

To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

"Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.

8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

To all our customers

Regarding the change of names mentioned in the document, such as Hitachi Electric and Hitachi XX, to Renesas Technology Corp.

The semiconductor operations of Mitsubishi Electric and Hitachi were transferred to Renesas Technology Corporation on April 1st 2003. These operations include microcomputer, logic, analog and discrete devices, and memory chips other than DRAMs (flash memory, SRAMs etc.) Accordingly, although Hitachi, Hitachi, Ltd., Hitachi Semiconductors, and other Hitachi brand names are mentioned in the document, these names have in fact all been changed to Renesas Technology Corp. Thank you for your understanding. Except for our corporate trademark, logo and corporate statement, no changes whatsoever have been made to the contents of the document, and these changes do not constitute any alteration to the contents of the document itself.

Renesas Technology Home Page: <http://www.renesas.com>

Renesas Technology Corp.
Customer Support Dept.
April 1, 2003

Cautions

Keep safety first in your circuit designs!

1. Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
2. Renesas Technology Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.
The information described here may contain technical inaccuracies or typographical errors.
Renesas Technology Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (<http://www.renesas.com>).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corporation for further details on these materials or the products contained therein.

H8/3857 Series

Application Note

Cautions

1. Hitachi neither warrants nor grants licenses of any rights of Hitachi's or any third party's patent, copyright, trademark, or other intellectual property rights for information contained in this document. Hitachi bears no responsibility for problems that may arise with third party's rights, including intellectual property rights, in connection with use of the information contained in this document.
2. Products and product specifications may be subject to change without notice. Confirm that you have received the latest product standards or specifications before final design, purchase or use.
3. Hitachi makes every attempt to ensure that its products are of high quality and reliability. However, contact Hitachi's sales office before using the product in an application that demands especially high quality and reliability or where its failure or malfunction may directly threaten human life or cause risk of bodily injury, such as aerospace, aeronautics, nuclear power, combustion control, transportation, traffic, safety equipment or medical equipment for life support.
4. Design your application so that the product is used within the ranges guaranteed by Hitachi particularly for maximum rating, operating supply voltage range, heat radiation characteristics, installation conditions and other characteristics. Hitachi bears no responsibility for failure or damage when used beyond the guaranteed ranges. Even within the guaranteed ranges, consider normally foreseeable failure rates or failure modes in semiconductor devices and employ systemic measures such as fail-safes, so that the equipment incorporating Hitachi product does not cause bodily injury, fire or other consequential damage due to operation of the Hitachi product.
5. This product is not designed to be radiation resistant.
6. No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without written approval from Hitachi.
7. Contact Hitachi's sales office for any questions regarding this document or Hitachi semiconductor products.

Preface

The H8/300L Series of single-chip microcomputers are based on the high-speed H8/300L CPU, and integrate all peripheral functions necessary for system configuration.

The H8/300L CPU uses an instruction set which is compatible with the H8/300 CPU.

The H8/3857 Series are provided with such peripheral functions for system configuration as an LCD controller/driver, four different timers, a 14-bit pulse width modulator (PWM), a two-channel serial communication interface, and an A/D converter. These models can be used as microcomputers for embedded systems where LCD display is required.

These H8/3857 Series application notes include a “Basic Operation” section with operation examples when using the built-in peripheral functions of the H8/3857 Series independently. They are provided in the hope that they will be of use for software and hardware design.

Operation of the programs and circuits described in these application notes has been verified, but their operation must be confirmed by the user as well before actually being used.

Contents

Section 1	Guide to Using the H8/3857 Series Application Notes	1
1.1	Contents of Basic Operation.....	2
Section 2	Basic Operation	5
2.1	Dot-matrix LCD Control—Character Display Mode	5
2.2	Dot-matrix LCD Control—Graphic Display Mode.....	83

Section 1 Guide to Using the H8/3857 Series Application Notes

These application notes consist of two sections, as follows.

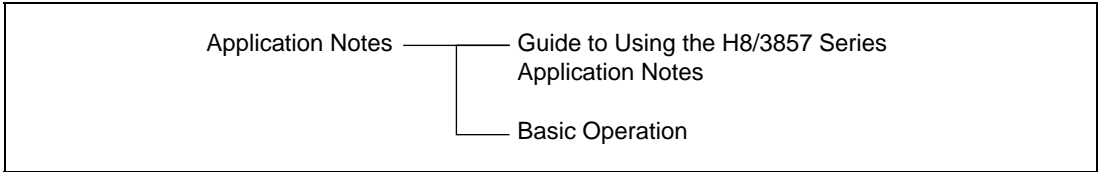


Figure 1 Contents of these Application Notes

Guide to Using the H8/3857 Series Application Notes

Explains how to use the H8/3857 Series application notes.

Basic Operation

Explains how to use the built-in peripheral functions of the H8/3857 Series through simple task examples.

1.1 Contents of Basic Operation

Basic Operation includes the sections shown below, explaining use of the built-in peripheral functions.

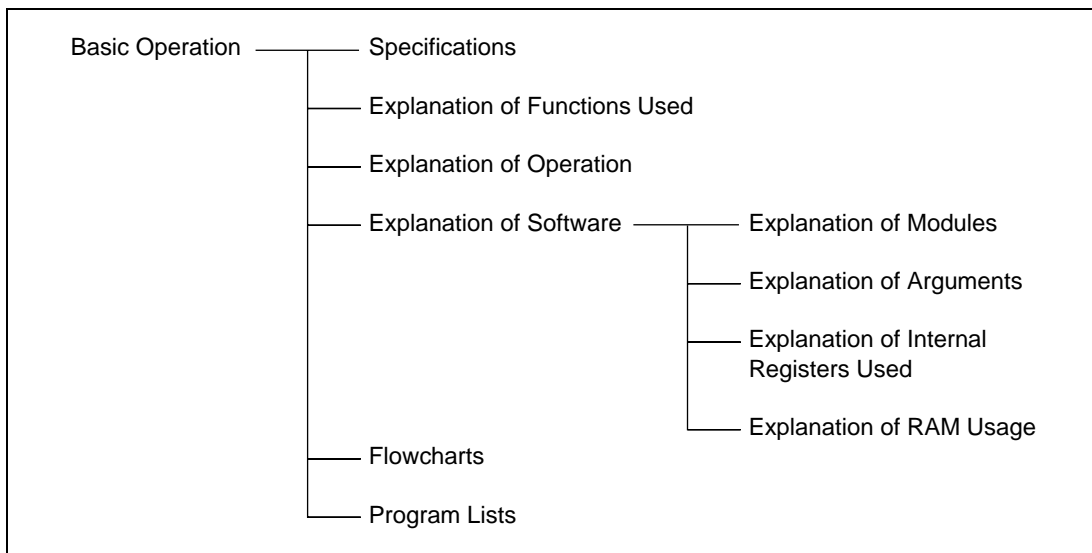


Figure 2 Contents of Basic Operation

Specifications

Explains system specifications for task examples.

Explanation of Functions Used

Explains the features of peripheral functions used in task examples, and allocation of the peripheral functions.

Explanation of Operation

Explains operation of task examples using timing charts.

Explanation of Software

1. Explanation of Modules

Explains the software modules used for operation in task examples.

2. Explanation of Arguments

Explains input arguments necessary for module execution, and arguments output following execution.

3. Explanation of Internal Registers Used

Explains internal registers such as a timer control register and serial mode register of peripheral functions used in modules.

4. Explanation of RAM Usage

Explains RAM label names and functions used in modules.

Flowcharts

Uses flowcharts to explain the software executed in task examples.

Program Lists

Gives program lists for software executed in task examples.

Section 2 Basic Operation

2.1 Dot-matrix LCD Control—Character Display Mode

Dot-matrix LCD Control— Character Display Mode	MCU: H8/3857 series	Functions Used: Dot-matrix LCD Controller
---	--------------------------------	--

Specifications

1. LCD display using a dot-matrix LCD controller function.
2. A 40 × 32 dot-matrix LCD is used in this task example.
3. A 1/5-bias, 1/32-duty display is used in character-display mode.
4. A four-line eight-character display with contrast control, scrolling, and blinking in selected areas.
5. Figure 1 shows an example of the LCD display.

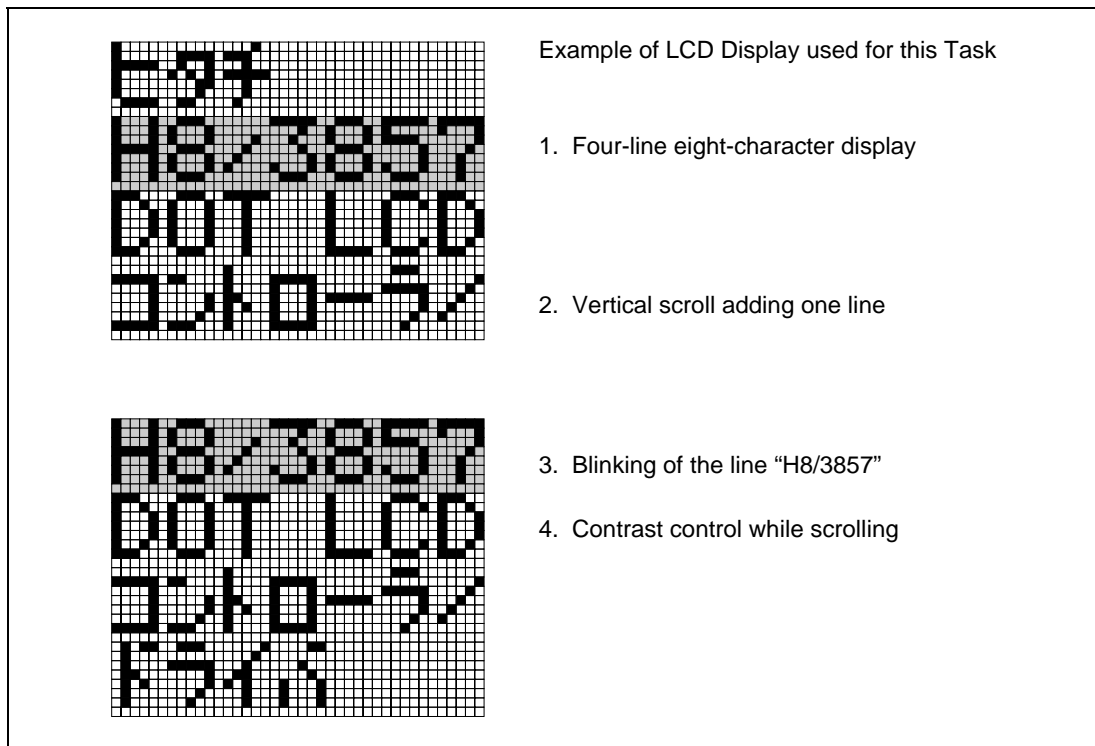


Figure 1 Example of LCD Display in this Task

Explanation of Functions Used

1. LCD display is implemented using a dot-matrix LCD controller in this example.

The features of the dot-matrix LCD controller are described below.

- Built-in bit-mapped display RAM (2048 bits). Maximum of 1280 display bits (choice of 40×32 bits, 56×16 bits, 64×8 bits, 40×16 bits, or 40×8 bits)
- Choice of 1/8, 1/16 or 1/32 duty
- Low power consumption enabling extended battery-powered operation (subclock operation and module standby)
- Built-in 2X or 3X LCD power supply step-up circuit
- Comprehensive display control functions (display data read/write, display on/off control, vertical display scrolling, arbitrary area blinking, and read-modify-write)
- CPU interface through I/O port interface.
- Built-in contrast-control circuit.

2. Figure 2 shows a block diagram of the dot-matrix LCD controller used in this example.

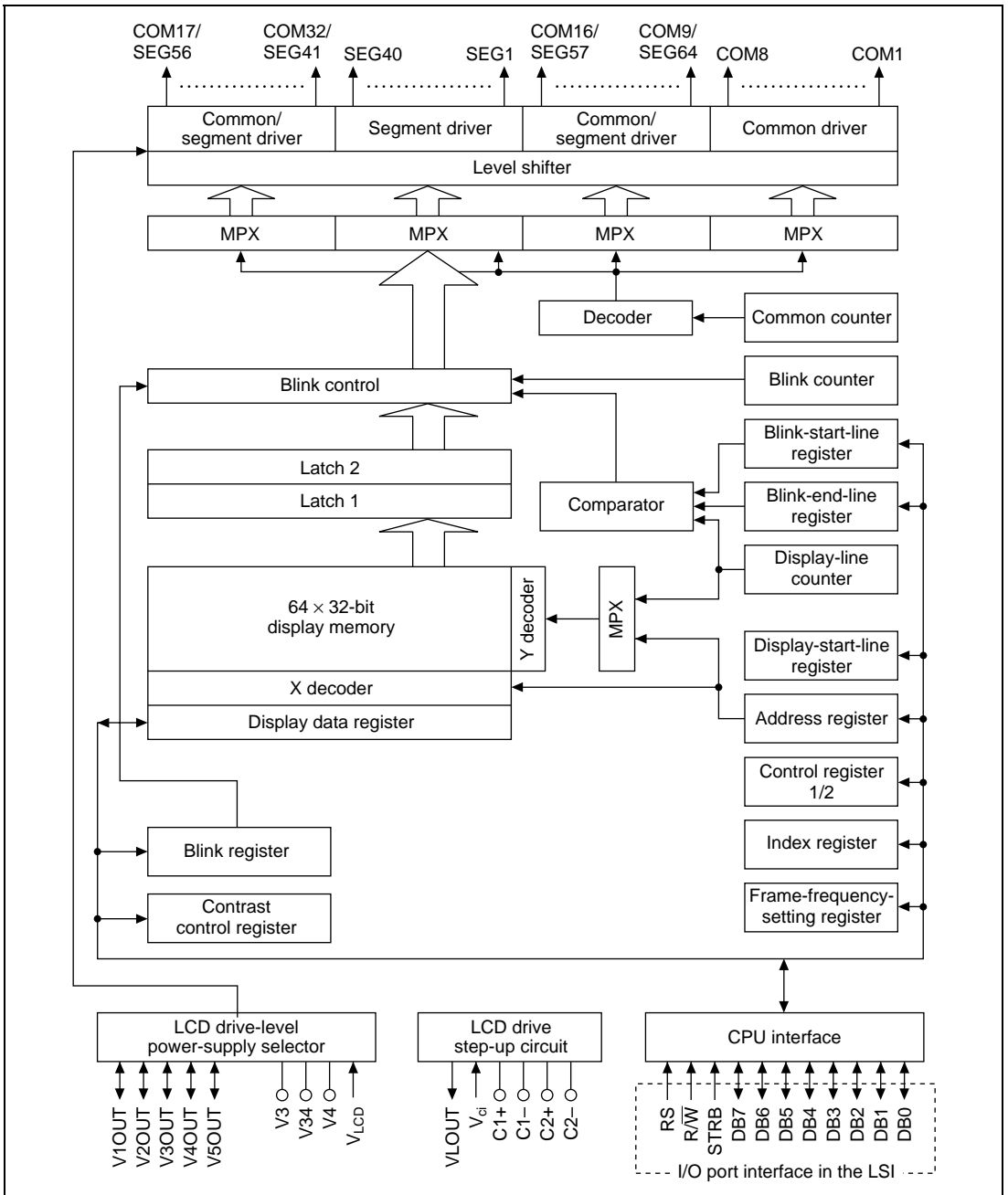


Figure 2 Block Diagram of the Dot-matrix LCD Controller

3. Table 1 describes the functions of the dot-matrix LCD controller

Table 1 Functions of the Dot-matrix LCD Controller

Index register (IR)

Function IR is an 8-bit write-only register that selects one of the LCD controller's ten control registers. IR is selected when RS is 0. Upon reset, IR is initialized to H'00.

Control register 1 (R0)

Function R0 is an 8-bit write-only register that sets the LCD module standby mode, controls the step-up circuit, switches between character display and graphic display, and selects the drive duty. Upon reset, R0 is initialized to H'00.

Control register 2 (R1)

Function R1 is an 8-bit write-only register that controls the on-off operation of the LCD display, the power-save mode and on-off operation of the op-amp circuit, the read-modify-write mode setting, and the selection of the address to be incremented in the display memory. Upon reset, R1 is initialized to H'00.

Address register (R2)

Function R2 is an 8-bit write-only register that sets the display-memory X- and Y-direction addresses accessed by the CPU. Upon reset, R2 is initialized to H'00.

Frame frequency setting register (R3)

Function R3 is an 8-bit write-only register that sets the frame frequency. Upon reset, R3 is initialized to H'00.

Display data register (R4)

Function R4 is an 8-bit read/write register used for read/write access to the display memory specified by XA2 to XA0 and YA4 to YA0 in R2. When the display memory is written to, the contents are written directly to the display memory via this register. When read, the data is temporarily latched into this register before being output to the bus. Upon reset, the display memory and R4 contents are undefined.

Display start line register (R5)

Function R5 is an 8-bit write-only register that specifies the line at which display starts. Upon reset, the display memory and R5 are undefined.

Blink register (R6)

Function R6 is an 8-bit write-only register that specifies blink areas. An area is made to blink by writing 1 to the corresponding bit in this register. There are no restrictions on which areas can blink simultaneously, and the entire screen can be made to blink by writing 1 to all the bits. The setting in this register is valid only when the BLK bit is set to 1. Upon reset, R6 is initialized to H'00.

Blink start line register (R8)

Function R8 is an 8-bit write-only register that specifies the start line of an area made to blink. Upon reset, R8 is initialized to H'00.

Table 1 Functions of the Dot-matrix LCD Controller (cont)

Blink end line register (R9)

Function R9 is an 8-bit write-only register that specifies the end line of an area made to blink. Upon reset, R9 is initialized to H'00.

Contrast control register (RA)

Function RA is an 8-bit write-only register that specifies the contrast control resistance value. Upon reset, RA is initialized to H'00.

Common output pins (COM1 to COM32)

Function These are LCD common drive output pins.

Segment output pins (SEG1to SEG64)

Function These are LCD segment drive output pins.

LCD bias setting pins (V3, V4)

Function These are LCD bias setting input pins.

LCD test pin (V34)

Function This is the internal resistance test pin. It is shorted to V3.

LCD step-up circuit capacitance connection pins (C1+, C1-, C2+, C2-)

Function These pins are used to connect external capacitances for LCD step-up.

LCD drive power supply level (V1OUT to V5OUT)

Function These are LCD drive power-supply level input/output pins.

LCD step-up circuit reference power supply (V_{ci})

Function This pin is an input pin for the reference input voltage to the LCD step-up circuit and step-up circuit power supply.

LCD step-up power supply output pin (VLOUT)

Function This is the LCD step-up voltage output pin.

LCD drive power supply (V_{LCD})

Function This is the LCD drive power supply input pin.

4. System overview of the dot-matrix LCD controller

The LCD controller operates at 1/32, 1/16, or 1/8 duty. The display size is a maximum of 40×32 dots (4 lines and 8 characters with a 5×8 -dot font). Because the LCD controller operates with a subclock used for display control, the time and date can be constantly displayed. This module includes a built-in 2X or 3X LCD power-supply step-up circuit, and an LCD system can be configured with just a few external parts (resistors and capacitors). Also, since data in the display RAM is retained even in module standby mode, and a step-up operation is not performed, low power consumption can be achieved without affecting the display.

Figure 3 shows a system block diagram of the LCD controller.

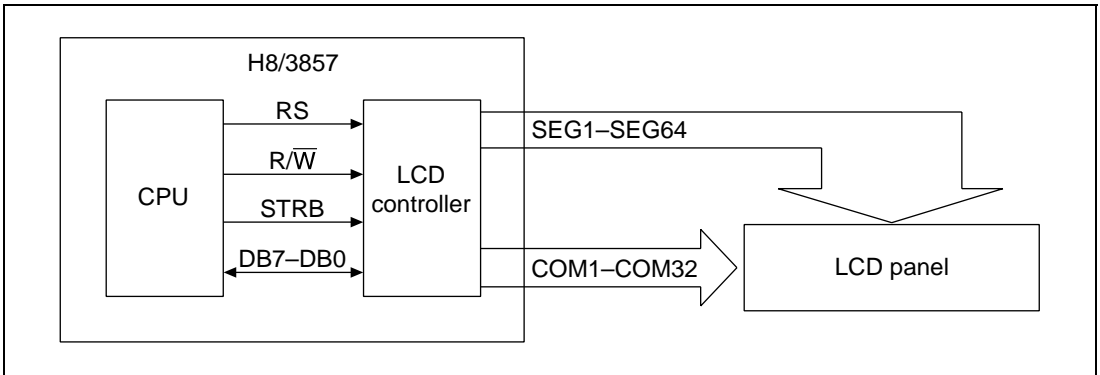


Figure 3 System Block Diagram of the LCD Controller

5. CPU interface

The LCD controller's registers are not included in the memory map of the H8/3857 CPU. They are controlled from the CPU by means of LSI internal LCD pins DB7 to DB0, RS, R/\overline{W} , and STRB via LSI internal I/O port 9 and port A. The pin configuration is shown in table 2, and an example of the timing of access to registers in the LCD controller is shown in figure 4.

Table 2 Configuration of LSI Internal LCD Pins

Pin Name	Symbol	I/O	Function
Data bus pins	DB7 to DB0	I/O	When $R/\overline{W} = 0$, these pins input data to be written to a register, when $R/\overline{W} = 1$, they output data read from a register
Register selection pin	RS	Input	When RS = 0, the index register is selected. When RS = 1, the control register is selected.
Read/write select pin	R/\overline{W}	Input	When $R/\overline{W} = 0$, write access is selected. When $R/\overline{W} = 1$; read access is selected
Strobe pin	STRB	Input	At the falling edge of STRB, read or write access as selected by R/\overline{W} , is performed on the register selected by RS.

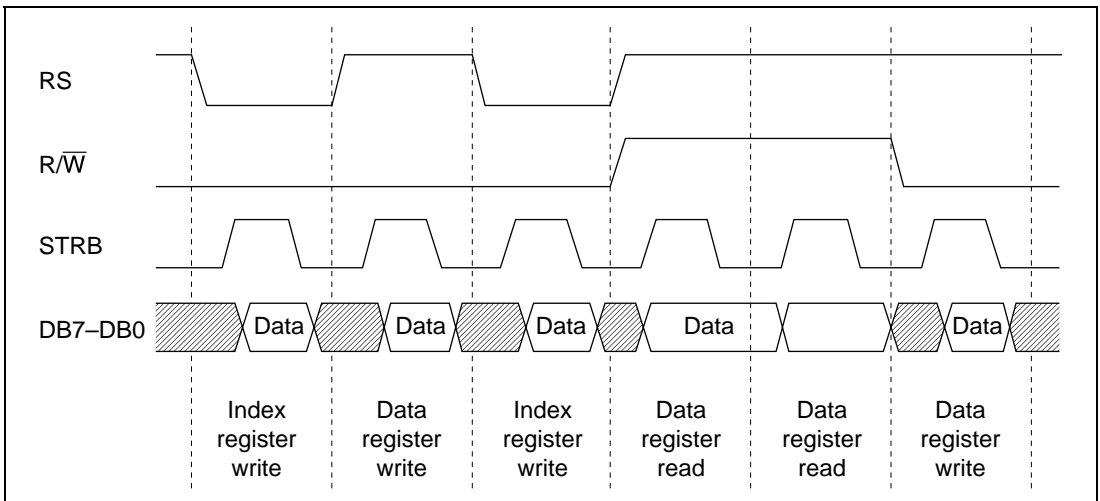


Figure 4 Timing Sequence Example for 8-Bit Data Transfer

a. Writing to the index register

When RS and R/\overline{W} are both cleared to 0, the data DB7 to DB0 is written to the index register (IR) at the falling edge of STRB. Do not change RS or R/\overline{W} at the fall of STRB.

b. Reading and writing to control registers

To access a control register, data indicating the number of the register to be accessed must be written to the index register (IR) beforehand. This data is shown in table 3. The register number written to the index register (IR) is retained until IR is written to again, so if the same control register is accessed repeatedly, it is not necessary to write to IR each time.

Table 3 Register Configuration

Register	Symbol	R/W	RS	Index Register			
				IR3	IR2	IR1	IR0
Index register	IR	W	0	—	—	—	—
Control register 1	R0	W	1	0	0	0	0
Control register 2	R1	W		0	0	0	1
Address register	R2	W		0	0	1	0
Frame frequency setting register	R3	W		0	0	1	1
Display data register	R4	R/W		0	1	0	0
Display start line register	R5	W		0	1	0	1
Blink register	R6	W		0	1	1	0
Blink start line register	R8	W		1	0	0	0
Blink end line register	R9	W		1	0	0	1
Contrast control register	RA	W		1	0	1	0

In a write to a control register, when RS has been set to 1 and R/\overline{W} cleared to 0, data DB7 to DB0 is written to the control register specified by the index register (IR) at the falling edge of STRB.

Except for the display data register (R4), control registers cannot be read. In a read of the display data register (R4), when data designating the display data register (R4) number is written to the index register (IR), and RS and R/\overline{W} are both set to 1, DB7 to DB0 are set to the output mode, and the display memory data at the address specified by the address register (R2) is output from DB7 to DB0 at the rising edge of STRB. If a read is also performed in the next cycle, the data output is held until the next rise of STRB. However, if a write is performed in the next cycle, DB7 to DB0 are set to the input mode from the point at which R/\overline{W} is cleared to 0, and the output is cleared.

In either case, do not change RS or R/\overline{W} at the falling edge of STRB.

6. LCD drive pin functions (character display mode)

Among the LCD controller's LCD drive outputs, COM9 to COM32 and SEG64 to SEG41 are switched according to the display duty and display mode.

Table 4 shows the switches of Common/Segment output for the character display mode (SOB = 0).

Table 4 Switches of Common/Segment Output in the Character Display Mode

Pin Name	SOB = 0 (Character Display Mode)		
	1/8 Duty	1/16 Duty	1/32 Duty
COM1 to COM8	COM1 to COM8	COM1 to COM16	COM1 to COM16
COM9/SEG64 to COM16/SEG57	Common signal non-selection waveform		
SEG1 to SEG40	SEG1 to SEG40	SEG1 to SEG40	SEG1 to SEG40
COM32/SEG41 to COM25/SEG48	Common signal non-selection waveform	COM16 to COM1	COM32 to COM17
COM24/SEG49 to COM17/SEG56	COM8 to COM1		

7. Display memory configuration and display (character display mode)

The LCD controller includes a 64×32 -bit bit-mapped display memory. The display memory configuration can be a 5-bit \times 8 or an 8-bit \times n (n = 5, 7, or 8) X-direction configuration with a 32-bit Y-direction configuration. The display data written from the CPU is stored horizontally in the display memory with the MSB at the left and the LSB at the right as shown in figure 5. On the display, data 1 corresponds to be lit (black) and data 0 to be unlit (achromatic).

