

**TFT LCD Module with
Metallized Projective Capacitive Touch Panel
(FLETAS touch panel)**

GT-CP series

(Command control type)

“General Function” Software Specification

**Model: GT-C9xxPA
GTxxxxxCxxxxPA**

Specification No.:	DS-2007-0009-01	
Date of Issue:	July 21, 2021	(00)
Revision:	August 11, 2021	(01)

Published by
NORITAKE ITRON Corp. / Japan
<https://www.noritake-itron.jp>

This specification is subject to change without prior notice.

Contents

1	General Description	5
1.1	Scope	5
1.2	Related specifications	5
1.3	Functions.....	5
2	Operating Mode	6
2.1	Normal Command Mode	6
2.2	User Setup Mode	6
2.3	Touch Panel Calibration Mode	6
2.4	Memory Re-writing Mode	6
2.5	Test Mode.....	6
2.6	Power-on Settings	6
2.7	Timing Unit	6
2.8	Display Memory Configuration	7
3	Normal Command Mode	8
3.1	Displayable Image Types	8
3.1.1	Graphic Display	8
3.1.2	Character Display.....	8
3.2	Display orientation.....	9
3.3	Memory	10
3.3.1	Pixel format	11
3.3.2	Display Memory < Firmware version: F200, or later >	12
3.3.3	Display Memory < Firmware version: less than F200 >	16
3.3.4	Bit Image and Font Definition Memory	18
3.3.5	General-purpose Memory.....	18
3.4	Cursor	19
3.5	Window	20
3.5.1	Base-window.....	20
3.5.2	User-window.....	20
3.6	Write Screen Mode.....	21
3.6.1	Display Screen Mode	21
3.6.2	All Screen Mode	21
3.7	Display Control Commands.....	22
3.7.1	Code Set	22
3.7.1.1	Character Code.....	22
3.7.1.2	Control Code.....	22
3.7.2	Detail of Code Set	22
3.7.2.1	Character Display	22
3.7.2.2	BS (Back Space).....	26
3.7.2.3	HT (Horizontal Tab)	26
3.7.2.4	LF (Line Feed)	27
3.7.2.5	HOM (Home Position)	27
3.7.2.6	CR (Carriage Return).....	27
3.7.2.7	CLR (Display Clear)	27
3.7.2.8	CAN (Line Clear).....	27
3.7.2.9	RCLR (Line End Clear)	27
3.7.3	Command Set	28
3.7.3.1	General Setting Commands.....	28
3.7.3.2	Character Display Setting Commands	28
3.7.3.3	Display Action Setting Commands	30
3.7.3.4	Bit Image Display Setting Commands.....	31
3.7.3.5	General Display Setting Commands	32
3.7.3.6	Window Display Setting Commands	32
3.7.3.7	Download Character Setting Commands	33
3.7.3.8	User Setup Mode Setting Commands.....	34
3.7.3.9	General-purpose I/O Port Control Commands	34
3.7.3.10	Macro Setting Commands.....	34
3.7.3.11	Other Setting Commands.....	35
3.7.3.12	Touch Panel Control Commands	36
3.7.4	Command Set Details.....	37
3.7.4.1	US X n (Brightness Level Setting (for Backlight)).....	37
3.7.4.2	ESC @ (Initialize Display).....	37
3.7.4.3	US \$ x y (Cursor Set).....	37
3.7.4.4	US (w 10h a (Write Screen Mode Select)	37
3.7.4.5	ESC R n (International Font Set)	38
3.7.4.6	ESC t n (Character Table Type)	38
3.7.4.7	US MD1 (Over-write Mode).....	39
3.7.4.8	US MD2 (Vertical Scroll Mode)	39
3.7.4.9	US MD3 (Horizontal Scroll Mode)	39
3.7.4.10	US MD5 (Horizontal Scroll Mode, Scroll ON)	39

3.7.4.11	US s n (Horizontal Scroll Speed)	39
3.7.4.12	US (g 01h m (Font Size Select)	39
3.7.4.13	US (g 02h m (2-byte Character)	40
3.7.4.14	US (g 03h m (2-byte Character Type)	40
3.7.4.15	US (g 06 yLS yLSH ySL ySH xSL xSH boL boH (Outline Font Size)	41
3.7.4.16	US (g 07 ad0 ad1 ad2 ad3 sz0 sz1 sz2 sz3 (User-supplied Font File Address and Size).....	42
3.7.4.17	US (g 08 n (Outline Font Type Select)	42
3.7.4.18	US (g 04h m (Font Width)	43
3.7.4.19	US (g 05h n (FROM Extended Font)	44
3.7.4.20	US (g 40h x y (Font Magnification).....	44
3.7.4.21	US (g 41h b (Character Style)	44
3.7.4.22	US (g 50h pR pG pB (Character Color).....	44
3.7.4.23	US (g 51h pR pG pB (Background Color)	44
3.7.4.24	US (g 52h pR pG pB (Shadow and Bordering Color)	45
3.7.4.25	US (g 58h b (Background Color Enable/ Disable)	45
3.7.4.26	US (a 01h t (Wait)	45
3.7.4.27	US (a 02h t (Short Wait)	45
3.7.4.28	US (a 11h p t1 t2 c (Blink)	46
3.7.4.29	US (a A0h sXL sXH sYL sYH cL cH s (Scroll Display Action XY)	46
3.7.4.30	US (a A2h v s pR pG pB (Curtain Display Action XY)	47
3.7.4.31	US (a A3h v s pXL pXH pYL pYH (Spring Display Action XY)	48
3.7.4.32	US (a A4h v s pXL pXH pYL pYH (Random Display Action XY)	49
3.7.4.33	US (a A5h s pXL pXH pYH pYL (Fade In Display Action XY)	50
3.7.4.34	US (a A6h s (Fade Out Display Action XY)	50
3.7.4.35	US (a 40h p (Display Power ON/ OFF)	50
3.7.4.36	US (d 10h pen xL xH yL yH (Pixel Drawing).....	50
3.7.4.37	US (d 11h mode pen x1L x1H y1L y1H x2L x2H y2L y2H (Line/ Box Pattern Drawing).....	51
3.7.4.38	US (f 11h xL xH yL yH fmt d(1)...d(n) (Real-time Bit Image Display)	52
3.7.4.39	US (f 21h xL xH yL yH fmt d(1)...d(n) (Packaged Real-time Bit Image Display)	53
3.7.4.40	US (f 01h aL aH aE sL sH sE d(1)...d(s) (RAM Bit Image Definition).....	54
3.7.4.41	US (e 10h aL aH aE sL sH sE d(1)...d(s) (Bit Image FROM1 Definition).....	55
3.7.4.42	US (f 10h m aL aH aE xSL xSH xL xH yL yH fmt (Downloaded Bit Image Display)	56
3.7.4.43	US (f 20h m aL aH aE pL pH fmt (Packaged Downloaded Bit Image Display)	59
3.7.4.44	US r n (Reverse Display)	61
3.7.4.45	US w n (Write Mixture Display Mode)	61
3.7.4.46	US (w 01h a (Current-window Select)	61
3.7.4.47	US (w 02h a b[xPL xPH yPL yPH xSL xSH ySL ySH] (User Window Define/ Cancel)	62
3.7.4.48	ESC % n (Download Character ON/ OFF)	63
3.7.4.49	ESC & a c1 c2 [x1 d1...d(y×x1)] ... [xk d1...d(y×xk) (Download Character Definition).....	63
3.7.4.50	ESC ? a c (Downloaded Character Delete).....	63
3.7.4.51	US (g 10h c1 c2 d1...dk (16×16 Download Character Definition).....	64
3.7.4.52	US (g 11h c1 c2 (16×16 Downloaded Character Delete)	64
3.7.4.53	US (g 14h c1 c2 d1...d128 (32×32 Download Character Definition).....	64
3.7.4.54	US (g 15h c1 c2 (32×32 Downloaded Character Delete)	65
3.7.4.55	US (e 11h a (Download Character Save)	65
3.7.4.56	US (e 21h a (Download Character Restore)	65
3.7.4.57	US (e 13h m P(80h-1) P(80h-2) ... P(FFh-n) (FROM User Font Definition)	65
3.7.4.58	US (e 15h a b p(1) ... p(65536) (FROM Extension Font Definition).....	66
3.7.4.59	US (e 01h d1 d2 (User Setup Mode Start)	66
3.7.4.60	US (e 02h d1 d2 d3 (User Setup Mode End)	66
3.7.4.61	US (p 01h n a (I/O Port Input/ Output Setting)	67
3.7.4.62	US (p 10h n a (I/O Port Output)	67
3.7.4.63	US (p 20h n (I/O Port Input)	67
3.7.4.64	US : pL pH [d1 ... dk] (RAM Macro Define/ Delete)	68
3.7.4.65	US (e 12h a pL pH t1 t2 [d(1) ... d(p)] (FROM Macro Define/ Delete)	68
3.7.4.66	US ^ a t1 t2 (Macro Execution)	70
3.7.4.67	US (i 20h a b c (Macro End Condition).....	70
3.7.4.68	US (e 03h a b (Memory SW Setting).....	71
3.7.4.69	US (e 04h a (Memory SW Data Send).....	71
3.7.4.70	US (e 18h sL sH sE m1 a1L a1H a1E d[1] ... d[s] (General-purpose Memory Store/ Bit Image FROM2 Store)	72
3.7.4.71	US (e 19h sL sH sE m1 a1L a1H a1E m2 a2L a2H a2E (General-purpose Memory Transfer)	73
3.7.4.72	US (e 28h sL sH sE m1 a1L a1H a1E (General-purpose Memory Send).....	74
3.7.4.73	US (e 40h a [b c] (Display Status Send).....	75
3.7.4.74	FS M m d1 ... d6 (Memory Re-writing Mode Start)	75
3.7.4.75	US (a 48h m w (Power Saving Mode)	76
3.8	Bit Image Data Format	77
3.8.1	Normal Bit Image Data Format	77
3.8.2	Packaged Bit Image Data Format	78
3.9	Color Combination Format	80

3.9.1	Character Display Color Combination Format	80
3.9.2	Monochrome Bit Image Color Combination Format	80
3.9.3	Color Image / BMP File Color Combination Format	80
3.9.4	Color Combination Examples	81
3.10	Download Character Format	82
3.11	Touch Panel	83
3.11.1	Outline	83
3.11.1.1	Touch Detection	83
3.11.2	Basic Operation	84
3.11.3	Touch Modes	84
3.11.4	Control Modes	87
3.11.4.1	Coordinates Mode	87
3.11.4.2	Matrix Switch Mode	87
3.11.4.3	Custom Switch Mode	87
3.12	Touch Panel Commands	88
3.12.1	US P 01h n (Touch Mode Selection: Single-Touch Mode/ Multi-Touch Mode)	88
3.12.2	US P 10h ch md (Coordinates Mode)	88
3.12.3	US P 10h ch md nx cx ny cy (Switch Matrix Mode)	89
3.12.4	US P 10h ch md sn px1L px1H py1L py1H sx1L sx1H sy1L sy1H [... px(sn)L px(sn)H py(sn)L py(sn)H sx(sn)L sx(sn)H sy(sn)L sy(sn)H] (Custom Switch Mode)	90
3.12.5	US P 20h m (Touch Panel Data Transmit ON/ OFF)	91
3.12.6	US P 21h ch (Touch Panel Channel Select)	92
3.12.7	US K 70h a [b [c]] (Touch Parameter Setting)	92
3.12.7.1	Threshold and Gain (a= 00h/ a = 04h)	92
3.12.7.2	Touch Standard Reference Related Command (a= 06h/ 07h/ 08h)	93
3.12.8	Touch Setting Package Data Store US (e 1Ch a d[1] ... d[1024]	94
3.12.9	Touch Setting Package Selection US K 70h 10h a	94
3.12.10	Touch Level Read US K 70h a	94
3.12.11	Touch Panel Control Data Transmit Mode	94
3.12.12	Touch Scan Period Setting for Power Saving Mode	94
4	Setup	95
4.1	Jumper	95
4.1.1	Display Address (I ² C Interface)	95
4.1.2	Baud Rate (UART Interface)	95
4.1.1	Serial Interface Type	95
4.1.2	Program Macro Start	95
4.1.3	Operating Mode	95
4.2	Memory SW	96
5	Lot No. / Firmware Version Specification	97
6	Firmware Version Revision History	98
	Revision Note	99

1 General Description

1.1 Scope

This specification covers the software aspects and supported functions of the Noritake Itron TFT-LCD display module with Metallized Projected-Capacitive Touch (MPCT) panel, GT-CP series.

The TFT-LCD module supports text and graphic display using simple ASCII-based commands, compatible with the GU-3000 series VFD modules produced by Noritake Itron.

Extra commands, and additional parameters to existing commands, are implemented in order to take advantage of the full-color display capability.

The supported firmware version varies depending on the part number. For the firmware version, please refer to 6 Firmware Version Revision History.

1.2 Related specifications

Hardware specification: (Refer to the following list)

Program macro specification: DS-1940-0005-xx

(Refer to 3.7.4.64 RAM Macro Define/ Delete)

(Refer to 3.7.4.65 FROM Macro Define/ Delete)

Character font specifications: (Refer to 3.1.2 Character Display)

Difference by type: (Product name listed is representative of the series)

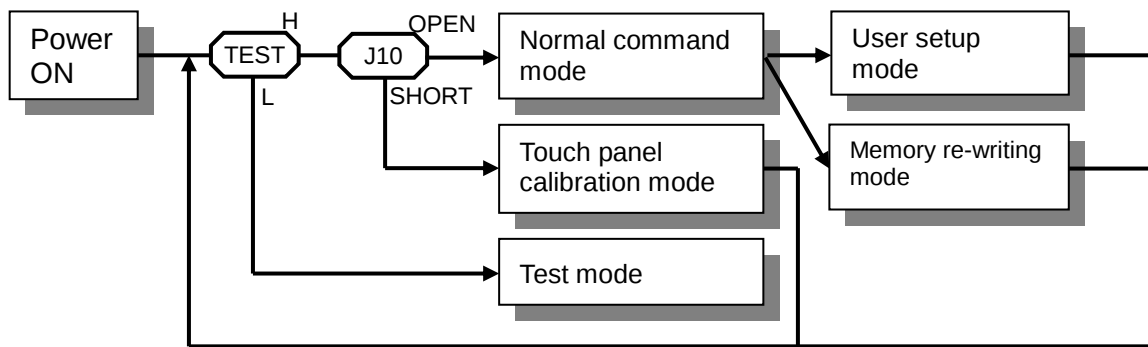
Product name	Display size	Pixels	Colors	Hardware specification number
GT480X272A-C903PA	4.3 inch equivalent	480×272	16.7M 24bit(RGB 8-8-8)	DS-2035-0000-xx
GT800X480A-C903PA	7.0 inch equivalent	800×480	65K 16bit(RGB 5-5-5 + 1bit)	DS-2007-0000-xx
GTWQ043C3A00PA	4.3 inch equivalent	480×272	16.7M 24bit(RGB 8-8-8)	DS-2077-0000-xx
GTWV050C3A00PA	5.0 inch equivalent	800×480	65K 16bit(RGB 5-5-5 + 1bit)	DS-2057-0000-xx
GTWV070C3A00PA	7.0 inch equivalent	800×480	65K 16bit(RGB 5-5-5 + 1bit)	DS-2082-0000-xx

1.3 Functions

Character display, graphic display, control command, display action command, download (user-definable) font, user-definable font table function, draw command, window function, general-purpose I/O port control, macro, program macro function, bit image download function, Memory SW, data storage, display orientation function, power saving function, touch panel control command.

2 Operating Mode

The operating modes are as follows, selected by jumper settings or software command.



2.1 Normal Command Mode

Normal operation mode – the module can receive commands and data via the various interfaces.

2.2 User Setup Mode

This mode is used for saving Memory SW and various data to FROM.

2.3 Touch Panel Calibration Mode

The display module includes a touch position correction function, for minimizing any inconsistency between actual touch position and the internally-calculated position (refer to hardware specification for details).

2.4 Memory Re-writing Mode

Mode for re-writing firmware and built-in font data. Not for routine use.

2.5 Test Mode

Test for display and internal operation. Used for factory test.

Can also be used to temporarily disable automatic start of the Program Macro. For details, refer to “2.3 Program Macro End” in the Program Macro specification.

2.6 Power-on Settings

At power-on, the various display settings are set to default value, or value stored in Memory SW (Refer to 4.2 Memory SW).

If “restore at power-on” is enabled, the applicable content in FROM is transferred to RAM before starting standard operation.

If “FROM macro execution at power-on” is enabled, Macro or Program Macro is automatically executed.

2.7 Timing Unit

Timing unit length varies between different modules. The timing unit length for each module display dot size is shown below.

Timing unit (Typ.) $\pm 5\%$ <i>IntTime</i>	16.2ms
--	--------

Timing unit affects the timing of the following commands and operations:

3.7.4.11 US s n	(Horizontal Scroll Speed)
3.7.4.27 US (a 02h t	(Short Wait)
3.7.4.28 US (a 11h p t1 t2 c	(Blink)
3.7.4.29 US (a A0h sXL sXH sYL sYH cL cH s	(Scroll Display Action XY)
3.7.4.30 US (a A2h v s pR pG pB	(Curtain Display Action XY)
3.7.4.31 US (a A3h v s pXL pXH pYL pYH	(Spring Display Action XY)
3.7.4.32 US (a A4h v s pXL pXH pYL pYH	(Random Display Action XY)
3.7.4.33 US (a A5h s pXL pXH pYH pYL	(Fade In Display Action XY)
3.7.4.34 US (a A6h s	(Fade Out Display Action XY)
3.7.4.65 US (e 12h a pL pH t1 t2 [d(1) ... d(p)]	(FROM Macro Define/ Delete)
3.7.4.66 US ^ a t1 t2	(Macro Execution)

2.8 Display Memory Configuration

Display memory size and configuration varies between different modules. For each module, the following module-specific values are referred to throughout this specification:

Item	Description
DispXdots	The number of dots (pixels) in the X-direction (horizontal) on the display screen.
DispYdots	The number of dots (pixels) in the Y-direction (vertical) on the display screen.
Xdots	The number of dots (pixels) in the X-direction (horizontal) for the entire display memory.
Ydots	The number of dots (pixels) in the Y-direction (vertical) for the entire display memory.
Max_Xdot	Valid X-coordinate values range from 0 to Max_Xdot . [Xdots - 1]
Max_Ydot	Valid Y-coordinate values range from 0 to Max_Ydot . [Ydots - 1]
Max_Xdot_CurtWin	Maximum valid X-coordinate value for current window.
Max_Ydot_CurtWin	Maximum valid Y-coordinate value for current window.
DispMemSize	Size of display memory in bytes. (number of bytes)
Max_DispmemAddr	Valid display memory addresses range from 0 to Max_DispmemAddr . [DispMemSize - 1]

- The configuration for each pixel size is shown below.

< Firmware version: F200, or later >

Item \ Pixels		Orientation: 0° / 180°		Orientation: 90° / 270°	
		480×272	800×480	480×272	800×480
Display area [DispXdots × DispYdots]		480×272	800×480	272×480	480×800
Total area [Xdots × Ydots]		960×544	1600×960	544×960	960×1600
Display memory	Xdots	960 (03C0h)	1600 (0640h)	544 (0220h)	960 (03C0h)
	Ydots	544 (0220h)	960 (03C0h)	960 (03C0h)	1600 (0640h)
	Max_Xdot	959 (03BFh)	1599 (063Fh)	543 (021Fh)	959 (03BFh)
	Max_Ydot	543 (021Fh)	959 (03BFh)	959 (03BFh)	1599 (063Fh)
	Max_Xdot_CurtWin	0 to 959 * (0000h to 03BFh)	0 to 1599 * (0000h to 063Fh)	0 to 543 * (0000h to 021Fh)	0 to 959 * (0000h to 03BFh)
	Max_Ydot_CurtWin	0 to 543 * (0000h to 021Fh)	0 to 959 * (0000h to 03BFh)	0 to 959 * (0000h to 03BFh)	0 to 1599 * (0000h to 063Fh)
	DispMemSize [byte]	2,088,960 (1FE000h)	3,072,000 (2EE000h)	2,088,960 (1FE000h)	3,072,000 (2EE000h)
	Max_DispmemAddr	2,088,959 (1FDFFFh)	3,071,999 (2EDFFFh)	2,088,959 (1FDFFFh)	3,071,999 (2EDFFFh)

* Depends on size of current window.

