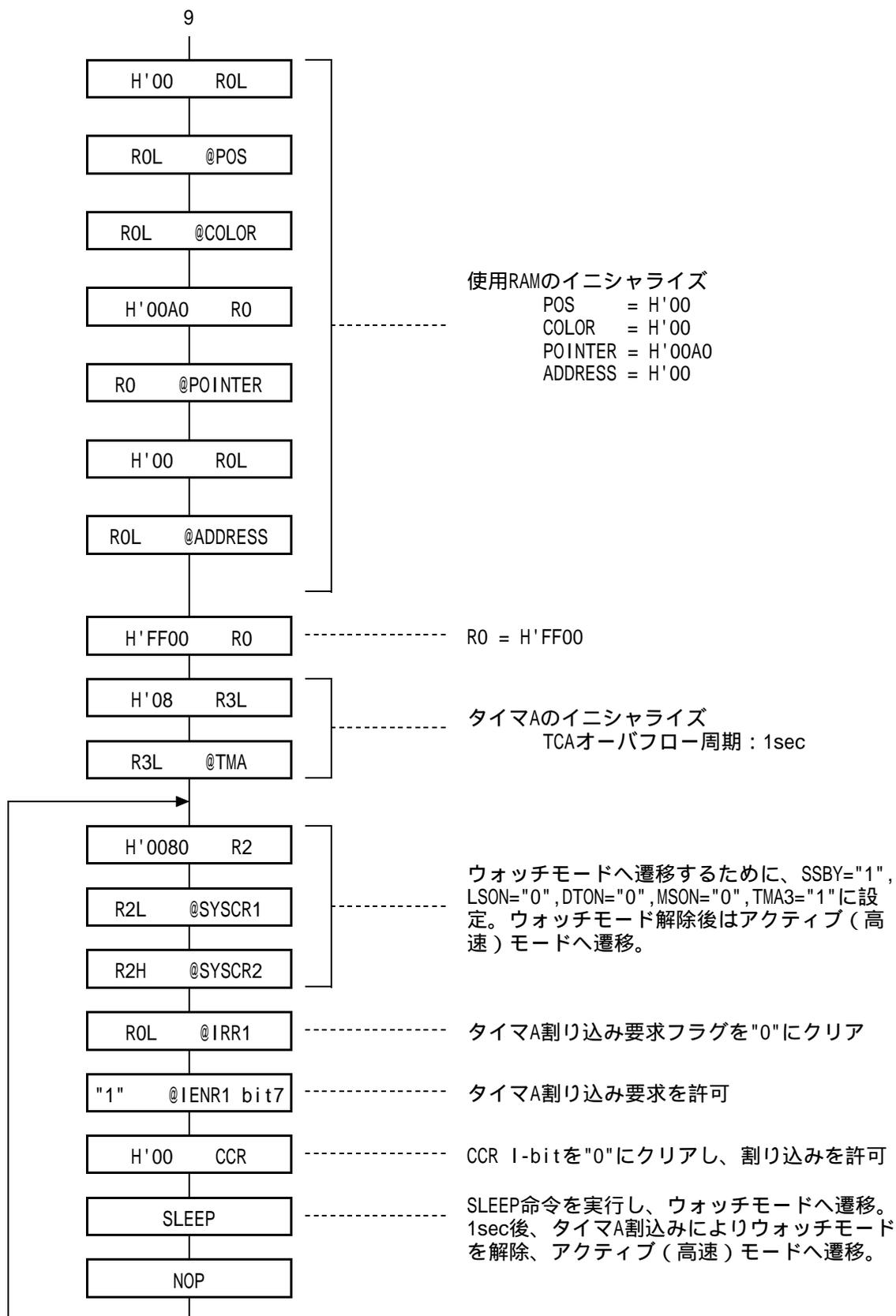
フローチャート



表示データ (1Byte×8×5×4) を表示RAMに設定

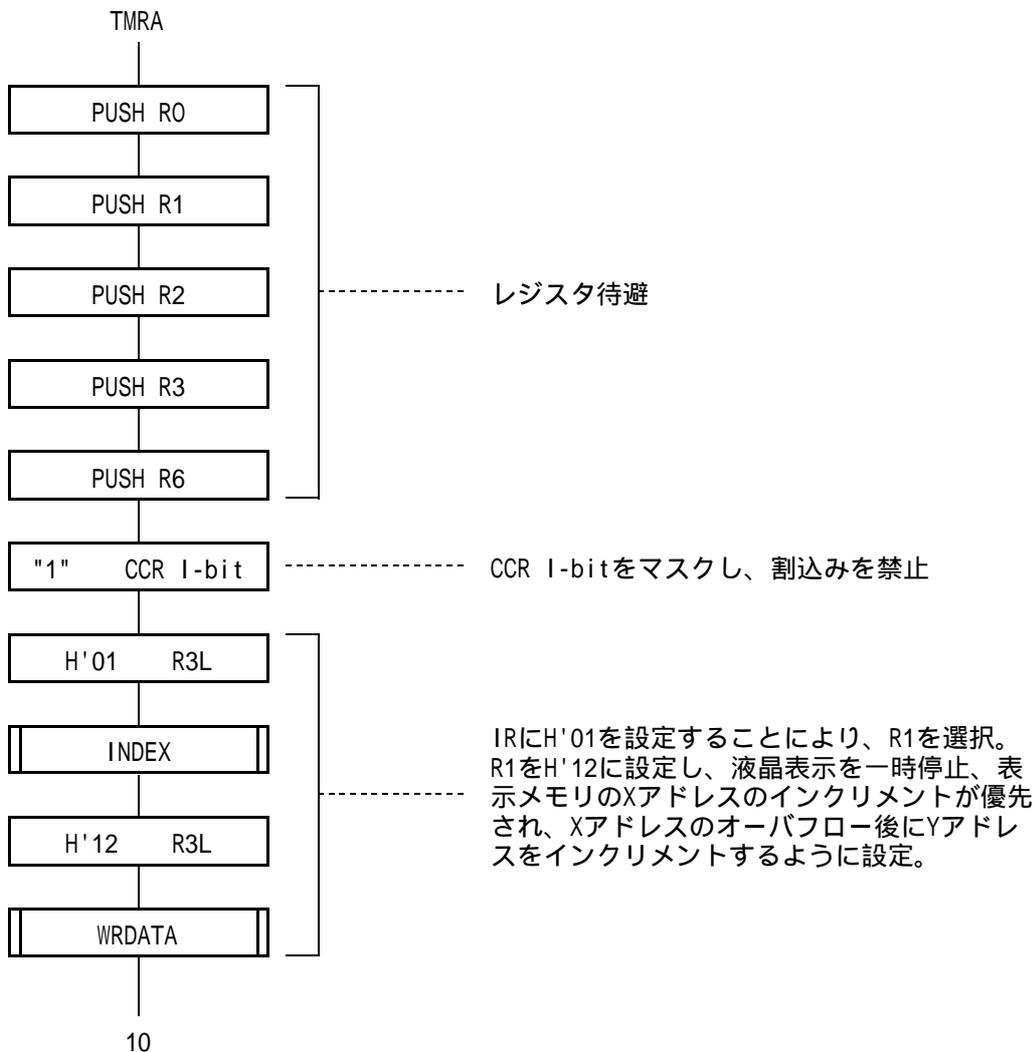
Yアドレス : H'00~H'1F
Xアドレス : H'00~H'04

フローチャート

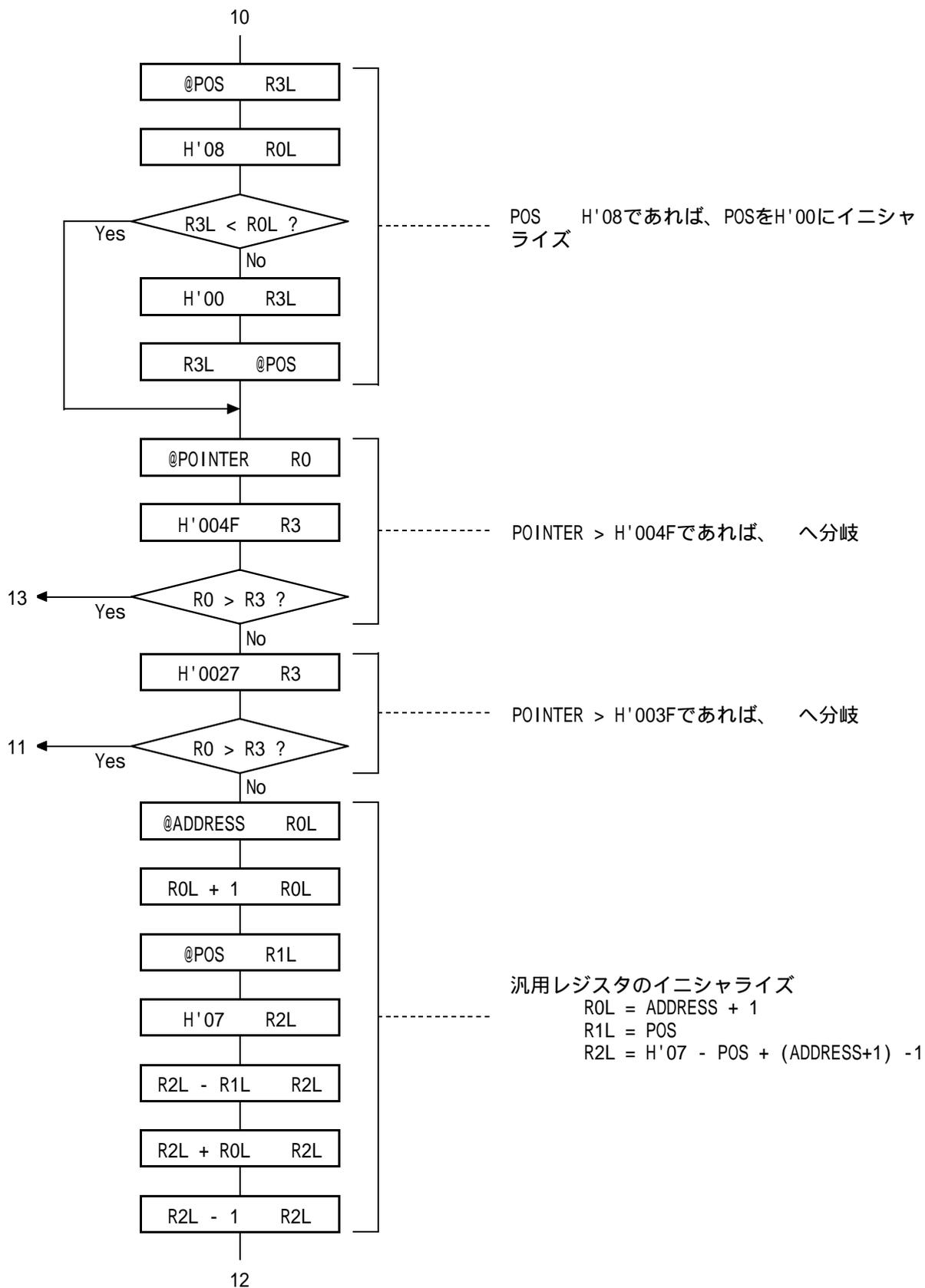


フローチャート

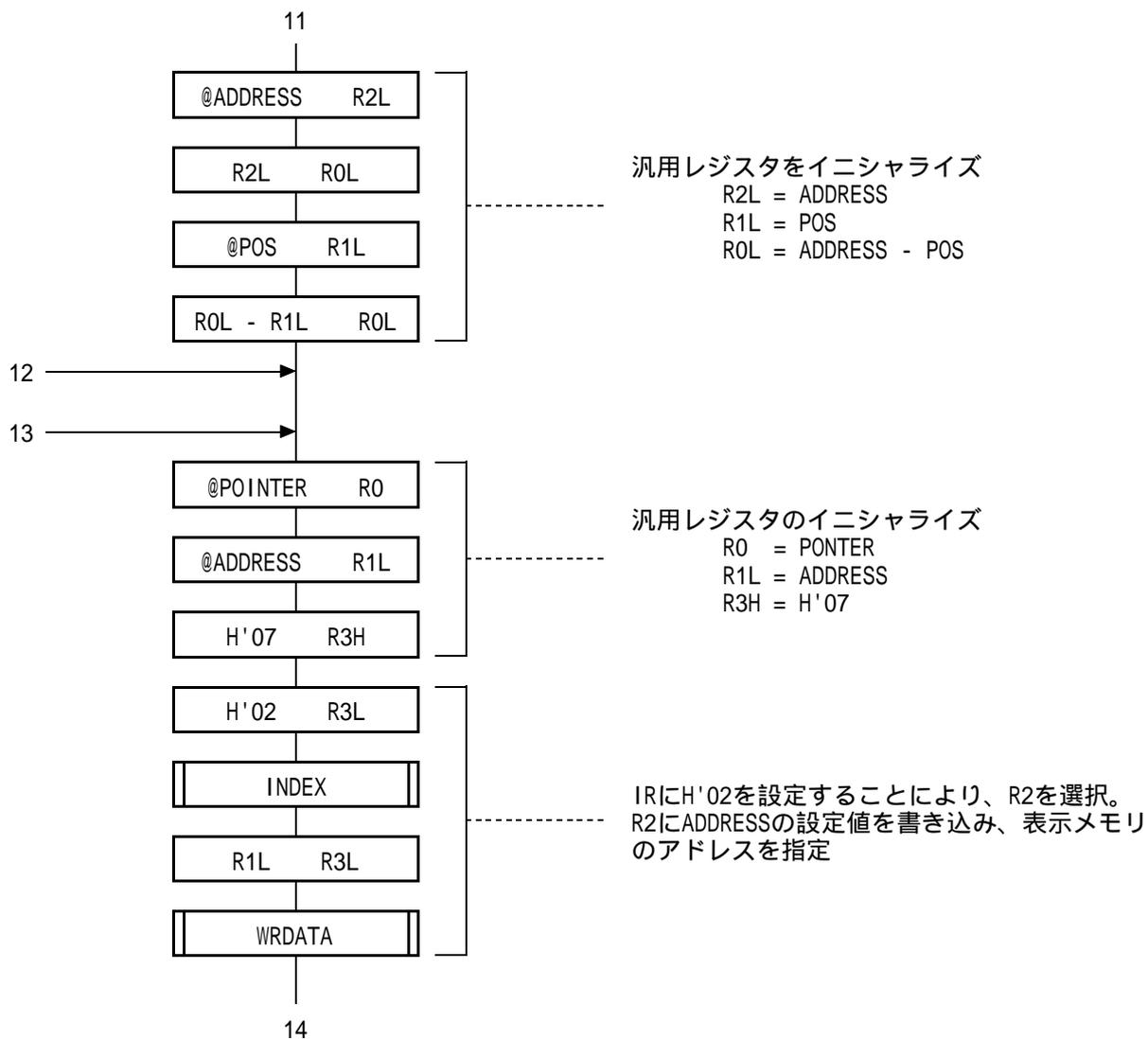
(2) タイマA割込み処理ルーチン



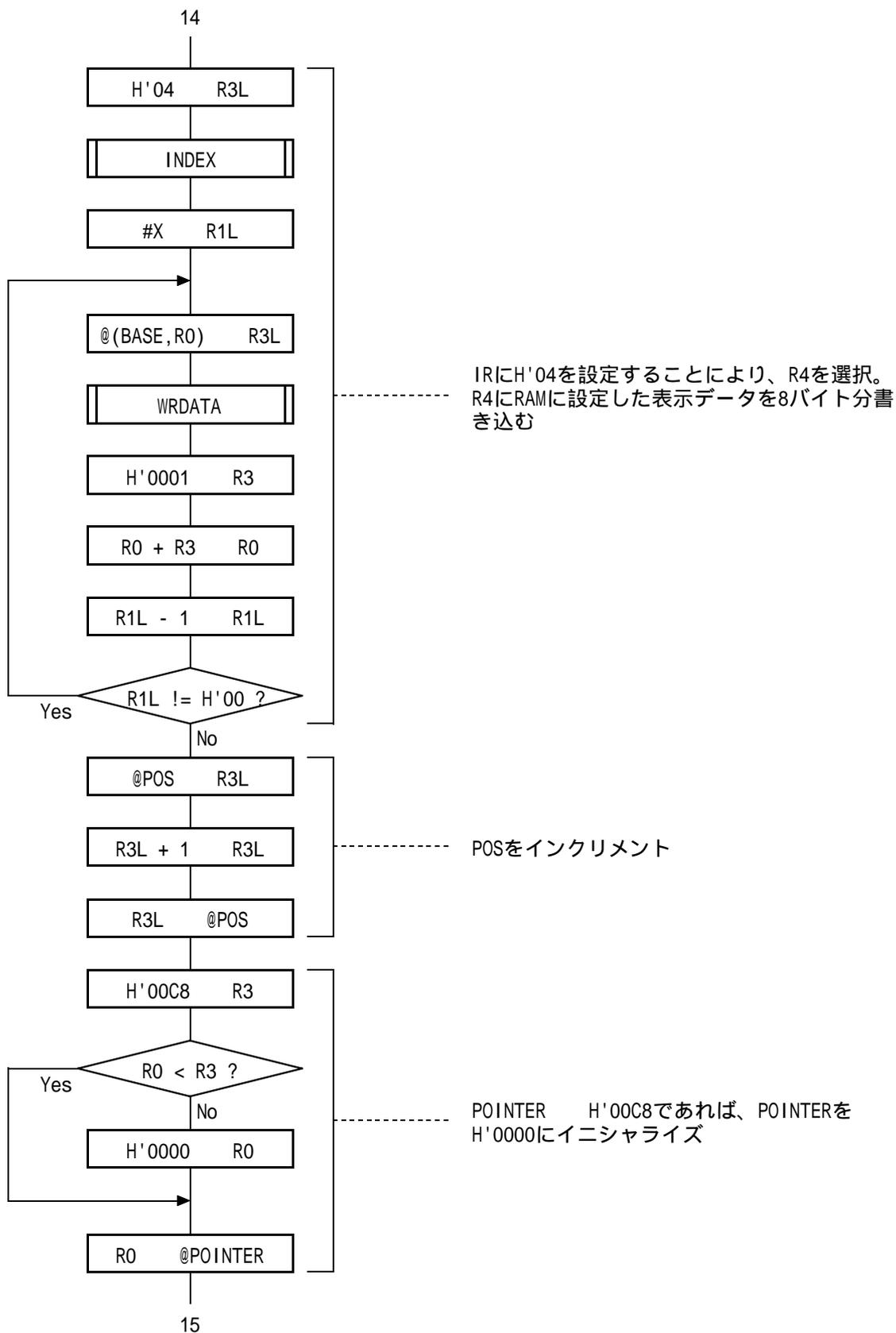
フローチャート



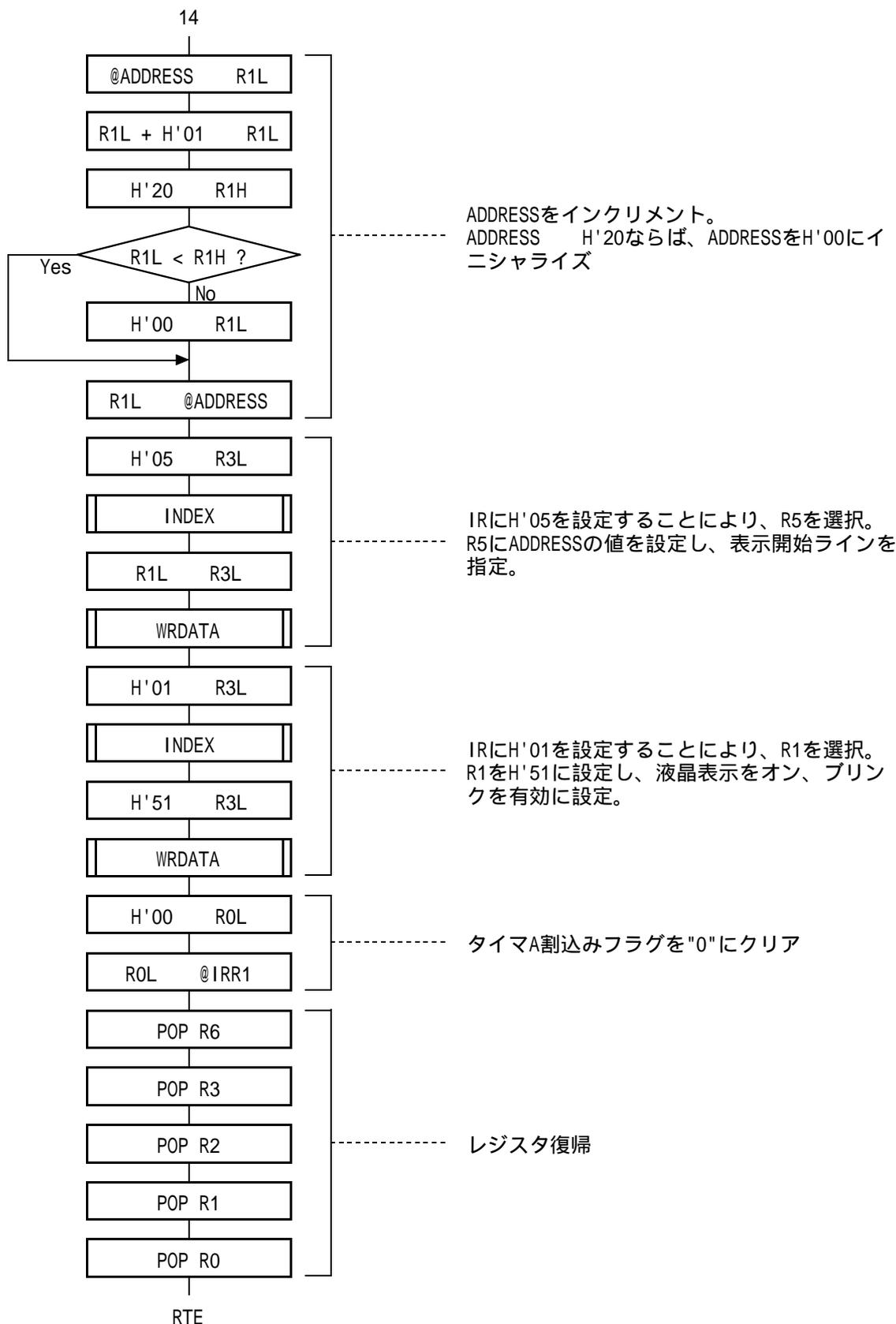
フローチャート



フローチャート

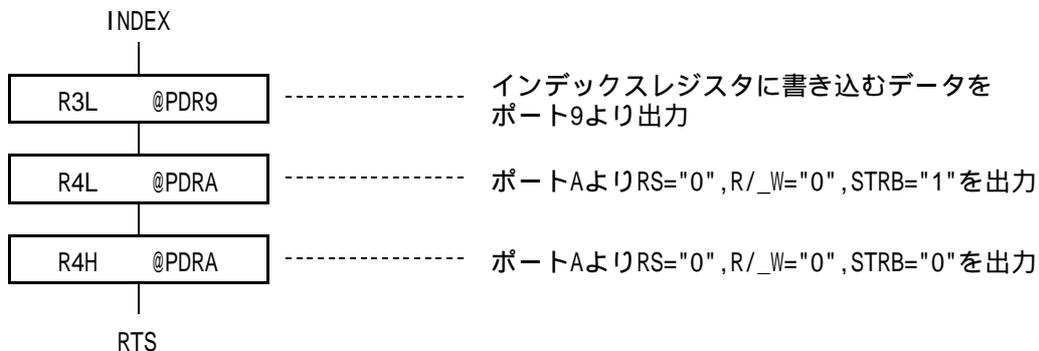


フローチャート

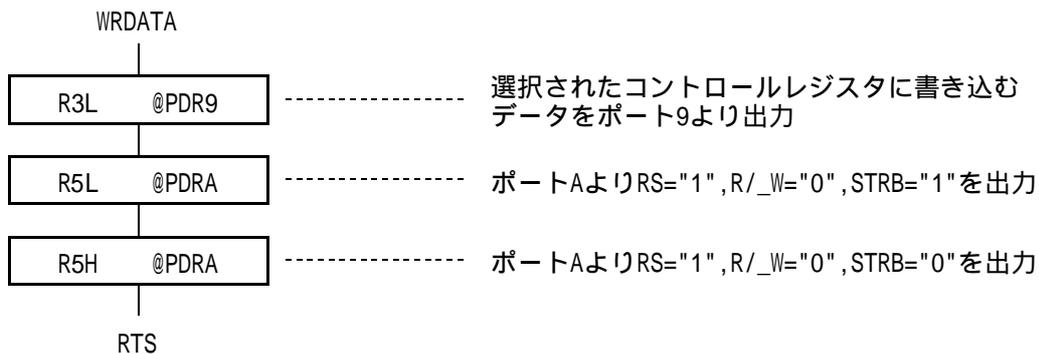


フローチャート

(3) インデックスレジスタ選択ルーチン



(4) LCDコントロールレジスタデータライトルーチン