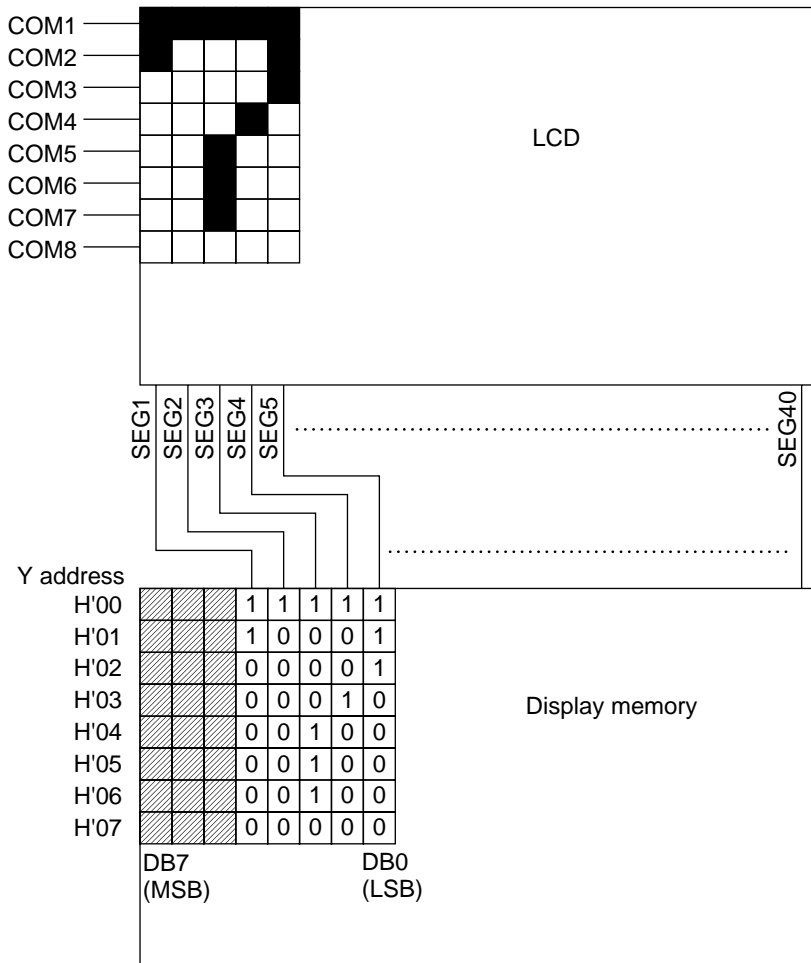


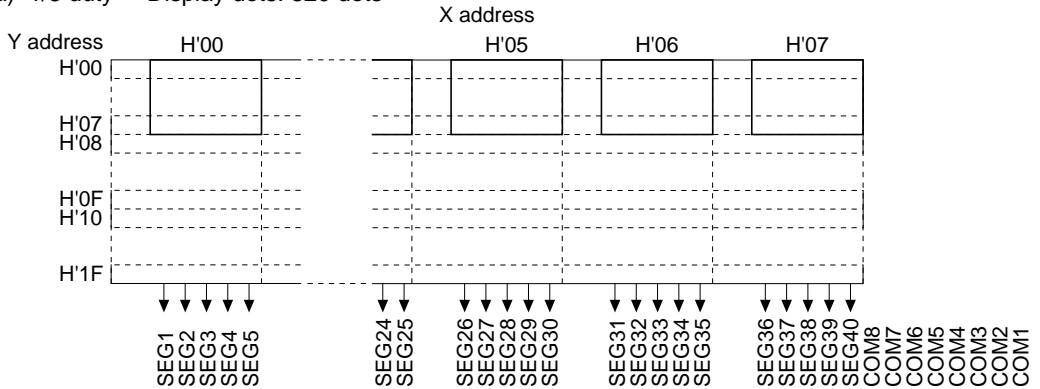
Figure 5 Memory Data and Display in Character Display Mode

8. Display data output (character display mode)

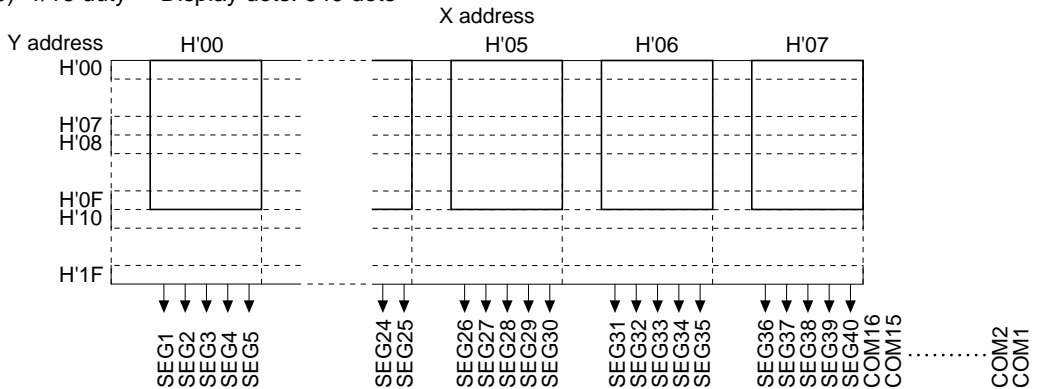
The LCD controller has a character display mode (SOB = 0) in which only 5 bits of each display-data byte can be output to enable efficient 5-dot × 8-dot character output. It also has a graphic display mode (SOB = 1) in which all the bits of a data byte can be output to provide an efficient full-dot graphic display.

The relationship between the display duty and the output pins in the character display mode (SOB = 0) is shown in figure 6.

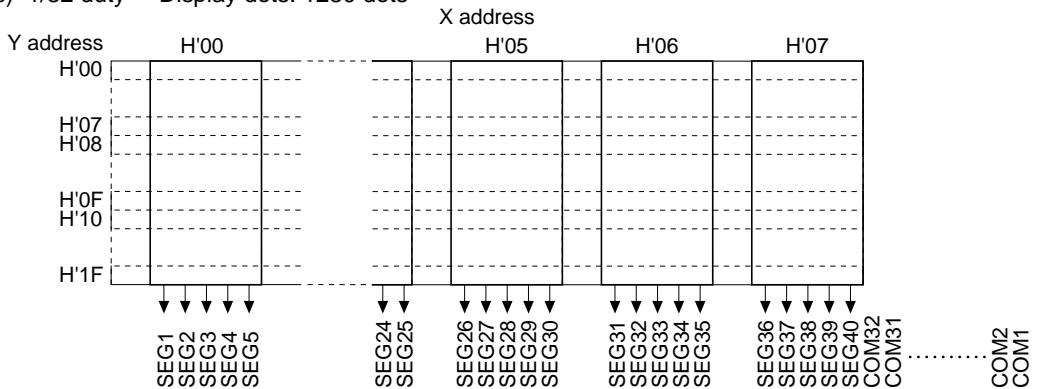
(a) 1/8 duty Display dots: 320 dots



(b) 1/16 duty Display dots: 640 dots



(c) 1/32 duty Display dots: 1280 dots



□ : Valid display data area

Figure 6 Display Duty and Valid Display Data Area in Character Display Mode

9. Register and display memory access

a. Register access

To access a register, RS is first cleared to 0, and the number of the register to be accessed is set in the index register. Then RS is set to 1, enabling the specified register to be accessed. Some internal registers have nonexistent bits; 0 must be written to these bits. The display data register (R4) is the only register that can be read.

b. Display memory access

To access the display memory, the address to be accessed is set in the address register (R2). The memory is then accessed via the display data register (R4). This access can be performed without awareness of the display-side read. Figure 7 shows the procedure under 1/32 duty.

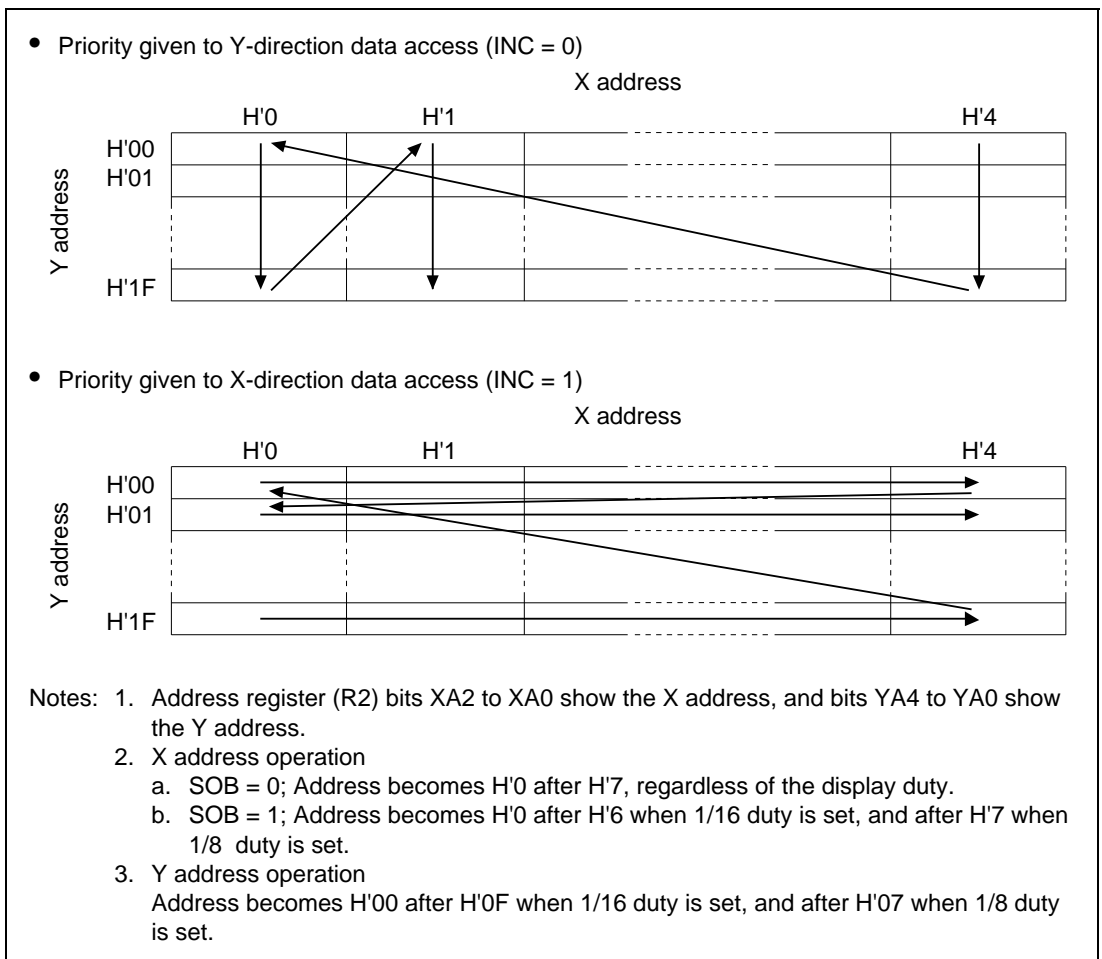


Figure 7 Display Memory Access Methods (1/32 Duty)

After each display data register (R4) access, the X and Y addresses are automatically incremented on the basis of the value set in the INC bit in control register 2 (R1), therefore, the address need not be set each time.

When 1/32 duty (DDTY1 = 0, DDTY0 = 0), SOB is 1, and INC is 0 in control register 1 (R0), the X address remains the same for each read/write access to the display data register (R4), while the Y address is automatically incremented up to H'1F. After reaching H'1F, the Y address returns to H'00, and the X address is simultaneously incremented. If INC is 1, on the other hand, the Y address remains the same for each read/write access to the display data register (R4), while the X address is automatically incremented up to H'4. After reaching H'4, the X address returns to H'0, and the Y address is simultaneously incremented. In this way, consecutive read/write access can be made to the entire display memory area.

c. Reading for display

The LCD controller has a dual-port display RAM, and accesses from the CPU and reads for the LCD display are independent of each other. This allows flexible interfacing.

Figure 8 shows the memory read procedure.

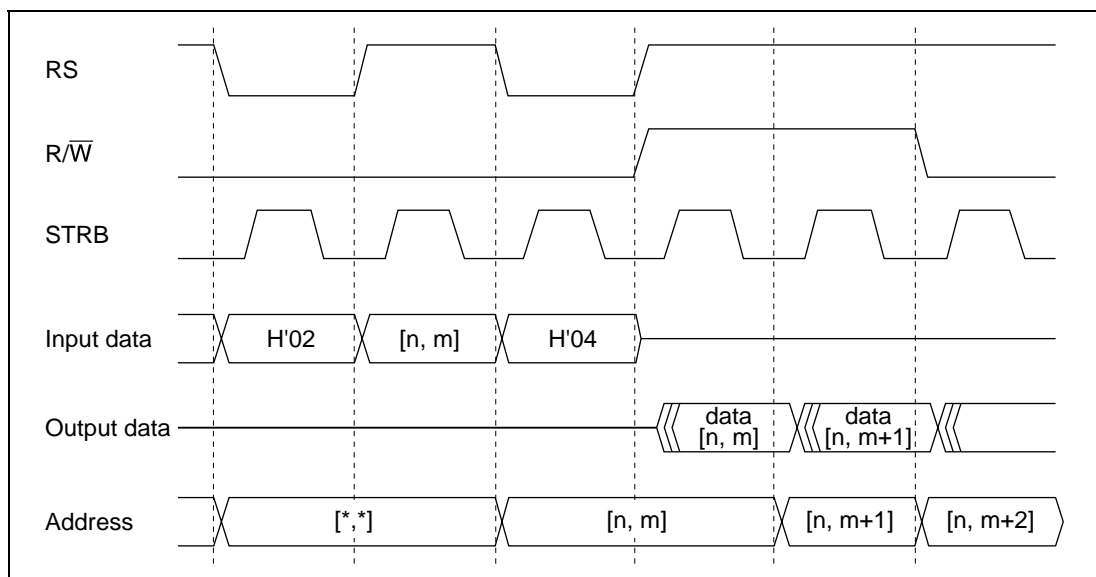


Figure 8 Memory Read Procedure

d. Read-modify-write mode

In the normal state, the X or Y address is incremented after both read and write accesses to the display memory. In the read-modify-write mode, the address is incremented only after a write, and remains the same after a read. By using this mode, previously written data can be read, processed then written back to the same address.

Figure 9 shows the read-modify-write mode flowchart.

When 1/32 duty (DDTY1 = 0, DDTY0 = 0), SOB is 1, and INC is 0 in control register 1 (R0), the X address remains the same for each read/write access to the display data register (R4), while the Y address is automatically incremented up to H'1F. After reaching H'1F, the Y address returns to H'00, and the X address is simultaneously incremented. If INC is 1, on the other hand, the Y address remains the same for each read/write access to the display data register (R4), while the X address is automatically incremented up to H'4. After reaching H'4, the X address returns to H'0, and the Y address is simultaneously incremented. In this way, consecutive read/write access can be made to the entire display memory area.

e. Reading for display

The LCD controller has a dual-port display RAM, and accesses from the CPU and reads for the LCD display are independent of each other. This allows flexible interfacing.

Figure 9 shows the memory read procedure.

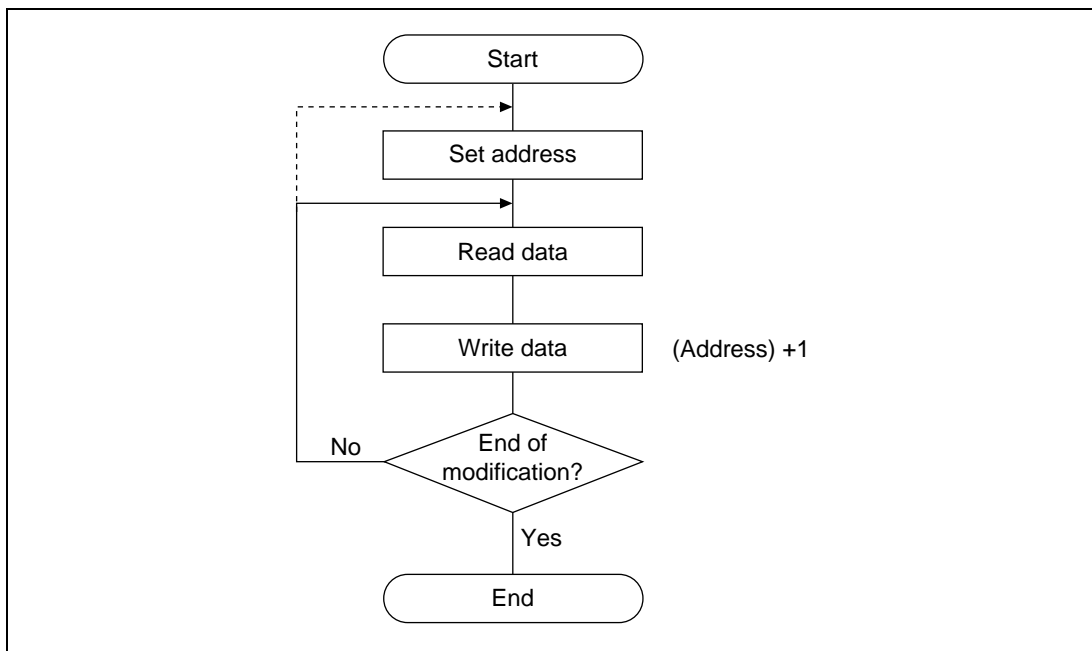


Figure 9 Read-Modify-Write Mode Flowchart

10. Scroll function

The LCD controller allows the user to vertically scroll any number of lines by specifying the display start line. Figure 10 shows the relationship, for the 1/16 duty and 1/8 duty settings, between the display memory and the Y address, and the display memory and the LCD display after scrolling. If the display start address is set to 1, data set at Y address 0 is displayed in the 16th line. Therefore, when the display is scrolled to show the next screen, the data at Y address 0 must be rewritten with the following display data. This data update must be carried out after the display is scrolled.

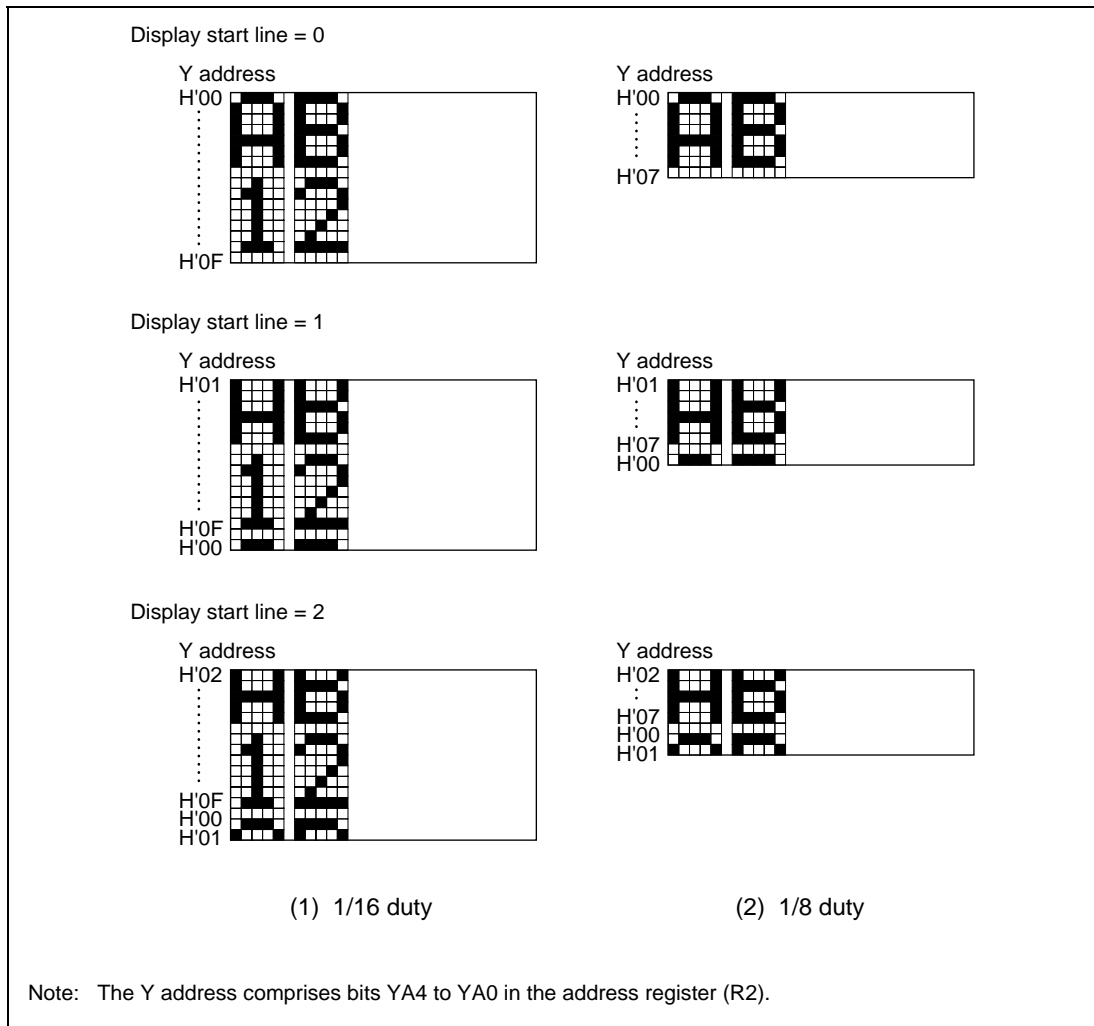


Figure 10 Vertical Scrolling

11. Blink function

The LCD controller can provide a blinking display in any area. With an 80-Hz frame frequency, the display goes on and off in a cycle of approximately 1.6 seconds.

To set a blink area, the line unit is specified in the horizontal direction by means of the blink start line register (R8) and the blink end line register (R9), while in the vertical direction a 5-bit unit (SOB = 0) or 8-bit unit (SOB = 1) is set in the blink register (R6). When 1 is set in the blink register (R6), blinking of the corresponding dot is controlled. After these register settings are made, blinking is started by setting the BLK bit in the control register 2 (R1) to 1. The blink area is designated by an absolute specification with respect to the display memory, so when the display is scrolled, the blink area shifts accordingly.

Figure 11 shows the blink register (R6) and the blinking locations, and figure 12 shows the blinking during display scrolling.

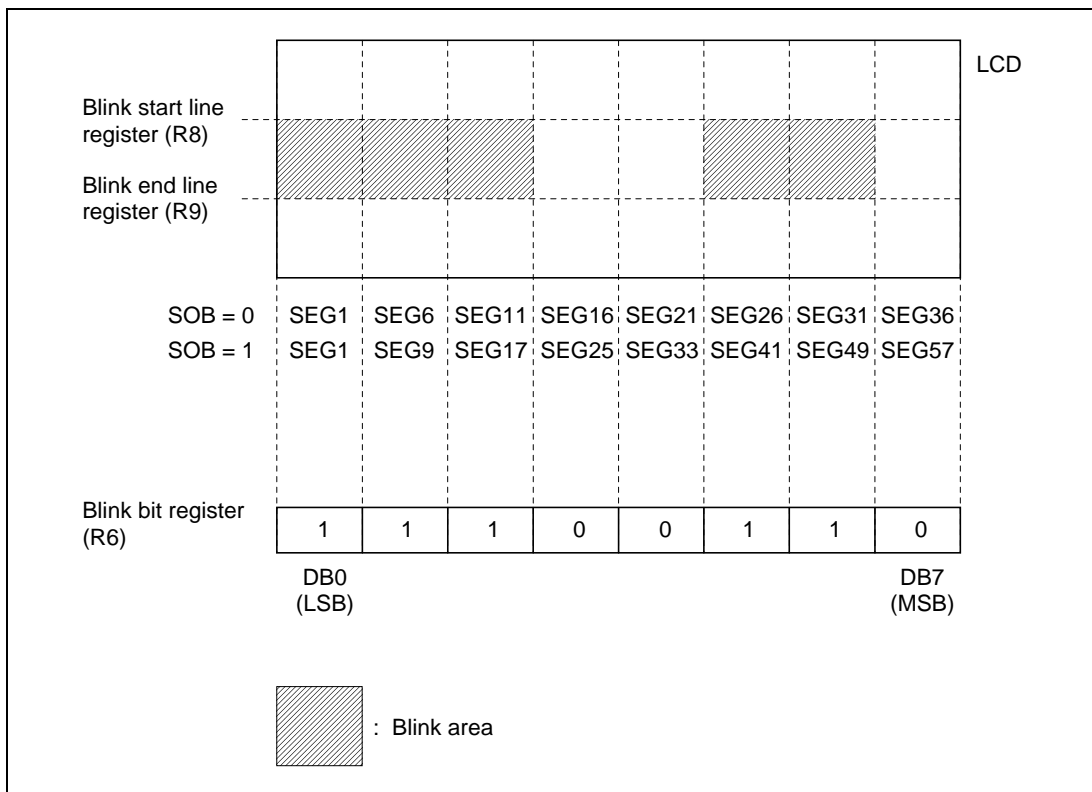
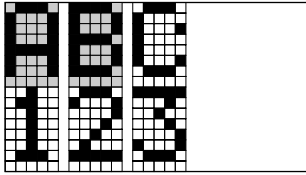
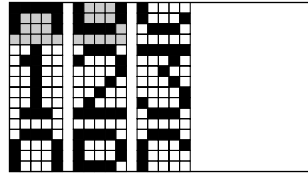


Figure 11 Blink Register (R6) and Blink Locations



Display start line = 0
Blink start line = 0
Blink end line = H'7



Display start line = H'4
Blink start line = H'4
Blink end line = H'7

Figure 12 Blinking during Display Scrolling (SOB = 0, 1/16 duty)

12. Module standby mode

The LCD controller has a module standby function that lowers power consumption. In the module standby mode, the built-in step-up circuit and the op-amps become inactive, and segment and common outputs go to the V_{ss} (display-off state) level. Display RAM and internal register data is retained, except for the DISP and OPON bits in control register 2 (R1). The control registers can still be accessed in the module standby state. Figure 13 shows the procedures for initiating and clearing the module standby mode. The initiation and clearing procedures must be followed exactly to protect the display memory contents. To place the CPU in standby mode, set the LSBY bit in control register 1 (R0) to 1 before executing the standby instruction. After clearing the standby mode, follow the module standby clearing procedure to start the display.

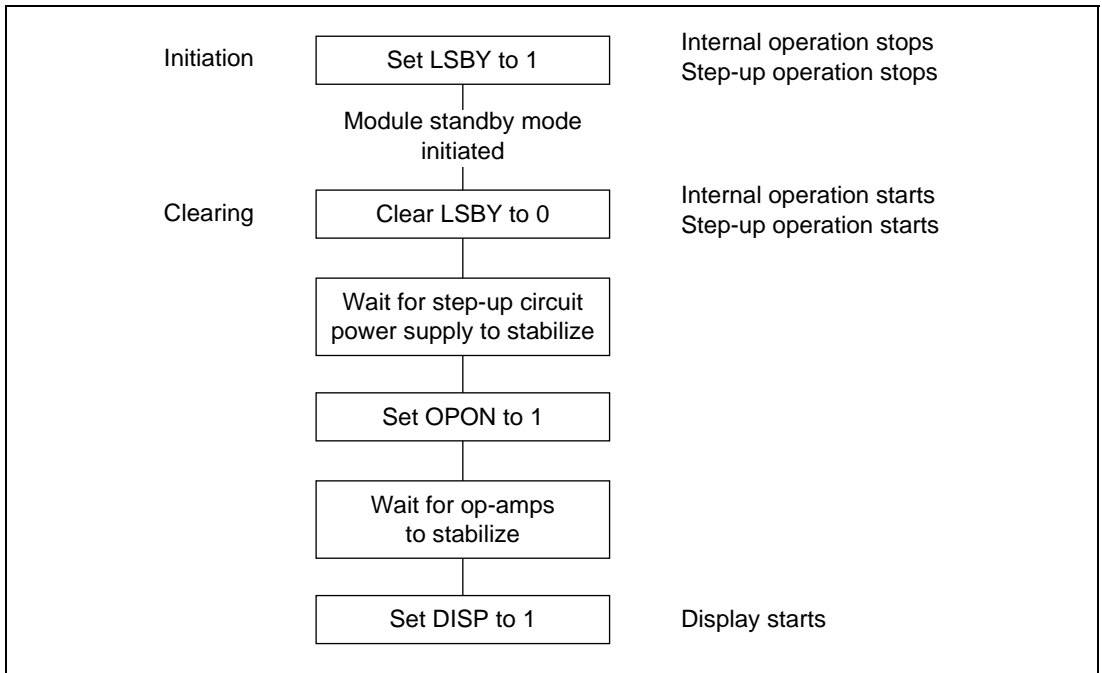


Figure 13 Module Standby Mode Initiation and Clearing Procedures

13. Power-on and power-off procedures

As the LCD controller includes a complete power supply circuit, the procedures shown in figure 14 must be followed when powering on and off. Failure to follow these procedures may result in an abnormal display.