< Firmware version: less than F200 >

Item \ Pixels		480×272	800×480
Display area [DispXdots × DispYdots]		480×272	800×480
Total area [Xdots × Ydots]		960×544	1600×960
Display memory	Xdots	960 (03C0h)	1600 (0640h)
	Ydots	544 (0220h)	960 (03C0h)
	Max_Xdot	959 (03BFh)	1599 (063Fh)
	Max_Ydot	543 (021Fh)	959 (03BFh)
	Max_Xdot_CurtWin	0 to 959 * (0000h to 03BFh)	0 to 1599 * (0000h to 063Fh)
	Max_Ydot_CurtWin	0 to 543 * (0000h to 021Fh)	0 to 959 * (0000h to 03BFh)
	DispMemSize [byte]	2,088,960 (1FE000h)	3,072,000 (2EE000h)
	Max_DispmemAddr	2,088,959 (1FDFFFh)	3,071,999 (2EDFFFh)

3 Normal Command Mode

3.1 Displayable Image Types

3.1.1 Graphic Display

Number of pixels: Refer to each specification.

3.1.2 Character Display

Character mode:

1-byte character: 6×8, 8×16, 12×24, 16×32 pixel character mode

2-byte character: 16×16, 32×32 pixel character mode

Outline (Scalable) font: Any displayable size

Built-in character font type:

1-byte character: 6×8, 8×16, 12×24, 16×32 pixel

– ANK, International font (specification DS-1600-0004-xx)

2-byte character: 16×16 pixel character

– Japanese Kanji (specification DS-906-0002-xx)

– Korean (specification DS-954-0008-xx)

– Simplified Chinese (specification DS-954-0006-xx)

– Traditional Chinese (specification DS-954-0007-xx)

2-byte character: 32×32 pixel character

– Japanese Kanji (specification DS-906-0003-xx)

Outline font (Pre-loaded): Source Han Sans (version 1.000)

– SourceHanSansCN-Normal.otf

– SourceHanSansJP-Normal.otf

– SourceHanSansKR-Normal.otf

– SourceHanSansTWHK-Normal.otf

(Refer to <https://github.com/adobe-fonts/source-han-sans/>)

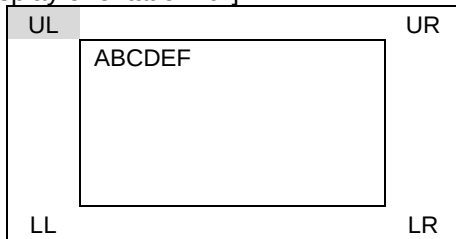
Standard fonts:

Font size	1-byte character		2-byte character			Outline Font
	International	Japanese	Korean	Simplified Chinese	Traditional Chinese	Source Han Sans
6×8	○	×	×	×	×	-
8×16	○	○ (16×16)	○ (16×16)	○ (16×16)	○ (16×16)	-
12×24	○	×	×	×	×	-
16×32	○	○ (32×32)	×	×	×	-
Any size	-	-	-	-	-	○

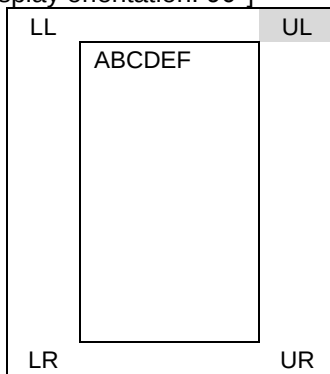
3.2 Display orientation

The orientation of the display can be set to "0° / 90° / 180° / 270°" by Memory SW No.6.
Refer to "4.2 Memory SW" for details.

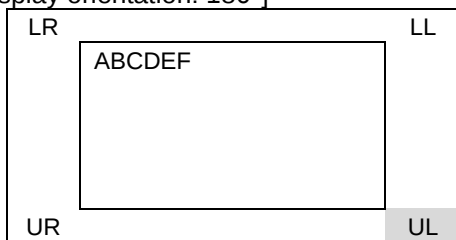
[Display orientation: 0°]



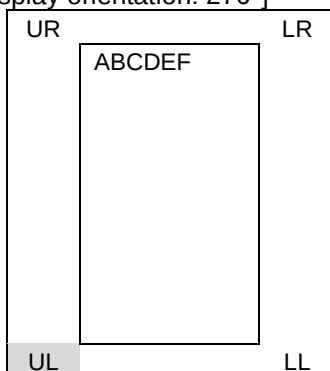
[Display orientation: 90°]



[Display orientation: 180°]



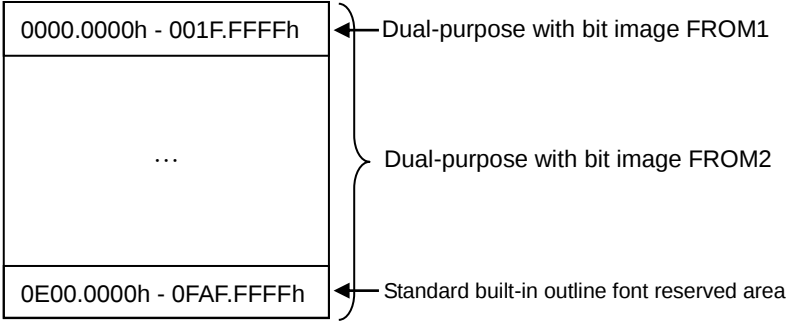
[Display orientation: 270°]



UL: Upper left, UR: Upper right, LR: Lower right, LL: Lower left

3.3 Memory

The following is the relationship of all installed memories.

Memory	Capacity
Display memory	For pixel 480×272: 2,040Kbytes (RAM) For pixel 800×480: 3,000Kbytes (RAM)
User-defined fonts memory	Download character: 16 characters of each size (RAM/ FROM) FROM user font: 128 characters of each size (FROM) FROM extension font: 64Kbytes (FROM)
Macro definition memory	RAM macro: 1Kbytes (RAM) FROM macro: < Firmware version: F220, or later > 64Kbytes (FROM) Note: 8k bytes × 8 < Firmware version: less than F220 > 32Kbytes (FROM) Note: 8k bytes × 4
Memory SW memory	64bytes (FROM)
General purpose memory	General purpose RAM: 1Kbytes (RAM) General purpose FROM: 64Kbytes (FROM) General purpose FROM2: 251Mbytes (FROM) 0000.0000h - 0FAF.FFFFh: Dual-purpose with 251Mbytes bit image FROM2 0000.0000h - 001F.FFFFh: Dual-purpose with 2Mbytes bit image FROM1 0E00.0000h - 0FAF.FFFFh: 27Mbytes standard built-in outline font reserved area <General purpose FROM2> 
Bit image memory	RAM bit image: 4Kbytes (RAM) Bit image FROM1: 2Mbytes Dual-purpose with general purpose FROM2 (0000.0000h - 001F.FFFFh) Bit image FROM2: 251Mbytes Dual-purpose with general purpose FROM2 (0000.0000h - 0FAF.FFFFh)

3.3.1 Pixel format

[For pixel 480×272]

Each pixel is 32-bits (4 bytes), composed of 8-bits for each of red, green, and blue as shown below.

Note that memory addressing is little-endian.

Display memory pixel format (4 bytes per pixel)

MSB															
b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
-	-	-	-	-	-	-	-	B bit7	B bit6	B bit5	B bit4	B bit3	B bit2	B bit1	B bit0
								LSB							
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
G bit7	G bit6	G bit5	G bit4	G bit3	G bit2	G bit1	G bit0	R bit7	R bit6	R bit5	R bit4	R bit3	R bit2	R bit1	R bit0

[For pixel 800×480]

Each pixel is 16-bits (2 bytes), composed of 5-bits for each of red, green, and blue, which are used as the upper 5 bits (b5...b1), and a single 'I' (Intensity) bit, which is used as a common least-significant bit (b0), as shown below. Note that memory addressing is little-endian.

Display memory pixel format (2 bytes per pixel)

MSB								LSB							
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
I bit0	B bit5	B bit4	B bit3	B bit2	B bit1	G bit5	G bit4	G bit3	G bit2	G bit1	R bit5	R bit4	R bit3	R bit2	R bit1

I: Intensity bit. Used as common least-significant bit (bit 0) for each of R, G, B.

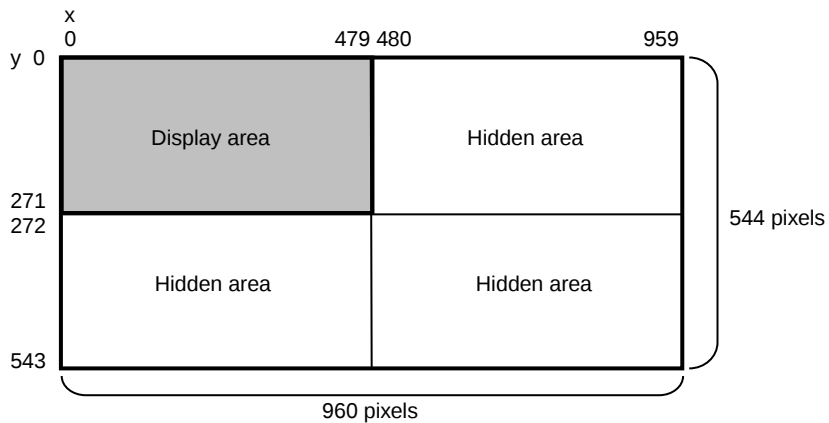
3.3.2 Display Memory < Firmware version: F200, or later >

Display memory is $Xdots \times Ydots$ – comprised of display area ($DispXdots \times DispYdots$) and hidden areas. By using “user window” function, the memory area can be separated, and each separate window can be controlled independently (refer to 3.7.4.47 User Window Define/ Cancel).

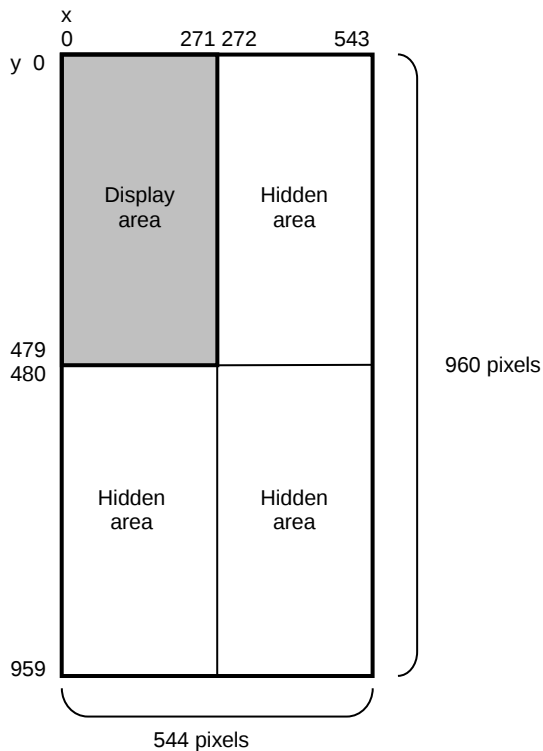
The “display area” shown below, at position $x=0$, $y=0$, is the portion of the display memory that is visible on the display screen. Data in hidden areas can be displayed using various display action commands.

[Display Memory Layout]

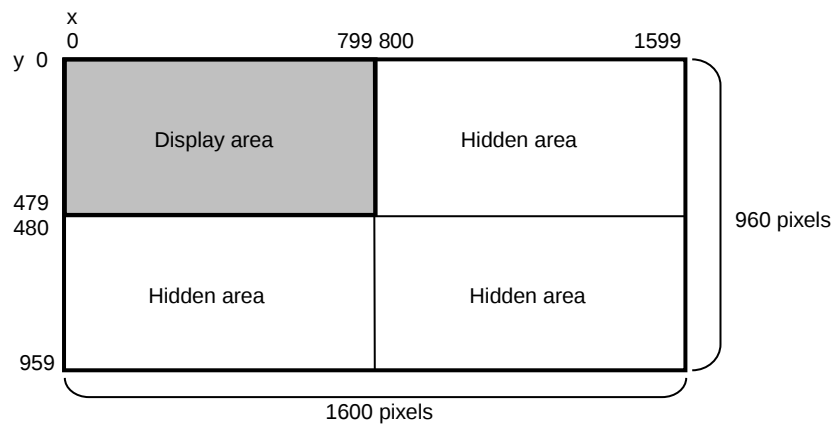
[For pixel 480×272 (Display orientation 0° / 180°)]



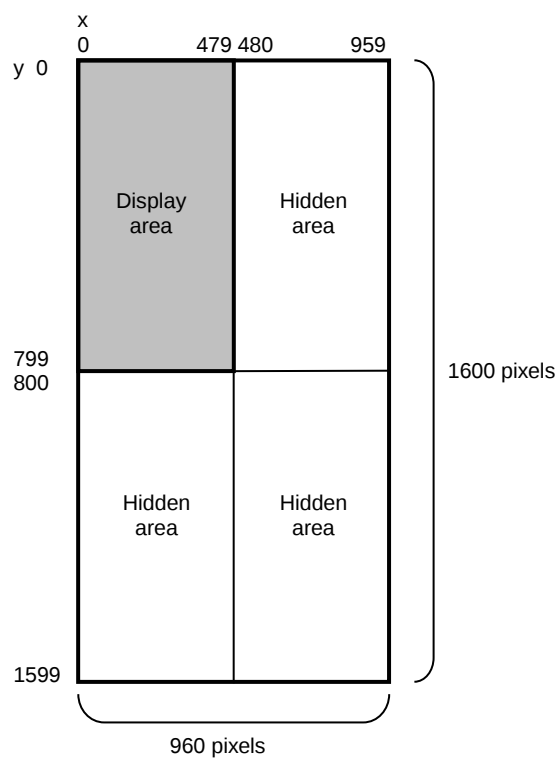
[For pixel 480×272 (Display orientation 90° / 270°)]



[For pixel 800×480 (Display orientation 0° / 180°)]



[For pixel 800×480 (Display orientation 90° / 270°)]



[Display Memory Address]

The following tables show the Display Memory address layout. The addresses shown within the table are expressed in units of one byte (eight bits).

Note: When using the Defined Bit Image Display command to display an image from the Display Memory (m=02), the address must be given in units of 1 pixel (not 1 byte). I.e., the addresses shown in the following tables would need to be divided by the number of bytes per pixel (4 or 2).

[For pixel 480×272]

Memory address zero corresponds to the top left pixel of the display memory, and memory addresses increment (by 4 for each pixel) to the right, then continuing on the next lower row, as shown below.

Display memory address layout [Display orientation 0° / 180°]:

	x 0	1	-	479	480	481	-	959
y 0	00.0000h	00.0004h	-	00.077Ch	00.0780h	00.0784h	-	00.0EFCh
1	00.0F00h	00.0F04h	-	00.167Ch	00.1680h	00.1684h	-	00.1DFCh
...	...	Display area	...	...	...	Hidden area	...	...
271	0F.E100h	0F.E104h	-	0F.87CEh	0F.E880h	0F.E884h	-	0F.EFFCh
272	0F.F000h	0F.F004h	-	0F.96CEh	0F.96D0h	0F.96D4h	-	0F.FEFCh
273	0F.FF00h	0F.FF04h	-	0F.A5CEh	0F.A5D0h	0F.A5D4h	-	10.0DFCh
...	...	Hidden area	...	...	...	Hidden area	...	...
543	1F.D100h	1F.D104h	-	1F.D87Ch	1F.D880h	1F.D884h	-	1F.DFFCh

Display memory address layout [Display orientation 90° / 270°]:

	x 0	1	-	271	272	273	-	543
y 0	00.0000h	00.0004h	-	00.043Ch	00.0440h	00.0444h	-	00.087Ch
1	00.0880h	00.0884h	-	00.0CBCh	00.0CC0h	00.0CC4h	-	00.10FCh
...	...	Display area	...	...	...	Hidden area	...	...
479	0F.E780h	0F.E784h	-	0F.EBBCh	0F.EBC0h	0F.EBC4h	-	0F.EFFCh
480	0F.F000h	0F.F004h	-	0F.F43Ch	0F.F440h	0F.F444h	-	0F.F87Ch
481	0F.F880h	0F.F884h	-	0F.FCBCh	0F.FCC0h	0F.FCC4h	-	10.00FCh
...	...	Hidden area	...	...	...	Hidden area	...	...
959	1F.D780h	1F.D784h	-	1F.DBBCh	1F.DBC0h	1F.DBC4h	-	1F.DFFCh

[For pixel 800×480]

Memory address zero corresponds to the top left pixel of the display memory, and memory addresses increment (by 2 for each pixel) to the right, then continuing on the next lower row, as shown below.

Display memory address layout [Display orientation 0° / 180°]:

	x	0	1	-	799	800	801	-	1599
y 0		00.0000h	00.0002h	-	00.063Eh	00.0640h	00.0642h	-	00.0C7Eh
1		00.0C80h	00.0C82h	-	00.12BEh	00.12C0h	00.12C2h	-	00.18FEh
...		...	Display area	...	...	Hidden area	...		...
479		17.6380h	17.6382h	-	17.69BEh	17.69C0h	17.69C2h	-	17.6FFEh
480		17.7000h	17.7002h	-	17.763Eh	17.7640h	17.7642h	-	17.7C7Eh
481		17.7C80h	17.7C82h	-	17.82BEh	17.82C0h	17.82C2h	-	17.88FEh
...		...	Hidden area	...	...	Hidden area	...		...
959		2E.D380h	2E.D382h	-	2E.D9BEh	2E.D9F0h	2E.D9F2h	-	2E.DFFEh

Display memory address layout [Display orientation 90° / 270°]:

	x	0	1	-	479	480	481	-	959
y 0		00.0000h	00.0002h	-	00.03BEh	00.03C0h	00.03C2h	-	00.077Eh
1		00.0780h	00.0782h	-	00.0B3Eh	00.0B40h	00.0B42h	-	00.0EFEh
...		...	Display area	...	...	Hidden area	...		...
799		17.6880h	17.6882h	-	17.6C3Eh	17.6C40h	17.6C42h	-	17.6FFEh
800		17.7000h	17.7002h	-	17.73BEh	17.73C0h	17.73C2h	-	17.777Eh
801		17.7780h	17.7782h	-	17.7B3Eh	17.7B40h	17.7B42h	-	17.7EFEh
...		...	Hidden area	...	...	Hidden area	...		...
1599		2E.D880h	2E.D882h	-	2E.DC3Eh	2E.DC40h	2E.DC42h	-	2E.DFFEh

3.3.3 Display Memory < Firmware version: less than F200 >

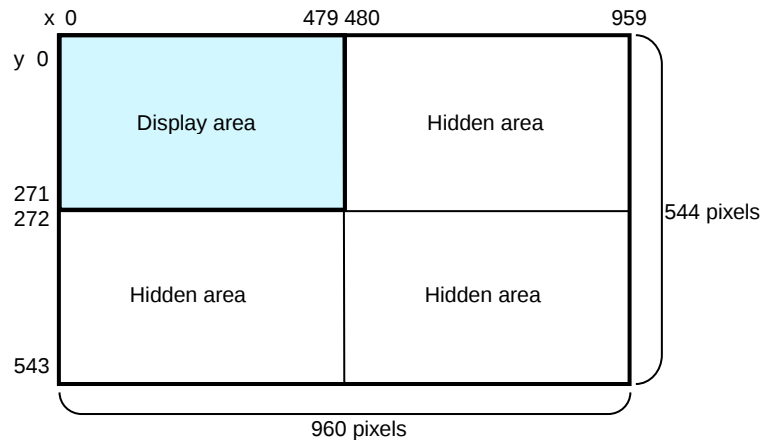
Display memory is $Xdots \times Ydots$ – comprised of display area ($DispXdots \times DispYdots$) and hidden areas.