プログラムリスト

```

;*****
;
;   H8/3857F Application Note
;
;   "Liquid Crystal Display - Graphic Display Mode-"
;
;       Function : Dot Matrix LCD Controller
;
;       External Clock : 10MHz
;       Internal Clock : 5MHz
;       Sub Clock      : 32.768kHz
;*****
;
;       .CPU          300L
;
;*****
;SYMBOL DEFINITION
;*****
;
TMA:      .EQU          H'FFB0      ;TIMER MODE REGISTER A
PDR9:    .EQU          H'FFDC      ;PORT DATA REGISTER 9
PDRA:    .EQU          H'FFDD      ;PORT DATA REGISTER A
PCR9:    .EQU          H'FFEC      ;PORT CONTROL REGISTER 9
PCRA:    .EQU          H'FFED      ;PORT CONTROL REGISTER A
SYSCR1:  .EQU          H'FFF0      ;SYSTEM CONTROL REGISTER 1
SYSCR2:  .EQU          H'FFF1      ;SYSTEM CONTROL REGISTER 2
IENR1:   .EQU          H'FFF3      ;INTERRUPT ENABLE REGISTER 1
IRR1:    .EQU          H'FFF6      ;INTERRUPT REQUEST REGISTER 1
;
;*****
;RAM ALLOCATION
;*****
;
BASE      .EQU          H'0F800     ;START ADDRESS OF THE DATA TABLE IN THE RAM
X         .EQU          H'5         ;DISPLAY CHARACTER NUMBER PER LINE