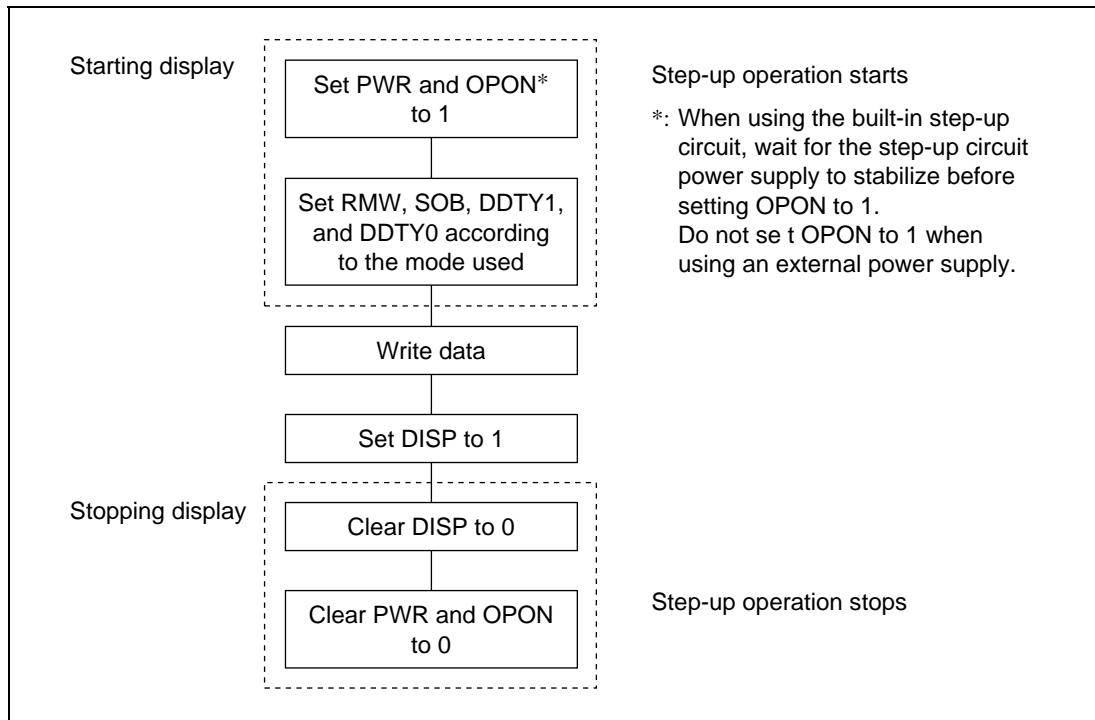


Figure 14 Power-On and Power-Off Procedures

14. Power supply circuit

The LCD controller has a built-in 2X or 3X step-up circuit for the LCD drive. In the standby mode, the power-supply circuit is automatically turned off after a maximum of two subclock cycles, and the power consumption of the step-up circuit falls to zero. The power-supply circuit can be turned on and off by a command, and an external power-supply circuit must be used if the current capacity of the built-in step-up circuit is insufficient.

a. Step-up circuit

When the reference voltage is input to V_{ci} ($V_{ci} \leq V_{CC}$), a capacitor is connected between V_{SS} and V1OUT, C1+ and C1–, and C2+ and C2–, and the PWR bit in control register 1 (R0) is set to 1, the potential between V_{ci} and V_{SS} is doubled or tripled. The subclock is used for voltage step-up, so step-up will not be done unless the subclock is supplied. Since V_{ci} is also used for the step-up circuit power supply, an adequate current must be provided.

If the step-up circuit is not used, connect V_{ci} to V_{CC} .

b. LCD drive level power supply

Six power supply levels—V1, V2, V3, V4, V5, and V_{SS} —are needed for the LCD drive. These levels are normally generated by means of resistive division. This power-supply circuit includes a voltage follower op-amp for each voltage level generated by the resistive division. These op-amps enable lower power consumption by limiting the driving current when using the OPS bit. In this case, insert a capacitor of approximately 0.1 to 0.5 μF between each of the V1OUT to V5OUT pins and V_{SS} .

When a 1/4 bias is used for the LCD display, the V3 and V4 pins must be shorted; when a 1/5 bias is used, the V3 and V4 pins must be left open. The V34 pin is an internal resistance test pin, and must always be externally shorted to the V3 pin.

c. Contrast control

The LCD controller provides two methods of contrast control.

- Using the built-in contrast control circuit

The LCD controller includes a programmable contrast control circuit. The LCD power-supply voltage can be adjusted for a given step-up voltage through the contrast control register (RA).

- The step-up voltage can be varied by changing the step-up reference voltage V_{ci} .

d. External power supply

- When an external power supply is input to V_{LCD}

V1 to V5 can be generated by inputting an external power supply to V_{LCD} , and using the built-in op-amps by setting the OPON bit in control register 2 (R1) to 1. The V_{LCD} input level must be between V_{CC} and 7.0 V.

- When an external power supply is input directly to V1 to V5

A power supply can be applied directly to V1, V2, V3, V4 and V5 from an external source by clearing the PWR bit in control register 1 (R0) and the OPON bit in control register 2 (R1) to 0, to stop the built-in step-up circuit and cut the built-in op-amp power supply. A potential equal to V1 must be input to V_{LCD} . The input level of V_{LCD} and V1 must be between V_{CC} and 7.0 V.

In either case, inputting a voltage exceeding the maximum rated voltage may reduce the reliability of the LSI.

15. LCD drive power-supply voltages

There are six LCD drive power-supply voltage values: V1 to V5, and V_{SS}. V1 is the highest voltage, and V_{SS} is the lowest. As shown in figure 15, the common waveforms are formed from a combination of V1, V2, V5, and V_{SS}, while the segment waveforms are formed from a combination of V1, V3, V4, and V_{SS}. V1 and V_{SS} are shared by both common and segment waveforms, but the intermediate voltages are different

In figure 15, waveform outputs of SEG1 to SEG40 differ according to the display data. In this example, the LCD panel lines to which COM1 is connected are lit, and all other dots are unlit.

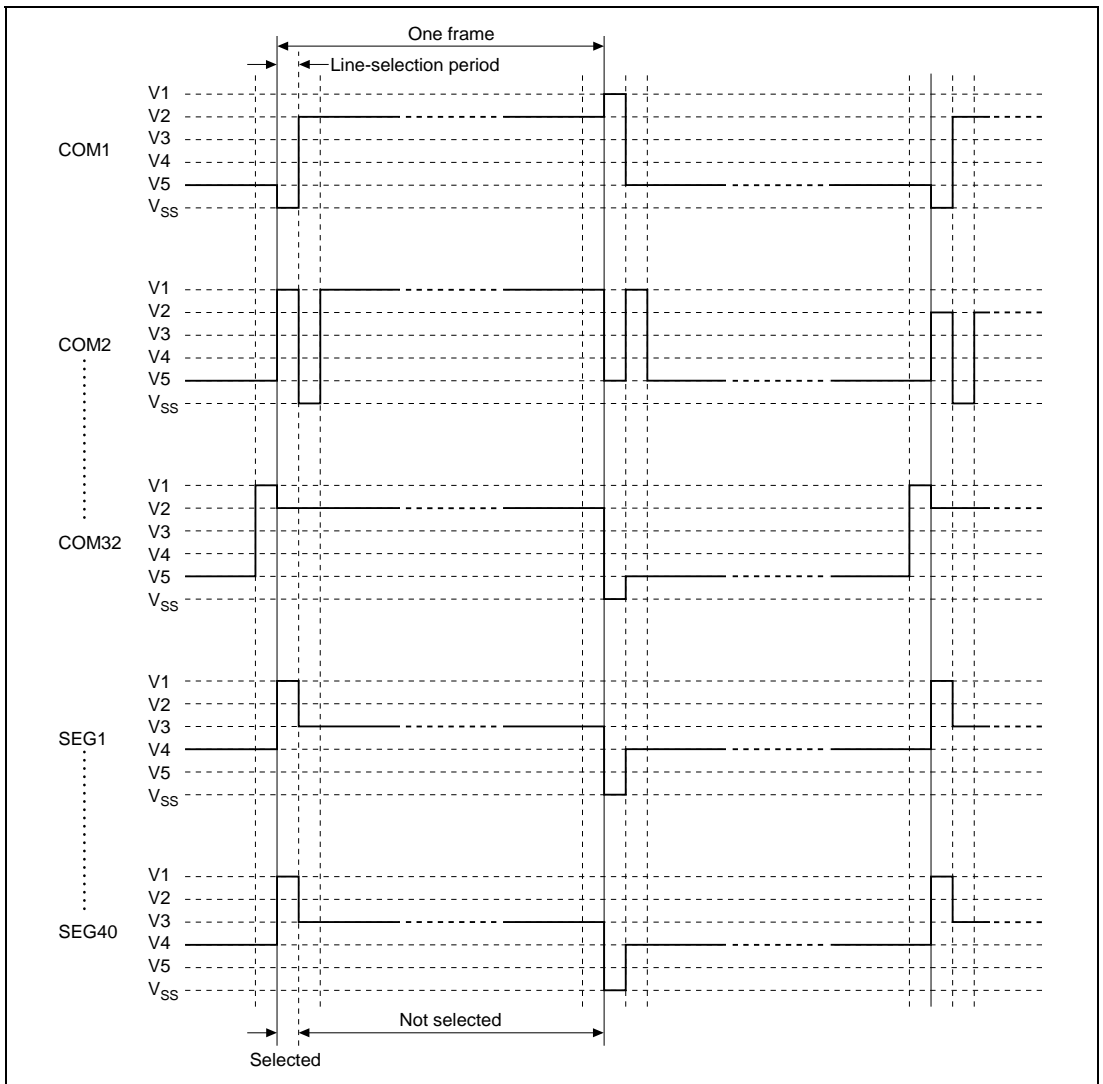
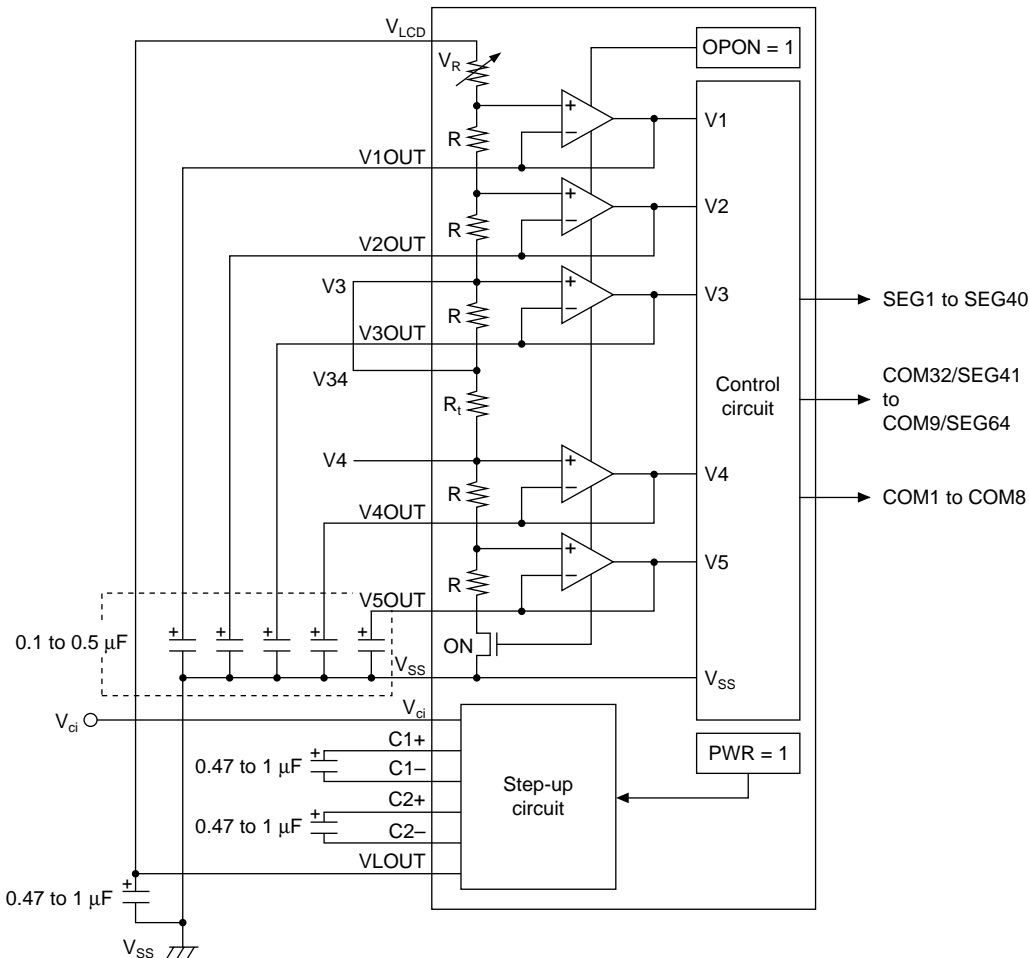


Figure 15 LCD Drive Voltage Waveform (1/32 duty)

16. LCD voltage-generation circuit

Figure 16 shows an example of the connection when the built-in step-up circuit is used (3X step-up, 1/5 bias). The LCD controller includes bleeder resistances that generate levels V1 to V5, and voltage follower op-amp circuits. When the built-in op-amps are used, the OPON bit in control register 2 (R1) must be set to 1. Contrast can be controlled through the software by using the contrast-control register (RA). Capacitors of about 0.1 to 0.5 μF must be inserted between each of the V1OUT to V5OUT built-in op-amp outputs and V_{SS} to provide stabilization. Capacitors must be inserted when a power-saving function is set by the OPS bit. For the op-amps to operate normally, the contrast control register (RA) must be set so that the potential difference between V_{LCD} and V1, and between V5 and V_{SS} , is at least 0.4 V. Since V_{ci} is also used for the step-up circuit power supply, ensure that an adequate current can be supplied when adjusting the reference voltage. The V_{ci} input level must not exceed V_{CC} .



- Notes:
1. R_t is a test resistance. The V3 and V34 pins must be shorted.
 2. The output voltage after step-up must not exceed 7.0 V. With 3X step-up, in particular, do not input a voltage of 2.3 V or above as the reference voltage (V_{ci}).
 3. V_{ci} is also used for the step-up circuit power supply. Use a transistor, etc., for current amplification to ensure an adequate LCD drive current.
 4. The V_{ci} input level must not exceed V_{CC}.
 5. Care is required with a connection using polarized capacitors.

Figure 16 Example of Connections When Using a Built-in Step-up Circuit (3X Step-up, 1/5 Bias)

17. Contrast-control circuit

Contrast control can be done by software (electronic control) by controlling the LCD drive voltage (the potential difference between V_{LCD} and $V1$) with the contrast-control register (RA). Variable resistance value V_R can be adjusted within $0.1 R$ to $1.6 R$, where R is the value of the basic dividing bleeder resistance between V_{LCD} and $V1$. The contrast-control settings using bits CCR3 to CCR0 in the contrast-control register (RA) are described in table 5.

To ensure stable operation of the voltage follower op-amp circuits that output levels $V1$ to $V5$, the contrast-control register (RA) must be set so that the potential difference between V_{LCD} and $V1$, and between $V5$ and V_{SS} , is at least $0.4 V$.

The contrast control ranges are given in table 6. If contrast control cannot be adequately performed by means of on-chip resistance V_R , it can be done by inserting a resistance between V_{LOUT} and V_{LCD} .

Table 5 Contrast Control Settings

Contrast-Control Register (RA)				Variable Resistance (V_R)	$V1-V_{SS}$ Potential Difference	Display Color
CCR3	CCR2	CCR1	CCR0			
0	0	0	0	1.6R	↑ ↓	↑ ↓
			1	1.5R		
		1	0	1.4R		
		1	1	1.3R		
	1	0	0	1.2R		
			1	1.1R		
		1	0	1.0R		
			1	0.9R		
			0	0.8R		
			1	0.7R		
1	0	0	0	0.6R		
			1	0.5R		
		1	0	0.4R		
			1	0.3R		
	1		0	0.2R		
			1	0.1R		

Table 6 Contrast-Control Range

Bias	LCD Drive Voltage (VDR)	Contrast-Control Range
1/5 bias drive	$\frac{5 \times R}{5 \times R + V_R} \times (V_{LCD} - V_{SS})$	<ul style="list-style-type: none"> • LCD Drive voltage adjustment range: $0.758 \times (V_{LCD} - V_{SS}) \leq V_{DR} \leq 0.980 \times (V_{LCD} - V_{SS})$ • $V_5 - V_{SS}$ potential difference limit: $\frac{R}{5 \times R + V_R} \times (V_{LCD} - V_{SS}) \geq 0.4 [V]$ • $V_{LCD} - V_1$ potential difference limit: $\frac{V_R}{5 \times R + V_R} \times (V_{LCD} - V_{SS}) \geq 0.4 [V]$
1/4 bias drive	$\frac{4 \times R}{4 \times R + V_R} \times (V_{LCD} - V_{SS})$	<ul style="list-style-type: none"> • LCD Drive voltage adjustment range: $0.714 \times (V_{LCD} - V_{SS}) \leq V_{DR} \leq 0.976 \times (V_{LCD} - V_{SS})$ • $V_5 - V_{SS}$ potential difference limit: $\frac{R}{4 \times R + V_R} \times (V_{LCD} - V_{SS}) \geq 0.4 [V]$ • $V_{LCD} - V_1$ potential difference limit: $\frac{V_R}{4 \times R + V_R} \times (V_{LCD} - V_{SS}) \geq 0.4 [V]$

18. LCD drive bias selection circuit

The bias value that gives the best contrast is calculated using the equation below. If a lower bias value is used, contrast will deteriorate, but the LCD drive voltage (the potential difference between V_1 and V_{SS}) can be kept low. If the LCD drive voltage is inadequate even with a low V_{ci} voltage and use of the 3X step-up circuit, or if the output voltage falls and the LCD display becomes faint as batteries wear out, for instance, the display can be made clearer by decreasing the LCD drive bias.

$$\text{Optimum bias value for } 1/N \text{ duty drive} = \frac{1}{(N)^{(1/2)} + 1}$$

- Notes:
1. When using 1/5 bias, leave the V3 and V4 pins open.
 2. When using 1/4 bias, short the V3 and V4 pins.
 3. The V3 and V34 pins must always be shorted.

19. Frame frequency

The LCD controller controls the subclock division ratio by setting FS5 to FS0 in the frame frequency setting register (R3), and sets the F frame frequency of the LCD. The relationship between the LCD frame frequency f_F (Hz), the subclock frequency f_W (Hz), the division ratio r , and the LCD duty $1/N$ is as follows.

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

Set a division ratio suitable for the characteristics of the LCD panel used. Table 7 shows the relationship between register setting values and division ratios. Table 8 lists examples of subclock frequency, LCD duty, division ratio setting, and frame frequency.

Table 7 Register Setting Values and Division Ratios.

FS						Divi- sion ratio	FS						Divi- sion ratio	FS						Divi- sion ratio							
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	0	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

Table 8 Sample Frame Frequency Settings

Display Duty 1/N		Subclock Frequency (kHz)		
		32.768	38.4	76.8
1/8	Division ratio r	48	56	112
	Frame frequency f_F (Hz)	85.3	85.7	85.7
1/16	Division ratio r	24	28	56
	Frame frequency f_F (Hz)	85.3	85.7	85.7
1/32	Division ratio r	12	14	28
	Frame frequency f_F (Hz)	85.3	85.7	85.7

Explanation of Operation

1. The basic operation for this task

a. The display memory and display data setting

The display memory and display data is shown in table 9. An LCD display is created by specifying the X-direction and Y-direction addresses to the address register (R2) and writing the display data to the display data register (R4).

Table 9 Display Data and Display Memory

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

Notes: A: Before scrolling
B: After scrolling

b. Basic operation

The basic operation for this task is shown in figure 17. An LCD display in the character display mode is created through the software and hardware processing shown in figure 17.

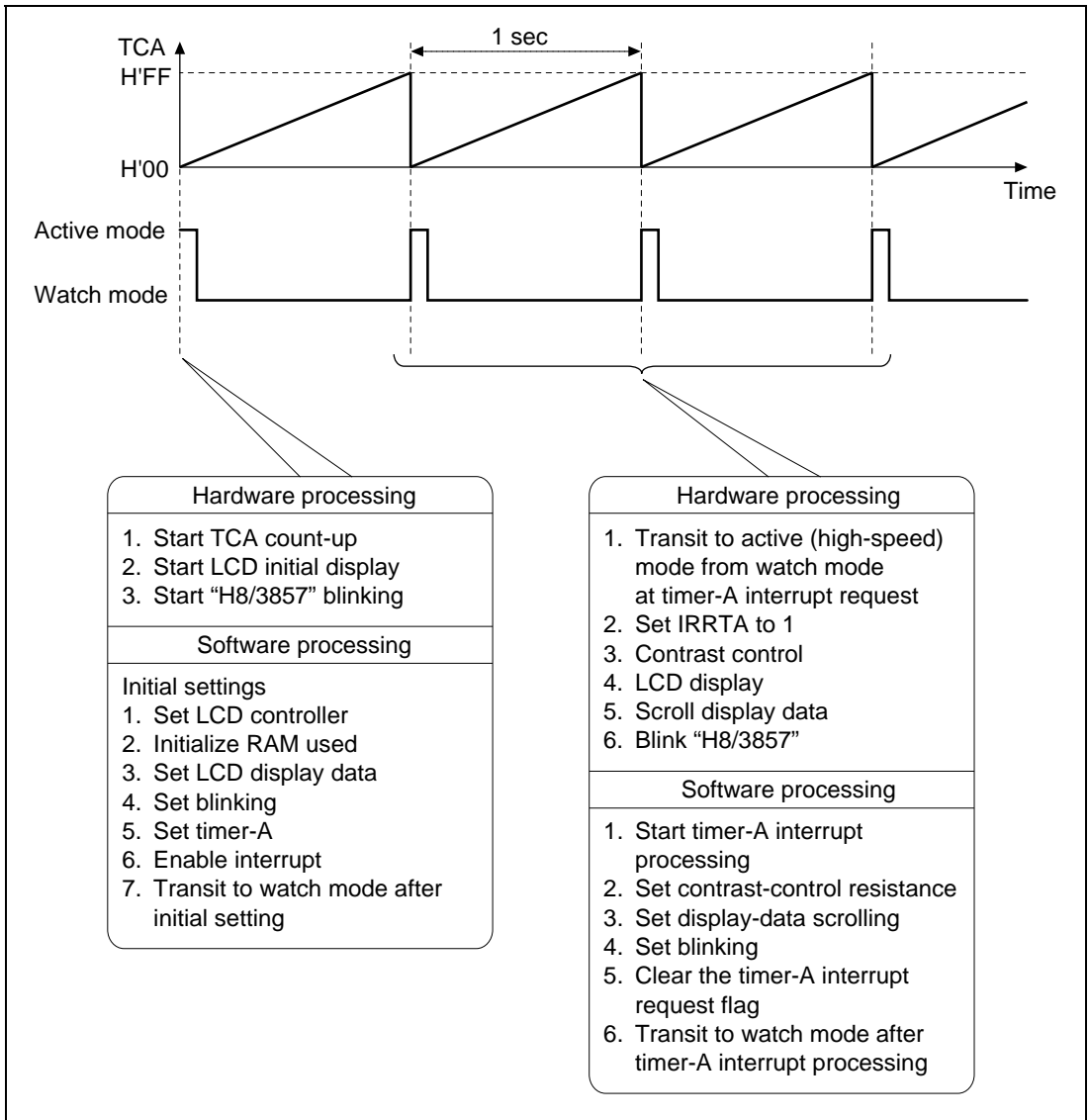


Figure 17 Basic Operation

Explanation of Software

1. Modules

Table 10 describes the modules in this task example

Table 10 **Modules**

Module	Label	Function
Main routine	INIT	Initializes the stack pointer, RAM used, the dot-matrix LCD controller, and timer-A, writes display data in the display memory, enables interrupt, and transits to watch mode.
Index register selection routine	INDEX	Selects the index register (IR)
LCD control-register write routine	WRDATA	Writes the data in the control register selected by IR.
Timer-A interrupt processing routine	TMRA	Scrolls display data and controls blinking.

2. Arguments

An argument is not used in this task

3. Internal Registers Used

Table 11 describes the internal registers used in this task example.

Table 11 Internal Registers Used

Register	Description	IR Setting	Setting Value
IR	IR3 to IR0 Index register (Index registers 3 to 0) Select one of ten control registers of the LCD controller. <ul style="list-style-type: none"> • R0 is selected when IR3 = 0, IR2 = 0, IR1 = 0, and IR0 = 0 • R1 is selected when IR3 = 0, IR2 = 0, IR1 = 0, and IR0 = 1 • R2 is selected when IR3 = 0, IR2 = 0, IR1 = 1, and IR0 = 0 • R3 is selected when IR3 = 0, IR2 = 0, IR1 = 1, and IR0 = 1 • R4 is selected when IR3 = 0, IR2 = 1, IR1 = 0, and IR0 = 0 • R5 is selected when IR3 = 0, IR2 = 1, IR1 = 0, and IR0 = 1 • R6 is selected when IR3 = 0, IR2 = 1, IR1 = 1, and IR0 = 0 • R8 is selected when IR3 = 1, IR2 = 0, IR1 = 0, and IR0 = 0 • R9 is selected when IR3 = 1, IR2 = 0, IR1 = 0, and IR0 = 1 • RA is selected when IR3 = 1, IR2 = 0, IR1 = 1, and IR0 = 0 	— Bit 3 to bit 0	—
Notes: 1. To access a register, set the register number to be accessed in IR after setting RS to 0. The specified register can then be accessed after setting RS to 1. 2. Some internal registers have nonexistent bits; 0 must be written to these bits.			

Table 11 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
R0	LSBY	Control register (Module standby) LSBY is the module standby setting bit. When LSBY is set to 1, the LCD controller enters the standby mode. When this occurs, the state of the PWR bit is not affected, but the DISP and OPON bits are reset. <ul style="list-style-type: none"> LSBY = 0; LCD controller operates normally. LSBY = 1; step-up and internal operations stop, display is turned off, and LCD controller enters standby mode. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 0 Bit 5	0
R0	PWR	Control register (Step-up circuit operation setting) The PWR is used to select whether the step-up circuit is active or not. <ul style="list-style-type: none"> PWR = 0; the step-up circuit is inactive. PWR = 1; the step-up circuit is active. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 0 Bit 4	1
R0	SOB	Control register (Display mode select) The SOB bit is used to select either the character display mode or the graphic display mode. <ul style="list-style-type: none"> SOB = 0; the character display mode is selected. Bits 4 to 0 of one display memory data byte are output to the segment pins. SOB = 1; the graphic display mode is selected. All bits in one display memory data byte are output to the segment pins. The X address that can be output is in the range H'0 to H'4 in the case of 1/32 duty, H'0 to H'6 in the case of 1/16 duty, and H'0 to H'7 in the case of 1/8 duty. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 0 Bit 2	0

Table 11 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value
R0	DDTY1, DDTY0 Control register (Display duty select) The DDTY1 and DDTY0 bits are used to select a display duty of 1/32, 1/16, or 1/8. <ul style="list-style-type: none"> DDTY1 = 0 and DDTY0 = 0; 1/32 duty is selected. DDTY1 = 0 and DDTY0 = 1; 1/16 duty is selected. Y-address H'10 to H'1F display data is invalid. DDTY1 = 1 and DDTY0 = *; 1/8 duty is selected. Y-address H'08 to H'1F display data is invalid. <p>*: Don't care.</p>	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 0 Bit 1 and bit 0	DDTY1 = 0 DDTY0 = 0
R1	DISP Control register 2 (LCD operation setting) The DISP bit is used to select whether the LCD display is active or not. When LSBY in R0 is set to 1, DISP is cleared. <ul style="list-style-type: none"> DISP = 0; LCD is inactive. All LCD outputs go to the VSS level. DISP = 1; LCD is active. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 6	1
R1	OPS Control register 2 (Op-amp circuit power-save setting) The OPS bit is used to select whether the op-amp power-save function is used or not. The power-save function is used if OPS is set to 1 when OPON is 1 and the built-in op-amp is used. Set OPS to 0 if the LCD output does not stabilize. <ul style="list-style-type: none"> OPS = 0; The power-save function is not used. OPS = 1; The power-save function is used and decreases the voltage-follower op-amp drive current. In this case, insert a 0.1- to 0.5-μF capacitor between VSS and each of V1OUT to V5OUT. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 5	0

Table 11 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
R1	OPON	Control register 2 (Op-amp circuit operation setting) The OPON bit is used to select whether the op-amp circuit is active or not. The op-amp circuit is active when OPON is set to 1. When the LCD drive power-supply level is applied to V1OUT to V5OUT from an external source, OPON must be cleared to 0. <ul style="list-style-type: none"> OPON = 0; built-in op-amps are inactive, and the output becomes high-impedance. The LCD drive voltage can be input from an external source. OPON = 1; built-in op-amps are active. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 4	1
R1	RMW	Control register 2 (Read-modify-write mode setting) The RMW bit used to select whether display memory X- and Y-address incrementing is done after a write/read access, or only after a write access (read-modify-write mode). <ul style="list-style-type: none"> RMW = 0; the address is incremented after write/read access to the display memory. RMW = 1; read-modify-write mode is set. In this mode, the address is incremented only after a write access to the display memory. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 3	0
R1	INC	Control register 2 (Increment address select) The INC is bit is used to select either the X or the Y address as the address to be incremented after the display data register (R4) access specified by RMW. The selected address is cleared after a display data register (R4) access with the maximum value for the valid display data area; in this case the other address is incremented. <ul style="list-style-type: none"> INC = 0; incrementing of the display-memory Y address has priority; the X address is incremented after Y-address overflow. INC = 1; incrementing of the display-memory X address has priority; the Y address is incremented after X-address overflow. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 1	0