By using “user window” function, the memory area can be separated, and each separate window can be controlled independently (refer to 3.7.4.47 User Window Define/ Cancel).

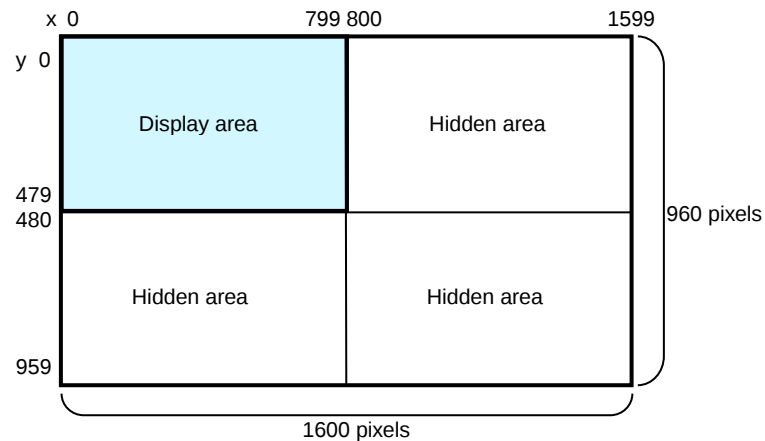
The “display area” shown below, at position $x=0$, $y=0$, is the portion of the display memory that is visible on the display screen. Data in hidden areas can be displayed using various display action commands.

[Display Memory Layout]

[For pixel 480×272]



[For pixel 800×480]



[Display Memory Address]

The following tables show the Display Memory address layout. The addresses shown within the table are expressed in units of one byte (eight bits).

Note: When using the Defined Bit Image Display command to display an image from the Display Memory (m=02), the address must be given in units of 1 pixel (not 1 byte). I.e., the addresses shown in the following tables would need to be divided by the number of bytes per pixel (4 or 2).

[For pixel 480×272]

Memory address zero corresponds to the top left pixel of the display memory, and memory addresses increment (by 4 for each pixel) to the right, then continuing on the next lower row, as shown below.

Display memory address layout

		x							
		0	1	-	479	480	481	-	959
y	0	00.0000h	00.0004h	-	00.077Ch	00.0780h	00.0784h	-	00.0EFCh
	1	00.0F00h	00.0F04h	-	00.167Ch	00.1680h	00.1684h	-	00.1DFCh
	...	...	Display area	...	...	...	Hidden area	...	...
	271	0F.E100h	0F.E104h	-	0F.87CEh	0F.E880h	0F.E884h	-	0F.EFFCh
	272	0F.F000h	0F.F004h	-	0F.96CEh	0F.96D0h	0F.96D4h	-	0F.FEFCCh
	273	0F.FF00h	0F.FF04h	-	0F.A5CEh	0F.A5D0h	0F.A5D4h	-	10.0DFCh
	...	...	Hidden area	...	...	...	Hidden area	...	...
	543	1F.D100h	1F.D104h	-	1F.D87Ch	1F.D880h	1F.D884h	-	1F.DFFCh

[For pixel 800×480]

Memory address zero corresponds to the top left pixel of the display memory, and memory addresses increment (by 2 for each pixel) to the right, then continuing on the next lower row, as shown below.

Display memory address layout

		x							
		0	1	-	799	800	801	-	1599
y	0	00.0000h	00.0002h	-	00.063Eh	00.0640h	00.0642h	-	00.0C7Eh
	1	00.0C80h	00.0C82h	-	00.12BEh	00.12C0h	00.12C2h	-	00.18FEh
	...	...	Display area	...	...	...	Hidden area	...	...
	479	17.6380h	17.6382h	-	17.69BEh	17.69C0h	17.69C2h	-	17.6FFEh
	480	17.7000h	17.7002h	-	17.763Eh	17.7640h	17.7642h	-	17.7C7Eh
	481	17.7C80h	17.7C82h	-	17.82BEh	17.82C0h	17.82C2	-	17.88FEh
	...	...	Hidden area	...	...	...	Hidden area	...	...
	959	2E.D380h	2E.D382h	-	2E.D9BEh	2E.D9F0h	2E.D9F2h	-	2E.DFFEh

3.3.4 Bit Image and Font Definition Memory

Bit image definition

Arbitrary bit image data can be defined and saved using bit image definition commands.

RAM bit image: 4,096 bytes

Bit image FROM1: 2,048K bytes

Bit image FROM2: 257,024K bytes (Note: Upper 27,648K used for standard built-in outline font)

The first 2MB of bit image FROM2 overlaps the bit FROM1 Bit Image memory; changing the content of either memory will affect the other.

Refer to 3.7.4.40 RAM Bit Image Definition, 3.7.4.41 Bit Image FROM1 Definition, and 3.7.4.70 Bit Image FROM2 Store. Note: Bit image FROM2 can also be used as general-purpose FROM2.

User-defined fonts

Memory for arbitrary user-defined fonts is available as follows.

Download character

For each of the font sizes 6×8, 8×16, 12×24, and 16×32 pixel (1-byte character), and 16×16 and 32×32 pixel (2-byte character), a maximum of 16 characters can be defined to memory space in RAM.

FROM user font

For each of the font sizes 6×8, 8×16, 12×24, and 16×32 pixel (1-byte character), a maximum of 128 characters can be defined to memory space in FROM.

FROM extension font

A number of custom font tables, for 1-byte character codes, with font Y-size 8, 16, 24, or 32 pixels, can be defined to memory space in FROM.

Refer to 3.7.4.49 Download Character Definition, 3.7.4.51 16×16 Download Character Definition, 3.7.4.53 32×32 Download Character Definition, 3.7.4.57 FROM User Font Definition, and 3.7.4.58 FROM Extension Font Definition.

Outline (Scalable) font

User-supplied font files can be used (refer to 3.7.4.16).

User-defined fonts summary:

Font size	1-byte character			2-byte character
	Download character	FROM user font	FROM extension font	Download character
6×8	○	○	○	×
8×16	○	○	○	○ (16×16)
12×24	○	○	○	×
16×32	○	○	○	○ (32×32)

3.3.5 General-purpose Memory

Arbitrary data can be stored to and retrieved from the memory.

General-purpose RAM: 1,024 bytes

General-purpose FROM: 4,096 bytes × 16 areas

General-purpose FROM2: 128K bytes × 2,008 areas

General-purpose RAM

000000h – 0003FFh

General-purpose FROM (1)

000000h – 000FFFh
001000h – 001FFFh
...
00F000h – 00FFFFh

General-purpose FROM2 (2)

00000000h – 0001FFFFh
00020000h – 0003FFFFh
...
0FAE0000h – 0FAFFFFFh

- (1) General-purpose FROM operation which would exceed a 4,096-byte memory area is not possible.
- (2) General-purpose FROM2 area from address 0E00.0000h is reserved for built-in outline fonts.

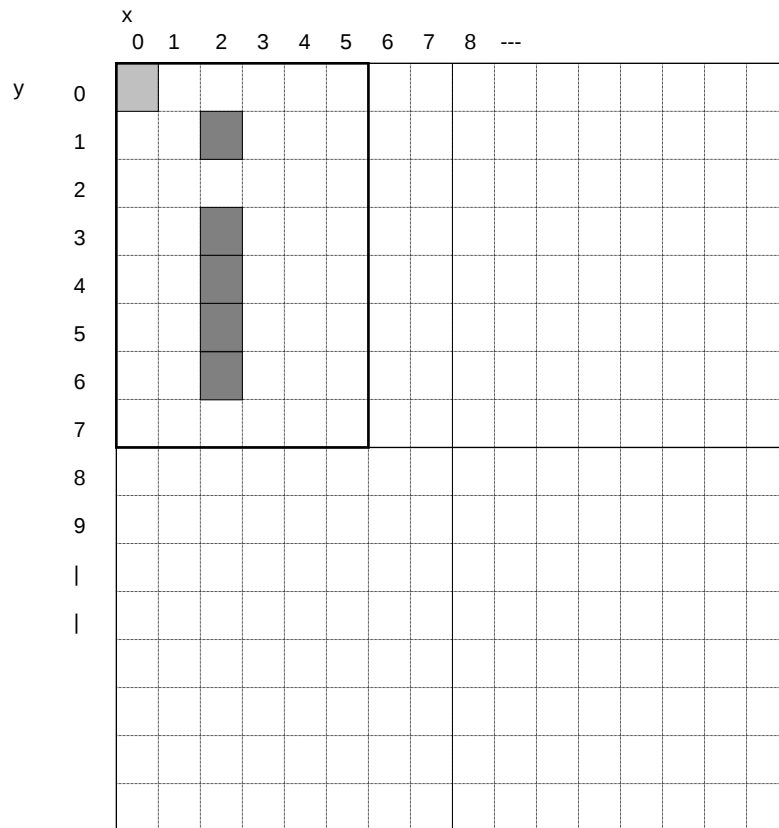
3.4 Cursor

Cursor indicates the write start position for displaying a character or image.

Characters and images are written to the right in the X direction and downwards in the Y direction from and including the Cursor position.

Cursor position can be moved by "Cursor set" command (refer to 3.7.4.3 Cursor Set).

Cursor position relates to display memory as shown below.



Light grey:	Cursor
Dark grey:	Character
Thick line frame:	Space for one character (6×8 pixel)

3.5 Window

Window function enables the display screen to be divided into “windows” each of which can be controlled and displayed independently.

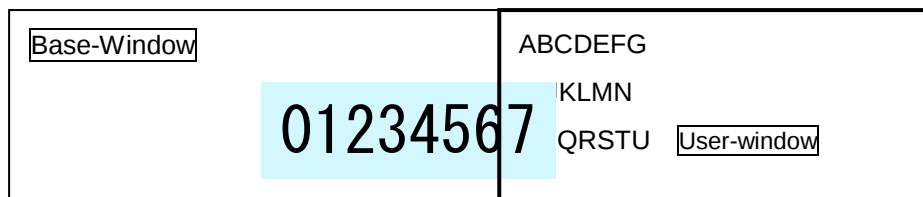
Display memory is shared by all windows; individual windows do not have their own display memory. There are 2 types of “window”: base-window and user-window.

Refer to 3.7.4.47 User Window Define / Cancel.

3.5.1 Base-window

Base-window covers the entire display screen. If no user-windows are defined, all display operation is processed on this window. If one or more user-windows are defined, display operation on any area not covered by a user-window is done by selecting base-window. When base-window is selected, even if user-window(s) are defined, all display operation is processed under base-window. Therefore, the current display contents of user-window(s) are overwritten.

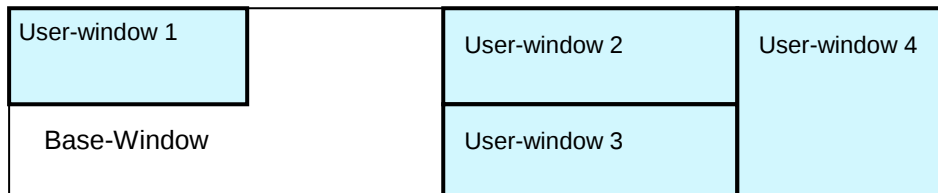
Operation on base-window depends on the setting of “Write Screen Mode” (refer to 3.6 Write Screen Mode).



3.5.2 User-window

User-window is defined by user-window definition command. Display operation is processed on the window selected by Current-window Select command.

A maximum of 4 user-windows can be defined.



3.6 Write Screen Mode

This setting is only applicable for Base-window.

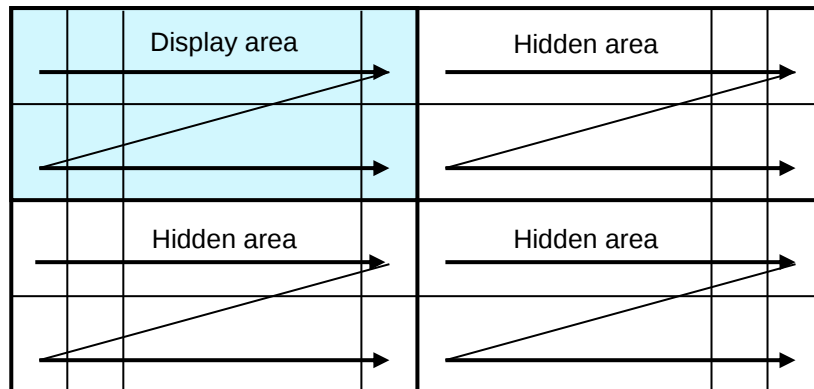
There are two Write screen modes, display screen mode and all screen mode. The mode is set by command (refer to 3.7.4.4 Write Screen Mode Select).

3.6.1 Display Screen Mode

When the cursor is located in the display area, all operation will be done within display area, and when cursor is located in a hidden area, it will be done within the hidden area.

Character write depends on the specified character display mode.

Bit images are written within the current area, and any data outside the area is ignored.

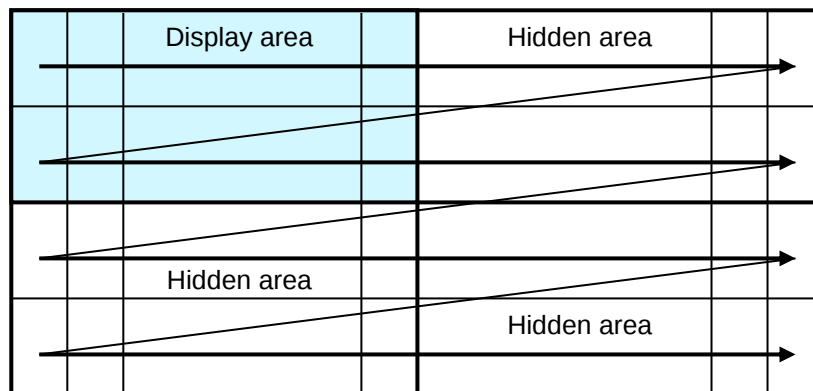


3.6.2 All Screen Mode

Regardless of the cursor position, operation will be done over the entire area.

Character write depends on the specified character display mode.

Bit images are written within the entire memory area, and any data outside the area is ignored.



3.7 Display Control Commands

This section describes the operation of each command.

Within these explanations, Character (x-pixel) and Line (y-pixel) refer to the number of pixels determined by the "Font size select" and "Font magnification" settings, etc.

For commands that produce response data from the display, this data is placed in the send buffer, then transmitted. **When DSR=MARK (BUSY), data transmission is halted, and during any time when there is no space in the transmit buffer, command processing is halted. Caution is needed when using these commands via any unidirectional interface (transmit-data is queued in the transmit buffer, but is not sent from the module).**

3.7.1 Code Set

3.7.1.1 Character Code

Command Name	Hex Code	Operation	Page
Character display	20h – FFh or 2-byte character code (or 1-4 byte UTF-8 byte sequence)	Display character at the current cursor position.	p22

3.7.1.2 Control Code

Command Name	Hex Code	Operation	Page
BS Back Space	08h	Cursor moves left by one character.	p26
HT Horizontal Tab	09h	Cursor moves right by one character.	p26
LF Line Feed	0Ah	Cursor moves down by one line.	p27
HOM Home Position	0Bh	Cursor moves to home position (top left).	p27
CR Carriage Return	0Dh	Cursor moves to left end of the current line.	p27
CLR Display Clear	0Ch	Display screen is cleared, cursor moves to home position.	p27
CAN Line Clear	18h	Current line is cleared and cursor moves to left end.	p27
RCLR Line end Clear	19h	Current line is cleared from cursor moves to right end.	p27

3.7.2 Detail of Code Set

3.7.2.1 Character Display

Code: 20h – FFh or 2-byte character code (or 1-4 byte UTF-8 byte sequence)

Function: Display character at cursor position.

Font size can be selected, 6×8, 8×16, 12×24 or 16×32 (refer to 3.7.4.12 Font Size Select), or any displayable size for Outline fonts (refer to 3.7.4.15 Outline Font Size).

To display 2-bytes characters, the following settings are required:

Font size select = 8×16 pixel, 16×32 pixel or Outline font (m=02h, m=04h or 00h *)

2-byte character = ON (m=01h)

2-byte character type = Japanese, Korean, Simplified or Traditional Chinese

Refer to 3.7.4.13 2-byte Character ON/OFF and 3.7.4.14 2-byte Character Type for Details.

The 2-byte character code depends on the type of built-in character fonts. This module has the following built-in 2-byte character fonts:

Font type	Code type	First byte	Second byte
Japanese	JIS X0208 (Shift-JIS)	81h ≤ c1 ≤ 9Fh, E0h ≤ c1 ≤ Ffh	40h ≤ c2 ≤ 7Eh, 80h ≤ c2 ≤ FCh
Korean	KSC5601-87	A1h ≤ c1 ≤ FEh	A1h ≤ c2 ≤ FEh
Simplified Chinese	GB2312-80	A1h ≤ c1 ≤ FEh	A1h ≤ c2 ≤ FEh
Traditional Chinese	Big-5	A1h ≤ c1 ≤ FEh	40h ≤ c2 ≤ 7Eh A1h ≤ c2 ≤ FEh

* Outline font type must be selected first (Refer to 3.7.4.17 Outline Font Type).

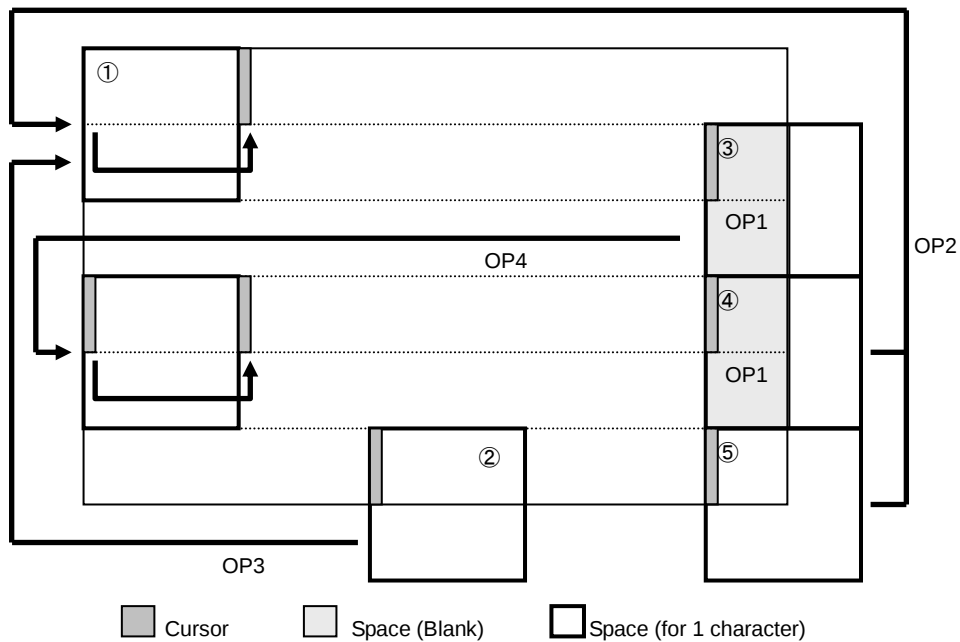
Alternatively, character codes can be entered in Unicode (UTF-8) format by setting Character Table type to FEh.