Y         .EQU          H'8         ;DISPLAY CHARACTER LENGTH IN VERTICAL DIRECTION
ROW       .EQU          H'5         ;THE LENGTH OF ALL THE DISPLAY DATAS
POINTER   .EQU          H'0FA00     ;POINTER OF THE CURRENT WRITEING DATA POSITION
ADDRESS   .EQU          H'0FA04     ;ADDRESS OF THE CURRENT WRITEING LCD RAM POSITION
POS       .EQU          H'0FA06     ;POINTER OF THE CURRENT WRITEING LCD RAM LINE
;
;*****
;VECTOR ADDRESS
;*****
;
;
;       .ORG          H'0000
;       .DATA.W      INIT           ;Reset Vector
;
;
;       .ORG          H'0008
;       .DATA.W      INIT           ;IRQ0 Vector
;       .DATA.W      INIT           ;IRQ1 Vector
;       .DATA.W      INIT           ;IRQ2 Vector
;       .DATA.W      INIT           ;IRQ3 Vector
;       .DATA.W      INIT           ;IRQ4 Vectoe
;       .DATA.W      INIT           ;WKPO Vector
;       .DATA.W      INIT           ;SC11 Vector
;       .DATA.W      TMRA          ;Timer A Vector
;       .DATA.W      INIT           ;Timer B Vector
;       .DATA.W      INIT           ;Timer C Vector
;       .DATA.W      INIT           ;Timer FH Vector