Table 11 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
R1	BLK	Control register 2 (Blink operation setting) The BLK bit is used to enable or disable the blink function. If BLK is set to 1 while the DISP bit is set to 1 and the LCD display is operating, the blink function is enabled and blinking operates in the range set by BK7 to BK0 in R6, BSL4 to BSL0 in R8, and BEL4 to BEL0 in R9. <ul style="list-style-type: none"> • BLK = 0; blinking is disabled. • BLK = 1; blinking is enabled. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 0	1
R2	XA2 to XA0	Address register (X-address setting) The XA2 to XA0 bits are used to set the display memory X-direction address. A value from H'0 to H'7 can be set, but if SOB = 1, display data H'7 is invalid with 1/16 duty, and display data from H'5 to H'7 is invalid with 1/32 duty. When INC is set to 1, the address is automatically incremented after the access specified by RMW, and is cleared after an access with the maximum value for the valid display-data area. When INC is 0 and YA4 to YA0 represent the maximum value for the valid display-data area, the address is incremented after the access specified by RMW.	IR3 = 0 IR2 = 0 IR1 = 1 IR0 = 0 Bit 7 to bit 5	XA2 = 0 XA1 = 0 XA0 = 0
R2	YA4 to YA0	Address register (Y-address setting) The YA4 to YA0 bits are used to set the display-memory Y-direction address. A value from H'00 to H'1F can be set, but display data from H'10 to H'1F is invalid with 1/16 duty, and display data from H'08 to H'1F is invalid with 1/8 duty. When INC is cleared to 0, the address is automatically incremented after the access specified by RMW, and is cleared after an access with the maximum value for the valid display-data area. When INC is 1 and XA2 to XA0 represent the maximum value for the valid display-data area, the address is incremented after the access specified by RMW.	IR3 = 0 IR2 = 0 IR1 = 1 IR0 = 0 Bit 4 to bit 0	YA4 = 0 YA3 = 0 YA2 = 0 YA1 = 0 YA0 = 0

Table 11 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value
R3	FS5 to FS0 Frame frequency setting register (Frame frequency setting 5 to 0) The FS5 to FS0 bits are used to set a division ratio suitable for the characteristics of the LCD panel used. The correspondence between register settings and division ratios is shown in table 8. <ul style="list-style-type: none"> When FS5 = 0, FS4 = 0, FS3 = 0, FS2 = 1, FS1 = 0, and FS0 = 1, the subclock division ratio is set to 12. After setting, the frame frequency is set to 85.3 Hz. 	IR3 = 0 IR2 = 0 IR1 = 1 IR0 = 1 Bit 5 to bit 0	FS5 = 0 FS4 = 0 FS3 = 0 FS2 = 1 FS1 = 0 FS0 = 1
R4	Display data register Used for read/write access to the display memory specified by XA2 to XA0 and YA4 to YA0 in R2.	IR3 = 0 IR2 = 1 IR1 = 0 IR0 = 0	—
R5	ST4 to ST0 Display start line register (Display start line setting 4 to 0) The ST4 to ST0 bits are used to specify the line at which display starts. Set a value of [display start line – 1]. Changing the setting in this register enables vertical scrolling. The possible settings are 0 to 31 for 1/32 duty, 0 to 15 for 1/16 duty, and 0 to 7 for 1/8 duty. The display will not operate normally if these ranges are exceeded.	IR3 = 0 IR2 = 1 IR1 = 0 IR0 = 1 Bit 4 to bit 0	ST4 = 0 ST3 = 0 ST2 = 0 ST1 = 0 ST0 = 0

Table 11 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value
R6	<p data-bbox="310 161 450 185">Blink register</p> <p data-bbox="310 204 844 448">Register R6 writes 1 to the bit corresponding to the area to blink. There are no restrictions on areas that can blink simultaneously, and the entire screen can be made to blink by writing 1 to all bits. The setting in this register is valid only when the BLK bit is set to 1. The blink areas corresponding to the register bits depend on the value of SOB, as shown below.</p> <ul data-bbox="310 467 816 1066" style="list-style-type: none"> <li data-bbox="310 467 816 756">• When SOB = 0; <ul data-bbox="346 512 816 756" style="list-style-type: none"> <li data-bbox="346 512 771 568">BK7: SEG36 to SEG50, BK6: SEG31 to SEG35 <li data-bbox="346 587 771 643">BK5: SEG26 to SEG30, BK4: SEG21 to SEG25 <li data-bbox="346 662 771 718">BK3: SEG16 to SEG20, BK2: SEG11 to SEG15 <li data-bbox="346 737 816 756">BK1: SEG6 to SEG10, BK0: SEG1 to SEG5 <li data-bbox="310 775 816 1066">• When SOB = 1; <ul data-bbox="346 820 816 1066" style="list-style-type: none"> <li data-bbox="346 820 771 876">BK7: SEG57 to SEG64, BK6: SEG49 to SEG56 <li data-bbox="346 895 771 951">BK5: SEG41 to SEG48, BK4: SEG33 to SEG40 <li data-bbox="346 970 771 1026">BK3: SEG25 to SEG32, BK2: SEG17 to SEG24 <li data-bbox="346 1045 816 1066">BK1: SEG9 to SEG16, BK0: SEG1 to SEG8 	IR3 = 0 IR2 = 1 IR1 = 1 IR0 = 0	H'FF

Table 11 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value	
R8	BSL4 to BSL0	Blink start line register (Blink start line setting 4 to 0)	IR3 = 0 IR2 = 1 IR1 = 1 IR0 = 0	BSL4 = 0 BSL3 = 1 BSL2 = 0 BSL1 = 0 BSL0 = 0
		The BSL4 to BSL0 bits are used to specify the start line of an area that will blink. Set a value of [blink start line – 1]. The possible settings are 0 to 31 for 1/32 duty, 0 to 15 for 1/16 duty, and 0 to 7 for 1/8 duty. Normal operation is not guaranteed if these ranges are exceeded.	Bit 4 to bit 0	
R9	BEL4 to BEL0	Blink end line register (Blink end line setting 4 to 0)	IR3 = 0 IR2 = 1 IR1 = 1 IR0 = 0	BEL4 = 0 BEL3 = 1 BEL2 = 1 BEL1 = 1 BEL0 = 1
		The BEL4 to BEL0 bits are used to specify the end line of an area that will blink. Set a value of [blink end line – 1]. The possible settings are 0 to 31 for 1/32 duty, 0 to 15 for 1/16 duty, and 0 to 7 for 1/8 duty. Normal operation is not guaranteed if these ranges are exceeded.	Bit 4 to bit 0	

Table 11 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
RA	CCR3	Contrast-control register	IR3 = 0	CCR3 = 0
	to CCR0	(contrast-control setting 3 to 0) The CCR3 to CCR0 bits are used to specify the value of the contrast-control resistance between the V_{LCD} and V1 levels. By adjusting the contrast-control resistance between the V_{LCD} and V1 levels, the LCD-panel contrast of can be adjusted. The contrast-control resistance can be set in the range from 0.1R to 1.6R, where R is the LCD bleeder resistance.	IR2 = 1 IR1 = 1 IR0 = 0	CCR2 = 0 CCR1 = 0 CCR0 = 0
			Bit 4 to bit 0	
		<ul style="list-style-type: none"> • CCR3 = 0, CCR2 = 0, CCR1 = 0, and CCR0 = 0; contrast control resistance is 1.6R • CCR3 = 0, CCR2 = 0, CCR1 = 0, and CCR0 = 1; contrast control resistance is 1.5R • CCR3 = 0, CCR2 = 0, CCR1 = 1, and CCR0 = 0; contrast control resistance is 1.4R • CCR3 = 0, CCR2 = 0, CCR1 = 1, and CCR0 = 1; contrast control resistance is 1.3R • CCR3 = 0, CCR2 = 1, CCR1 = 0, and CCR0 = 0; contrast control resistance is 1.2R • CCR3 = 0, CCR2 = 1, CCR1 = 0, and CCR0 = 1; contrast control resistance is 1.1R • CCR3 = 0, CCR2 = 1, CCR1 = 1, and CCR0 = 0; contrast control resistance is 1.0R • CCR3 = 0, CCR2 = 1, CCR1 = 1, and CCR0 = 1; contrast control resistance is 0.9R • CCR3 = 1, CCR2 = 0, CCR1 = 0, and CCR0 = 0; contrast control resistance is 0.8R • CCR3 = 1, CCR2 = 0, CCR1 = 0, and CCR0 = 1; contrast control resistance is 0.7R • CCR3 = 1, CCR2 = 0, CCR1 = 1, and CCR0 = 0; contrast control resistance is 0.6R • CCR3 = 1, CCR2 = 0, CCR1 = 1, and CCR0 = 1; contrast control resistance is 0.5R • CCR3 = 1, CCR2 = 1, CCR1 = 0, and CCR0 = 0; contrast control resistance is 0.4R • CCR3 = 1, CCR2 = 1, CCR1 = 0, and CCR0 = 1; contrast control resistance is 0.3R • CCR3 = 1, CCR2 = 1, CCR1 = 1, and CCR0 = 0; contrast control resistance is 0.2R • CCR3 = 1, CCR2 = 1, CCR1 = 1, and CCR0 = 1; contrast control resistance is 0.1R 		

Table 11 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value
PDR9	Port data register 9 Stores data for port 9 pins (P9 ₇ to P9 ₀). <ul style="list-style-type: none"> • P9_n = 0; P9_n pin data is 0 • P9_n = 1; P9_n pin data is 1 	H'FFDC	H'00
PDRA	Port data register A Stores data for port A pins (PA ₃ to PA ₀). <ul style="list-style-type: none"> • PA_n = 0; PA_n pin data is 0 • PA_n = 1; PA_n pin data is 1 	H'FFDD	H'00
PCR9	Port control register 9 Controls whether each of the port 9 pins (P9 ₇ to P9 ₀) functions as an input or output pin. <ul style="list-style-type: none"> • PCR9_n = 0; P9_n functions as an input pin • PCR9_n = 1; P9_n functions as an output pin 	H'FFEC	H'FF
PCRA	Port control register A Controls whether each of the port A pins (PA ₃ to PA ₀) functions as an input or output pin. <ul style="list-style-type: none"> • PCRA_n = 0; PA_n functions as an input pin • PCRA_n = 1; PA_n functions as an output pin 	H'FFED	H'FF
TMA	TMA3 Timer mode register A (Timer mode register A3) Selects the clock source for input for TCA. <ul style="list-style-type: none"> • TMA3 = 0; the input clock source is PSS • TMA3 = 1; the input clock source is PSW 	H'FFB0 Bit 3	1
IENR1	IENTA Interrupt enable register 1 (Timer-A interrupt enable) Enables or disables timer-A interrupt requests. <ul style="list-style-type: none"> • IENTA = 0; disables timer-A interrupt • IENTA = 1; enables timer-A interrupt 	H'FFF3 Bit 7	1

Table 11 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
IRR1	IRRTA	<p>Interrupt request register 1 (Timer-A interrupt request flag)</p> <p>Reflects the existence of timer-A interrupt request.</p> <ul style="list-style-type: none"> • IRRTA = 0; Timer-A interrupt is not requested • IRRTA = 1; Timer-A interrupt is requested 	<p>H'FFF6</p> <p>Bit 7</p>	0
SYSCR1	SSBY	<p>System control register 1 (Software standby)</p> <p>Designates transition to standby mode or watch mode.</p> <ul style="list-style-type: none"> • SSBY = 0; After a SLEEP instruction is executed in active mode, a transition is made to sleep mode • SSBY = 1; After a SLEEP instruction is executed in active mode, a transition is made to watch or standby mode 	<p>H'FFF0</p> <p>Bit 7</p>	1
SYSCR1	STS2 to STS0	<p>System control register 1 (Standby timer select 2 to 0)</p> <p>Designates the time the CPU and peripheral modules wait for stable clock operation after exiting from the standby or watch mode to the active mode due to a specific interrupt.</p> <ul style="list-style-type: none"> • STS2 = 0, STS1 = 0, and STS0 = 0; wait time is 8,192 states • STS2 = 0, STS1 = 0, and STS0 = 1; wait time is 16,384 states. • STS2 = 0, STS1 = 1, and STS0 = 0; wait time is 32,768 states • STS2 = 0, STS1 = 1, and STS0 = 10; wait time is 65,536 states • STS2 = 1, STS1 = *, and STS0 = *; wait time is 131,972 states <p>*: Don't care.</p>	<p>H'FFF0</p> <p>Bit 6 to bit 4</p>	<p>STS2 = 0</p> <p>STS1 = 0</p> <p>STS0 = 0</p>

Table 11 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
SYSCR1	LSON	System control register 1 (Low speed on flag) Chooses the CPU operating clock when the watch mode is cleared. <ul style="list-style-type: none"> LSON = 0; the CPU operates on the system clock LSON = 1; the CPU operates on the subclock 	H'FFF0 Bit 3	0
SYSCR2	MSON	System control register 2 (Medium speed on flag) After standby, watch, or sleep mode is cleared, this bit selects active (high-speed) or active (medium-speed) mode. <ul style="list-style-type: none"> MSON = 0; operation is in active (high-speed) mode MSON = 1; operation is in active (medium-speed) mode 	H'FFF1 Bit 2	0

4. RAM Used

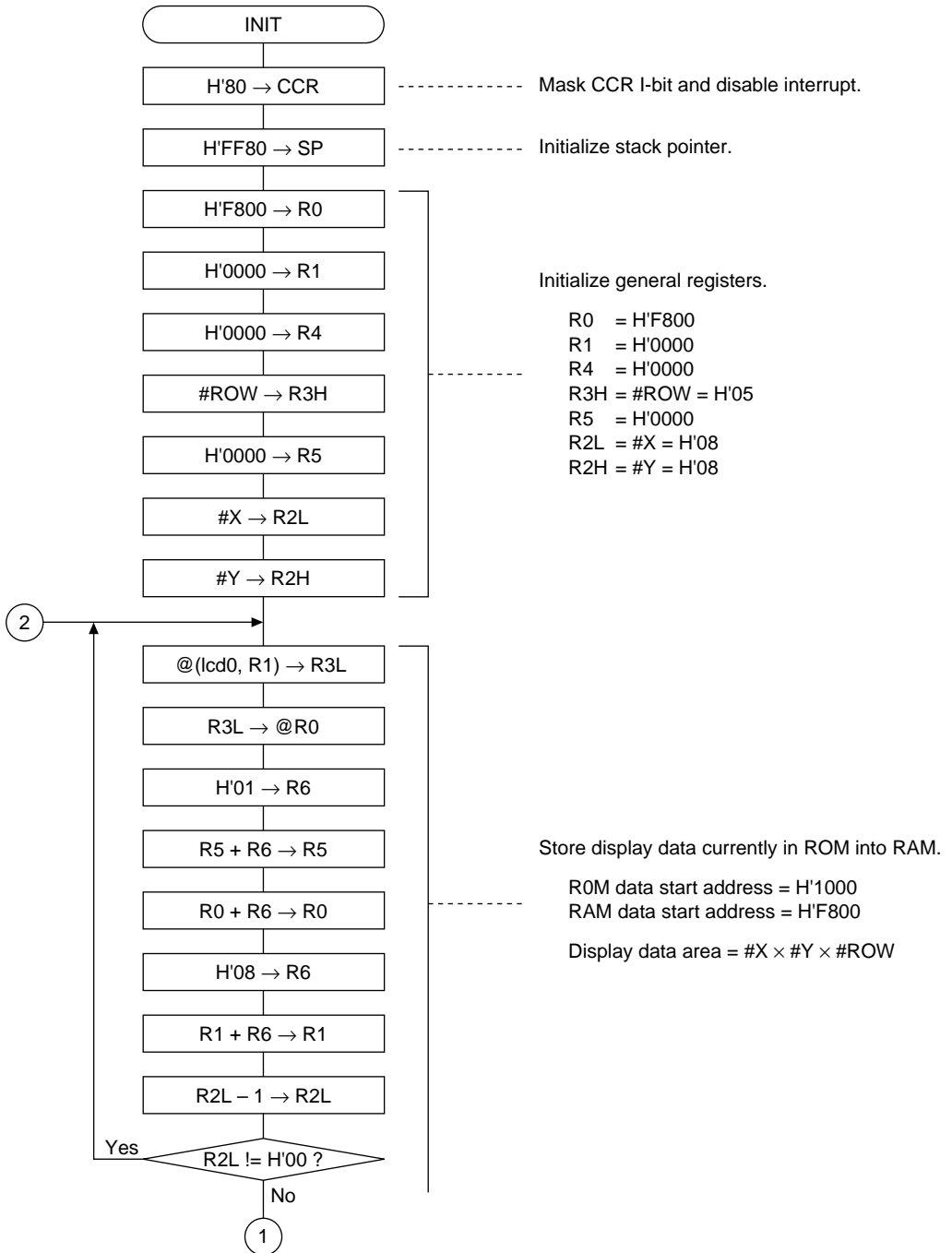
The RAM functions in this task example are described in table 12.

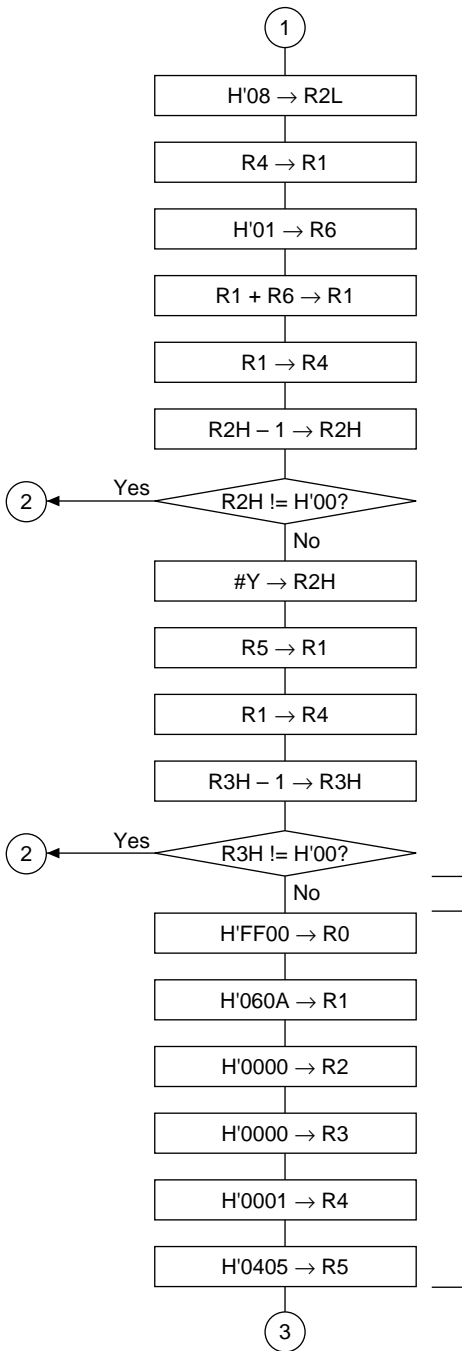
Table 12 RAM Functions

Label	Function	RAM Address	Module Used
POINTER	The pointer that shows the address of the RAM that is storing display data being written.	H'FA00	INIT, TMRA
ADDRESS	Stores the address of the display memory being written to.	H'FA04	INIT, TMRA
POS	The pointer that shows the address of the display memory being written to.	H'FA06	INIT, TMRA
COLOR	Stores the data for setting the contrast control resistance value to be set in the contrast-control register (RA).	H'FA08	INIT, TMRA

Flowcharts

1. Main routine





Store display data currently in ROM into RAM.

ROM data start address = H'1000

RAM data start address = H'F800

Display data area = X × Y × Row byte

Initialize general registers.