This command operates on the currently-selected window (refer to 3.7.4.46 Current-window Select). Regardless of the cursor position, if the character size (x and/or y) exceeds the window size, the command is ignored. Details of operation are as follows:

MD1 (Over-write mode)

Cursor position		Figure Number	Display Operation
X direction	Y direction		
Space for character on right side.	Space for character at current cursor position.	①	Display character on cursor. Horizontal Tab (HT).
	No space for character at current cursor position.	②	Cursor moves to the left end of top line (OP3). Display character at cursor. Horizontal Tab (HT).
No space for character on right side.	Space for character in next lower line.	③	Display space at cursor (OP1). Cursor moves to left end of next lower line (OP4). Display character at cursor. Horizontal Tab (HT).
	No space for character in next lower line.	④	Display space at cursor (OP1). Cursor moves to left end of top line (OP2). Display character at cursor. Horizontal Tab (HT).
	No space for character at current cursor position.	⑤	Cursor moves to left end of top line (OP2). Display character at cursor. Horizontal Tab (HT).

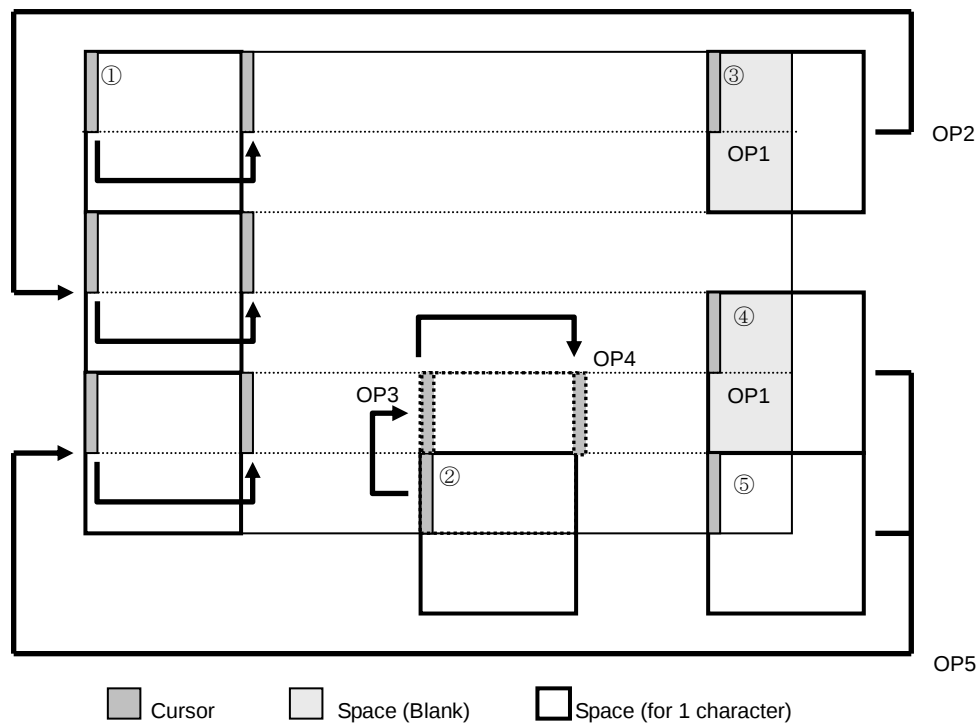
Note: HT operation depends on cursor position (refer to 3.7.2.3 Horizontal Tab).



MD2 (Vertical scroll mode)

Cursor position		Figure Number	Display Operation
X direction	Y direction		
Space for character on right side.	Space for character at current cursor position.	①	Display character at cursor. Horizontal Tab (HT) (OP4).
	No space for character at current cursor position.	②	Display contents are scrolled up the required number of pixels, and the bottom line is cleared. Cursor moves to the displayable upper position (OP3). Display character at cursor. Horizontal Tab (HT).
No space for character on right side.	Space for character in next lower line.	③	Display space at cursor (OP1). Cursor moves to left end of next lower line (OP2). Display character at cursor. Horizontal Tab (HT).
	No space for character in next lower line.	④	Display space at cursor (OP1). Display contents are scrolled up the required number of pixels, and the bottom line is cleared. Cursor moves to left end of bottom line (OP5). Display character at cursor. Horizontal Tab (HT).
	No space for character at current cursor position.	⑤	Display contents are scrolled up the required number of pixels, and the bottom line is cleared. Cursor moves to left end of bottom line (OP5). Display character at cursor. Horizontal Tab (HT).

Note: HT operation depends on cursor position (refer to 3.7.2.3 Horizontal Tab).



MD3 (Horizontal scroll mode)

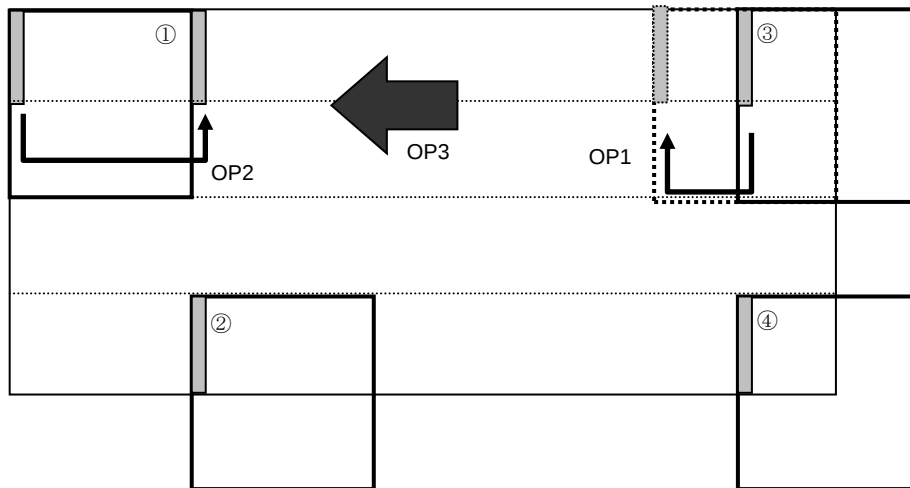
Cursor position			Figure Number	Display Operation
X direction		Y direction		
Space for character on right side.	Not right end.	-	①	Display character at cursor. Horizontal Tab (HT) (OP2).
	Right end (Refer Figure 2).		-	Display character at cursor. Shift to Scroll ON*.
	-	No space for character at current cursor position.	②	No action. Cursor does not move.
No space for character on right side.	-	-	③	Contents of current line scroll to left until sufficient space for character is available at the right end (OP3). Cursor moves to left edge of newly-created space (OP1). Display character at cursor. Shift to Scroll ON*.
	-	No space for character at current cursor position.	④	No action. Cursor does not move.

Note: HT operation depends on cursor position (refer to 3.7.2.3 Horizontal Tab).

* Note: Operation during "Scroll ON":

Contents of current line scroll left until sufficient space for character is available at the right end, then character is displayed at cursor.

"Scroll ON" condition is cancelled by any command that moves the cursor except Character display or Horizontal Tab.



Cursor
 Space (Blank)
 Space (for 1 character)

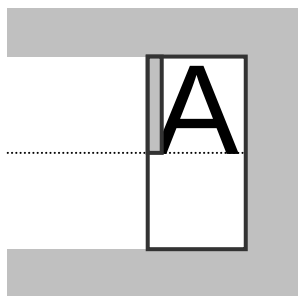


Figure 2

3.7.2.2 BS (Back Space)

Code: 08h

Function: Cursor moves to the left by one character.

This command has effect for the currently-selected window.

MD1 (Over-write mode) and MD2 (Vertical scroll mode)

Cursor position		Display operation
X direction	Y direction	
Space for character on left side.	-	Cursor moves to left by one character.
No space for character on left end.	Space for one line above.	Cursor moves to right end of next upper line.
	No space for one line above.	Cursor does not move.

MD3 (Horizontal scroll mode)

Cursor position		Display operation
X direction	Y direction	
Space for character on left side.	-	Cursor moves to left by one character.
No space for character on left end.	-	Cursor does not move.

3.7.2.3 HT (Horizontal Tab)

Code: 09h

Function: Cursor moves to the right by one character.

This command has effect for the currently-selected window.

MD1 (Over-write mode)

Cursor position		Display operation
X direction	Y direction	
Space for character on right side.	-	Cursor moves to right by one character.
No space for character on right side.	Space for character in next lower line.	Cursor moves to left end of next lower line.
	No space for character in next lower line.	Cursor moves to left end of top line.

MD2 (Vertical scroll mode)

Cursor position		Display operation
X direction	Y direction	
Space for character on right side.	-	Cursor moves to right by one character.
No space for character on right side.	Space for character in next lower line.	Cursor moves to left end of next lower line.
	No space for character in next lower line.	Display contents are scrolled up the required number of pixels, and the bottom line is cleared. Cursor moves to left end of bottom line.

MD3 (Horizontal scroll mode)

Cursor position			Display operation
X direction		Y direction	
Space for character on right side.	Not right end	-	Cursor moves right by one character.
	Right end (refer to Figure 2, page 13).		Shift to Scroll ON*
No space for character on right side.	-	-	Contents on current line scroll left until sufficient space for character is available at the right end. Cursor moves to the left edge of newly-created space. Shift to Scroll ON*.

* Note: Operation during "Scroll ON":

Contents of current line scroll left until sufficient space for character is available at the right end (cursor does not move).

"Scroll ON" condition is cancelled by any command that moves the cursor except Character display or horizontal tab.

3.7.2.4 LF (Line Feed)**Code: 0Ah**

Function: Cursor moves to next lower line.

This command has effect for the currently-selected window.

MD1 (Over-write mode)

Cursor position		Display operation
X direction	Y direction	
-	Space for character in next lower line.	Cursor moves to the same position on next lower line.
	No space for character in next lower line.	Cursor moves to the same position on top line.

MD2 (Vertical scroll mode)

Cursor position		Display operation
X direction	Y direction	
-	Space for character in next lower line.	Cursor moves to the same position on next lower line.
	No space for character in next lower line.	Display contents are scrolled up to the required number of pixels, and the bottom line is cleared. Cursor does not move.

MD3, MD5 (Horizontal scroll mode)

Cursor position		Display operation
X direction	Y direction	
-	-	Cursor does not move.

3.7.2.5 HOM (Home Position)**Code: 0Bh**

Function: Cursor moves to home position (top left).

This command has effect for the currently-selected window.

3.7.2.6 CR (Carriage Return)**Code: 0Dh**

Function: Cursor moves to left end of current line.

This command has effect for the currently-selected window.

3.7.2.7 CLR (Display Clear)**Code: 0Ch**

Function: Display screen is cleared (filled in the background color) and cursor moves to home position.

This command has effect for the currently-selected window.

3.7.2.8 CAN (Line Clear)**Code: 18h**

Function: Current line is cleared (filled in the background color) and cursor moves to left end of current line.

This command has effect for the currently-selected window.

3.7.2.9 RCLR (Line End Clear)**Code: 19h**

Function: Current line is cleared (filled in the background color) from cursor position to end of line (right end).

Cursor does not move.

This command has effect for the currently-selected window.

3.7.3 Command Set

3.7.3.1 General Setting Commands

Command Name	Hex Code	Operation	Page
Brightness Level Setting (for Backlight)	1Fh,58h,n Default: n=FFh or Memory SW setting.	Set backlight brightness level for entire display screen. Brightness level = $(n / 255) * 100$ [%]	p37
Initialize Display	1Bh,40h	Clear entire display screen and initialize all settings.	p37
Cursor Set	1Fh,24h, xL,xH,yL,yH	Cursor moves to specified x,y position on display memory. xL: Cursor position x, lower byte. xH: Cursor position x, upper byte. yL: Cursor position y, lower byte. yH: Cursor position y, upper byte.	p37
Write Screen Mode	1Fh,28h,77h,10h,a Default a=00h or Memory SW setting.	Sets the write screen mode for base window. a=00h: Display screen mode a=01h: All screen mode	p37

3.7.3.2 Character Display Setting Commands

Command Name	Hex Code	Operation	Page
International Font Set	1Bh,52h,n Default n=00h or Memory SW setting.	Some character codes within the range 20h – 7Fh are selected from the types listed below. n=00h: America n=01h: France n=02h: Germany n=03h: England n=04h: Denmark1 n=05h: Sweden n=06h: Italy n=07h: Spain1 n=08h: Japan n=09h: Norway n=0Ah: Denmark2 n=0Bh: Spain2 n=0Ch: Latin America n=0Dh: Korean	p38
Character Table Type	1Bh,74h,n Default n=00h or Memory SW setting.	Character codes in the range 80h – FFh are selected from the types listed below. n=00h: PC437 (USA:Standard Europe) n=01h: Katakana, n=02h: PC850 (Multilingual) n=03h: PC860 (Portuguese) n=04h: PC863 (Canadian-French) n=05h: PC865 (Nordic), n=10h: WPC1252 n=11h: PC866 (Cyrillic #2) n=12h: PC852 (Latin 2), n=13h: PC858 n=FEh: UTF-8 input n=FFh: User table	p38
Over-write Mode	1Fh,01h	Set Over-write mode.	p39
Vertical Scroll Mode	1Fh,02h	Set Vertical scroll mode.	p39
Horizontal Scroll Mode	1Fh,03h	Set Horizontal scroll mode.	p39
Horizontal Scroll Mode, Scroll ON	1Fh,05h	Set Horizontal scroll mode, scroll ON.	p39
Horizontal Scroll Speed	1Fh,73h,n Default n=00h or Memory SW setting.	Set Horizontal scroll speed.	p39
Font Size Select	1Fh,28h,67h,01h,m Default m=01h or Memory SW setting.	Select font size of a character. m=00h: Outline font m=01h: 6×8 font m=02h: 8×16 font m=03h: 12×24 font m=04h: 16×32 font	p39
2-byte Character	1Fh,28h,67h,02h,m Default m=00h or Memory SW setting.	Sets 2-byte character ON/OFF. m=01h: 2-byte character mode ON m=00h: 2-byte character mode OFF	p40
2-byte Character Type	1Fh,28h,67h,03h,m Default: m=00h or Memory SW setting.	Sets 2-byte character type. m=00h: Japanese m=01h: Korean m=02h: Simplified Chinese m=03h: Traditional Chinese	p40

Command Name	Hex Code	Operation	Page
Outline Font Size	1Fh,28h,67h,06h,ylsL,ylsH, ysL,ysH,xsL,xsH,boL,boH Default: yls = 64, ys, xs, bo = 0 (auto-calculated)	Set character size and line spacing and alignment for outline (scalable) font display. ylsL: Y-size (height) of character line (line spacing), in pixels (x1h) ylsH: Y-size (height) of character line (line spacing), in pixels (x100h) ysL: Nominal character Y-size (height), in pixels (x1h) ysH: Nominal character Y-size (height), in pixels (x100h) xsL: Nominal character X-size (width), in pixels (x1h) xsH: Nominal character X-size (width), in pixels (x100h) boL: Baseline Y-offset from cursor position, in pixels (x1h) boH: Baseline Y-offset from cursor position, in pixels (x100h)	p41
User-supplied Font File Address and Size	1Fh,28h,67h,07h,ad0,ad1, ad2,ad3,sz0,sz1,sz2,sz3 Default: ad, sz = FFFFFFFFh (no font file specified)	Set address and size in General-purpose FROM2 for the user-supplied font file (binary image). ad3:ad2:ad1:ad0: Address of start of font file (binary image) in General-purpose FROM2. sz3:sz2:sz1:sz0: Size in bytes of font file (binary image) in General-purpose FROM2.	p42
Outline Font Type Select	1Fh,28h,67h,08h,n Default: n=00h	Sets the outline font type. n = 00h: Japanese n = 01h: Korean n = 02h: Simplified Chinese n = 03h: Traditional Chinese n = 80h: None (no outline font selected) n = FFh: User-supplied font file.	p42
Font Width	1Fh,28h,67h,04h,m Default: m=00h	Character width select m=00h: Fixed width m=02h: Proportional 1 m=03h: Proportional 2 m=04h: Proportional 3	p43
FROM Extended Font	1Fh,28h,67h,05h,n Default: n=00h	FROM extended font select. n=00h: Normal font n=01h – FFh: FROM extended font	p44
Font Magnification	1Fh,28h,67h,40h, x,y Default: x=01h, y=01h or Memory SW setting.	Magnify the character by x times on the right, y times downward. x: X magnification factor y: Y magnification factor	p44
Character Style	1Fh,28h,67h,41h,b Default: b=00h or Memory SW setting.	Character style setting. b=00h: Normal b=01h: Bold b=02h: Shadow b=03h: Bordering	p44
Character Color (Fore Color)	1Fh,28h,67h,50h,pR,pG,pB Default: pR=FFh, pG=FFh, pB=FFh	Specify character color. pR = 00h–FFh: Red color level. pG = 00h–FFh: Green color level. pB = 00h–FFh: Blue color level.	p44
Background Color	1Fh,28h,67h,51h,pR,pG,pB Default: pR=00h, pG=00h, pB=00h	Specify character background color. pR = 00h–FFh: Red color level. pG = 00h–FFh: Green color level. pB = 00h–FFh: Blue color level.	p44
Shadow and Bordering Color	1Fh,28h,67h,52h,pR,pG,pB Default: pR=00h, pG=00h, pB=00h	Specify character shadow and bordering color. pR = 00h–FFh: Red color level. pG = 00h–FFh: Green color level. pB = 00h–FFh: Blue color level.	p45
Background Color Enable / Disable	1Fh,28h,67h,58h,m Default m=00h	Background color enable / disable. m=01h: Enabled m=00h: Disabled	p45

3.7.3.3 Display Action Setting Commands

Command Name	Hex Code	Operation	Page
Wait	1Fh,28h,61h,01h,t	Processing is stopped for the specified time. t: Wait time (x approximately 0.47s)	p45
Short Wait	1Fh,28h,61h,02h,t	Processing is stopped for the specified time. t: Wait time (x IntTime)	p45
Blink	1Fh,28h,61h,11h, p,t1,t2,c	Blink display action on display screen. p: Blink pattern t1: Normal display time t2: Blank display time c: Number of cycles	p46
Scroll Display Action XY	1Fh,28h,61h,A0h, sXL,sXH,sYL, sYH,cL,cH,s	Shifts the display screen, enabling horizontal and/or vertical display screen scroll action. sXL: Number of display screen X shift (×1h) sXH: Number of display screen X shift (×100h) sYL: Number of display screen Y shift (×1h) sYH: Number of display screen Y shift (×100h) cL: Number of cycles, lower byte cH: Number of cycles, upper byte s: Scroll speed	p46
Curtain Display Action	1Fh,28h,61h,A2h, v,s,pR,pG,pB	Curtain display action on display screen. v: Direction of curtain action s: Curtain action speed pR: Red color level. pG: Green color level. pB: Blue color level.	p47
Spring Display Action XY	1Fh,28h,61h,A3h, v,s,pXL,pXH,pYL,pYH	Spring display action on display screen. v: Direction of spring action s: Spring action speed pXL: Display memory X position (×1h) pXH: Display memory X position (×100h) pYL: Display memory Y position (×1h) pYH: Display memory Y position (×100h)	p48
Random Display Action XY	1Fh,28h,61h,A4h,v,s,pXL,pX H,pYL,pYH	Random display action on display screen. v: Random display action type s: Random display action speed pXL: Display memory X position (×1h) pXH: Display memory X position (×100h) pYL: Display memory Y position (×1h) pYH: Display memory Y position (×100h)	p49
Fade In Display Action XY	1Fh,28h,61h,A5h,s,pXL,pXH, pYL,pYH	Fade in display action on display screen. s: Fade in display action speed pXL: Display memory X position (×1h) pXH: Display memory X position (×100h) pYL: Display memory Y position (×1h) pYH: Display memory Y position (×100h)	p50
Fade Out Display Action XY	1Fh,28h,61h,A6h,s	Fade out display action on display screen. s: Fade out display action speed	p50
Display Power ON/OFF	1Fh,28h,61h,40h,p Default: p=01h	Set display power ON / OFF / Auto-ON. p: Set display power ON / OFF / Auto-ON p = 00h: Display power OFF p = 01h: Display power ON p = 80h: Display power OFF, automatic ON when touch detected.	p50
Power Saving Mode	1Fh,28h,61h,48h,m,w	Transition to Power Saving Mode and set the wakeup method m: mode m = 01h: Sleep Mode w: Wakeup method w = b7, b6, b5, b4, b3, b2, b1, b0	P76