```

プログラムリスト

```

;
;          .DATA.W      INIT          ;Timer FL Vector
;
;          .ORG          H'0024
;          .DATA.W      INIT          ;SCI3 Vector
;          .DATA.W      INIT          ;A/D Vector
;          .DATA.W      INIT          ;DT Vector
;
;*****
; INIT : MAIN ROUTINE
;*****
;
;          .ORG          H'0100
;
; INIT:      LDC          #H'80,CCR      ;INT DISABLE
;          MOV.W        #H'FF80,SP      ;SP INIT
;
; PTOP1:     MOV.W        #H'0F80,R0      ;STORE THE ROM DATA IN THE RAM STARTING
;                                     ;FROM #0F80
;
;          MOV.W        #0,R1
;          MOV.W        #0,R4
;          MOV.B        #ROW,R3H
;          MOV.W        #0,R5
;          MOV.B        #X,R2L
;          MOV.B        #Y,R2H          ;THE DISPLAY DATA AREA SIZE IS #X * #Y * ROW
;                                     ;BYTES
;
; JJ1:      MOV.B        @(Icd0,R1),R3L ;THE DISPALY DATA IN THE ROM
;                                     ;STARTING FROM #LCDO
;
;          MOV.B        R3L,@R0
;          MOV.W        #1,R6
;          ADD.W        R6,R5
;          ADD.W        R6,R0
;          MOV.W        #Y,R6
;          ADD.W        R6,R1
;          DEC          R2L
;          BNE          JJ1
;          MOV.B        #X,R2L
;          MOV.W        R4,R1
;          MOV.W        #1,R6
;          ADD.W        R6,R1
;          MOV.W        R1,R4
;          DEC          R2H
;          BNE          JJ1
;          MOV.B        #Y,R2H
;          MOV.W        R5,R1
;          MOV.W        R1,R4
;          DEC          R3H
;          BNE          JJ1
;
;*****
; SET THE GENERAL REGISTER
;*****
;
;
; PTOP:     MOV.W        #H'OFF00,R0      ;GENERAL
;          MOV.W        #H'060A,R1      ;WATCH
;          MOV.W        #H'0000,R2      ;MODE TRANSITION
;          MOV.W        #H'0000,R3      ;TEMP
;          MOV.W        #H'0001,R4      ;LCD I/F INDEX
;          MOV.W        #H'0405,R5      ;LCD I/F DATA
;          MOV.B        ROH,@PCR9:8      ;
;          MOV.B        ROH,@PCRA:8      ;

```

プログラムリスト

```

MOV.B          ROL,@SYSCR2:8 ;NC ON
;
;*****
; INITIALIZE THE LCD CONTROL REGISTER
;*****
;
MOV.B          #H'00,R3L
JSR            @INDEX
MOV.B          #H'14,R3L
JSR            @WRDATA
;
MOV.B          #H'01,R3L
JSR            @INDEX
MOV.B          #H'10,R3L
JSR            @WRDATA
;
MOV.B          #H'03,R3L
JSR            @INDEX
MOV.B          #H'05,R3L
JSR            @WRDATA
;
MOV.B          #H'0A,R3L
JSR            @INDEX
MOV.B          #H'08,R3L
JSR            @WRDATA
;
MOV.B          #H'06,R3L
JSR            @INDEX
MOV.B          #H'04,R3L
JSR            @WRDATA
;
MOV.B          #H'08,R3L
JSR            @INDEX
MOV.B          #H'0,R3L
JSR            @WRDATA
;
MOV.B          #H'09,R3L
JSR            @INDEX
MOV.B          #H'1f,R3L
JSR            @WRDATA
;
MOV.B          #H'01,R3L
JSR            @INDEX
MOV.B          #H'51,R3L
JSR            @WRDATA
;
MOV.B          #H'04,R3L
JSR            @INDEX
MOV.B          #H'0FF,R3H
MOV.B          #H'00,R3L ;WRITE DATA
LRAMI: JSR      @WRDATA ;CLEAR ALL THE LCD RAM
DEC          R3H
BNE          LRAMI
;
;*****
;WRITE THE DISPLAY DATA INTO THE LCD RAM
; IN THIS PART , THE LCD RAM INCREASEMENT IS Y ADDRESS FIRST
;*****
mov.w        #h'0000,r1
mov.w        #h'0000,r2

```

プログラムリスト

```

mov.w      #h'0800, r6
mov.b      #h'02, r3l
jsr        @INDEX
mov.b      r1l, r3l
jsr        @WRDATA
mov.b      #h'4, r3l
jsr        @INDEX
mov.b      #h'0, r3h
mov.b      #h'00, r1l
mov.b      #8, r6h
;
gg1:
mov.b      @(lcd0, r2), r3l
jsr        @WRDATA
mov.w      #h'1, r0
add.w      r0, r2
add.b      #h'20, r3h
bcc        gg1
gg2:
mov.b      #h'0, r3h
add.b      #h'20, r1l
MOV.B     #H'0A0, R3L
CMP.B     R3L, R1L
bcc        gg3
gg23:
mov.b      #h'02, r3l
jsr        @INDEX
mov.b      r1l, r3l
jsr        @WRDATA
mov.b      #h'4, r3l
jsr        @INDEX
nop
nop
jmp        @gg1
gg3:
mov.b      #h'1f, r3l
and.b      r3l, r1l
add.b      #h'08, r1l
add.b      #h'1, r6l
mov.b      #h'3, r3l
cmp.b      r3l, r6l
bhi        TIMA
;
mov.b      #h'02, r3l
jsr        @INDEX
mov.b      r1l, r3l
jsr        @WRDATA
mov.b      #h'4, r3l
jsr        @INDEX
nop
nop
mov.b      #0, r3h
jmp        @gg1
;
; *****
; SET THE VARIABLE
; *****
;
TIMA:
MOV.B     #H'0, ROL