R0 = H'FF00

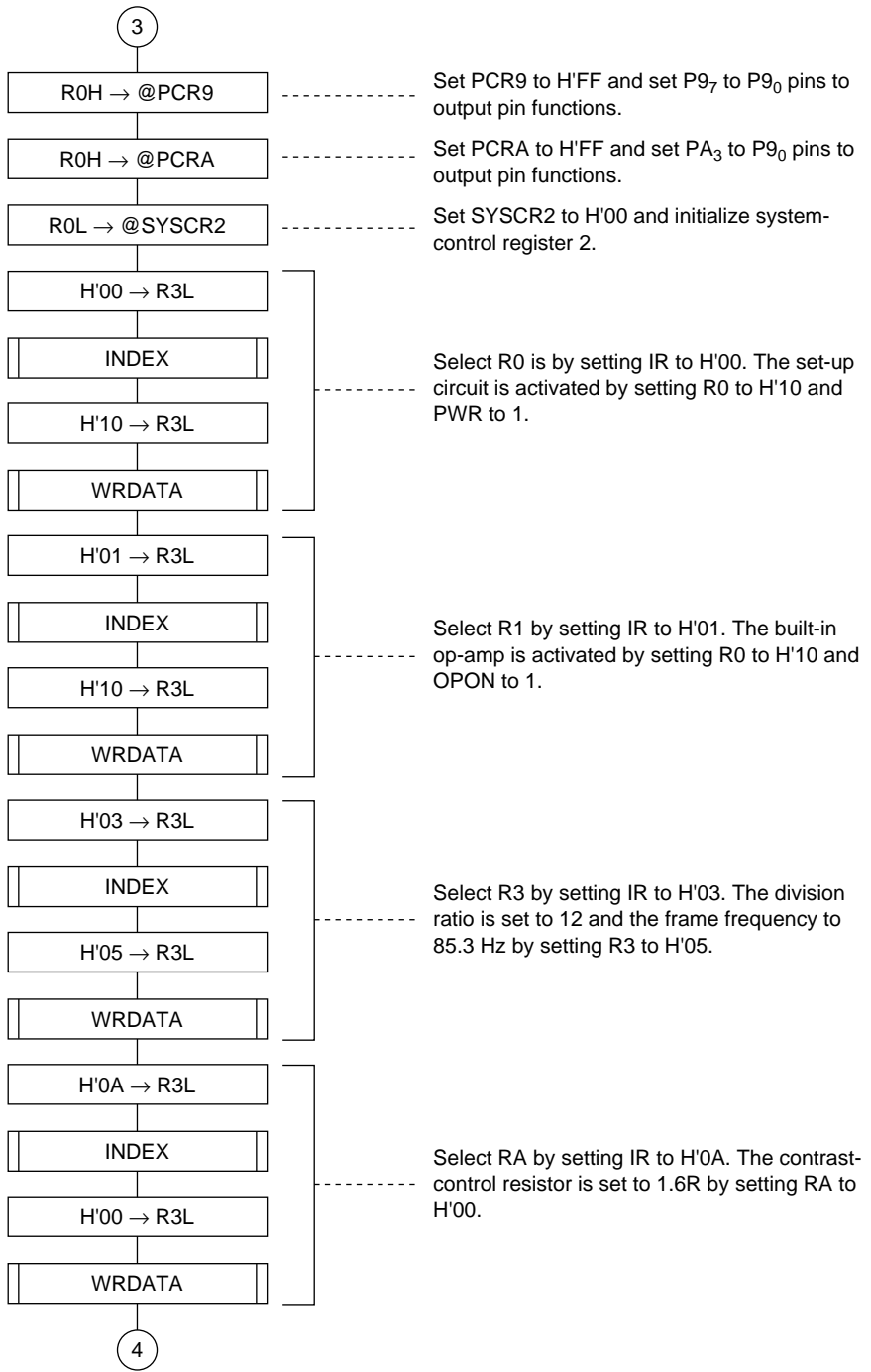
R1 = H'060A

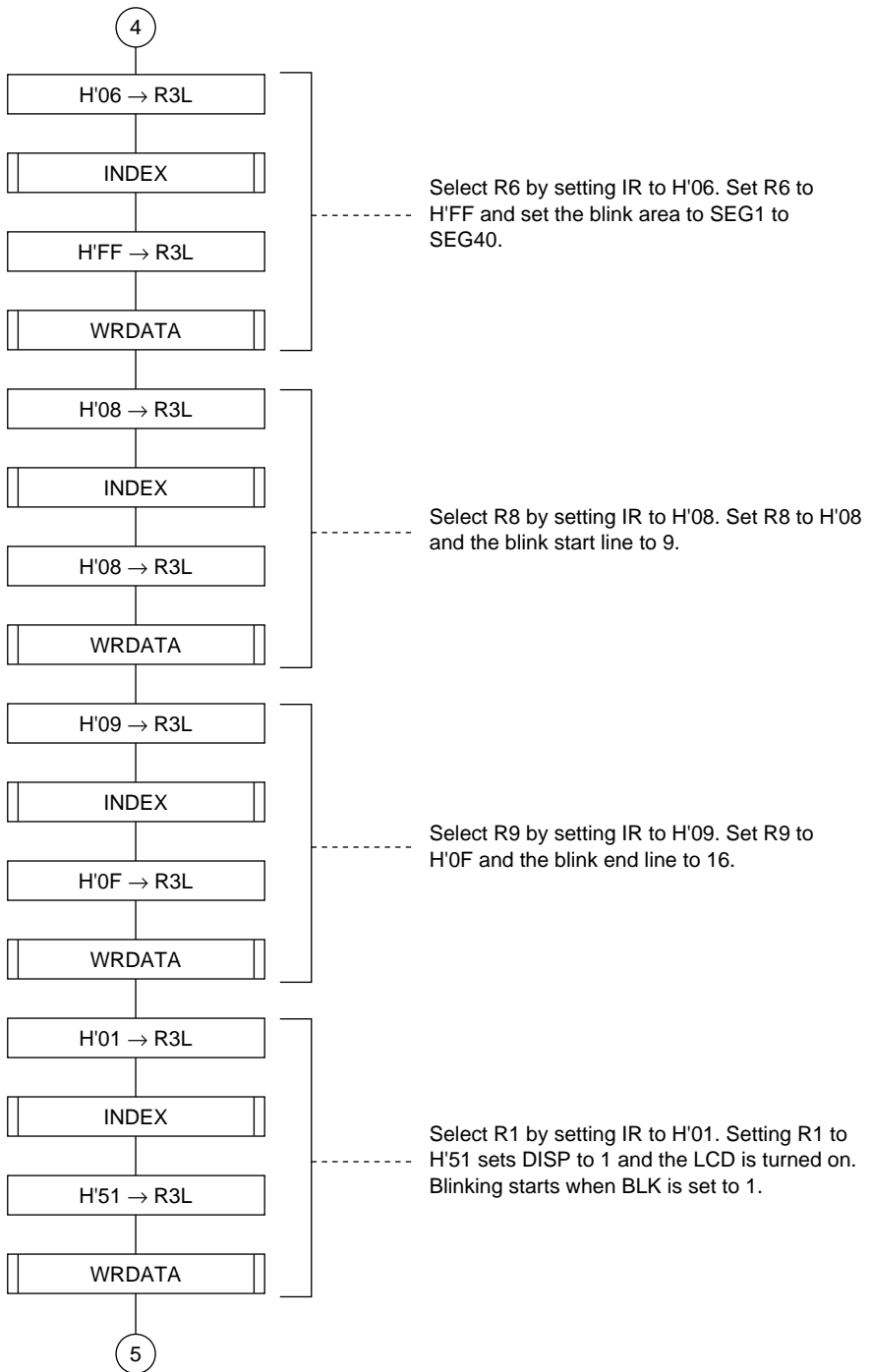
R2 = H'0000

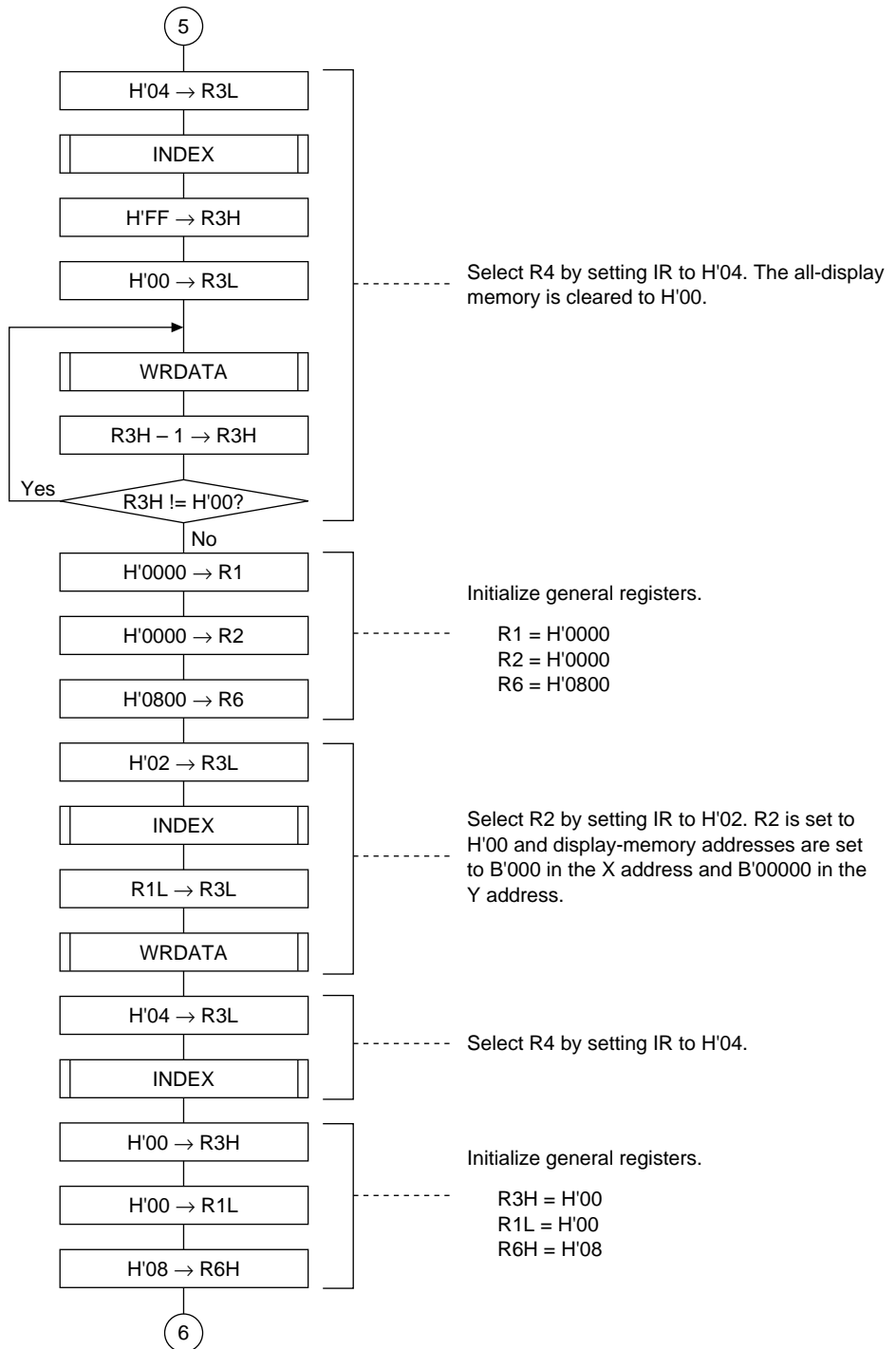
R3 = H'0000

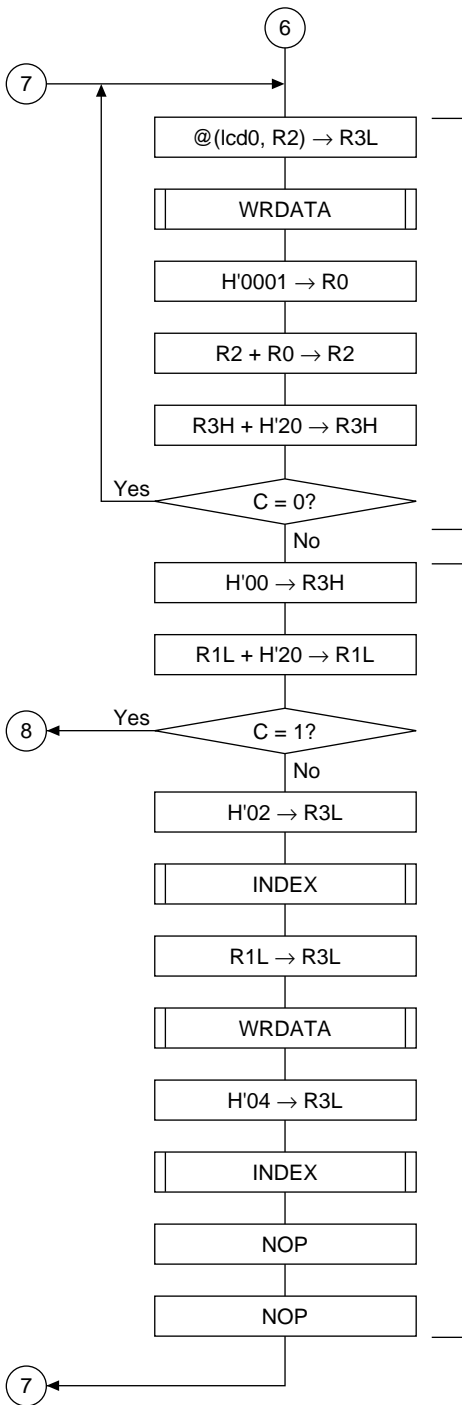
R4 = H'0001

R5 = H'0405







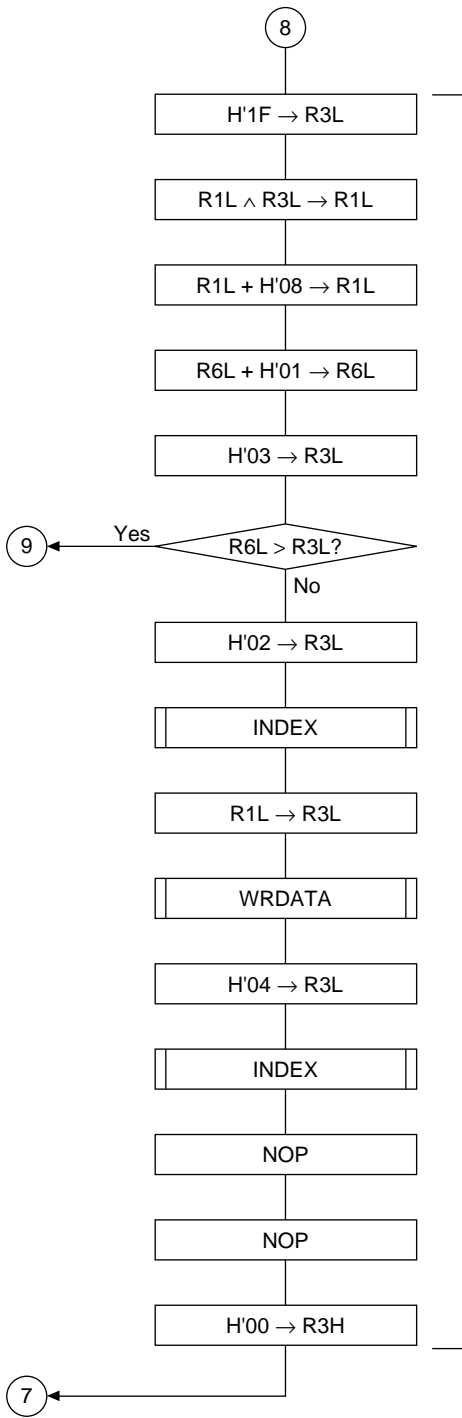


Set one-character display data (1 byte × 8) in the display RAM.

Y address: H'00 to H'07
X address: H'00

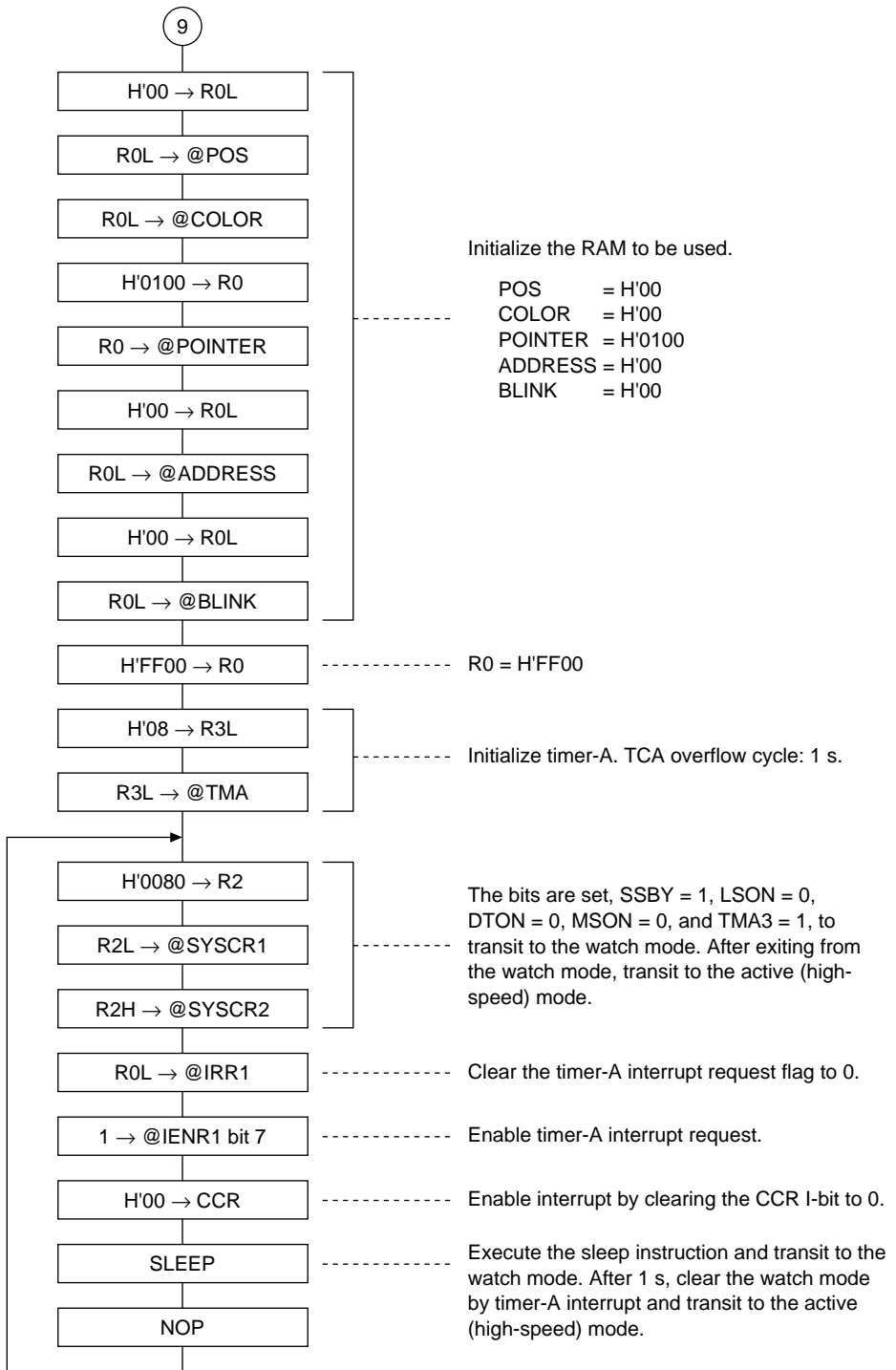
Set one-line display data (1 character × 8) in the display RAM.

Y address: H'00 to H'07
X address: H'00 to H'07

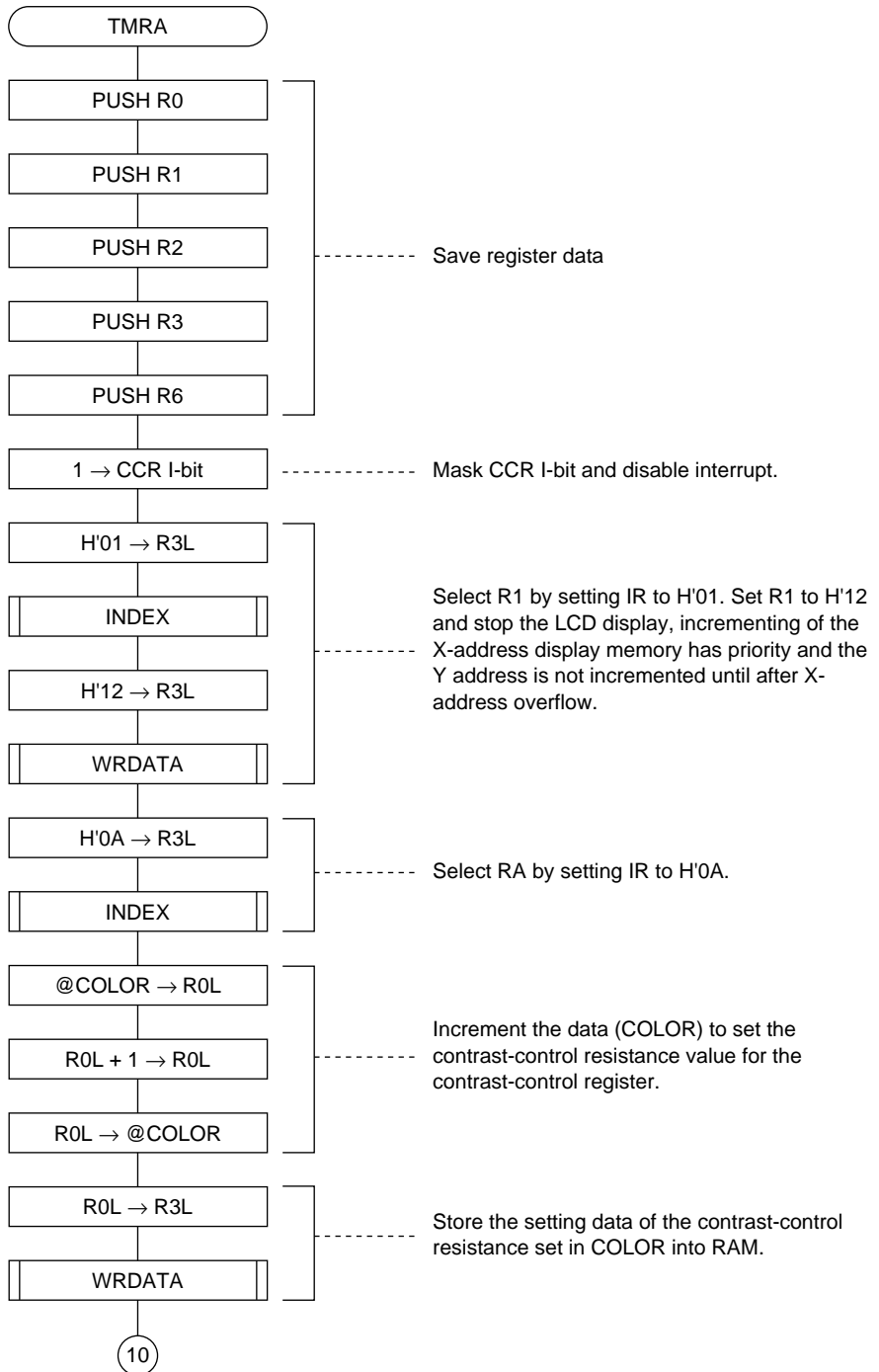


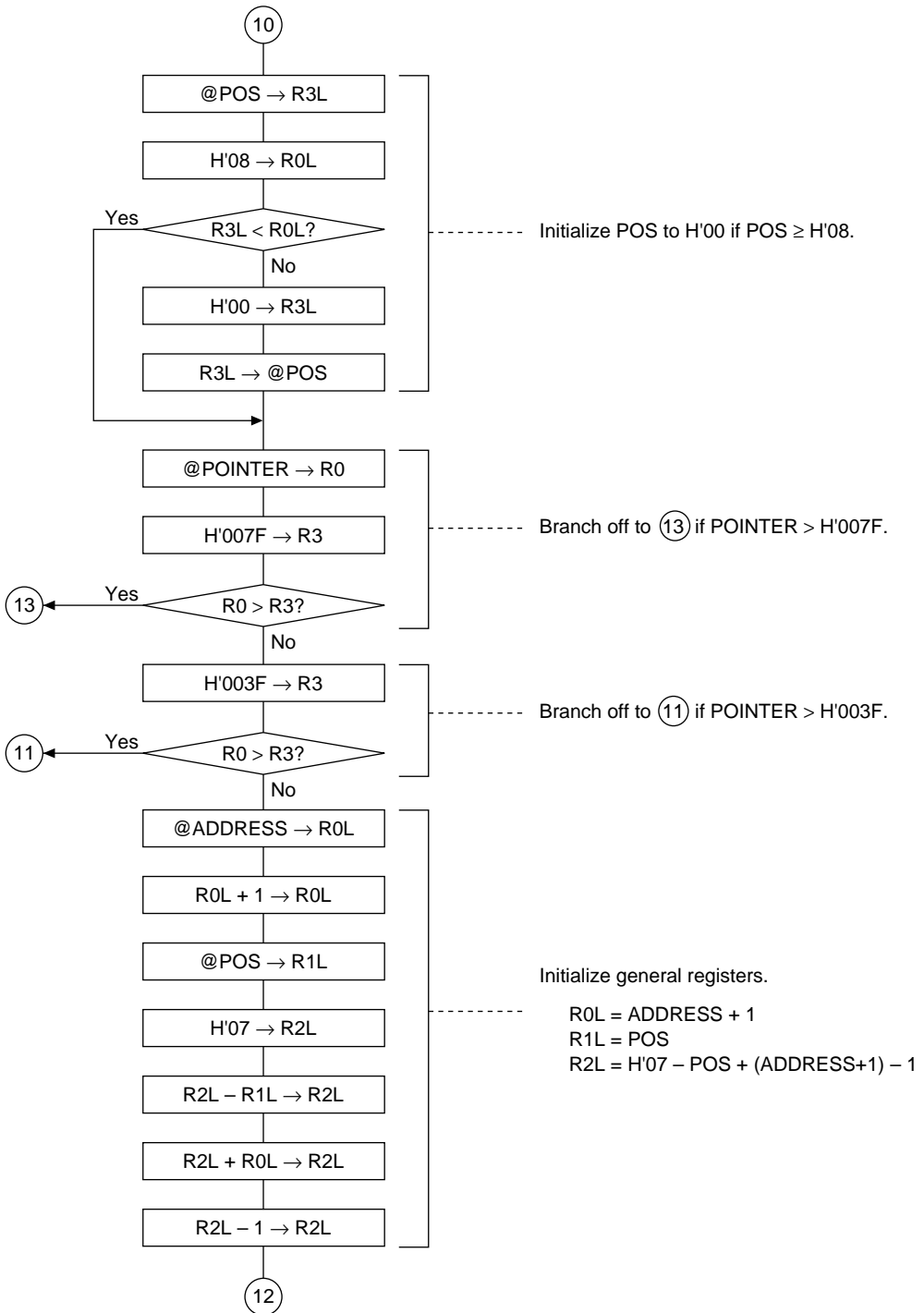
Set four-line display data (8 characters × 4) in the display RAM.

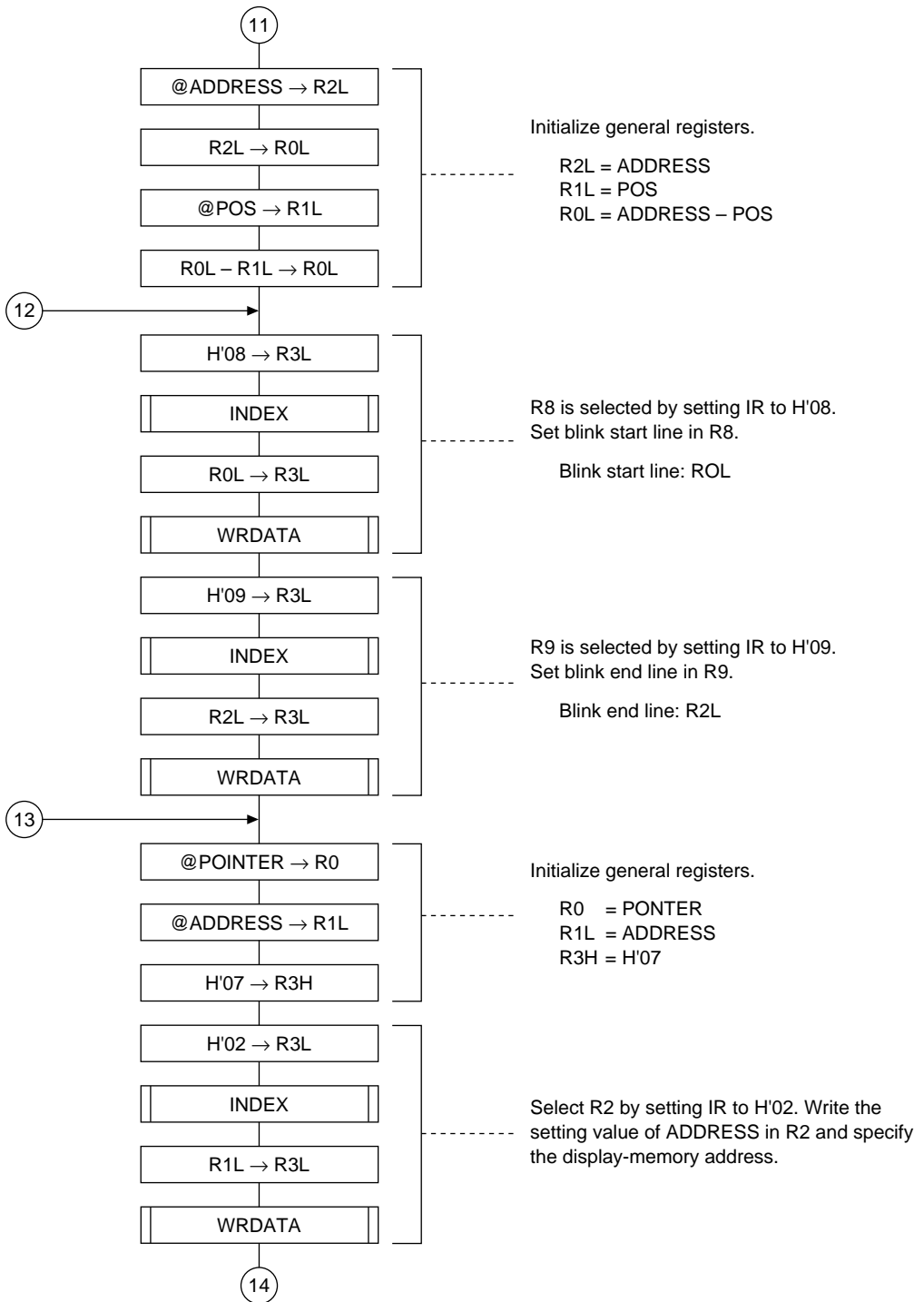
Y address: H'00 to H'1F
X address: H'00 to H'07

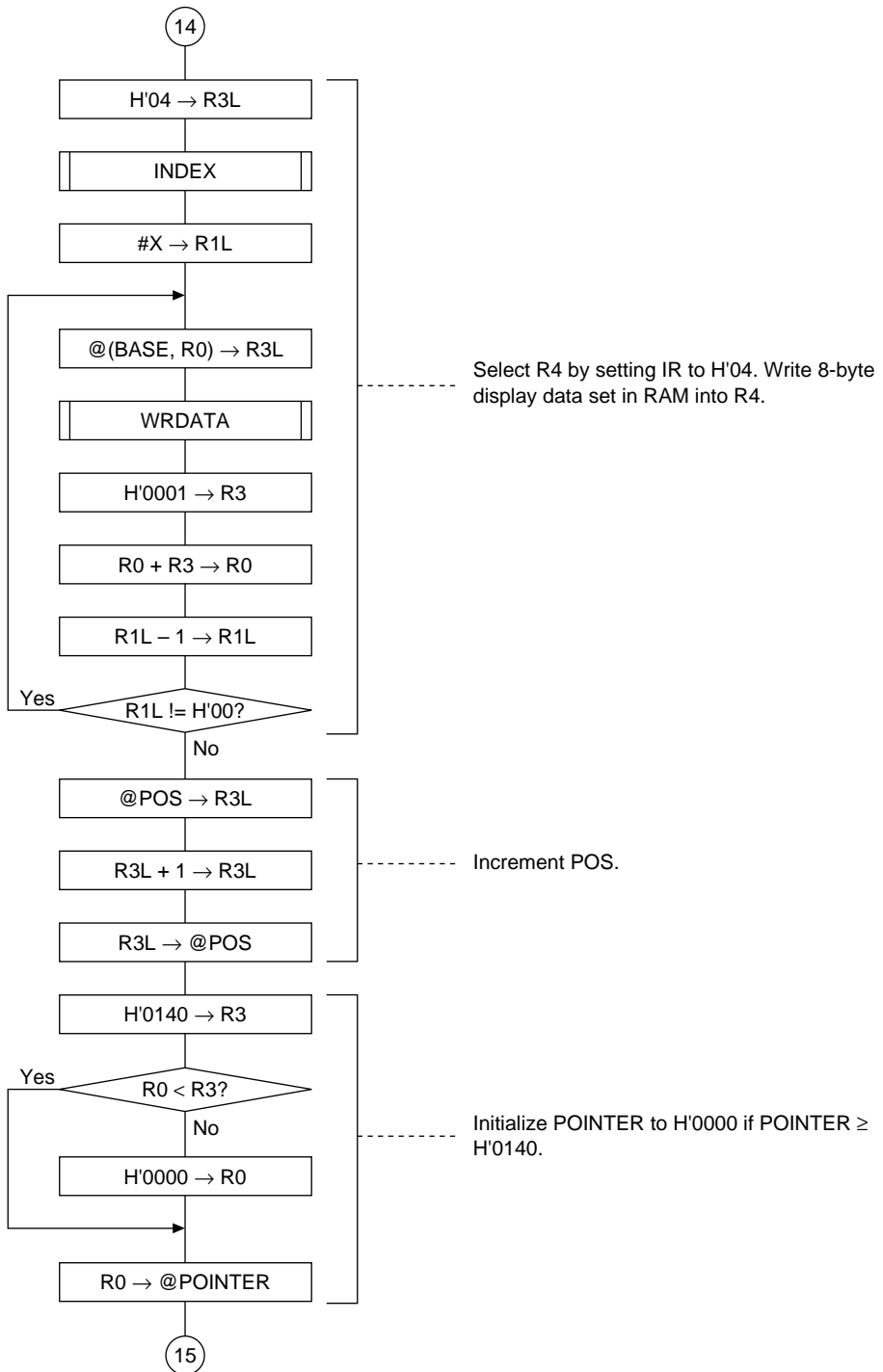


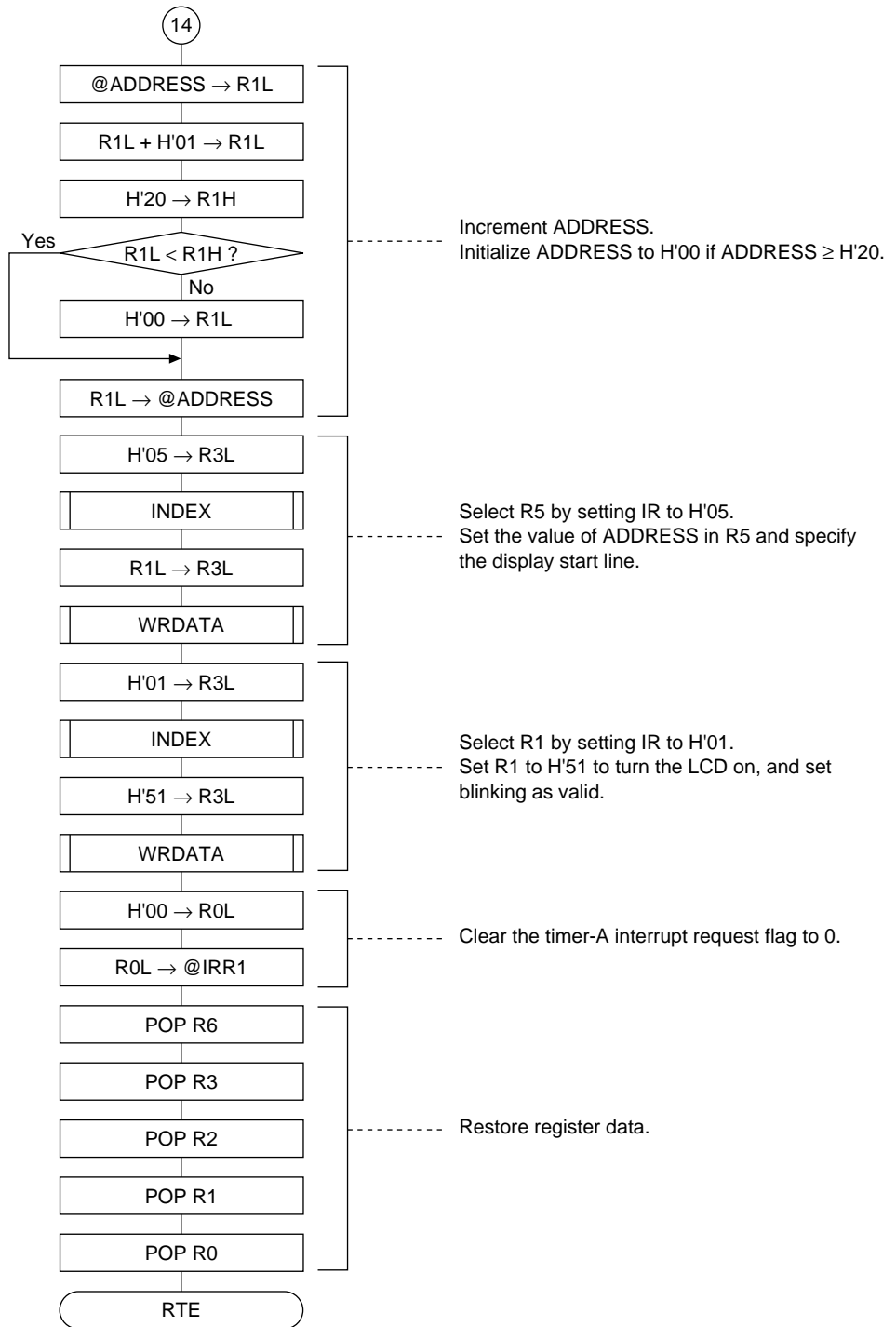
2. Timer-A interrupt processing routine



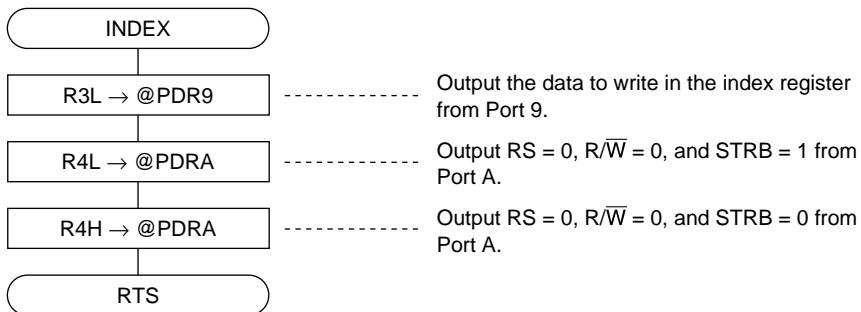




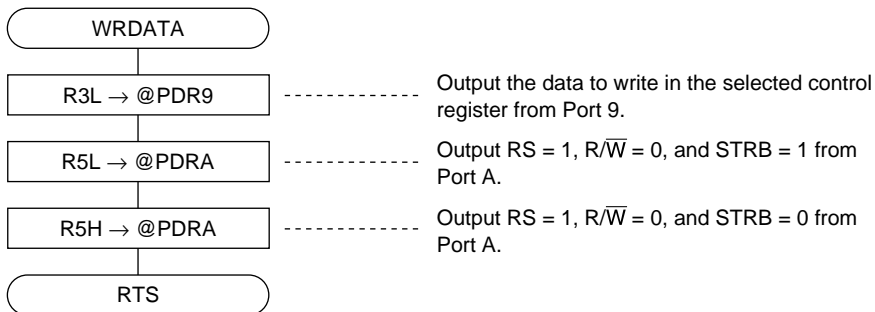




3. Index-register selection routine



4. LCD-control register data write routine



Program Lists

```
;*****
;   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 ADRESS 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 ;WKP0 Vector
.DATA.W INIT ;SCI1 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
.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'0F800,R0     ;STORE THE ROM DATA IN THE RAM STARTING
                                   ;FROM #0F800
        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     @(lcd0,R1),R3L   ;THE DISPALY DATA IN THE ROM
                                   ; STARTING FROM #LCD0
        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

;
;*****
;SET THE GENERAL REGISTER
;*****
;
PTOP:    MOV.W    #H'0FF00,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    R0H,@PCR9:8
        MOV.B    R0H,@PCRA:8
        MOV.B    R0L,@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'0FF,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'0FF,R3H   ;BYTE NUMBER
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
    mov.w    #h'0800,r6
    mov.b    #h'02,r31
    jsr      @INDEX
    mov.b    r11,r31
    jsr      @WRDATA
    mov.b    #h'4,r31
    jsr      @INDEX
    mov.b    #h'0,r3h
    mov.b    #h'00,r11
    mov.b    #8,r6h

;
gg1:
    mov.b    @(lcd0,r2),r31
    jsr      @WRDATA          ;STORE ONE CHARACTER
    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,r11        ;STORE 8 CHARACTER
    bcs     gg3

gg23:
    mov.b    #h'02,r31
    jsr      @INDEX
    mov.b    r11,r31
    jsr      @WRDATA
    mov.b    #h'4,r31