3.7.3.4 Bit Image Display Setting Commands

Command Name	Hex Code	Operation	Page
Pixel Drawing	1Fh,28h,64h,10h,pen, xL,xH,yL,yH	Display the pixel pattern on a drawing position or delete the pixel pattern already displayed. pen: Pixel display ON or OFF xL: Pixel pattern drawing position x, lower byte xH: Pixel pattern drawing position x, upper byte yL: Pixel pattern drawing position y, lower byte yH: Pixel pattern drawing position y, upper byte	p50
Line/Box Pattern Drawing	1Fh,28h,64h,11h,mode, pen, x1L,x1H,y1L,y1H, x2L,x2H,y2L,y2H	Display the Line, Box, Box FILL on the drawing area specified by x1,y1, x2,y2 or delete the pixel pattern already displayed. mode: Drawing mode select pen: Pixel ON or OFF x1L: Line/Box pattern drawing start position x1, lower byte x1H: Line/Box pattern drawing start position x1, upper byte y1L: Line/Box pattern drawing start position y1, lower byte y1H: Line/Box pattern drawing start position y1, upper byte x2L: Line/Box pattern drawing end position x2, lower byte x2H: Line/Box pattern drawing end position x2, upper byte y2L: Line/Box pattern drawing end position y2, lower byte y2H: Line/Box pattern drawing end position y2, upper byte	p51
Real-time Bit Image Display	1Fh,28h,66h,11h,xL,xH, yL,yH,fmt,d(1)...d(n)	Display the supplied bit image data on the cursor position in real-time. xL: Bit image X size, lower byte (by 1 pixel) xH: Bit image X size, upper byte (by 1 pixel) yL: Bit image Y size, lower byte (by 1 pixel) yH: Bit image Y size, upper byte (by 1 pixel) fmt: Bit image format fmt=81h: Monochrome (1-bit) format fmt=86h: Color 6-bit format fmt=8Ch: Color 12-bit format fmt=90h: Color 16-bit format fmt=98h: Color 24-bit format fmt=F0h: BMP file format d(1)...d(n): Image data	p52
Packaged Real-time Bit Image Display	1Fh,28h,66h,21h, xL,xH,yL,yH,fmt, d(1)...d(n)	Display the supplied packaged bit image data on the cursor position in real-time. xL: Bit image X size, lower byte (by 1 pixel) xH: Bit image X size, upper byte (by 1 pixel) yL: Bit image Y size, lower byte (by 1 pixel) yH: Bit image Y size, upper byte (by 1 pixel) fmt: Bit image format fmt=86h: Color 6-bit format (packaged) fmt: 8Ch: Color 12-bit format (packaged) fmt: 98h: Color 24-bit format (packaged) d(1)...d(n): Packaged image data	p53
RAM Bit Image Definition	1Fh,28h,66h,01h, aL,aH,aE,sL,sH,sE, d(1)...d(s)	Define user bit image to RAM. aL: Bit image data definition address, lower byte aH: Bit image data definition address, upper byte aE: Bit image data definition address, extension byte sL: Bit image data length, lower byte sH: Bit image data length, upper byte sE: Bit image data length, extension byte d(1)...d(s): Image data	p54
FROM Bit Image Definition (Only valid in User setup mode)	1Fh,28h,65h,10h, aL,aH,aE,sL,sH,sE, d(1)...d(s)	Define user bit image to FROM1. aL: Bit image data definition address, lower byte aH: Bit image data definition address, upper byte aE: Bit image data definition address, extension byte sL: Bit image data length, lower byte sH: Bit image data length, upper byte sE: Bit image data length, extension byte d(1)...d(s): Image data	p55

Command Name	Hex Code	Operation	Page
Downloaded Bit Image Display	1Fh,28h,66h,10h,m, aL,aH,aE,xSL,xSH, xL,xH,yL,yH, fmt	Display the RAM, FROM1, FROM2, or display memory bit image defined on cursor position. m: Select bit image data memory aL: Bit image data definition address, lower byte aH: Bit image data definition address, upper byte aE: Bit image data definition address, extension byte xSL: Bit image defined X size, lower byte (by 1 pixel) xSH: Bit image defined X size, upper byte (by 1 pixel) xL: Bit image display X size, lower byte (by 1 pixel) xH: Bit image display X size, upper byte (by 1 pixel) yL: Bit image display Y size, lower byte (by 1 pixel) yH: Bit image display Y size, upper byte (by 1 pixel) fmt: Bit image format fmt=81h: Monochrome (1-bit) format fmt=86h: Color 6-bit format fmt=8Ch: Color 12-bit format fmt=90h: Color 16-bit format fmt=91h: Color 16-bit high-speed format fmt=98h: Color 24-bit format fmt=F0h: BMP file format	p56
Packaged Downloaded Bit Image Display	1Fh,28h,66h,20h,m,aL,aH,a E,pL,pH, fmt	Display the packaged RAM, FROM1, or FROM2 bit image defined on cursor position. m: Select bit image data memory aL: Bit image data library address, lower byte aH: Bit image data library address, upper byte aE: Bit image data library address, extension byte pL: Bit image display number, lower byte pH: Bit image display number, upper byte fmt: Bit image format fmt=00h = Refer to Bit Image Index memory fmt=81h = Monochrome (1-bit) format fmt=86h = Color 6-bit format fmt=8Ch = Color 12-bit format fmt=98h = Color 24-bit format fmt=F0h = BMP file format	p59

3.7.3.5 General Display Setting Commands

Command Name	Hex Code	Operation	Page
Reverse Display	1Fh,72h,n Default: n=00h or Memory SW setting.	Reverse display setting ON/OFF. n=00h: Reverse display OFF n=01h: Reverse display ON	p61
Write Mixture Display Mode	1Fh,77h,n Default: n=10h or Memory SW setting.	Sets the write mixture mode. New character or graphic image data is mixed with the current display image when written to the display memory. n=10h: Normal display write (Background pixels are written to display memory) n=11h: Thru write (Background pixels are skipped when writing to display memory)	p61

3.7.3.6 Window Display Setting Commands

Command Name	Hex Code	Operation	Page
Current-window Select	1Fh,28h,77h,01h,a	Selects current window a=00h: Base-window a=01h: User-window 1 a=02h: User-window 2 a=03h: User-window 3 a=04h: User-window 4	p61
User-window Define / Cancel	1Fh,28h,77h,02h,a,b, xPL,xPH,yPL,yPH, xSL,xSH,ySL,ySH	Define or cancel user-window a: Definable window No. b: Define or cancel xPL: Left position of window, lower byte (by 1 pixel) xPH: Left position of window, upper byte (by 1 pixel) yPL: Top position of window, lower byte (by 1 pixel) yPH: Top position of window, upper byte (by 1 pixel) xSL: X size of window, lower byte (by 1 pixel) xSH: X size of window, upper byte (by 1 pixel) ySL: Y size of window, lower byte (by 1 pixel) ySH: Y size of window, upper byte (by 1 pixel)	p62

3.7.3.7 Download Character Setting Commands

Command Name	Hex Code	Operation	Page
Download Character ON/OFF	1Bh,25h,n Default: n=00h	Specify enable or disable for download character n=01h: Enable n=00h: Disable	p63
Download Character Definition	1Bh,26h,a,c1,c2, x1,d1...dx1, ... , xk,d1...dxk	Define 6×8, 8×16, 12×24 or 16×32 pixel download characters into RAM. a: Select character type c1: Start character code c2: End character code x: Number of pixels for X-direction d1...dxk: Defined data	p63
Downloaded Character Delete	1Bh,3Fh,a,c	Delete defined 6×8, 8×16, 12×24 or 16×32 pixel download character. a: Select character type c: Delete Character code	p63
16×16 Download Character Definition	1Fh,28h,67h,10h, c1,c2,d1...d32	Defines the 16×16 download character in specified code. c1: Character code upper byte c2: Character code lower byte d: Definition data	p64
16×16 Downloaded Character Delete	1Fh,28h,67h,11h, c1,c2	Delete the 16×16 download character defined in the specified code. c1: Character code, upper byte c2: Character code, lower byte	p64
32×32 Download Character Definition	1Fh,28h,67h,14h, c1,c2,d1...d128	Define 32×32 pixel download characters into RAM. c1: Character code, upper byte c2: Character code, lower byte d: Definition data	p64
32×32 Downloaded Character Delete	1Fh,28h,67h,15h, c1,c2	Delete defined 32×32 pixel download character. c1: Delete character code, upper byte c2: Delete character code, lower byte	p65
Download Character Save (Only valid in User setup mode)	1Fh,28h,65h,11h,a	Save download characters defined on RAM to FROM. a: Font size a=01h: 6×8 pixel a=02h: 8×16 pixel a=03h: 16×16 pixel a=04h: 16×32 pixel a=05h: 32×32 pixel a=06h: 12×24 pixel	p65
Download Character Restore	1Fh,28h,65h,21h,a	Transfer the download characters saved in FROM to RAM. a: Font size a=01h: 6×8 pixel a=02h: 8×16 pixel a=03h: 16×16 pixel a=04h: 16×32 pixel a=05h: 32×32 pixel a=06h: 12×24 pixel	p65
FROM User Font Definition (Only valid in User setup mode)	1Fh,28h,65h,13h,m, P(80h-1),P(80h-2), ... ,P(FFh-n)	Define the user font for each size of 1-byte code to the user table. m: User table m=01h: 6×8 pixel m=02h: 8×16 pixel m=03h: 12×24 pixel m=04h: 16×32 pixel p: Definition data	p65
FROM Extension Font Definition	1Fh,28h,65h,15h,a,b, p(1), ... p(65536)	Define or delete FROM extension font a: Bank b: Define/Delete p: Definition data (if Define)	p66

3.7.3.8 User Setup Mode Setting Commands

Command Name	Hex Code	Operation	Page
User Setup Mode Start	1Fh,28h,65h,01h,49h,4Eh	User setup mode start.	p66
User Setup Mode End (Only valid in User setup mode)	1Fh,28h,65h,02h,4Fh,55h,54h	User setup mode end.	p66

3.7.3.9 General-purpose I/O Port Control Commands

Command Name	Hex Code	Operation	Page
I/O Port Input / Output Setting	1Fh,28h,70h,01h,n,a	Set input or output for general-purpose I/O ports. n: I/O port number n=00h: Port 0 n=01h: Port 1 a: Set Input / Output (bit-wise) bit = 0: Input, bit = 1: Output	p67
I/O Port Output	1Fh,28h,70h,10h,n,a	Output data to general-purpose I/O port. n: I/O port number n=00h: Port 0 n=01h: Port 1 a: Output data value	p67
I/O Port Input	1Fh,28h,70h,20h,n	The state of a general-purpose I/O port is transmitted. n: I/O port number n=00h: Port 0 n=01h: Port 1	p67

3.7.3.10 Macro Setting Commands

Command Name	Hex Code	Operation	Page
RAM Macro Define / Delete	1Fh,3Ah,pL,pH,d1...dk	Define or delete RAM macro or RAM program macro. pL: RAM macro data length, lower byte pH: RAM macro data length, upper byte d1...dk: RAM macro data	p68
FROM Macro Define / Delete (Only valid in User setup mode)	1Fh,28h,65h,12h,a,pL,pH,t1,t2,d(1)...d(p)	Define or delete FROM macro or FROM program macro. a: FROM macro definition number pL: FROM macro data length, lower byte pH: FROM macro data length, upper byte t1: Display time interval t2: Idle time of macro repetition d(1)...d(p): FROM macro data	p68
Macro Execution	1Fh,5Eh,a,t1,t2	Continuously execute macro. a: Macro processing definition number < Firmware version: F220, or later > a=00h: RAM macro a=01h~08h: FROM macro 1~8 a=80h: RAM program macro a=81h~88h: FROM program macro 1~8 < Firmware version: less than F220 > a=00h: RAM macro a=01h~04h: FROM macro 1~4 a=80h: RAM program macro a=81h~84h: FROM program macro 1~4 t1: Display time interval t2: Idle time for macro repetition	p70
Macro End Condition	1Fh,28h,69h,20h,a,b,c Default: a=00h, b=00h, c=00h or Memory SW setting.	Macro end condition set. (Not applicable for program macro) a: Macro end code enable/ disable b: Macro end code c: Macro end clear screen setting	p70

3.7.3.11 Other Setting Commands

Command Name	Hex Code	Operation	Page
Memory SW Setting (Only valid in User setup mode)	1Fh,28h,65h,03h,a,b	Set Memory SW. a: Memory SW number b: Setting data	p70
Memory SW Data Send	1Fh,28h,65h,04h,a	Send the contents of Memory SW data. a: Memory SW number	p71
General-purpose Memory Store / Bit Image FROM2 Store	1Fh,28h,65h,18h, sL,sH,sE,m1, aL,aH,aE, d[1]...d[s]	Store the supplied data into general-purpose memory. sL: Data size, lower byte sH: Data size, upper byte sE: Data size, extension byte m1: Memory select aL: Memory address, lower byte aH: Memory address, upper byte aE: Memory address, extension byte d: Data to store	p72
General-purpose Memory Transfer	1Fh,28h,65h,19h, sL,sH,sE, m1,a1L,a1H,a1E, m2,a2L,a2H,a2E	Transfer data between general-purpose memory areas. sL: Transfer size, lower byte sH: Transfer size, upper byte sE: Transfer size, extension byte m1: Destination memory select a1L: Destination address, lower byte a1H: Destination address, upper byte a1E: Destination address, extension byte m2: Source memory select a2L: Source address, lower byte a2H: Source address, upper byte a2E: Source address, extension byte	p73
General-purpose Memory Send	1Fh,28h,65h,28h, sL,sH,sE,m1, aL,aH,aE	Send data stored in general-purpose memory. sL: Data size, lower byte sH: Data size, upper byte sE: Data size, extension byte m1: Memory select aL: Memory address, lower byte aH: Memory address, upper byte aE: Memory address, extension byte	p73
Display Status Send	1Fh,28h,65h,40h, a,b,c	Send display status information a: Information name a=01h: Boot version information a=02h: Firmware version information a=10h: Kanji font information a=20h: Memory check sum information a=30h: Product information a=40h: Display x pixel information a=41h: Display y pixel information a=70h: Touch setting package name a=71h: Touch setting package ID a=72h: Touch sensitivity (Gain) a=73h: Touch sensitivity (Threshold) b: Start address c: Data length	p75
Memory Re-write Mode Shift	1Ch,7Ch,4Dh, D0h,4Dh,4Fh,44h, 45h,49h,4Eh	Shift to "Memory re-writing mode" from "Normal mode".	p75

3.7.3.12 Touch Panel Control Commands

Command Name	Hex Code	Operation	Page
Touch Mode Selection: Single-Touch Mode/ Multi-Touch Mode	1Fh,50h,01h,n	Touch mode select (Single / Multi-Touch mode). n: Select Touch Mode and maximum simultaneous touch detection (for Multi-Touch Mode)	p88
Touch Panel Control Mode Coordinates Mode	1Fh,50h,10h,ch,00h	Set the specified channel to Coordinates mode. ch: Channel	p88
Touch Panel Control Mode Switch Matrix Mode	1Fh,50h,10h,ch,01h, nx,cx,ny,cy	Sets the specified channel to Switch Matrix mode. ch: Channel nx: Number of switch cx: Clearance ny: Number of switch cy: Clearance	p89
Touch Panel Control Mode Custom Switch Mode	1Fh,50h,10h,ch,02h, sn, px1L, px1H, py1L, py1H, sx1L, sx1H, sy1L, sy1H, [...px(sn)L,px(sn)H, py(sn)L,py(sn)H, sx(sn)L,sx(sn)H , sy(sn)L,sy(sn)H]	Sets the specified channel to Custom Switch mode. ch: Channel md: Mode sn: Number of Switch buttons pxL: Switch button left horizontal position, lower byte pxH: Switch button left horizontal position, upper byte pyL: Switch button top vertical position, lower byte pyH: Switch button top vertical position, lower byte sxL: Switch button x-size, lower byte sxH: Switch button x-size, upper byte syL: Switch button y-size, lower byte syH: Switch button y-size, upper byte	p90
Touch Panel Data Transmit ON/OFF	1Fh,50h,20h,m	Sets whether or not touch operation data is transmitted. m: Transmit ON/OFF	p91
Touch Panel Channel Select	1Fh,50h,21h,ch	Selects the currently-active touch panel control channel. ch: Channel	p92
Touch Parameter Setting (*1)	1Fh 4Bh 70h a [b [c]]	Sets the touch parameters. a: Parameter selection / operation designation a = 00h : Threshold setting a = 04h : Gain setting a = 06h : Touch standard reference setting a = 07h : Touch standard reference use a = 08h : Touch standard reference status read b, c : Set value	p92
Touch Setting Package Data Store	1Fh 28h 65h 1Ch a d[1] ... d[1024]	Touch setting package data is stored. a: Touch setting package data storing destination d: Package data 01h ≤ a ≤ 04h 00h ≤ d ≤ FFh	p94
Touch Setting Package Selection (*1)	1Fh 4Bh 70h 10h a	Select touch setting package to use. a: Touch setting package a = 00h : Factory setting 01h ≤ a ≤ 04h : Touch setting package 1 to 4.	p94
Touch Level Read	1Fh 4Bh 70h a	Send touch level information of FLETAS touch panel. a: Touch level read mode	p94
Touch Panel Control Data Transmit Mode (*1)	1Fh 50h 22h a	Select transmit mode of the touch / release report for the FLETAS touch panel. a: Data transmit mode	p94
Touch Scan Period Setting at Power Saving Mode (*1)	1Fh,28h,61h,49h,p	Set the touch scan period for Power Saving Mode p: Touch scan period 05h (5ms) ≤ p ≤ FEh (254ms)	P94

*1: For the default memory SW setting value, refer to the hardware specifications of each model.

3.7.4 Command Set Details

3.7.4.1 US X n (Brightness Level Setting (for Backlight))

Code: 1Fh 58h n

n: Brightness level setting

Definable area: 00h ≤ n ≤ FFh

Default: n = FFh or Memory SW setting.

Function: Set display brightness level.

$$\text{Brightness level} = (n / 255) \times 100 [\%]$$

3.7.4.2 ESC @ (Initialize Display)

Code: 1Bh 40h

Function: Clear entire display screen and initialize all settings.

Settings return to default values.

Jumper settings are not reloaded.

Contents of receive buffer remain in memory.

General-purpose I/O port settings are unchanged.

Settings that are not reset by this command are only reset by cycling the power or executing the "Exit user setting mode" command.

3.7.4.3 US \$ x y (Cursor Set)

Code: 1Fh 24h xL xH yL yH

xL: Cursor position x, lower byte (1 pixel / unit)

xH: Cursor position x, upper byte (1 pixel / unit)

yL: Cursor position y, lower byte (1 pixel / unit)

yH: Cursor position y, upper byte (1 pixel / unit)

Definable area: 0000h ≤ (xL + xH×100h) ≤ **Max_Xdot**
0000h ≤ (yL + yH×100h) ≤ **Max_Ydot**

Function: Cursor moves to the specified (X, Y) position on display memory.