```

プログラムリスト

```

MOV.B      ROL,@POS
MOV.W      MOV.W #H'A0,R0
            RO,@POINTER
MOV.B      #H'00,ROL
MOV.B      ROL,@ADDRESS
mov.w      #h'0f00,r0
MOV.B      #H'08,R3L      ;SET THE TIMERA
MOV.B      R3L,@TMA:8
;
;*****
; SLEEP
;*****
SLEEP:
MOV.W      #H'0080,R2
MOV.B      R2L,@SYSCR1:8 ;7:SSBY 3:LSON 3:DTON 2:MSON TMA3
MOV.B      R2H,@SYSCR2:8 ; 1 0 0 0 1
MOV.B      ROL,@IRR1:8 ;
BSET       #H'7,@IENR1:8 ;IENTA=1
LDC        #H'00,CCR
SLEEP      ;TO WATCH MODE (CKW/2)
NOP
BRA        SLEEP
;
;*****
;TMRA : TIMER A INTERRUPT ROUTINE
;*****
TMRA:
PUSH.W     R0
PUSH.W     R1
PUSH.W     R2
PUSH.W     R3
PUSH.W     R6
ORC        #H'80,CCR      ;INT DISABLE
MOV.B      #H'01,R3L
JSR        @INDEX        ;STOP THE LCD DISPLAY
MOV.B      #H'12,R3L     ;CHANGE THE INC MODE TO SET THE X ADDRESS
                        ;FIRST
JSR        @WRDATA
;
;*****
; CONTROL THE BLINK AREA POSITION
;*****
NOPOS:
MOV.B      @POS,R3L
MOV.B      #H'8,ROL
CMP.B      ROL,R3L
BLO
MOV.B      #H'0,R3L
MOV.B      R3L,@POS
MOV.W      @POINTER,R0
MOV.W      #H'4F,R3
CMP.W      R3,R0
BHI        NOTHING
MOV.W      #H'27,R3
CMP.W      R3,R0
BHI        CHAN1
MOV.B      @ADDRESS,ROL
INC        ROL

```

プログラムリスト

```

MOV.B      @POS,R1L
MOV.B      #H'7,R2L
SUB.B      R1L,R2L
ADD.B      R0L,R2L
dec        r2l
JMP        @SETB

CHAN1:
MOV.B      @ADDRESS,R2L
MOV.B      R2L,R0L
MOV.B      @POS,R1L
SUB.B      R1L,R0L

SETB:
NOTHING:
MOV.W      @POINTER,R0
MOV.B      @ADDRESS,R1L
MOV.B      #7,R3H
mov.b      #h'02,r3l
jsr        @INDEX
mov.b      r1l,r3l
jsr        @WRDATA
mov.b      #h'4,r3l
jsr        @INDEX
MOV.B      #X,R1L

TMRA2:
MOV.B      @(BASE,R0),R3L
JSR        @WRDATA
MOV.W      #1,R3
ADD.W      R3,R0
DEC        R1L
BNE        TMRA2
MOV.B      @POS,R3L
INC        R3L
MOV.B      R3L,@POS
MOV.W      #H'C8,R3
CMP.W      R3,R0
BLO        TMRA3
MOV.W      #0,R0

TMRA3:
MOV.W      R0,@POINTER
MOV.B      @ADDRESS,R1L
ADD.B      #1,R1L
MOV.B      #H'20,R1H
CMP.B      R1H,R1L
BLO        TMRA1
MOV.B      #H'0,R1L

TMRA1:
MOV.B      R1L,@ADDRESS
MOV.B      #5,R3L
JSR        @INDEX
MOV.B      R1L,R3L
JSR        @WRDATA
;
;*****
;
BLEND1:
MOV.B      #1,R3L
JSR        @INDEX
MOV.B      #H'51,R3L
JSR        @WRDATA
mov.b      #0,r0l
mov.b      r0l,@IRR1

```

プログラムリスト

```

;
BTMRA:
        POP.W      R6
        POP.W      R3
        POP.W      R2
        POP.W      R1
        POP.W      R0
        RTE

;*****
;PROGRAM SUBROUTINE
;*****
;
INDEX:
        MOV.B      R3L,@PDR9:8 ;
        MOV.B      R4L,@PDRA:8 ;RS = RW = 0 , STRB = 1
        MOV.B      R4H,@PDRA:8 ;RS = RW = 0 , STRB = 0
        RTS

;
WRDATA:
        MOV.B      R3L,@PDR9:8 ;
        MOV.B      R5L,@PDRA:8 ;RS = 1 , RW = 0 , STRB = 1
        MOV.B      R5H,@PDRA:8 ;RS = 1 , RW = 0 , STRB = 0
        RTS

;*****
; LCD DATA AREA
;*****
;
;
        .ORG      H'1000

lcd0:
;1
        .DATA.B    H'80
        .DATA.B    H'40
        .DATA.B    H'20
        .DATA.B    H'10
        .DATA.B    H'08
        .DATA.B    H'04
        .DATA.B    H'02
        .DATA.B    H'01

;2
        .DATA.B    H'01
        .DATA.B    H'02
        .DATA.B    H'04
        .DATA.B    H'08
        .DATA.B    H'10
        .DATA.B    H'20
        .DATA.B    H'40
        .DATA.B    H'80

;3
        .DATA.B    H'55
        .DATA.B    H'AA
        .DATA.B    H'55
        .DATA.B    H'AA
        .DATA.B    H'55
        .DATA.B    H'AA
        .DATA.B    H'55
        .DATA.B    H'AA

;4
        .DATA.B    H'80
        .DATA.B    H'40