```

```

    jsr      @INDEX
    nop
    nop
    jmp      @gg1
;
gg3:
    mov.b    #h'1f,r31
    and.b    r31,r11
    add.b    #h'08,r11
    add.b    #h'1,r61
    mov.b    #h'3,r31          ;STORE 32 CHARACTER
    cmp.b    r31,r61
    bhi     TIMA
    mov.b    #h'02,r31
    jsr      @INDEX
    mov.b    r11,r31
    jsr      @WRDATA
    mov.b    #h'4,r31
    jsr      @INDEX
    nop
    nop
    mov.b    #0,r3h
    jmp      @gg1
;
;*****
;SET THE VARIABLE
;*****
;
TIMA:
    MOV.B    #H'0,R0L
    MOV.B    R0L,@POS
    MOV.B    R0L,@COLOR
    MOV.W    #H'100,R0
    MOV.W    R0,@POINTER
    MOV.B    #H'00,R0L
    MOV.B    R0L,@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    R0L,@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,R0L
INC          R0L
MOV.B        R0L,@COLOR
MOV.B        R0L,R3L      ;CCR = 8
JSR         @WRDATA
;
;*****
; CONTROL THE BLINK AREA POSITION
;*****
;
MOV.B        @POS,R3L
MOV.B        #H'8,R0L
CMP.B        R0L,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          R0L
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    ;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    R0L,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,r31
jsr      @INDEX          ;SET THE REPLACE DATA POSITION
mov.b    r11,r31
jsr      @WRDATA
mov.b    #h'4,r31
jsr      @INDEX
MOV.B    #X,R1L

;
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,r01        ;CLEAR THE TIMERA INTERRUPT FLAG
mov.b   r01,@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

;

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:

i1

.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

```


;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
```

i2

.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

i3

.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

i4

.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
.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

i5

.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

i6

```

.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'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'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'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
.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'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
.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

```

;2

```

.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
.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
.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
;
.END