If the specified X, Y position (X and/or Y) is outside the definable area, the command is ignored and the cursor remains in the same position.

This command has effect for the currently-selected window.

3.7.4.4 US (w 10h a (Write Screen Mode Select)

Code: 1Fh 28h 77h 10h a

a: Write screen mode

Definable area: 00h ≤ a ≤ 01h

a = 00h: Display screen mode

a = 01h: All screen mode

Default: a = 00h or Memory SW setting.

Function: Select the write screen mode.

This setting is only applicable for base-window.

Display screen mode: Display action is valid within area of either display area or hidden area, depending on cursor position.

All screen mode: Display action is valid over the entire display memory.

3.7.4.5 ESC R n (International Font Set)**Code:** 1Bh 52h n**Definable area:** 00h ≤ n ≤ 0Dh**Default:** n = 00h or Memory SW setting.**Function:** Select international font set.

Characters already displayed are not affected.

n	Font set
00h	America
01h	France
02h	Germany
03h	England
04h	Denmark 1
05h	Sweden
06h	Italy
07h	Spain1
08h	Japan
09h	Norway
0Ah	Denmark2
0Bh	Spain2
0Ch	Latin America
0Dh	Korea

Note: This setting is not used if UTF-8 input is selected (refer to 3.7.4.6 Character Table Type).

3.7.4.6 ESC t n (Character Table Type)**Code:** 1Bh 74h n**Definable area:** n = 00h, 01h, 02h, 03h, 04h, 05h, 10h, 11h, 12h, 13h, FEh, FFh**Default:** n = 00h or Memory SW setting.**Function:** Select Character table type.

Characters already displayed are not affected.

FFh (User table): User-defined font table (refer to 3.7.4.57 FROM User Font Definition).

n	Font code type
00h	PC437(USA – Euro std)
01h	Katakana – Japanese
02h	PC850 (Multilingual)
03h	PC860 (Portuguese)
04h	PC863 (Canadian-French)
05h	PC865 (Nordic)
10h	WPC1252
11h	PC866 (Cyrillic #2)
12h	PC852 (Latin 2)
13h	PC858
FEh	UTF-8 input
FFh	User table

3.7.4.7 US MD1 (Over-write Mode)**Code:** 1Fh 01h**Function:** Display mode set to Over-write mode.
This command has effect for the currently-selected window.**3.7.4.8 US MD2 (Vertical Scroll Mode)****Code:** 1Fh 02h**Function:** Display mode set to Vertical scroll mode.
This command has effect for the currently-selected window.**3.7.4.9 US MD3 (Horizontal Scroll Mode)****Code:** 1Fh 03h**Function:** Display mode set to Horizontal scroll mode.
This command has effect for the currently-selected window.**3.7.4.10 US MD5 (Horizontal Scroll Mode, Scroll ON)****Code:** 1Fh 05h**Function:** Display mode set to Horizontal scroll mode, scroll ON state.
After this command, operation is same as MD3 mode.
This command has effect for the currently-selected window.**3.7.4.11 US s n (Horizontal Scroll Speed)****Code:** 1Fh 73h n**Definable area:** $00h \leq n \leq 1Fh$ **Default:** n = 00h or Memory SW setting.**Function:** Set speed for Horizontal scroll mode.

Scroll speed is set by 'n'.

Subsequent commands are not processed until scroll is completed.

Scroll base time period 'T' is **IntTime**, but may be longer due to screen mode or character size, etc.

n	Speed
00h	Instantaneous
01h	T ms / 2 pixels
02h – 1Fh	(n-1) × T ms / pixel

Note: Scroll speed is approximate.

Depending on the scrolling area, scroll may reduce in speed or flicker.

3.7.4.12 US (g 01h m (Font Size Select)**Code:** 1Fh 28h 67h 01h m**Definable area:** m = 00h, 01h, 02h, 03h, 04h**Default:** m = 01h or Memory SW setting.**Function:** Sets the font size for 1-byte characters.

m	Function
00h	Outline font*
01h	6×8 pixel character
02h	8×16 pixel character
03h	12×24 pixel character
04h	16×32 pixel character

*Outline font type must be selected first, or else setting is ignored (refer to 3.7.4.17).

3.7.4.13 US (g 02h m (2-byte Character)**Code:** 1Fh 28h 67h 02h m**Definable area:** m = 00h, 01h**Default:** m = 00h or Memory SW setting.**Function:** Sets 2-byte character ON/OFF.

m	Function
00h	2-byte character mode OFF
01h	2-byte character mode ON

Note: This setting is not used if UTF-8 input is selected (refer to 3.7.4.6 Character Table Type).

3.7.4.14 US (g 03h m (2-byte Character Type)**Code:** 1Fh 28h 67h 03h m

or 1Fh 28h 67h 0Fh m

Definable area: m = 00h, 01h, 02h, 03h**Default:** m = 00h or Memory SW setting.**Function:** Sets 2-byte character type.

m	Function	Code Type
00h	Japanese	JIS X0208 (Shift-JIS)
01h	Korean	KSC5601-87
02h	Simplified Chinese	GB2312-80
03h	Traditional Chinese	Big-5

To display a 16×16 pixel, 2-byte character:Font size select: Code: 1Fh 28h 67h 01h **02h**2-byte character ON: Code: 1Fh 28h 67h 02h **01h**

2-byte character type: Code: 1Fh 28h 67h 03h 00h Japanese

1Fh 28h 67h 03h 01h Korean

1Fh 28h 67h 03h 02h Simplified Chinese

1Fh 28h 67h 03h 03h Traditional Chinese

2-byte character code input: Code: 88h 9Fh (“亜” Example Japanese character)

Note: This setting is not used if UTF-8 input is selected (refer to 3.7.4.6 Character Table Type).

3.7.4.15 US (g 06 ylsL ylsH ysL ysH xsL xsH boL boH (Outline Font Size)

Code: 1Fh 28h 67h 06h ylsL ylsH ysL ysH xsL xsH boL boH

ylsL: Y-size (height) of character line (line spacing), in pixels (x1h)
 ylsH: Y-size (height) of character line (line spacing), in pixels (x100h)
 ysL: Nominal character Y-size (height), in pixels (x1h)
 ysH: Nominal character Y-size (height), in pixels (x100h)
 xsL: Nominal character X-size (width), in pixels (x1h)
 xsH: Nominal character X-size (width), in pixels (x100h)
 boL: Baseline Y-offset from cursor position, in pixels (x1h)
 boH: Baseline Y-offset from cursor position, in pixels (x100h)

Definable area: $0001h \leq (ylsL + ylsH \times 100h) \leq \text{Ydots}$ (other values: Default or existing setting)
 $0001h \leq (ysL + ysH \times 100h) \leq 03FFh$ (if higher: capped, if zero: use xs or auto)
 $0001h \leq (xsL + xsH \times 100h) \leq 03FFh$ (if higher: capped, if zero: use ys or auto)
 $FC00h \leq (boL + boH \times 100h) \leq 03FFh$, 8000h (16-bit signed integer)
 (If higher or lower: capped, if zero: auto, if 8000h: zero offset)

Default: yls = 64, ys, xs, bo = 0 (auto-calculated)

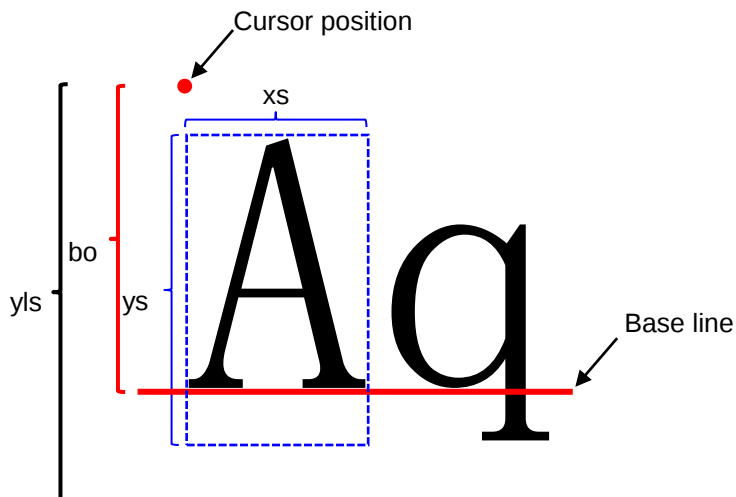
Function: Set character size and line spacing and alignment for outline (scalable) font display.

For displaying outline (scalable) font characters, the size and alignment is specified using four 16-bit parameters, yls, ys, xs, and bo.

'yls' is the most basic parameter, specifying the number of pixels in the y-direction for the region within which the characters will be written (and background cleared, if applicable). This is used as the y-line size (character size), when determining character display operation, and cursor behavior, etc., as described in section 3.7.2.1. When multiple lines of text are displayed, this value determines the (vertical) line pitch. This parameter value is capped (if necessary) at **Ydots**. If set to zero, the previous setting (or default setting, 64) is used.

'ys' and 'xs' set the nominal character size in pixels for the y- and x-directions respectively. This is normally less than 'yls'. If either parameter is zero, the value is set to the same as the other parameter. If both are zero, both are set to values automatically calculated with reference to 'yls'. The actual size of displayed characters depends on the font design, so 'ys' and 'xs' are relative values.

'bo' sets the offset in pixels, from the cursor position (top, left-hand corner) to the baseline on which the outline font text is placed. This is a 16-bit signed value. Typically, positive values are used, which places the baseline below the cursor (negative values place the baseline above the cursor). If set to zero, the value is automatically calculated with reference to the other parameters to attempt to place the text vertically centered within the line. To explicitly set an offset value of zero, set 'bo' to 8000h.



3.7.4.16 US (g 07 ad0 ad1 ad2 ad3 sz0 sz1 sz2 sz3 (User-supplied Font File Address and Size)**Code:** 1Fh 28h 67h 07h ad0 ad1 ad2 ad3 sz0 sz1 sz2 sz3

ad3:ad2:ad1:ad0: Address of start of font file (binary image) in General-purpose FROM2.

sz3:sz2:sz1:sz0: Size in bytes of font file (binary image) in General-purpose FROM2.

Definable area: 0000h ≤ (ad3:ad2:ad1:ad0) ≤ 0FAFFFFFFh
0001h ≤ (sz3:sz2:sz1:sz0) ≤ 0FB00000h – (ad3:ad2:ad1:ad0)**Default:** ad, sz = FFFFFFFFh (no font file specified)**Function:** Set address and size in General-purpose FROM2 for the user-supplied font file (binary image).

Loading of the specified font file is attempted when FFh is selected as the outline font type (refer to 3.7.4.17).

Regarding the file size, in addition to the above definition area, there is a difference in the upper limit of the file size that can be used depending on the product.

4.3 inch product: 8MB

Other products: 32MB

If the above definition area is exceeded, this command is ignored.

3.7.4.17 US (g 08 n (Outline Font Type Select)**Code:** 1Fh 28h 67h 08h n

n: Outline font type

Definable area: n = 00h: Japanese
n = 01h: Korean
n = 02h: Simplified Chinese
n = 03h: Traditional Chinese
n = 80h: None (no outline font selected)
n = FFh: User-supplied font file (refer to 3.7.4.16).**Default:** n = 80h**Function:** Sets the outline font type.

If using outline font, please send commands by the following procedure.

<When using standard font of 2-byte character>

Step 1: Character Table Type (3.7.4.6)... n = 00h (other than UTF-8)

Step 2: Outline Font Type Select (3.7.4.17)... Japanese n = 00h

Step 3: Font Size Select (3.7.4.12)... Outline font m = 00h

Step 4: 2-byte Character (3.7.4.13)... Designation m = 01h

Step 5: 2-byte Character Type (3.7.4.14)... Japanese m = 00h

Step 6: Outline Font Size (3.7.4.15)... Yls = 64, yx, xs, bo = auto-calculated 1Fh, 28h, 67h, 06h, 00h, 40h, 00h, 00h, 00h, 00h, 00h

<When using standard font of UTF-8>

Step 1: Character Table Type (3.7.4.6)... n = FEh

Step 2: Outline Font Type Select (3.7.4.17)... Japanese n = 00h

Step 3: Font Size Select (3.7.4.12)... Outline font m = 00h

Step 4: Outline Font Size (3.7.4.15)... Yls = 64, yx, xs, bo = auto-calculated 1Fh, 28h, 67h, 06h, 00h, 40h, 00h, 00h, 00h, 00h, 00h

<When using user fonts>

Step 1: User-supplied Font File Address and Size (3.7.4.16)

Step 2: Outline Font Type Select (3.7.4.17)... User font file n = FFh

Step 3: Font Size Select (3.7.4.12)... Outline font m = 00h

Step 4: Outline Font Size (3.7.4.15)... Yls = 64, yx, xs, bo = automatic calculation 1Fh, 28h, 67h, 06h, 00h, 40h, 00h, 00h, 00h, 00h, 00h

The command must be re-executed when the power is turned ON / OFF or the initialization command is executed.

If font loading fails (e.g., due to incompatible or corrupted font file), outline font is set to "none".

For n = 01h (Korean), a small number of symbol characters, available in the built-in pixel font, are not available in the built-in outline font.

The outline font is read after this command execution, but it will take several seconds. Please do not send any data during that time.

3.7.4.18 US (g 04h m (Font Width)**Code:** 1Fh 28h 67h 04h m

m: Font width setting

Definable area: m = 00h, 02h, 03h, 04h**Default:** m = 00h**Function:** Sets the character width.

For fixed-width font, all characters are displayed as font size.

For proportional font, characters are displayed as:

Display width = right blank + character width + left blank

Note: This command is applicable for pixel character font only, not outline (scalable) font.

6×8 pixel character

m	Character width	Left blank pixels	Right blank pixels	Space (20h) character width
00h	Fixed-width	0	0	6
02h	Proportional 1	0	1	2
03h	Proportional 2	1	1	2
04h	Proportional 3	2	2	2

8×16 pixel character

m	Character width	Left blank pixels	Right blank pixels	Space (20h) character width
00h	Fixed-width	0	0	8
02h	Proportional 1	0	1	4
03h	Proportional 2	1	1	4
04h	Proportional 3	2	2	4

12×24 pixel character

m	Character width	Left blank pixels	Right blank pixels	Space (20h) character width
00h	Fixed-width	0	0	12
02h	Proportional 1	0	2	6
03h	Proportional 2	2	2	6
04h	Proportional 3	4	4	6

16×32 pixel character

m	Character width	Left blank pixels	Right blank pixels	Space (20h) character width
00h	Fixed-width	0	0	16
02h	Proportional 1	0	2	8
03h	Proportional 2	2	2	8
04h	Proportional 3	4	4	8

Example: Proportional 2, 6×8 pixel character 'l'

Left blank	Character width			Right blank

3.7.4.19 US (g 05h n (FROM Extended Font)**Code:** 1Fh 28h 67h 05h n

n: FROM Extended Font select

Definable area: 00h ≤ n ≤ FFh

n = 00h: Normal font.

n = 01h–FFh: FROM Extended Font (if FROM Extended Font is defined).

Default: n = 00h**Function:** FROM extended font select.**3.7.4.20 US (g 40h x y (Font Magnification)****Code:** 1Fh 28h 67h 40h x y

x: X magnification factor

y: Y magnification factor

Definable area: 01h ≤ x ≤ 04h

01h ≤ y ≤ 04h

Default: x = 01h or Memory SW setting.

y = 01h or Memory SW setting.

Function: Sets character magnification 'x' times to the right and 'y' times downward.

Note: This command is applicable for pixel character font only, not outline (scalable) font.

3.7.4.21 US (g 41h b (Character Style)**Code:** 1Fh 28h 67h 41h b

b: Style

Definable area: 00h ≤ b ≤ 03h

b = 00h: Normal

b = 01h: Bold

b = 02h: Shadow

b = 03h: Bordering

Default: b = 00h or Memory SW setting.**Function:** Character style setting (*Settings may reduce legibility*).

Note: This command is applicable for pixel character font only, not outline (scalable) font.

3.7.4.22 US (g 50h pR pG pB (Character Color)**Code:** 1Fh 28h 67h 50h pR pG pB

pB: Blue brightness

pG: Green brightness

pR: Red brightness

Definable area: 00h ≤ pR, pG, pB ≤ FFh

00h: OFF

:

7Fh: 50%

:

FFh: Maximum

Default: pR = FFh, pG = FFh, pB = FFh**Function:** Specifies the character color.**3.7.4.23 US (g 51h pR pG pB (Background Color)****Code:** 1Fh 28h 67h 51h pR pG pB

pB: Blue brightness

pG: Green brightness

pR: Red brightness

Definable area: 00h ≤ pR, pG, pB ≤ FFh

00h: OFF

:

7Fh: 50%

:

FFh: Maximum

Default: pR = 00h, pG = 00h, pB = 00h**Function:** Specifies the character background color.

This setting is valid when background color is enabled (Refer to 3.7.4.25 Background Color Enable/Disable).

3.7.4.24 US (g 52h pR pG pB (Shadow and Bordering Color)**Code:** 1Fh 28h 67h 52h pR pG pB

pB: Blue brightness

pG: Green brightness

pR: Red brightness

Definable area: 00h ≤ pR, pG, pB ≤ FFh

00h: OFF

:

7Fh: 50%

:

FFh: Maximum

Default: bB = 00h, pG = 00h, pR = 00h**Function:** Specifies the character shadow and bordering color.

This setting is valid when shadow or bordering is selected (Refer to 3.7.4.21 Character Style).

Note: This command is applicable for pixel character font only, not outline (scalable) font.

3.7.4.25 US (g 58h b (Background Color Enable/ Disable)**Code:** 1Fh 28h 67h 58h b

b: Background enable/disable

Definable area: b = 00h, 01h

b = 00h: Background disabled

b = 01h: Background enabled

Default: b = 00h**Function:** Background color for character display enabled or disabled.

Background color disabled is functionally equivalent to background color enabled with Background color = black (pR = 00h, pG = 00h, pB = 00h).

Background setting is not used if Write mixture display mode is set to "Thru write" mode (background is transparent in this case). For details, refer to 3.9 Color Combination Format.

3.7.4.26 US (a 01h t (Wait)**Code:** 1Fh 28h 61h 01h t

t: Wait time

Definable area: 00h ≤ t ≤ FFh**Function:** Waits for the specified time (command and data processing is stopped).

Wait time = t × approximately 0.47s

Command / data processing does not resume until wait time is completed.

It is possible to interrupt this command if the command is defined and run in a macro.

3.7.4.27 US (a 02h t (Short Wait)**Code:** 1Fh 28h 61h 02h t

t: Wait time

Definable area: 00h ≤ t ≤ FFh**Function:** Waits for the specified time (command and data processing is stopped).Wait time = t × **IntTime** (refer to 2.7 Timing Unit)

Command / data processing does not resume until wait time is completed.

It is possible to interrupt this command if the command is defined and run in a macro.

3.7.4.28 US (a 11h p t1 t2 c (Blink)**Code: 1Fh 28h 61h 11h p t1 t2 c**

p: Blink pattern
 t1: Normal display time
 t2: Blank display time
 c: Number of cycles

Definable area: $00h \leq p \leq 02h$
 p = 00h: Normal display.
 p = 01h: Blink display (alternately Normal and Blank display).
 p = 02h: Reserved.
 $01h \leq t1 \leq FFh$
 $01h \leq t2 \leq FFh$
 $00h \leq c \leq FFh$

Function: Blink display action Blink pattern specified by 'p'.

Time specified by 't1', 't2'.

A: $t1 \times \text{IntTime}$ Normal displayB: $t2 \times \text{IntTime}$ Blank display

Repeated 'c' times.

This command does not affect display memory.

c = 00h: Blink continues during subsequent command and data processing, until c=01h–FFh is set, or Initialize command.

c = 01h–FFh: Blink display is repeated 1–255 times while command and data processing is stopped. After display blinking is completed, Normal display returns and command and data processing resumes.