```

プログラムリスト

```

        .DATA .B      H' 20
        .DATA .B      H' 10
        .DATA .B      H' 08
        .DATA .B      H' 04
        .DATA .B      H' 02
        .DATA .B      H' 01
;5
        .DATA .B      H' 01
        .DATA .B      H' 02
        .DATA .B      H' 04
        .DATA .B      H' 08
        .DATA .B      H' 10
        .DATA .B      H' 20
        .DATA .B      H' 40
        .DATA .B      H' 80
LCD1:
;1
        .DATA .B      H' 80
        .DATA .B      H' 40
        .DATA .B      H' 20
        .DATA .B      H' 10
        .DATA .B      H' 08
        .DATA .B      H' 04
        .DATA .B      H' 02
        .DATA .B      H' 01
;2
        .DATA .B      H' 01
        .DATA .B      H' 02
        .DATA .B      H' 04
        .DATA .B      H' 08
        .DATA .B      H' 10
        .DATA .B      H' 20
        .DATA .B      H' 40
        .DATA .B      H' 80
;3
        .DATA .B      H' 55
        .DATA .B      H' AA
        .DATA .B      H' 55
        .DATA .B      H' AA
        .DATA .B      H' 55
        .DATA .B      H' AA
        .DATA .B      H' 55
        .DATA .B      H' AA
;4
        .DATA .B      H' 80
        .DATA .B      H' 40
        .DATA .B      H' 20
        .DATA .B      H' 10
        .DATA .B      H' 08
        .DATA .B      H' 04
        .DATA .B      H' 02
        .DATA .B      H' 01
;5
        .DATA .B      H' 01
        .DATA .B      H' 02
        .DATA .B      H' 04
        .DATA .B      H' 08
        .DATA .B      H' 10
        .DATA .B      H' 20
        .DATA .B      H' 40

```

プログラムリスト

LCD2:	.DATA.B	H'80
;1		
	.DATA.B	H'80
	.DATA.B	H'40
	.DATA.B	H'20
	.DATA.B	H'10
	.DATA.B	H'08
	.DATA.B	H'04
	.DATA.B	H'02
	.DATA.B	H'01
;2		
	.DATA.B	H'01
	.DATA.B	H'02
	.DATA.B	H'04
	.DATA.B	H'08
	.DATA.B	H'10
	.DATA.B	H'20
	.DATA.B	H'40
	.DATA.B	H'80
;3		
	.DATA.B	H'55
	.DATA.B	H'AA
	.DATA.B	H'55
	.DATA.B	H'AA
	.DATA.B	H'55
	.DATA.B	H'AA
	.DATA.B	H'55
	.DATA.B	H'AA
;4		
	.DATA.B	H'80
	.DATA.B	H'40
	.DATA.B	H'20
	.DATA.B	H'10
	.DATA.B	H'08
	.DATA.B	H'04
	.DATA.B	H'02
	.DATA.B	H'01
;5		
	.DATA.B	H'01
	.DATA.B	H'02
	.DATA.B	H'04
	.DATA.B	H'08
	.DATA.B	H'10
	.DATA.B	H'20
	.DATA.B	H'40
	.DATA.B	H'80
LCD3		
;1		
	.DATA.B	H'80
	.DATA.B	H'40
	.DATA.B	H'20
	.DATA.B	H'10
	.DATA.B	H'08
	.DATA.B	H'04
	.DATA.B	H'02
	.DATA.B	H'01
;2		
	.DATA.B	H'01
	.DATA.B	H'02

プログラムリスト

```

        .DATA .B      H' 04
        .DATA .B      H' 08
        .DATA .B      H' 10
        .DATA .B      H' 20
        .DATA .B      H' 40
        .DATA .B      H' 80
;3
        .DATA .B      H' 55
        .DATA .B      H' AA
        .DATA .B      H' 55
        .DATA .B      H' AA
        .DATA .B      H' 55
        .DATA .B      H' AA
        .DATA .B      H' 55
        .DATA .B      H' AA
;4
        .DATA .B      H' 80
        .DATA .B      H' 40
        .DATA .B      H' 20
        .DATA .B      H' 10
        .DATA .B      H' 08
        .DATA .B      H' 04
        .DATA .B      H' 02
        .DATA .B      H' 01
;5
        .DATA .B      H' 01
        .DATA .B      H' 02
        .DATA .B      H' 04
        .DATA .B      H' 08
        .DATA .B      H' 10
        .DATA .B      H' 20
        .DATA .B      H' 40
        .DATA .B      H' 80
LCD4:
;1
        .DATA .B      H' 80
        .DATA .B      H' 40
        .DATA .B      H' 20
        .DATA .B      H' 10
        .DATA .B      H' 08
        .DATA .B      H' 04
        .DATA .B      H' 02
        .DATA .B      H' 01
;2
        .DATA .B      H' 01
        .DATA .B      H' 02
        .DATA .B      H' 04
        .DATA .B      H' 08
        .DATA .B      H' 10
        .DATA .B      H' 20
        .DATA .B      H' 40
        .DATA .B      H' 80
;3
        .DATA .B      H' 55
        .DATA .B      H' AA
        .DATA .B      H' 55
        .DATA .B      H' AA
        .DATA .B      H' 55
        .DATA .B      H' AA
        .DATA .B      H' 55

```

プログラムリスト

```

;4          .DATA.B      H'AA
           .DATA.B      H'80
           .DATA.B      H'40
           .DATA.B      H'20
           .DATA.B      H'10
           .DATA.B      H'08
           .DATA.B      H'04
           .DATA.B      H'02
           .DATA.B      H'01
;5
           .DATA.B      H'01
           .DATA.B      H'02
           .DATA.B      H'04
           .DATA.B      H'08
           .DATA.B      H'10
           .DATA.B      H'20
           .DATA.B      H'40
           .DATA.B      H'80
;
           .END

```

H8/3857シリーズ アプリケーションノート

発行年月 平成11年3月 第1版

発行 株式会社 日立製作所

半導体事業本部統括営業本部

編集 株式会社 超Lメディア

技術ドキュメントグループ

©株式会社 日立製作所

H8/3857 シリーズ アプリケーションノート



ルネサスエレクトロニクス株式会社
神奈川県川崎市中原区下沼部1753 〒211-8668

ADJ-502-077