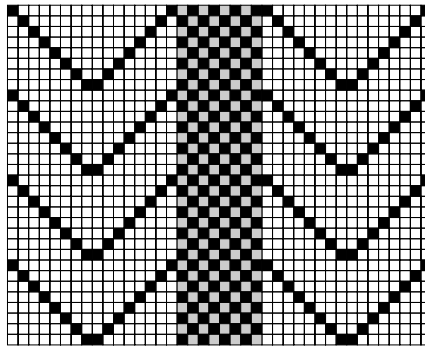
```

2.2 Dot-matrix LCD Control—Graphic Display Mode

Dot-matrix LCD Control— Graphic Display Mode	MCU: H8/3857 series	Functions Used: Dot-matrix LCD Controller
---	--------------------------------	--

Specifications

1. LCD display using a dot-matrix LCD controller function.
2. A 40×32 dot-matrix LCD is used in this task example.
3. A 1/5-bias, 1/32-duty display is used in graphic display mode.
4. Scrolling and blinking.
5. Figure 1 shows an example of the LCD display.



The picture shown above scrolls vertically by one line upward with blinking the shaded area.

Figure 1 Example of LCD Display in this Task Example

Explanation of Functions Used

1. The functions commonly used with the character-display mode in the functions of the dot-matrix LCD controller are explained in “2.1 Dot-matrix LCD Control—Character Display Mode.”

2. LCD drive pin functions (graphic display mode)

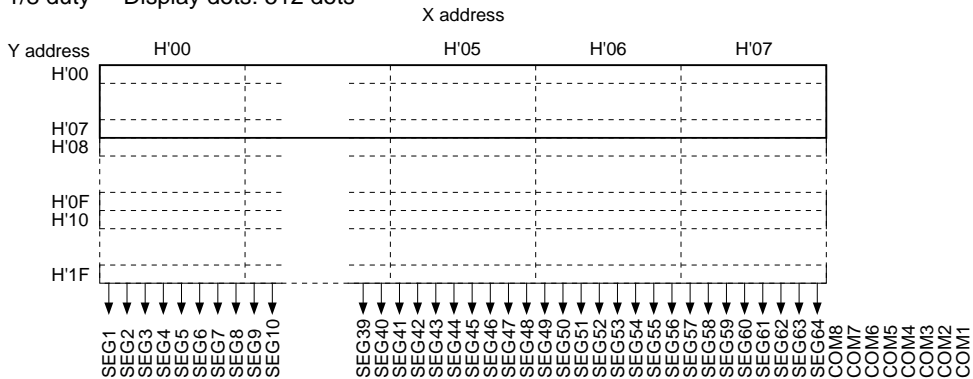
The LCD controller’s LCD drive outputs, COM9 to COM32 and SEG64 to SEG41, are switched according to the display duty and display mode.

Table 1 shows the switch settings for common/segment output in the graphic-display mode (SOB = 1).

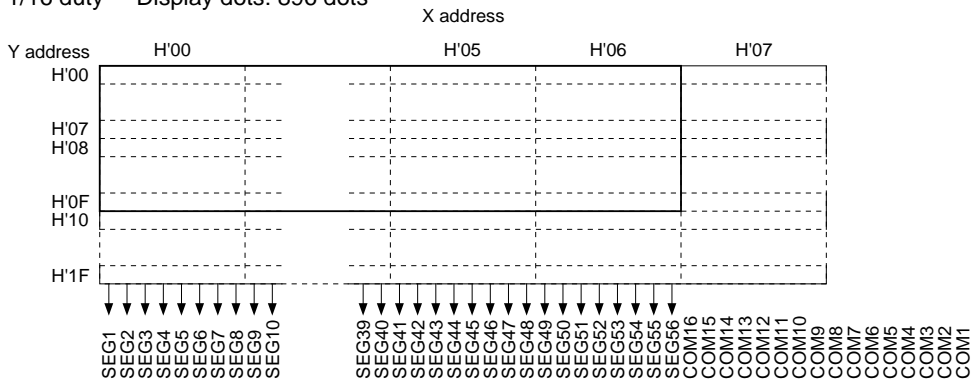
Table 1 Switches of Common/Segment Output in the Graphic-display Mode

Pin Name	SOB = 1 (Graphic-display Mode)		
	1/8 Duty	1/16 Duty	1/32 Duty
COM1 to COM8	COM1 to COM8	COM1 to COM16	COM1 to COM16
COM9/SEG64 to COM16/SEG57	SEG64 to SEG57		
SEG1 to SEG40	SEG1 to SEG56	SEG1 to SEG56	SEG1 to SEG40
COM32/SEG41 to COM25/SEG48			COM32 to COM17
COM24/SEG49 to COM17/SEG56			

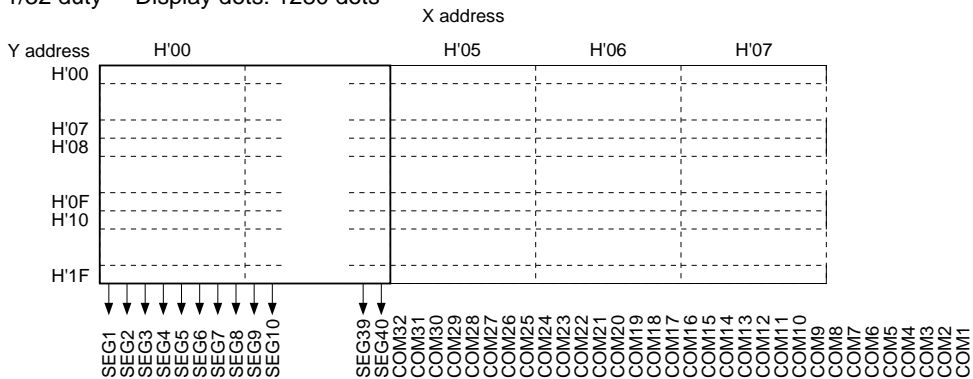
(a) 1/8 duty Display dots: 512 dots



(b) 1/16 duty Display dots: 896 dots



(c) 1/32 duty Display dots: 1280 dots



: Valid display data area

Figure 3 Display Duty and Valid Display Data Area in Graphic-display Mode

5. Table 2 shows the allocation of function for this task example.

Table 3 Function Allocation

Pin	Function Allocation
IR	Selects one of the LCD controller's ten control registers.
R0	Performs the LCD module-standby mode setting, step-up circuit control, switching between character display and graphic display, and drive duty selection.
R1	Selects whether the LCD display is activated, sets the power-save function of the op-amp circuit, selects whether the op-amp circuit is active, performs read-modify-write mode setting, and selects the address to be incremented in the display memory.
R2	Sets the display memory X- and Y-direction addresses accessed by the CPU.
R3	Sets the frame frequency.
R4	Performs read/write access to the display memory specified in R2.
R5	Specifies the line at which display starts.
R6	Specifies blinking areas.
R8	Specifies the start line of an area that will blink.
R9	Specifies the end line of an area that will blink.
RA	Specifies the contrast-control resistance value.
COM1 to COM32	LCD common drive output pins.
SEG1 to SEG40	LCD segment drive output pins.
V3, V4	Sets the LCD bias.
V34	Internal resistance test pin. Shorted to V3.
C1+, C1-, C2+, C2-	Connect external capacitances for LCD step-up.
V1OUT to V5OUT	LCD drive power-supply level input/output pins.
V_{ci}	Doubles as the reference input voltage and step-up circuit power supply.
VLOUT	LCD step-up voltage output pin.
V_{LCD}	LCD drive power-supply input pin.
Port 9	8-bit I/O port that interfaces to the on-chip LCD controller.
Port A	4-bit I/O port that interfaces to the on-chip LCD controller.

Explanation of Operation

1. The basic operation for this task

a. The display-memory and display-data settings

The display memory and display data are shown in table 3. An LCD display is created by specifying the X- and Y-direction addresses to the address register (R2) and writing the display data to the display data register (R4).

Table 3 Display Data and Display Memory

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

b. Basic operation

The basic operation for this task is shown in figure 4. An LCD display in the graphic-display mode is created through software and hardware processing as shown in figure 4.

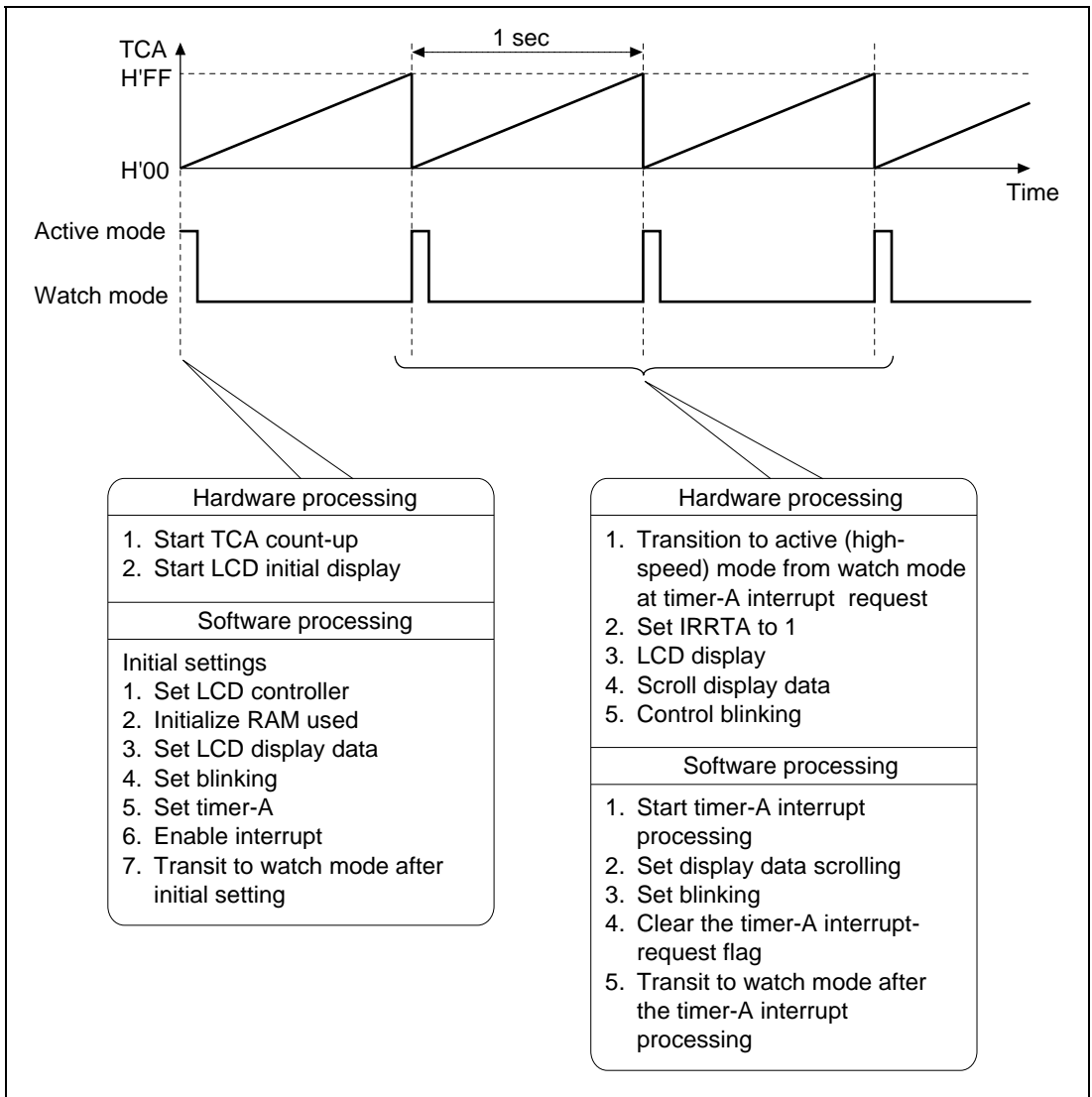


Figure 4 Basic Operation

Explanation of Software

1. Modules

Table 4 shows the modules for this task example.

Table 4 **Modules**

Module	Label	Function
Main routine	INIT	Initializes the stack pointer, RAM used, dot-matrix LCD controller, and timer-A, writes display data in the display memory, enables interrupt, and transits to watch mode.
Index register selection routine	INDEX	Selects the index register (IR)
LCD control-register write routine	WRDATA	Writes the data in the control register selected by IR.
Timer-A interrupt processing routine	TMRA	Scrolls display data and controls blinking.

2. Argument

An argument is not used in this task

3. Internal registers used

Table 5 shows the internal registers used for this task example.

Table 5 Internal Registers Used

Register	Description	IR Setting	Setting Value
IR IR3 to IR0	<p>IR Index register (Index registers 3 to 0)</p> <p>Select one of ten control registers of the LCD controller.</p> <ul style="list-style-type: none"> • R0 is selected when IR3 = 0, IR2 = 0, IR1 = 0, and IR0 = 0 • R1 is selected when IR3 = 0, IR2 = 0, IR1 = 0, and IR0 = 1 • R2 is selected when IR3 = 0, IR2 = 0, IR1 = 1, and IR0 = 0 • R3 is selected when IR3 = 0, IR2 = 01, IR1 = 1, and IR0 = 1 • R4 is selected when IR3 = 0, IR2 = 1, IR1 = 0, and IR0 = 0 • R5 is selected when IR3 = 0, IR2 = 1, IR1 = 0, and IR0 = 1 • R6 is selected when IR3 = 0, IR2 = 1, IR1 = 1, and IR0 = 0 • R8 is selected when IR3 = 1, IR2 = 0, IR1 = 0, and IR0 = 0 • R9 is selected when IR3 = 1, IR2 = 0, IR1 = 0, and IR0 = 1 • RA is selected when IR3 = 1, IR2 = 0, IR1 = 1, and IR0 = 0 <p>Notes: 1. To access a register, set the register number to be accessed in IR after setting RS to 0. The specified register can then be accessed after resetting RS to 1.</p> <p>2. Some internal registers have nonexistent bits; 0 must be written to these bits.</p>	— Bit 3 to bit 0	—

Table 5 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
R0	LSBY	Control register (Module standby) LSBY is the module standby setting bit. When LSBY is set to 1, the LCD controller enters the standby mode. When this happens, the state of the PWR bit is not affected, but the DISP and OPON bits are reset. <ul style="list-style-type: none"> LSBY = 0; LCD controller operates normally. LSBY = 1; step-up and internal operations stop, the display is turned off, and the LCD controller enters the standby mode. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 0 Bit 5	0
R0	PWR	Control register (Step-up circuit operation setting) The PWR bit is used to select whether the step-up circuit is activated or not. <ul style="list-style-type: none"> PWR = 0; the step-up circuit is inactive. PWR = 1; the step-up circuit is active. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 0 Bit 4	1
R0	SOB	Control register (Display-mode select) The SOB bit is used to select either the character-display mode or the graphic-display mode. <ul style="list-style-type: none"> SOB = 0; the character-display mode is selected. Bits 4 to 0 of one display memory data byte are output to the segment pins. SOB = 1; the graphic-display mode is selected. All bits in one display memory data byte are output to the segment pins. The X address that can be output is in the range H'0 to H'4 in the case of 1/32 duty, H'0 to H'6 in the case of 1/16 duty, and H'0 to H'7 in the case of 1/8 duty. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 0 Bit 2	1

Table 5 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value
R0	DDTY1, DDTY0 Control register (Display duty select) The DDTY1 and DDTY0 bits are used to select a display duty of 1/32, 1/16, or 1/8. <ul style="list-style-type: none"> DDTY1 = 0 and DDTY0 = 0; 1/32 duty is selected. DDTY1 = 0 and DDTY0 = 1; 1/16 duty is selected. Y-address H'10 to H'1F display data is invalid. DDTY1 = 1 and DDTY0 = *; 1/8 duty is selected. Y-address H'08 to H'1F display data is invalid. *: Don't care.	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 0 Bit 1 and bit 0	DDTY1 = 0 DDTY0 = 0
R1	DISP Control register 2 (LCD operation setting) The DISP bit is used to select whether the LCD display is activated or not. When LSBY in R0 is set to 1, DISP is cleared. <ul style="list-style-type: none"> DISP = 0; the LCD is turned off. All LCD outputs go to the VSS level. DISP = 1; the LCD is turned on. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 6	1
R1	OPS Control register 2 (Op-amp circuit power-save setting) The OPS bit is used to select whether the op-amp power-save function is used or not. The power-save function is used if OPS is set to 1 when OPON is 1 and the built-in op-amp operates. Set OPS to 0 is the LCD output does not stabilize. <ul style="list-style-type: none"> OPS = 0; the power-save function is not used. OPS = 1; the power-save function is used and decreases the voltage follower op-amp drive current. In this case, insert a 0.1- to 0.5-μF capacitor between VSS and each of V1OUT to V5OUT. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 5	0

Table 5 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
R1	OPON	Control register 2 (Op-amp circuit operation setting) The OPON bit is used to select whether the op-amp circuit is activated or not. The op-amp circuit operates when OPON is set to 1. When the LCD drive power-supply level is applied to V1OUT to V5OUT from an external source, OPON must be cleared to 0. <ul style="list-style-type: none"> • OPON = 0; built-in op-amps are inactive, and output becomes high-impedance. The LCD drive voltage can be input from an external source. • OPON = 1; built-in op-amps are active 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 4	1
R1	RMW	Control register 2 (Read-modify-write mode setting) The RMW bit is used to select whether display-memory X- or Y-address incrementing is done after a write/read access, or only after a write access (read-modify-write mode). <ul style="list-style-type: none"> • RMW = 0; the address is incremented after write/read access to the display memory. • RMW = 1; read-modify-write mode is set. In this mode, the address is incremented only after write access to the display memory. 	IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1 Bit 3	0

Table 5 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
R1	INC	<p>Control register 2 (Increment address select)</p> <p>The INC bit is used to select either the X address or the Y address as the address to be incremented after the display-data register (R4) access specified by RMW. The selected address is cleared after a display-data register (R4) access with the maximum value for the valid display-data area; in this case the other address is incremented.</p> <ul style="list-style-type: none"> • INC = 0; incrementing of the display-memory Y address has priority; the X address is incremented after Y-address overflow. • INC = 1; incrementing of the display memory X address has priority; the Y address is incremented after X address overflow. 	<p>IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1</p> <p>Bit 1</p>	0
R1	BLK	<p>Control register 2 (Blink operation setting)</p> <p>The BLK bit is used to enable or disable the blink function. If BLK is set to 1 while the DISP bit is set to 1 and the LCD display is operating, the blink function is enabled and blinking occurs in the range set by BK7 to BK0 in R6, BSL4 to BSL0 in R8, and BEL4 to BEL0 in R9.</p> <ul style="list-style-type: none"> • BLK = 0; blinking is disabled. • BLK = 1; blinking is enabled. 	<p>IR3 = 0 IR2 = 0 IR1 = 0 IR0 = 1</p> <p>Bit 0</p>	1
R2	XA2 to XA0	<p>Address register (X-address setting)</p> <p>The XA2 to XA0 bits are used to set the display-memory X-direction address. A value from H'0 to H'7 can be set, but if SOB = 1, display data H'7 is invalid with 1/16 duty, and display data from H'5 to H'7 is invalid with 1/32 duty. When INC is set to 1, the address is automatically incremented after the access specified by RMW, and is cleared after an access with the maximum value for the valid display-data area. When INC is 0 and YA4 to YA0 represent the maximum value for the valid display-data area, the address is incremented after the access specified by RMW.</p>	<p>IR3 = 0 IR2 = 0 IR1 = 1 IR0 = 0</p> <p>Bit 7 to bit 5</p>	<p>XA2 = 0 XA1 = 0 XA0 = 0</p>

Table 5 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value
R2	YA4 to YA0 Address register (Y-address setting) The YA4 to YA0 bits are used to set the display memory Y-direction address. A value from H'00 to H'1F can be set, but display data from H'10 to H'1F is invalid with 1/16 duty, and display data from H'08 to H'1F is invalid with 1/8 duty. When INC is cleared to 0, the address is automatically incremented after the access specified by RMW, and is cleared after an access with the maximum value for the valid display-data area. When INC is 1 and XA2 to XA0 represent the maximum value for the valid display-data area, the address is incremented after the access specified by RMW.	IR3 = 0 IR2 = 0 IR1 = 1 IR0 = 0 Bit 4 to bit 0	YA4 = 0 YA3 = 0 YA2 = 0 YA1 = 0 YA0 = 0
R3	FS5 to FS0 Frame-frequency setting register (Frame frequency setting 5 to 0) FS5 to FS0 set a division ratio suitable for the characteristics of the LCD panel used. The correspondence between the register settings and the division ratios is shown in table 8. <ul style="list-style-type: none"> When FS5 = 0, FS4 = 0, FS3 = 0, FS2 = 1, FS1 = 0, and FS0 = 1, the subclock division ratio is set to 12. After setting, the frame frequency is set to 85.3 Hz. 	IR3 = 0 IR2 = 0 IR1 = 1 IR0 = 1 Bit 5 to bit 0	FS5 = 0 FS4 = 0 FS3 = 0 FS2 = 1 FS1 = 0 FS0 = 1
R4	Display-data register Performs read/write access to the display memory specified by XA2 to XA0 and YA4 to YA0 in R2.	IR3 = 0 IR2 = 1 IR1 = 0 IR0 = 0	—
R5	ST4 to ST0 Display start line register (Display start line setting 4 to 0) The ST4 to ST0 bits are used to specify the line at which the display starts. Set a value of [display start line – 1]. Changing the setting in this register enables vertical scrolling. The possible settings are 0 to 31 for 1/32 duty, 0 to 15 for 1/16 duty, and 0 to 7 for 1/8 duty. The display will not perform normally if these ranges are exceeded.	IR3 = 0 IR2 = 1 IR1 = 0 IR0 = 1 Bit 4 to bit 0	ST4 = 0 ST3 = 0 ST2 = 0 ST1 = 0 ST0 = 0

Table 5 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value
R6	<p>Blink register</p> <p>R6 writes 1 to the bit corresponding to the area that will blink. There are no restrictions on areas that can blink simultaneously, and the entire screen can be made to blink by writing 1 to all bits. The setting in this register is valid only when the BLK bit is set to 1. The blink area corresponding to the register bits depends on the value of SOB, as shown below.</p> <ul style="list-style-type: none">• When SOB = 0; BK7: SEG36 to SEG40, BK6: SEG31 to SEG35 BK5: SEG26 to SEG30, BK4: SEG21 to SEG25 BK3: SEG16 to SEG20, BK2: SEG11 to SEG15 BK1: SEG6 to SEG10, BK0: SEG1 to SEG5• When SOB = 1; BK7: SEG57 to SEG64, BK6: SEG49 to SEG56 BK5: SEG41 to SEG48, BK4: SEG33 to SEG40 BK3: SEG25 to SEG32, BK2: SEG17 to SEG24 BK1: SEG9 to SEG16, BK0: SEG1 to SEG8	IR3 = 0 IR2 = 1 IR1 = 1 IR0 = 0	H'FF

Table 5 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value
R8	BSL4 to BSL0 Blink start line register (Blink start line setting 4 to 0) The BSL4 to BSL0 bits are used to specify the start line of an area that will blink. Set a value of [blink start line – 1]. The possible settings are 0 to 31 for 1/32 duty, 0 to 15 for 1/16 duty, and 0 to 7 for 1/8 duty. Normal operation is not guaranteed if these ranges are exceeded.	IR3 = 0 IR2 = 1 IR1 = 1 IR0 = 0 Bit 4 to bit 0	BSL4 = 0 BSL3 = 0 BSL2 = 0 BSL1 = 0 BSL0 = 0
R9	BEL4 to BEL0 Blink end line register (Blink end line setting 4 to 0) The BEL4 to BEL0 bits are used to specify the end line of an area that will blink. Set a value of [blink end line – 1]. The possible settings are 0 to 31 for 1/32 duty, 0 to 15 for 1/16 duty, and 0 to 7 for 1/8 duty. Normal operation is not guaranteed if these ranges are exceeded.	IR3 = 0 IR2 = 1 IR1 = 1 IR0 = 0 Bit 4 to bit 0	BEL4 = 1 BEL3 = 1 BEL2 = 1 BEL1 = 1 BEL0 = 1

Table 5 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
RA	CCR3 to CCR0	Contrast-control register (contrast-control setting 3 to 0)	IR3 = 0 IR2 = 1 IR1 = 1 IR0 = 0	CCR3 = 1 CCR2 = 0 CCR1 = 0 CCR0 = 0
		The CCR3 to CCR0 bits are used to specify the contrast-control resistance between the V_{LCD} and V1 levels. By adjusting the contrast-control resistance between the V_{LCD} and V1 levels, the LCD panel contrast can be adjusted. The contrast-control resistance can be set from 0.1R to 1.6R, where R is the LCD bleeder resistance.	Bit 4 to bit 0	
		<ul style="list-style-type: none"> • CCR3 = 0, CCR2 = 0, CCR1 = 0, and CCR0 = 0; contrast control resistance is 1.6R • CCR3 = 0, CCR2 = 0, CCR1 = 0, and CCR0 = 1; contrast control resistance is 1.5R • CCR3 = 0, CCR2 = 0, CCR1 = 1, and CCR0 = 0; contrast control resistance is 1.4R • CCR3 = 0, CCR2 = 0, CCR1 = 1, and CCR0 = 1; contrast control resistance is 1.3R • CCR3 = 0, CCR2 = 1, CCR1 = 0, and CCR0 = 0; contrast control resistance is 1.2R • CCR3 = 0, CCR2 = 1, CCR1 = 0, and CCR0 = 1; contrast control resistance is 1.1R • CCR3 = 0, CCR2 = 1, CCR1 = 1, and CCR0 = 0; contrast control resistance is 1.0R • CCR3 = 0, CCR2 = 1, CCR1 = 1, and CCR0 = 1; contrast control resistance is 0.9R • CCR3 = 1, CCR2 = 0, CCR1 = 0, and CCR0 = 0; contrast control resistance is 0.8R • CCR3 = 1, CCR2 = 0, CCR1 = 0, and CCR0 = 1; contrast control resistance is 0.7R • CCR3 = 1, CCR2 = 0, CCR1 = 1, and CCR0 = 0; contrast control resistance is 0.6R • CCR3 = 1, CCR2 = 0, CCR1 = 0, and CCR0 = 1; contrast control resistance is 0.5R • CCR3 = 1, CCR2 = 1, CCR1 = 0, and CCR0 = 0; contrast control resistance is 0.4R • CCR3 = 1, CCR2 = 1, CCR1 = 0, and CCR0 = 1; contrast control resistance is 0.3R • CCR3 = 1, CCR2 = 1, CCR1 = 1, and CCR0 = 0; contrast control resistance is 0.2R • CCR3 = 1, CCR2 = 1, CCR1 = 1, and CCR0 = 1; contrast control resistance is 0.1R 		

Table 5 Internal Registers Used (cont)

Register	Description	IR Setting	Setting Value
PDR9	Port data register 9 Stores data for port 9 pins P9 ₇ to P9 ₀ . <ul style="list-style-type: none"> • P9_n = 0; P9_n pin data is 0 • P9_n = 1; P9_n pin data is 1 	H'FFDC	H'00
PDRA	Port data register A Stores data for port A pins PA ₃ to PA ₀ . <ul style="list-style-type: none"> • PA_n = 0; PA_n pin data is 0 • PA_n = 1; PA_n pin data is 1 	H'FFDD	H'00
PCR9	Port control register 9 Stores data for port 9 pins P9 ₇ to P9 ₀ . <ul style="list-style-type: none"> • PCR9_n = 0; P9_n functions as an input pin • PCR9_n = 1; P9_n functions as an output pin 	H'FFEC	H'FF
PCRA	Port control register A Stores data for port A pins PA ₃ to PA ₀ . <ul style="list-style-type: none"> • PCRA_n = 0; PA_n functions as an input pin • PCRA_n = 1; PA_n functions as an output pin 	H'FFED	H'FF
TMA	TMA3 Timer mode register A (Timer mode register A3) Selects the clock source for TCA input. <ul style="list-style-type: none"> • TMA3 = 0; the input-clock source is PSS • TMA3 = 1; the input-clock source is PSW 	H'FFB0 Bit 3	1
IENR1	IENTA Interrupt enable register 1 (Timer-A interrupt enable) Enables or disables timer-A interrupt requests. <ul style="list-style-type: none"> • IENTA = 0; disables timer-A interrupts • IENTA = 1; enables timer-A interrupts 	H'FFF3 Bit 7	1

Table 5 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
IRR1	IRRTA	Interrupt request register 1 (Timer-A interrupt request flag) Reflects the existence of timer-A interrupt request. <ul style="list-style-type: none"> • IRRTA = 0; timer-A interrupt is not requested • IRRTA = 1; timer-A interrupt is requested 	H'FFF6 Bit 7	0
SYSCR1	SSBY	System control register 1 (Software standby) Designates transition to standby mode or watch mode. <ul style="list-style-type: none"> • SSBY = 0; After a SLEEP instruction is executed in the active mode, a transition is made to the sleep mode • SSBY = 1; After a SLEEP instruction is executed in the active mode, a transition is made to the watch or standby mode 	H'FFF0 Bit 7	1
SYSCR1	STS2 to STS0	System control register 1 (Standby timer select 2 to 0) Designates the time the CPU and peripheral modules wait for stable clock operation after exiting from the standby or watch mode to the active mode due to a specific interrupt. <ul style="list-style-type: none"> • STS2 = 0, STS1 = 0, and STS0 = 0; wait time is 8,192 states • STS2 = 0, STS1 = 0, and STS0 = 1; wait time is 16,384 states • STS2 = 0, STS1 = 1, and STS0 = 0; wait time is 32,768 states • STS2 = 0, STS1 = 1, and STS0 = 1; wait time is 65,536 states • STS2 = 0, STS1 = *, and STS0 = *; wait time is 131,972 states <p>*: Don't care</p>	H'FFF0 Bit 6 to bit 4	STS2 = 0 STS1 = 0 STS0 = 0

Table 5 Internal Registers Used (cont)

Register		Description	IR Setting	Setting Value
SYSCR1	LSON	System control register 1 (Low speed on flag) Chooses the CPU operating clock when the watch mode is cleared. <ul style="list-style-type: none"> LSON = 0; the CPU operates on the system clock LSON = 1; the CPU operates on the subclock 	H'FFF0 Bit 3	0
SYSCR2	MSON	System control register 2 (Medium-speed on flag) After standby, watch, or sleep mode is cleared, this bit selects the active (high-speed) or active (medium-speed) mode. <ul style="list-style-type: none"> MSON = 0; operation is in the active (high-speed) mode MSON = 1; operation is in the active (medium-speed) mode 	H'FFF1 Bit 2	0

4. RAM Used

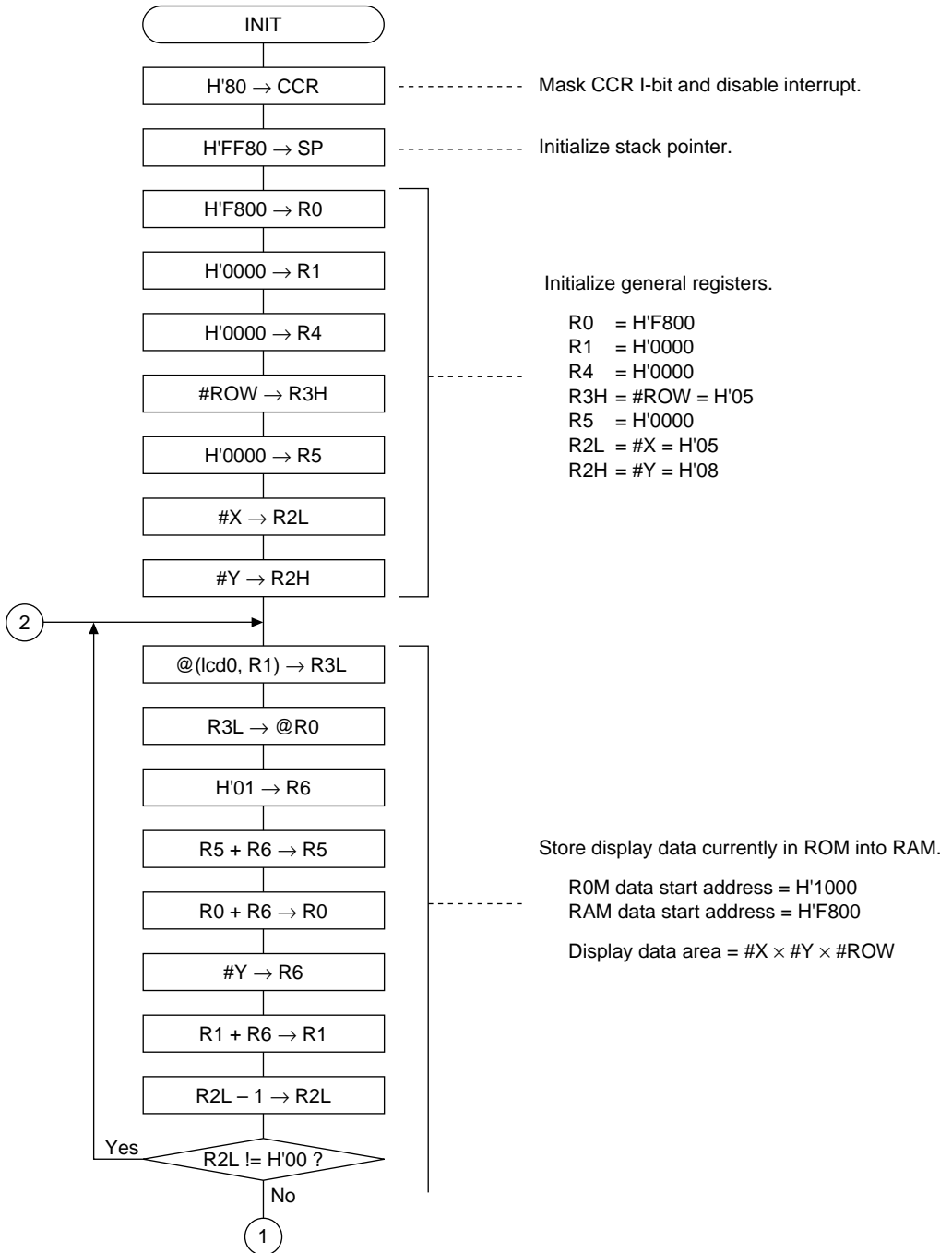
The RAM functions for this task example are shown in table 12.

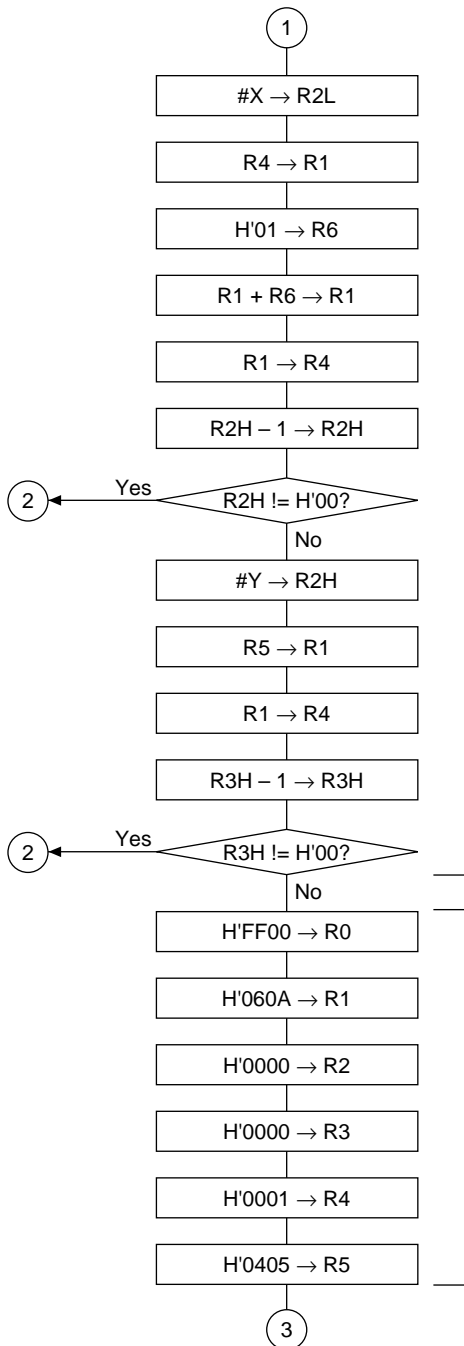
Table 6 RAM Functions

Label	Function	RAM Address	Module Used
POINTER	The pointer that shows the address of the RAM storing the display data currently being written.	H'FA00	INIT, TMRA
ADDRESS	Stores the address of the display memory being written.	H'FA04	INIT, TMRA
POS	The pointer that shows the address of the display memory being written.	H'FA06	INIT, TMRA

Flowcharts

1. Main routine





Store display data currently in ROM into RAM.

ROM data start address = H'1000

RAM data start address = H'F800

Display data area = X × Y × Row byte

Initialize general registers.

R0 = H'FF00

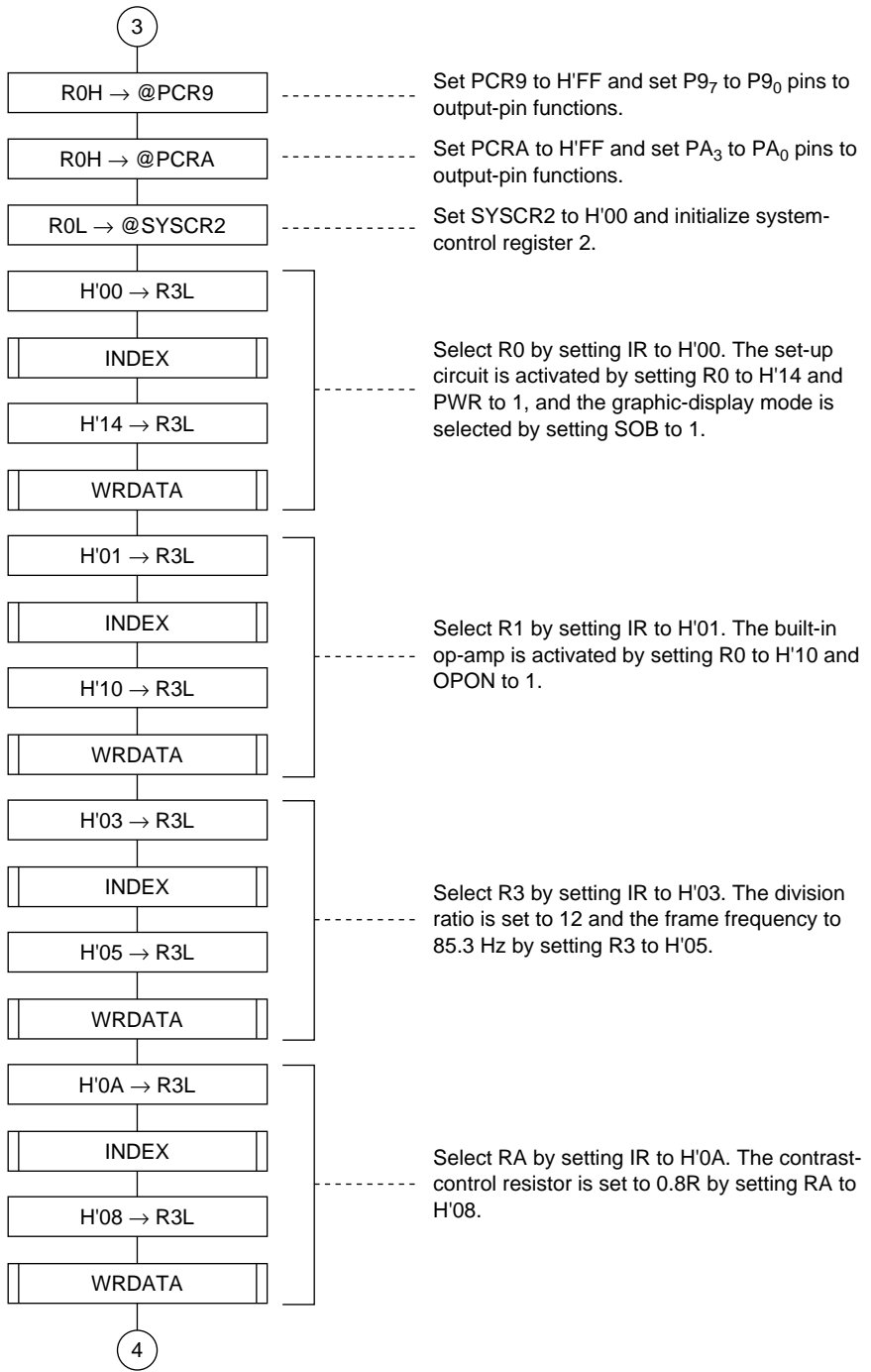
R1 = H'060A

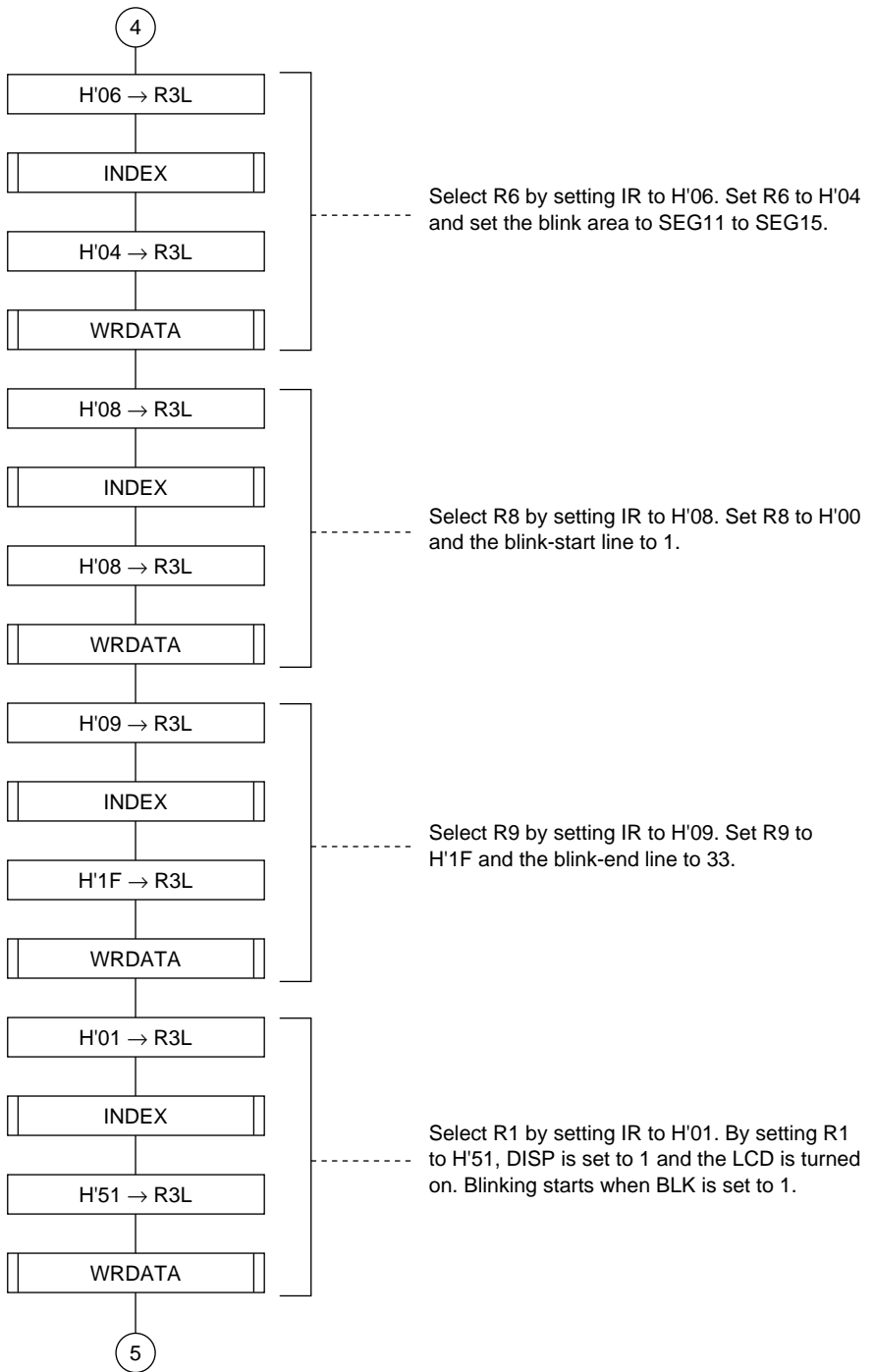
R2 = H'0000

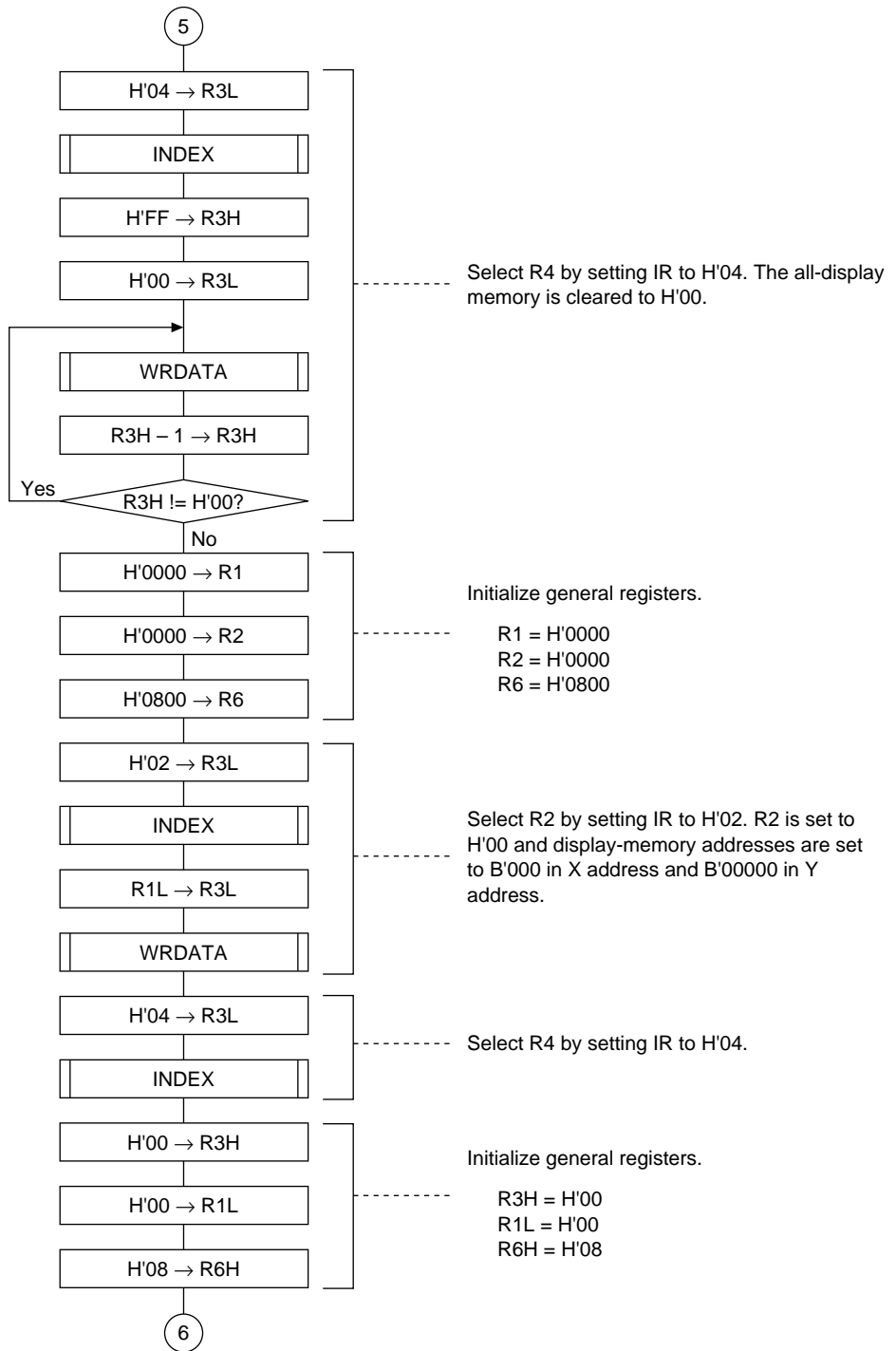
R3 = H'0000

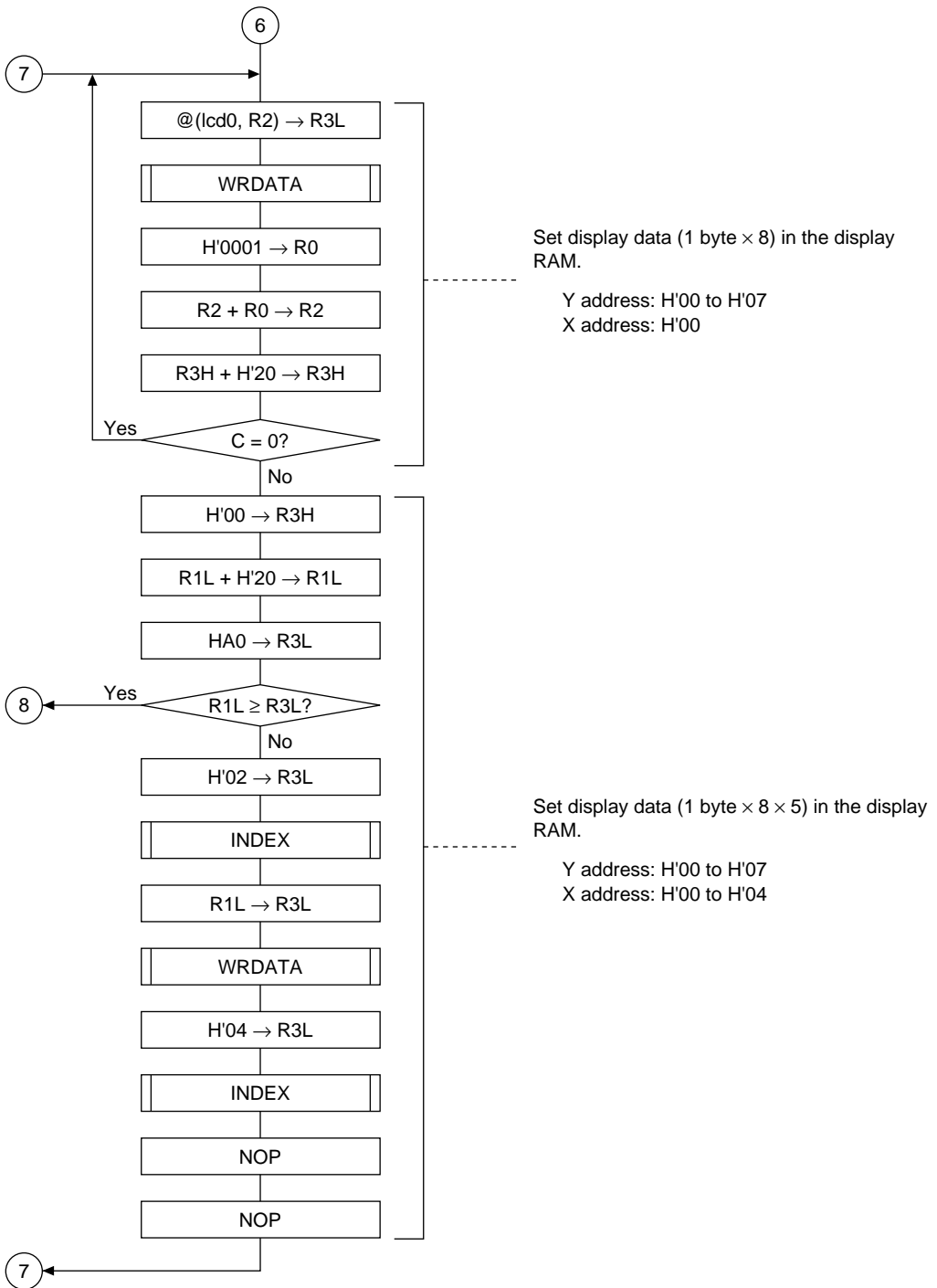
R4 = H'0001

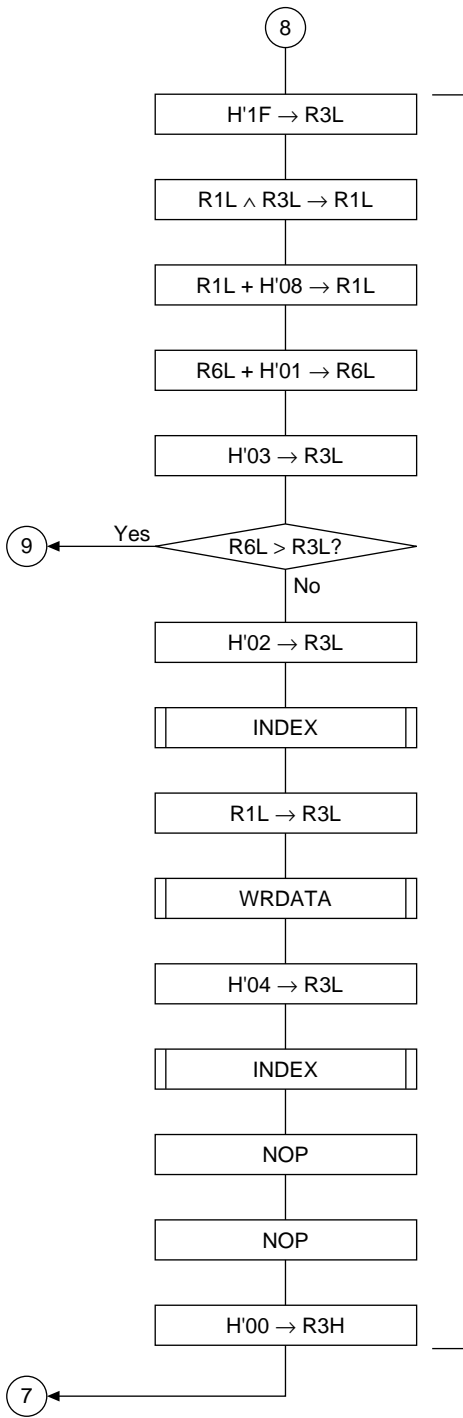
R5 = H'0405





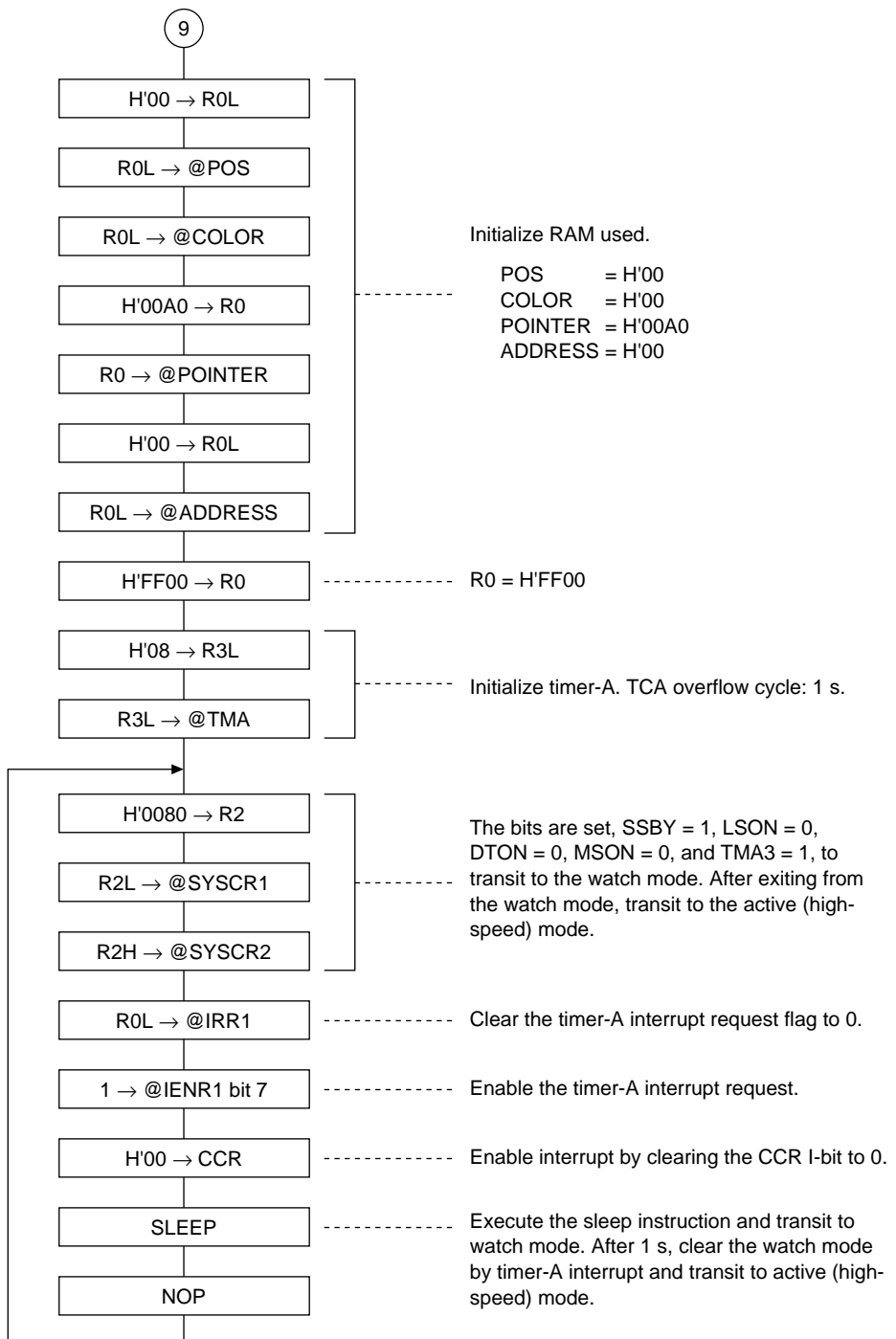




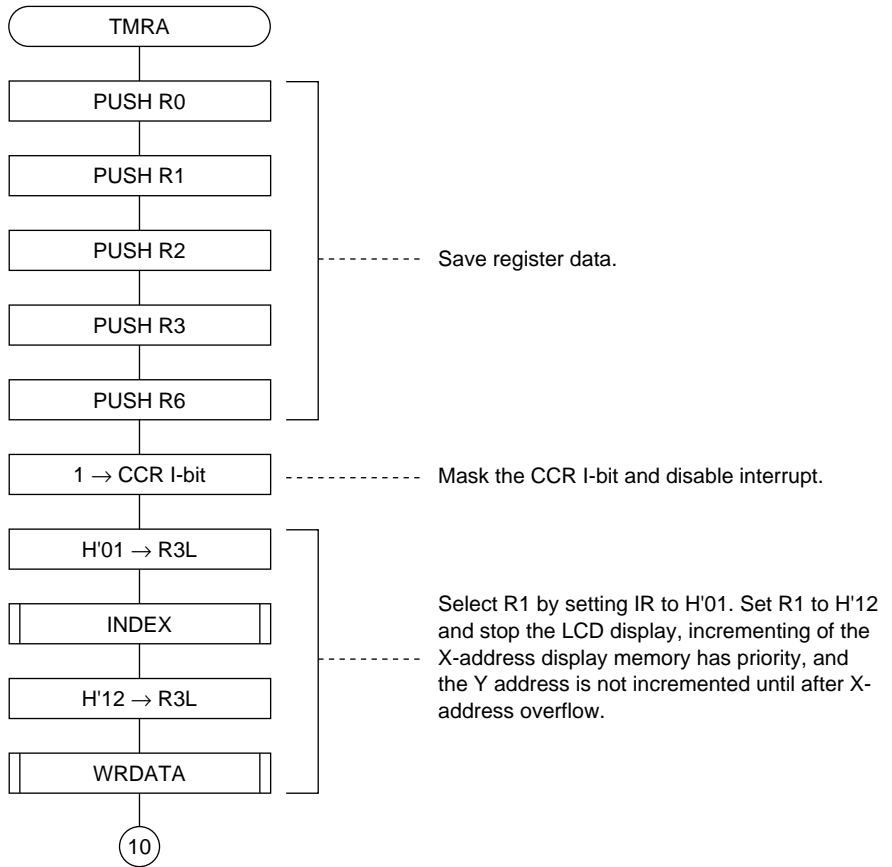


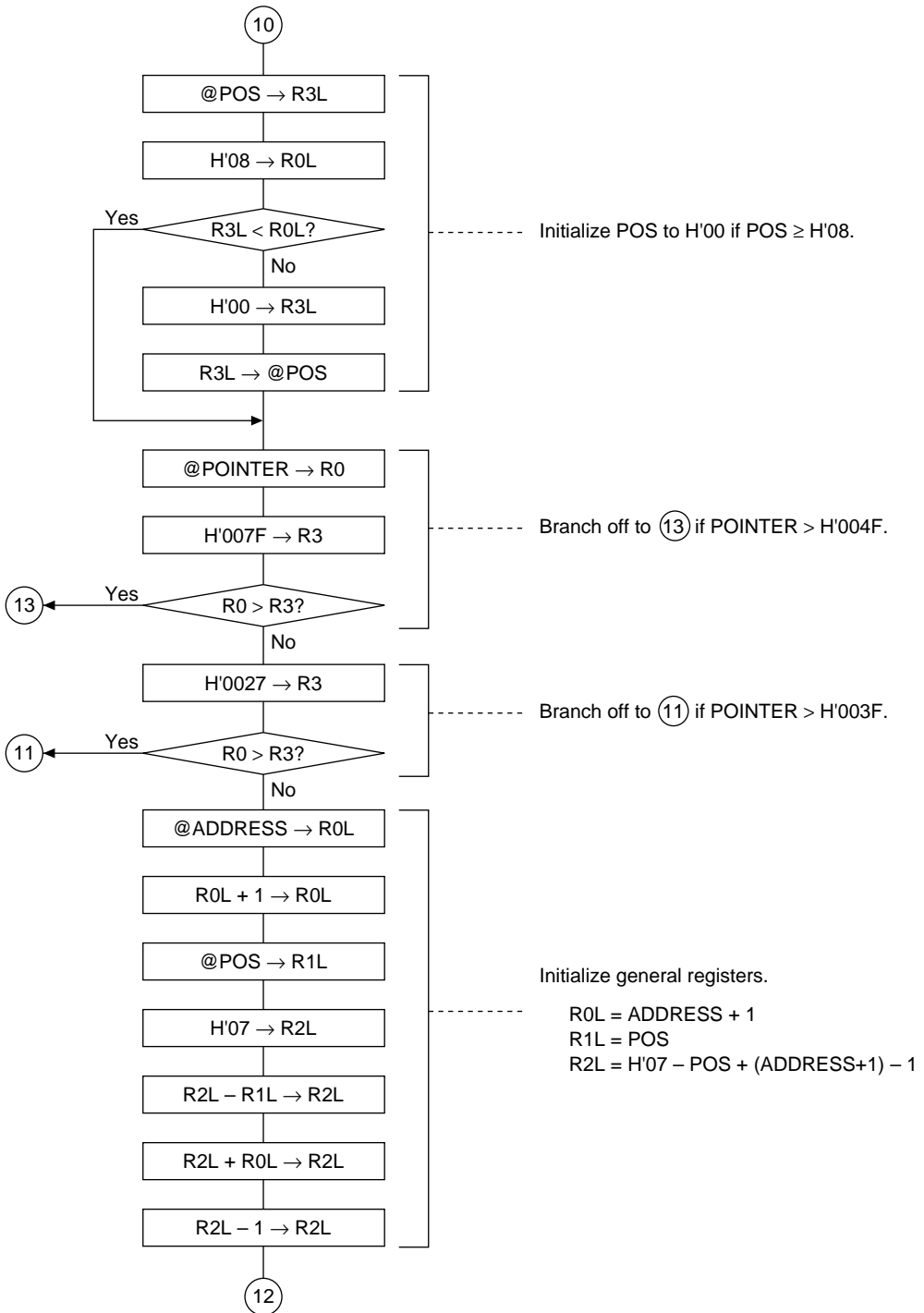
Set display data (1 byte × 8 × 5 × 4) in the display RAM.

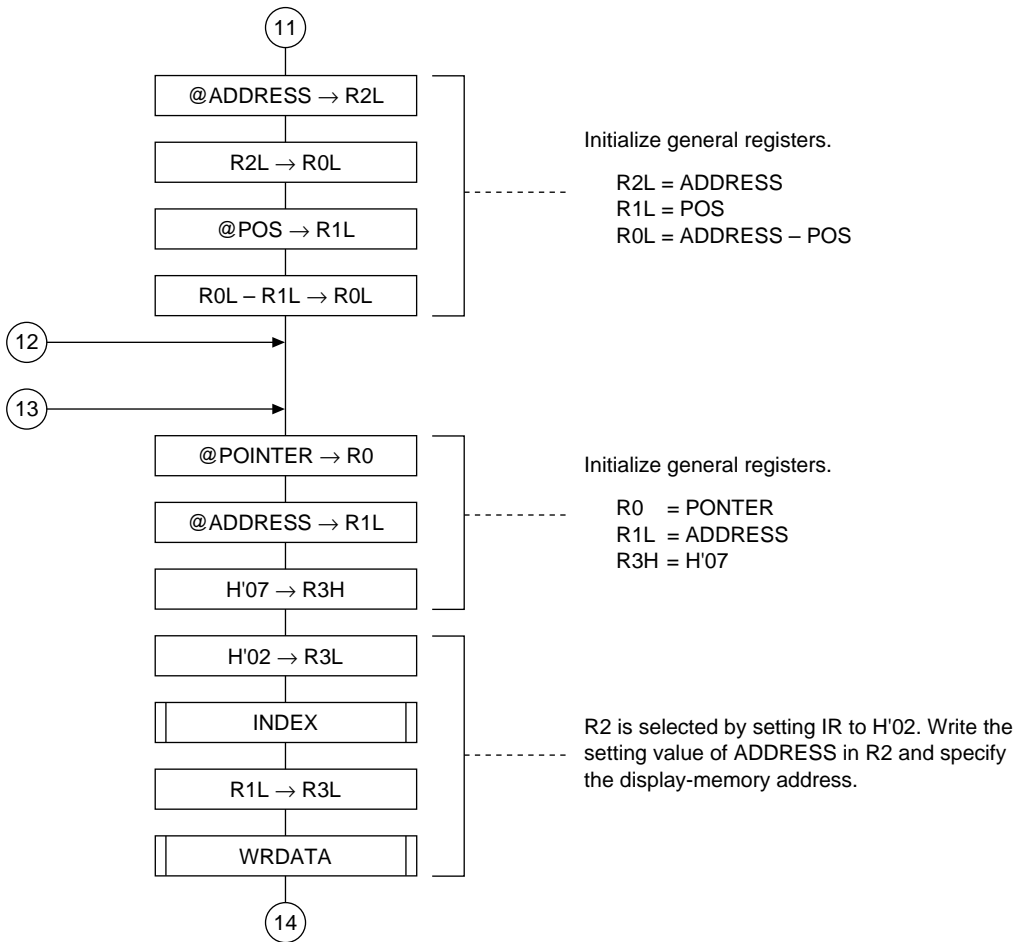
Y address: H'00 to H'1F
X address: H'00 to H'04

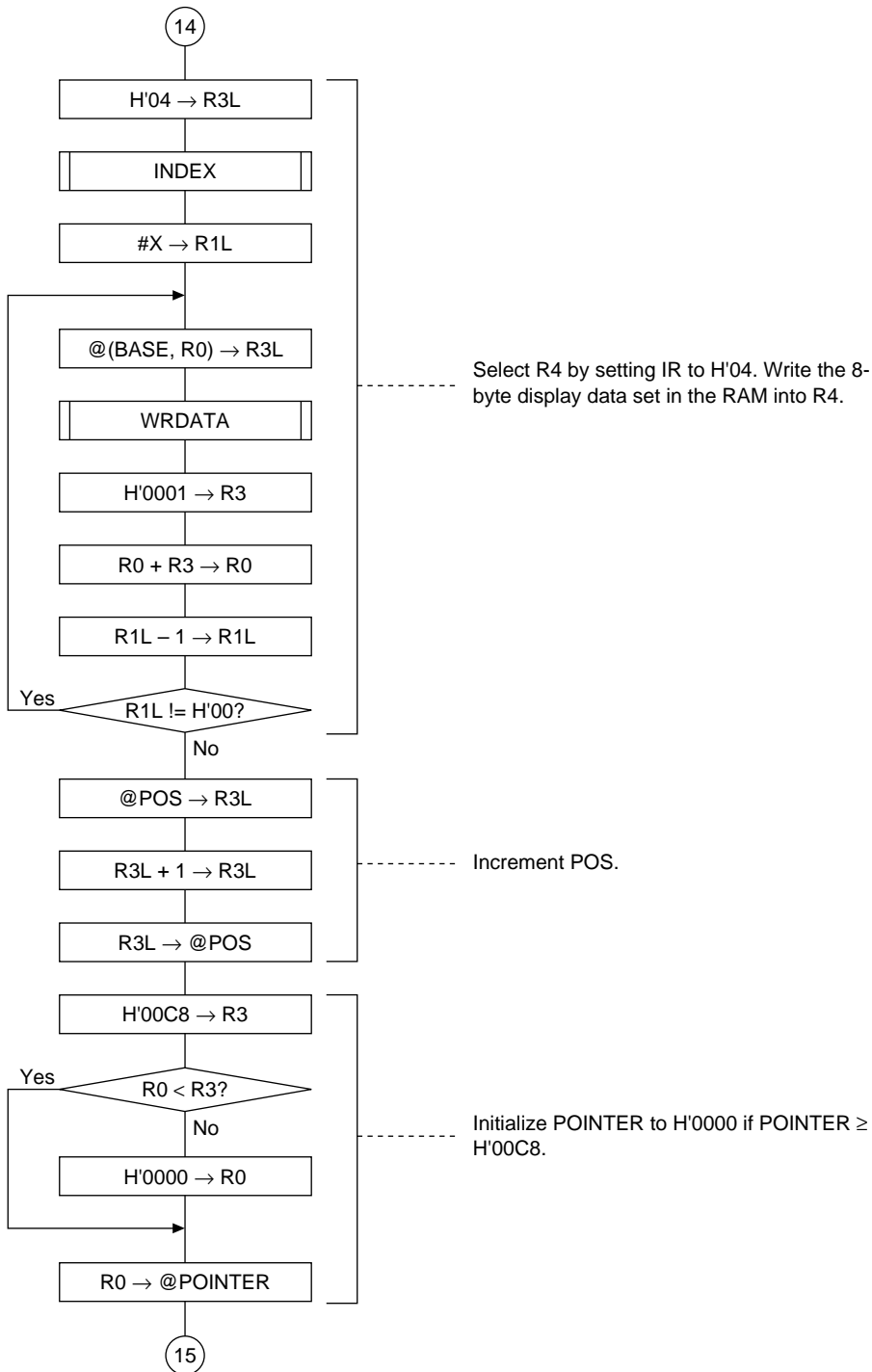


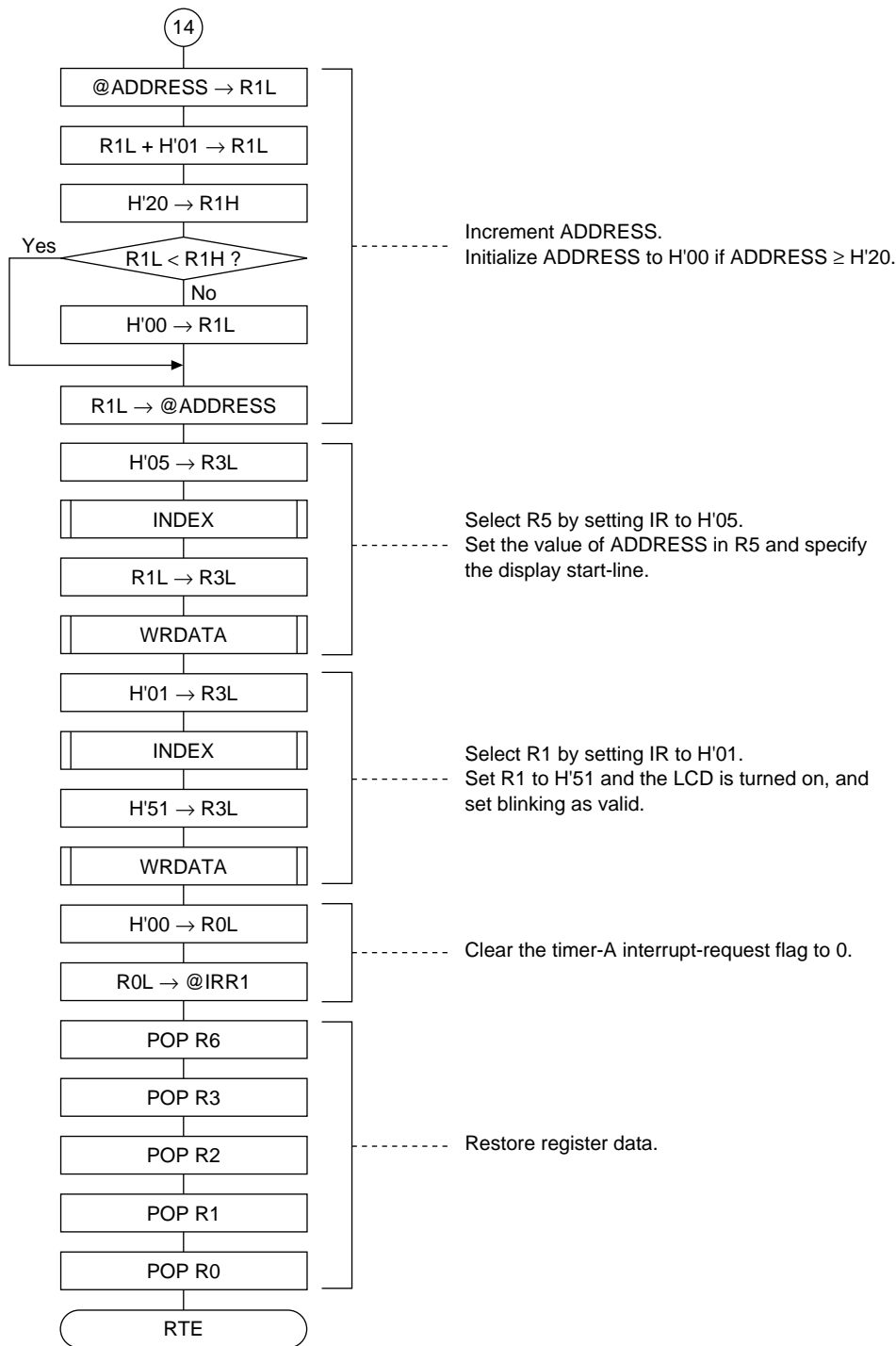
2. Timer-A interrupt-processing routine



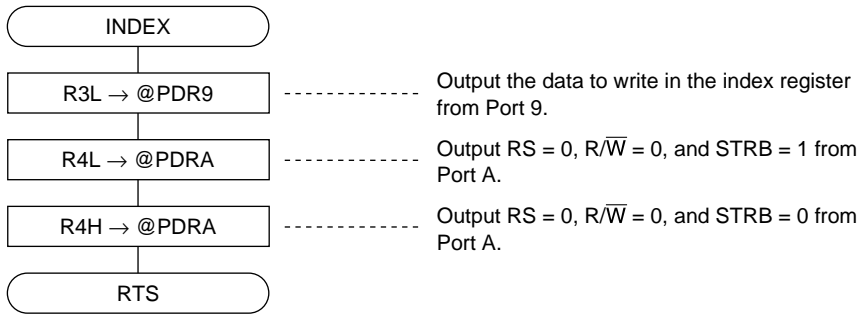




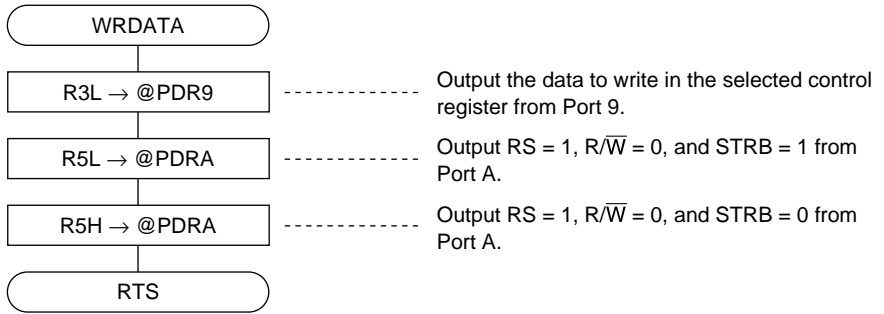




3. Index-register select routine



4. LCD-control register data write routine



Program Lists