Command / data processing does not resume until operation is completed.

This command cannot be interrupted when running in a macro.

3.7.4.29 US (a A0h sXL sXH sYL sYH cL cH s (Scroll Display Action XY)**Code: 1Fh 28h 61h A0h sXL sXH sYL sYH cL cH s**

sXL: Number of display screen X shift (x1h)
 sXH: Number of display screen X shift (x100h)
 sYL: Number of display screen Y shift (x1h)
 sYH: Number of display screen Y shift (x100h)
 cL: Number of repetition lower byte
 cH: Number of repetition upper byte
 s: Scroll speed

Definable area: $0000h \leq (sXL + sXH \times 100h) \leq FFFFh$
 $0000h \leq (sYL + sYH \times 100h) \leq FFFFh$
 $0001h \leq (cL + cH \times 100h) \leq FFFFh$
 $00h \leq s \leq FFh$

Function: Shift the display screen.

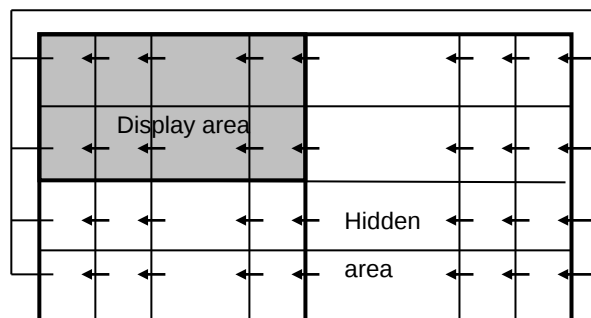
Scroll speed is specified by 's'.

Scroll speed = $s \times \text{IntTime} / \text{shift}$

Command / data processing does not resume until scroll is completed.

It is possible to interrupt this command if the command is defined and run in a macro.

For example: 1-pixel scroll to the left: sXL=01h, sXH=00h, sYL=00h, sYH=00h



3.7.4.30 US (a A2h v s pR pG pB (Curtain Display Action XY)**Code: 1Fh 28h 61h A2h v s pR pG pB**

v: Direction of Curtain action
 s: Curtain action speed
 pR: Red brightness
 pG: Green brightness
 pB: Blue brightness

Definable area: $00h \leq v \leq 03h$
 v = 00h: To the Right from the Left edge
 v = 01h: To the Left from the Right edge
 v = 02h: To the Left and Right separately from the Center
 v = 03h: To the Center from Left edge and Right edge
 $00h \leq s \leq FFh$
 $00h \leq pR \leq FFh$
 $00h \leq pG \leq FFh$
 $00h \leq pB \leq FFh$

Function: Curtain display action on display screen.

Curtain action is displayed from the direction specified by 'v'.

Curtain action speed = $60 \times s \times \text{IntTime}$

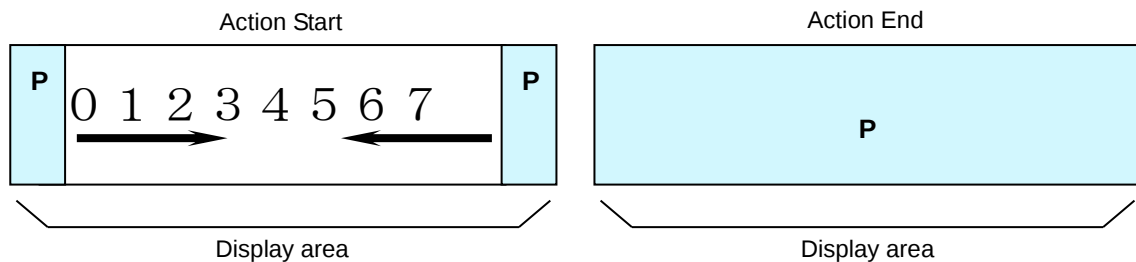
This command only affects the display area.

The non-display area memory is not affected.

Command / data processing does not resume until operation is completed.

If it is necessary to be able to cancel the display action during processing, this is possible if the command is defined and run in a macro.

For examples: v = 03h



3.7.4.31 US (a A3h v s pXL pXH pYL pYH (Spring Display Action XY)

Code: 1Fh 28h 61h A3h v s pXL pXH pYL pYH

v: Direction of spring action
s: Spring action speed
pXL: Display memory X position (x1h)
pXH: Display memory X position (x100h)
pYL: Display memory Y position (x1h)
pYH: Display memory Y position (x100h)

Definable area: $00h \leq v \leq 03h$
v=00h: To the right from the left edge.
v=01h: To the left from the right edge.
v=02h: To the left and right separately from the center.
v=03h: To the center from left edge and right edge.
 $00h \leq s \leq FFh$
 $0000h \leq (pXL + pXH \times 100h) \leq \text{DispXdots}$
 $0000h \leq (pYL + pYH \times 100h) \leq \text{DispYdots}$

Function: Spring display action on display screen.

Pattern 'p' specified by display memory pattern address is displayed from the direction specified by 'v'.

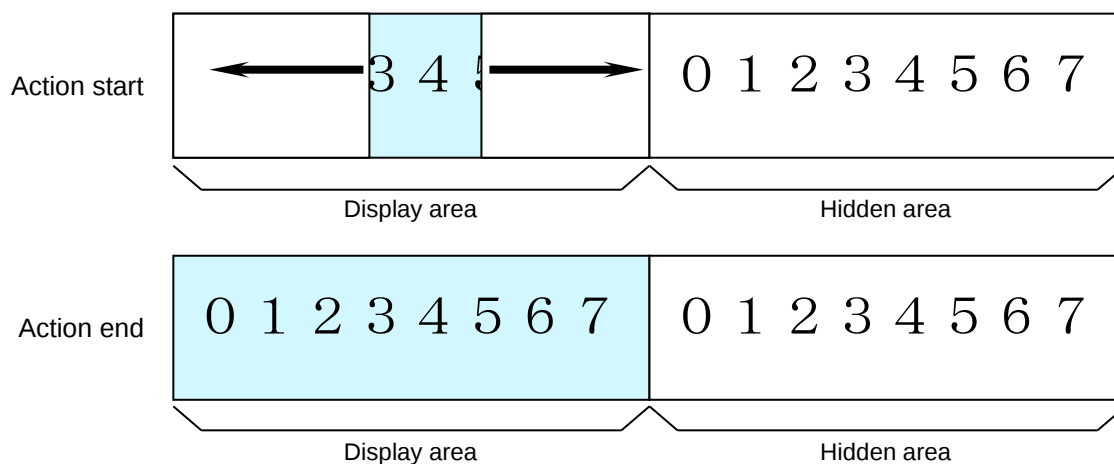
Spring action speed = $60 \times s \times \text{IntTime}$

This command affects only the display area, not hidden area.

Command / data processing does not resume until operation is completed.

If it is necessary to be able to cancel the display action during processing, this is possible if the command is defined and run in a macro.

Example is shown below.



3.7.4.32 US (a A4h v s pXL pXH pYL pYH (Random Display Action XY)

Code: 1Fh 28h 61h A4h v s pXL pXH pYL pYH

v: Random display action type
 s: Random display action speed
 pXL: Display memory X position ($\times 1h$)
 pXH: Display memory X position ($\times 100h$)
 pYL: Display memory Y position ($\times 1h$)
 pYH: Display memory Y position ($\times 100h$)p

Definable area: $00h \leq v \leq 02h$
 $00h \leq s \leq FFh$
 $0000h \leq (pXL + pXH \times 100h) \leq \text{DispXdots}$
 $0000h \leq (pYL + pYH \times 100h) \leq \text{DispYdots}$

Function: Random display action on display screen.

Pattern 'p' specified by display memory pattern address is displayed randomly, using one of three different transition patterns selected by 'v'.

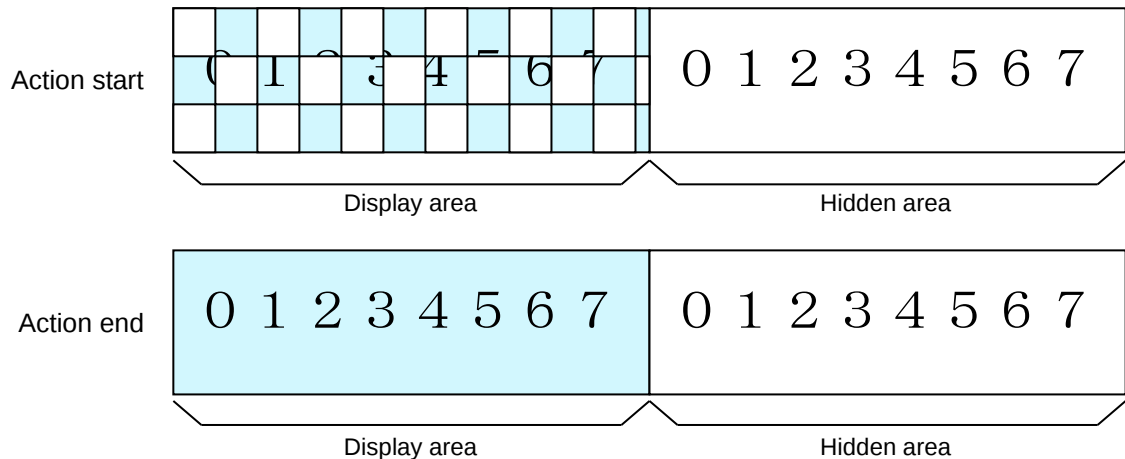
Random display action is completed in 16 steps, at approximately $(s \times 4 \times \text{IntTime}) / \text{step}$.

This command affects only the display area, not hidden area.

Command / data processing does not resume until operation is completed.

If it is necessary to be able to cancel the display action during processing, this is possible if the command is defined and run in a macro.

Example is shown below.



Transition patterns:

15	5	8	13	1	2	3	4	3	5	7	9
1	12	4	6	12	13	14	5	5	7	9	11
9	7	16	2	11	16	15	6	7	9	11	13
14	3	11	10	10	9	8	7	9	11	13	15
v=00h				v=01h				v=02h			

Within each 16×16-pixel area on the display screen, sixteen 4×4-pixel cells are arranged in a grid. These transition to the new display pattern on the timing step (1 – 16) indicated by the number in that cell.

3.7.4.33 US (a A5h s pXL pXH pYH pYL (Fade In Display Action XY)**Code: 1Fh 28h 61h A5h s pXL pXH pYL pYH**

s: Fade in display action speed
 pXL: Display memory X position (×1h)
 pXH: Display memory X position (×100h)
 pYL: Display memory Y position (×1h)
 pYH: Display memory Y position (×100h)

Definable area: $00h \leq s \leq FFh$
 $0000h \leq (pXL + pXH \times 100h) \leq \text{DispXdots}$
 $0000h \leq (pYL + pYH \times 100h) \leq \text{DispYdots}$

Function: Fade in display action on display screen.

Pattern 'p' specified by display memory pattern address is displayed by fade in.

Fade in time = s × approximately 0.5s

This command affects only the display area, not hidden area.

Command / data processing does not resume until operation is completed.

If it is necessary to be able to cancel the display action during processing, this is possible if the command is defined and run in a macro.

3.7.4.34 US (a A6h s (Fade Out Display Action XY)**Code: 1Fh 28h 61h A6h s**

s: Fade out display action speed

Definable area: $00h \leq s \leq FFh$ **Function:** Fade out display action on display screen.

Display area is cleared by fade out.

Fade out time = s × approximately 0.5s

This command affects only the display area, not hidden area.

Command / data processing does not resume until operation is completed.

If it is necessary to be able to cancel the display action during processing, this is possible if the command is defined and run in a macro.

3.7.4.35 US (a 40h p (Display Power ON/ OFF)**Code: 1Fh 28h 61h 40h p**

p: Set display power ON / OFF / Auto-ON

Definable area: p = 00h: Display power OFF (Display OFF)
 p = 01h: Display power ON (Display ON)
 p = 80h: Display power OFF, automatic ON when touch detected.

Default: p = 01h**Function:** Set display power ON / OFF / Auto-ON.

To change from display power OFF (p = 00h) to display power ON (p=01h), set this command to display power ON or set the initialization command.

3.7.4.36 US (d 10h pen xL xH yL yH (Pixel Drawing)**Code: 1Fh 28h 64h 10h pen xL xH yL yH**

pen: Pixel display ON or OFF
 xL: Pixel position x, lower byte
 xH: Pixel position x, upper byte
 yL: Pixel position y, lower byte
 yH: Pixel position y, upper byte

Definable area: $00h \leq pen \leq 01h$
 pen = 00h: Pixel display OFF
 * Pen color is set to the background color (refer to 3.7.4.23).
 pen = 01h: Pixel display ON
 * Pen color is set to the character color (refer to 3.7.4.22).
 $0000h \leq (xL + xH \times 100h) \leq \text{Max_Xdot}$
 $0000h \leq (yL + yH \times 100h) \leq \text{Max_Ydot}$

Function: Display the pixel at the position specified, or set pixel to background color.

This command has effect for the currently-selected window.

If pixel display ON/ OFF, or pixel position is outside the definable area, the command is cancelled at that point and the following data is treated as standard data.

3.7.4.37 US (d 11h mode pen x1L x1H y1L y1H x2L x2H y2L y2H (Line/ Box Pattern Drawing)**Code: 1Fh 28h 64h 11h mode pen x1L x1H y1L y1H x2L x2H y2L y2H**

mode: Line / Box / Box FILL select (Refer to illustration)
 pen: Pixel display ON or OFF
 x1L: Line/Box pattern drawing start position x, lower byte
 x1H: Line/Box pattern drawing start position x, upper byte
 y1L: Line/Box pattern drawing start position y, lower byte
 y1H: Line/Box pattern drawing start position y, upper byte
 x2L: Line/Box pattern drawing end position x, lower byte
 x2H: Line/Box pattern drawing end position x, upper byte
 y2L: Line/Box pattern drawing end position y, lower byte
 y2H: Line/Box pattern drawing end position y, upper byte

Definable area: $00h \leq mode \leq 02h$
 mode = 00h: Line
 mode = 01h: Box
 mode = 02h: Box fill
 $00h \leq pen \leq 01h$
 pen = 00h: Line/ box display OFF
 * Pen color is set to the background color (refer to 3.7.4.23).
 pen = 01h: Line/ box display ON
 * Pen color is set to the character color (refer to 3.7.4.22).
 $0000h \leq (x1L + x1H \times 100h) \leq \text{Max_Xdot_CurtWin}$
 $0000h \leq (y1L + y1H \times 100h) \leq \text{Max_Ydot_CurtWin}$
 $0000h \leq (x2L + x2H \times 100h) \leq \text{Max_Xdot_CurtWin}$
 $0000h \leq (y2L + y2H \times 100h) \leq \text{Max_Ydot_CurtWin}$

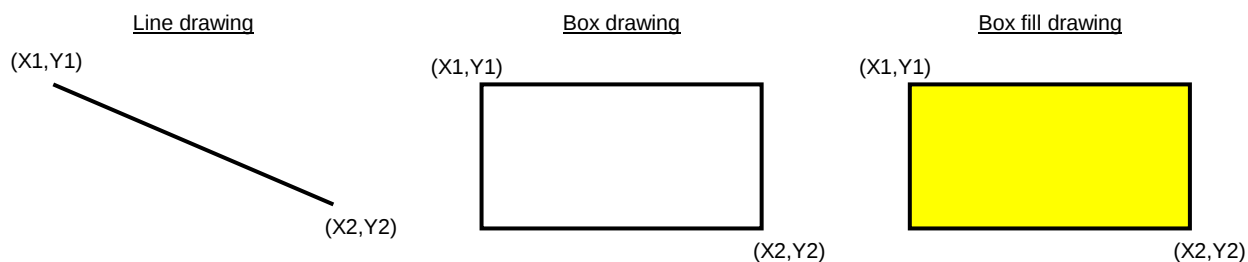
Function: Display a line, box, or box fill on the drawing area specified by (x1,y1)–(x2,y2) or set the pixels to the background color.

This command has effect for the currently-selected window.

If display ON/OFF or pixel pattern drawing position is outside the definable area, the command is cancelled at that point and the following data is treated as standard data.

If a diagonal line is specified, parts of the line may be 2 or more pixels in width.

Drawing mode



3.7.4.38 US (f 11h xL xH yL yH fmt d(1)...d(n) (Real-time Bit Image Display)**Code:** 1Fh 28h 66h 11h xL xH yL yH fmt d(1)...d(n)

xL: Bit image X size, lower byte (by 1 pixel)

xH: Bit image X size, upper byte (by 1 pixel)

yL: Bit image Y size, lower byte (by 1 pixel)

yH: Bit image Y size, upper byte (by 1 pixel)

fmt: Image format

d(1) – d(n): Image data (see below)

Definable area: $0001h \leq (xL + xH \times 100h) \leq Xdots$ $0001h \leq (yL + yH \times 100h) \leq Ydots$

fmt = 81h, 86h, 8Ch, 90h, 98h, F0h

fmt = 81h: Monochrome (1-bit) format*

*: 1 byte per 8 pixels or part thereof for each horizontal line.

(Refer to "3.8.1 Normal Bit Image Data Format")

fmt = 86h: Color 6-bit format

fmt = 8Ch: Color 12-bit format

fmt = 90h: Color 16-bit format

fmt = 98h: Color 24-bit format

fmt = F0h: BMP file format

 $00h \leq d \leq FFh$ (Refer to 3.8.1 Normal Bit Image Data Format)**Function:** Display the bit image data at the cursor position in real-time.
Cursor position does not change.

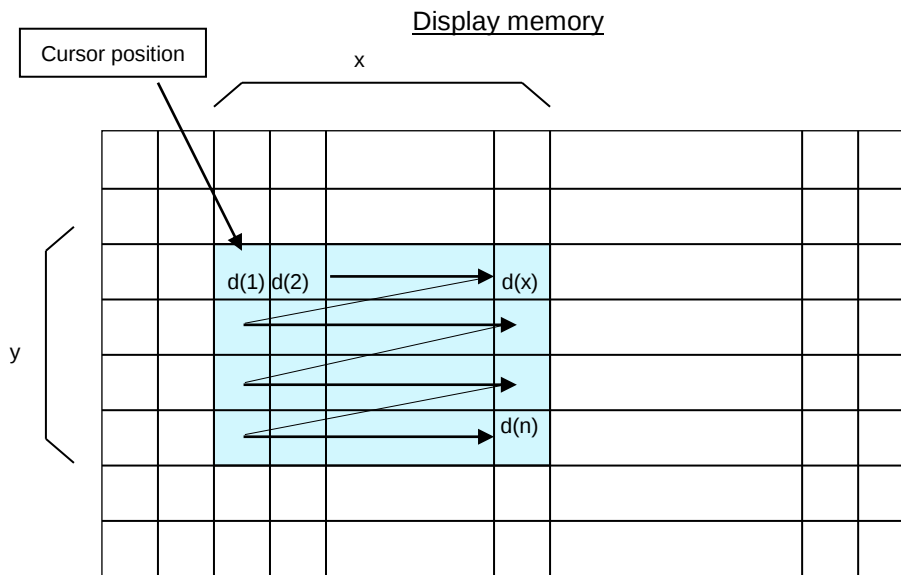
If bit image exceeds the bounds of the current window, only the portion within the currently-selected window is displayed.

If display position or display size etc., are outside the definable area, the command is cancelled at the point where the error is detected, and the remaining data is treated as standard data.

Display the bit image data at the cursor position in real-time.

Note: for BMP file format, the image size specified in the file header is used. 1, 4, 8, 24, and 32 bit BMP files are supported. 16 bit BMP files are not supported.