```
;*****
;   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 ;WKP0 Vector
.DATA.W INIT ;SCI1 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
.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'0F800,R0        ;STORE THE ROM DATA IN THE RAM STARTING
                                           ;FROM #0F800

        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        @(lcd0,R1),R3L       ;THE DISPALY DATA IN THE ROM
                                           ;STARTING FROM #LCD0

        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'0FF00,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    R0H,@PCR9:8   ;
        MOV.B    R0H,@PCRA:8   ;
        MOV.B    R0L,@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
        mov.w   #h'0800,r6
        mov.b   #h'02,r31
        jsr     @INDEX
        mov.b   r11,r31

```



```

jsr      @WRDATA
mov.b    #h'4,r31
jsr      @INDEX
mov.b    #h'0,r3h
mov.b    #h'00,r11
mov.b    #8,r6h

;
gg1:
mov.b    @(lcd0,r2),r31
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,r11
MOV.B    #H'0A0,R3L
CMP.B    R3L,R1L
bcc      gg3
gg23:
mov.b    #h'02,r31
jsr      @INDEX
mov.b    r11,r31
jsr      @WRDATA
mov.b    #h'4,r31
jsr      @INDEX
nop
nop
jmp      @gg1
gg3:
mov.b    #h'1f,r31
and.b    r31,r11
add.b    #h'08,r11
add.b    #h'1,r61
mov.b    #h'3,r31
cmp.b    r31,r61

```

```

        bhi            TIMA
;
        mov.b         #'02,r31
        jsr           @INDEX
        mov.b         r11,r31
        jsr           @WRDATA
        mov.b         #'4,r31
        jsr           @INDEX
        nop
        nop
        mov.b         #0,r3h
        jmp           @gg1
;
;*****
; SET THE VARIABLE
;*****
;
TIMA:
        MOV.B         #'0,R0L
        MOV.B         R0L,@POS
        MOV.W         #'A0,R0
        MOV.W         R0,@POINTER
        MOV.B         #'00,R0L
        MOV.B         R0L,@ADDRESS
        mov.w         #'0ff00,r0
        MOV.B         #'08,R3L           ;SET THE TIMERA
        MOV.B         R3L,@TMA:8
;
;*****
; SLEEP
;*****
;
SLEEP:
        MOV.W         #'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         R0L,@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
;*****
;
    MOV.B     @POS,R3L
    MOV.B     #H'8,R0L
    CMP.B     R0L,R3L
    BLO       NOPOS
    MOV.B     #H'0,R3L
    MOV.B     R3L,@POS

NOPOS:
    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 ,R0L
INC        R0L
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,r01
mov.b    r01,@IRR1
;
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

.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
.DATA.B H'80

LCD2:

;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

.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
.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

;

.END

H8/3857 Series Application Note

Publication Date: 1st Edition, August 1999

Published by: Electronic Devices Sales & Marketing Group
Semiconductor & Integrated Circuits
Hitachi, Ltd.

Edited by: Technical Documentation Group
UL Media Co., Ltd.

Copyright © Hitachi, Ltd., 1999. All rights reserved. Printed in Japan.