Example:



3.7.4.39 US (f 21h xL xH yL yH fmt d(1)...d(n) (Packaged Real-time Bit Image Display)**Code:** 1Fh 28h 66h 21h xL xH yL yH fmt d(1)...d(n)

xL: Bit image X size, lower byte (by 1 pixel)

xH: Bit image X size, upper byte (by 1 pixel)

yL: Bit image Y size, lower byte (by 1 pixel)

yH: Bit image Y size, upper byte (by 1 pixel)

fmt: Image format

d(1) – d(n): Image data (refer to figure as follows)

Definable area: $0001h \leq (xL + xH \times 100h) \leq \text{Xdots}$ $0001h \leq (yL + yH \times 100h) \leq \text{Ydots}$

fmt = 86h, 8Ch, 98h

fmt = 86h: Color 6-bit format

fmt = 8Ch: Color 12-bit format

fmt = 98h: Color 24-bit format

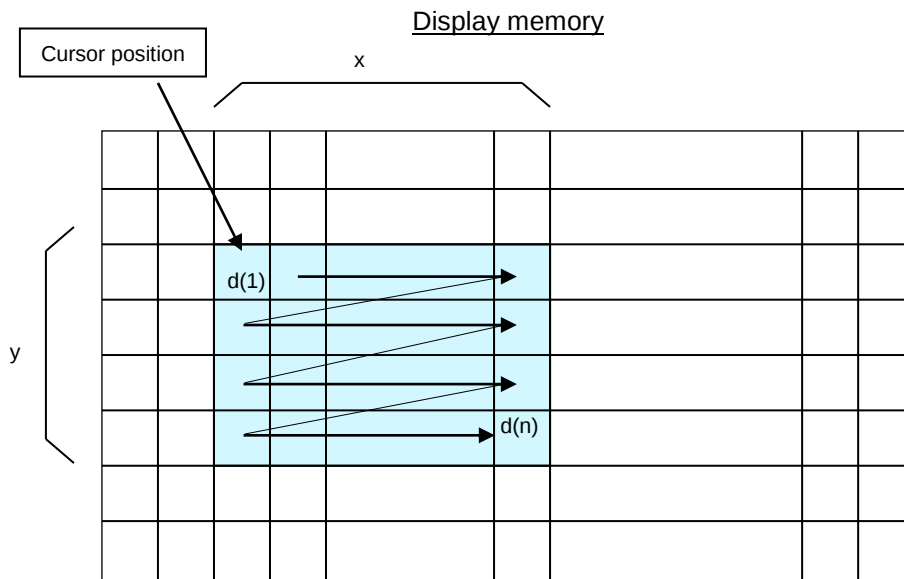
 $00h \leq d \leq FFh$ (Refer to 3.8.2 Packaged Bit Image Data Format)**Function:** Display the bit image data at the cursor position in real-time.

Cursor position does not change.

If bit image exceeds the bounds of the current window, only the portion within the currently-selected window is displayed.

If display position or display size etc., are outside the definable area, the command is cancelled at the point where the error is detected, and the remaining data is treated as standard data.

Example:



3.7.4.40 US (f 01h aL aH aE sL sH sE d(1)...d(s) (RAM Bit Image Definition)

Code: 1Fh 28h 66h 01h aL aH aE sL sH sE d(1)...d(s)

aL: Bit image data definition address, lower byte
 aH: Bit image data definition address, upper byte
 aE: Bit image data definition address, extension byte
 sL: Bit image data length, lower byte
 sH: Bit image data length, upper byte
 sE: Bit image data length, extension byte

d(1) – d(s): Image data (see below)

Definable area: $000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq 000FFFh$
 $000001h \leq (sL + sH \times 100h + sE \times 10000h) \leq 001000h$
 $00h \leq d \leq FFh$

Function: Define user bit image to the RAM.

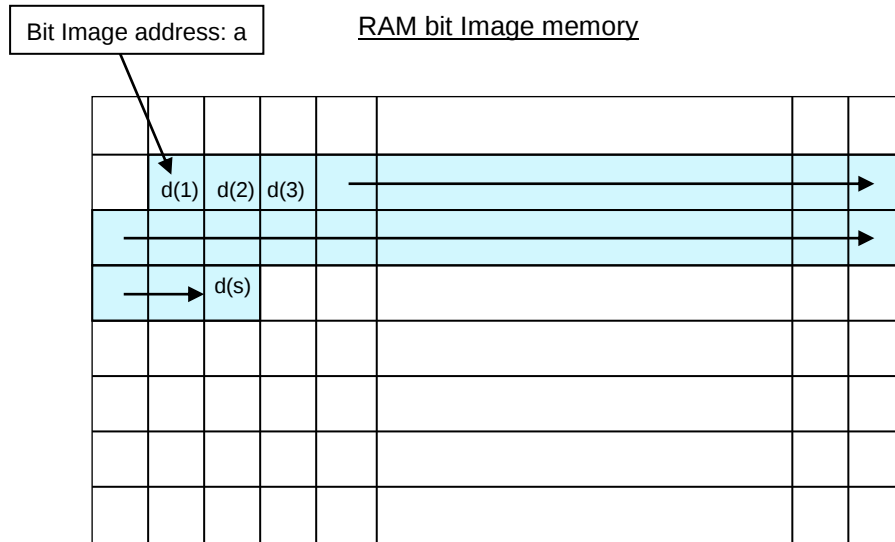
RAM bit image capacity is 4,096 bytes.

Bit image data at arbitrary addresses can be defined or changed by appropriately setting bit image data definition address and bit image data length.

Bit images defined in RAM can be displayed using 3.7.4.42 Downloaded Bit Image Display.

If bit image data definition address or bit image data length is outside the definable area, the command is cancelled at that point, and the remaining data is treated as standard data.

Example:



3.7.4.41 US (e 10h aL aH aE sL sH sE d(1)...d(s) (Bit Image FROM1 Definition)**Code: 1Fh 28h 65h 10h aL aH aE sL sH sE d(1)...d(s)**

aL: Bit image data definition address, lower byte (bit 0 ignored)

aH: Bit image data definition address, upper byte

aE: Bit image data definition address, extension byte

sL: Bit image data length, lower byte (bit 0 ignored)

sH: Bit image data length, upper byte

sE: Bit image data length, extension byte

d(1) – d(s): Image data (see below)

Definable area: aE = 00h–1Fh $000000h \leq ((aL \& FEh) + aH \times 100h + aE \times 10000h) \leq 1FFFFEh$ $000002h \leq ((sL \& FEh) + sH \times 100h + sE \times 10000h) \leq 200000h$ $00h \leq d \leq FFh$ **Function:** Define user bit image to the bit image FROM1.

Bit image FROM1 bit image capacity is 2048Kbytes.

Data is defined or changed from the specified Bit image data definition address for the number of bytes specified by Bit image data length.

Bit images defined in bit image FROM1 can be displayed using 3.7.4.42 Downloaded Bit Image Display.

The least significant bit for both bit image data definition address and bit image data length is ignored – these are processed as even values.

If bit image data definition address or bit image data length is outside the definable area, the command is cancelled at that point, and the remaining data is treated as standard data.

This command is only valid in user setup mode.

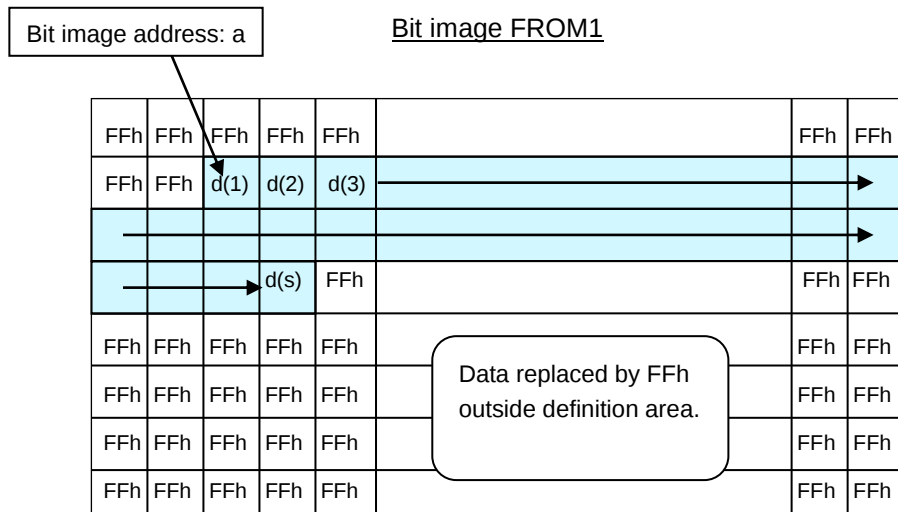
Total definable area is 000000h to 1FFFFFh (2,097,152 bytes). Bit image definition is performed in units of 131,072 bytes (128KB). For example, if 10Kbytes of bit image data is defined, the remaining 118Kbytes data is set to FFh.

Bit image data definition address cannot be specified in such a way that the definition data would overflow the bit imageFROM1 area. The command is cancelled if this situation is detected, and the remaining data is treated as standard data.

Defined contents are not guaranteed if an error occurs.

To store image data in bit image FROM2, refer to 3.7.4.70 General-purpose memory Store / Bit Image FROM2 Store.

Note that the first 2Mbytes memory of bit image FROM2 overlaps the bit image FROM1; changing the content of either memory will affect the other.



3.7.4.42 US (f 10h m aL aH aE xSL xSH xL xH yL yH fmt) (Downloaded Bit Image Display)

Code: 1Fh 28h 66h 10h m aL aH aE xSL xSH xL xH yL yH fmt

m: Select bit image data memory
aL: Bit image data definition address, lower byte
aH: Bit image data definition address, upper byte
aE: Bit image data definition address, extension byte
xSL: Bit image defined X size, lower byte (by 1 pixel)
xSH: Bit image defined X size, upper byte (by 1 pixel)
xL: Bit image display X size, lower byte (by 1 pixel)
xH: Bit image display X size, upper byte (by 1 pixel)
yL: Bit image display Y size, lower byte (by 1 pixel)
yH: Bit image display Y size, upper byte (by 1 pixel)
fmt: Image format

Definable area: $00h \leq m \leq 02h, 10h \leq m \leq 1Fh$

m = 00h:	RAM bit image
m = 01h:	Bit image FROM1 bit image
m = 02h:	Display memory
m = 10h:	Bit image FROM2 base address 0000.0000h
m = 11h:	Bit image FROM2 base address 0100.0000h
m = 12h:	Bit image FROM2 base address 0200.0000h
m = 13h:	Bit image FROM2 base address 0300.0000h
m = 14h:	Bit image FROM2 base address 0400.0000h
m = 15h:	Bit image FROM2 base address 0500.0000h
m = 16h:	Bit image FROM2 base address 0600.0000h
m = 17h:	Bit image FROM2 base address 0700.0000h
m = 18h:	Bit image FROM2 base address 0800.0000h
m = 19h:	Bit image FROM2 base address 0900.0000h
m = 1Ah:	Bit image FROM2 base address 0A00.0000h
m = 1Bh:	Bit image FROM2 base address 0B00.0000h
m = 1Ch:	Bit image FROM2 base address 0C00.0000h
m = 1Dh:	Bit image FROM2 base address 0D00.0000h
m = 1Eh:	Bit image FROM2 base address 0E00.0000h
m = 1Fh:	Bit image FROM2 base address 0F00.0000h

When RAM bit image is selected:

$000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq 000FFFh$

When bit image FROM1 is selected:

$000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq 1FFFFFFh$

When display memory is selected:

$000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq 176FFFh$

When bit image FROM2 is selected:

$000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq FFFFFFFh$ (m=10h–1Eh)

$000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq AFFFFFFh$ (m=1Fh)

$0000h \leq (xSL + xSH \times 100h) \leq FFFFh$

$0001h \leq (xL + xH \times 100h) \leq \text{Xdots}$

$0001h \leq (yL + yH \times 100h) \leq \text{Ydots}$

fmt = 81h, 86h, 8Ch, 90h, 91h, 98h, F0h

fmt = 81h: Monochrome (1-bit) format

fmt = 86h: Color 6-bit format

fmt = 8Ch: Color 12-bit format

fmt = 90h: Color 16-bit format

fmt = 91h: Color 16-bit high-speed format

fmt = 98h: Color 24-bit format

fmt = F0h: BMP file format

Function: Display, at the cursor position, the bit image defined in RAM, bit image FROM1/ FROM2, or in display memory.
Cursor position does not change.

Select RAM, bit image FROM1/ FROM2 or display memory bit image by select bit image data memory 'm'.

Set bit image defined X size to the same X size of the bit image defined in memory.

A portion of the defined bit image can be displayed by setting bit image display X size less than defined bit image X size, or by changing bit image display Y size and/ or bit image data definition address.

If the bit image extends beyond the currently-selected window, only the portion within the current window is displayed.

When the image is being written to the display memory, if the image memory area is exceeded,

undefined data is displayed.

Note: To use $m=02h$ (read from display memory) to copy images within the display memory, use $fmt=98h$, bit image defined X size = ***Xdots***, and data definition address = $X + (Y \times \mathbf{Xdots})$, where (X,Y) is the start (top, left) point of the image.

Note: For BMP file format, the image size specified in the file header is used. 1, 4, 8, 24, and 32 bit BMP files are supported. 16 bit BMP files are not supported.

Note: Display orientation 90° and 270° may take longer to transfer to display memory than display orientation 0° and 180° .

Note for FROM1 memory:

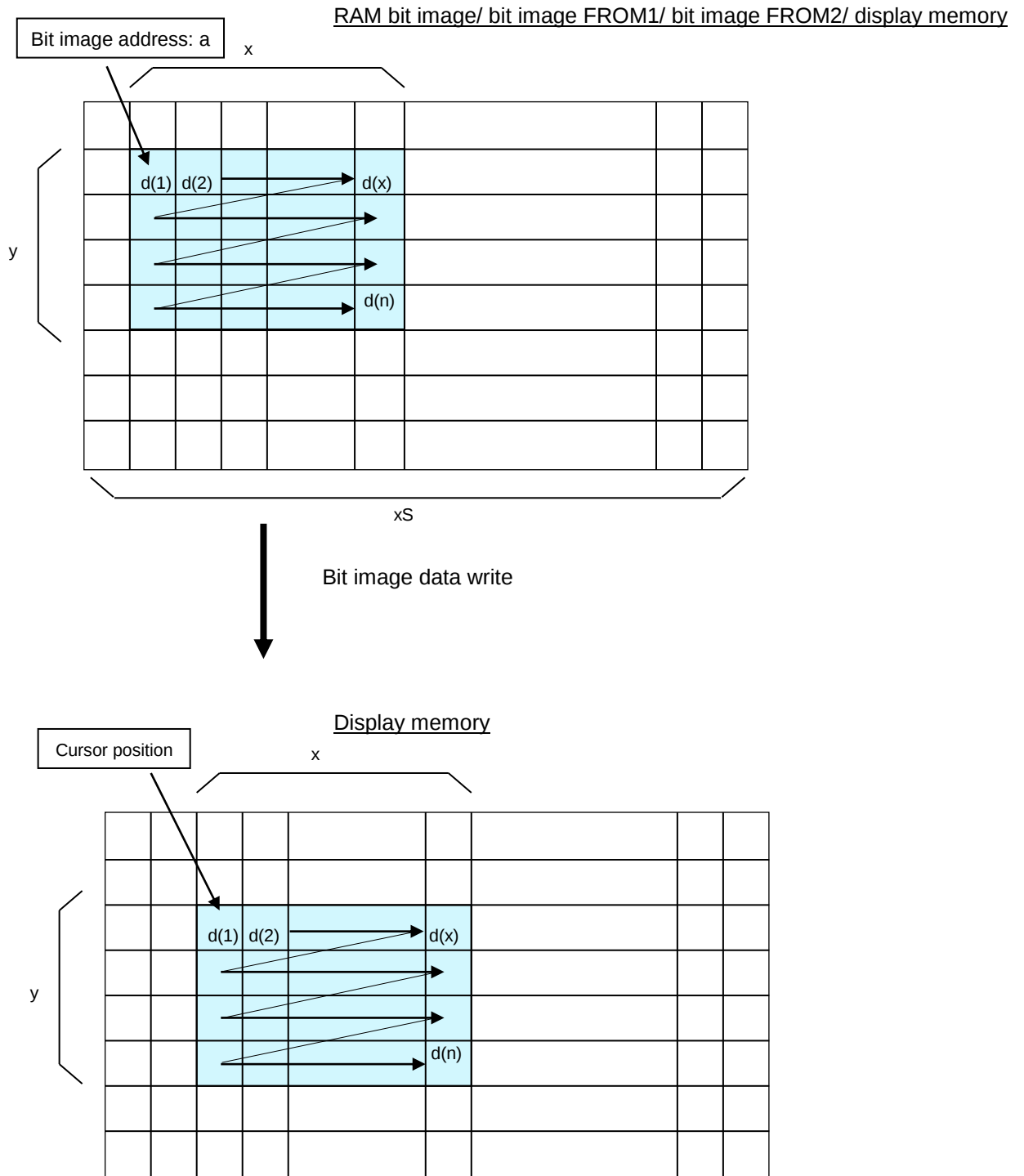
Bit image data can be read from $000000h$ to $1FFFFFFh$ continuously (bit image display can cross a 128KB block boundary).

For $fmt=91h$ (Color 16-bit high-speed format):

Data is transferred directly to display memory, so reverse and write mixture settings are not used.

Bit image data display memory selection (m) must be FROM2 ($10h \leq m \leq 1Fh$).

The least-significant bit of the lower byte (aL) of the bit image data display address must be 0 (address must be even).



3.7.4.43 US (f 20h m aL aH aE pL pH fmt (Packaged Downloaded Bit Image Display)

Code: 1Fh 28h 66h 20h m aL aH aE pL pH fmt
m: Select bit image data display memory
aL: Bit image index start address, lower byte
aH: Bit image index start address, upper byte
aE: Bit image index start address, extension byte
pL: Bit image number, lower byte
pH: Bit image number, upper byte
fmt: Image format

Definable area: $00h \leq m \leq 01h, 10h \leq m \leq 1Fh$

m = 00h: RAM bit image
 m = 01h: Bit image FROM1 bit image
 m = 10h: Bit image FROM2 base address 0000.0000h
 m = 11h: Bit image FROM2 base address 0100.0000h
 m = 12h: Bit image FROM2 base address 0200.0000h
 m = 13h: Bit image FROM2 base address 0300.0000h
 m = 14h: Bit image FROM2 base address 0400.0000h
 m = 15h: Bit image FROM2 base address 0500.0000h
 m = 16h: Bit image FROM2 base address 0600.0000h
 m = 17h: Bit image FROM2 base address 0700.0000h
 m = 18h: Bit image FROM2 base address 0800.0000h
 m = 19h: Bit image FROM2 base address 0900.0000h
 m = 1Ah: Bit image FROM2 base address 0A00.0000h
 m = 1Bh: Bit image FROM2 base address 0B00.0000h
 m = 1Ch: Bit image FROM2 base address 0C00.0000h
 m = 1Dh: Bit image FROM2 base address 0D00.0000h
 m = 1Eh: Bit image FROM2 base address 0E00.0000h
 m = 1Fh: Bit image FROM2 base address 0F00.0000h

When RAM bit image is selected;

$000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq 000FFFh$

When bit image FROM1 is selected;

$000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq 1FFFFFFh$

When bit image FROM2 is selected;

$000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq FFFFFFFh$ (m=10h–1Eh)

$000000h \leq (aL + aH \times 100h + aE \times 10000h) \leq AFFFFFFh$ (m=1Fh)

$0000h \leq (pL + pH \times 100h) \leq FFFFh$

fmt = 00h, 81h, 86h, 8Ch, 98h, F0h

fmt = 00h: Refer to bit image index memory

fmt = 81h: Monochrome (1-bit) format

fmt = 86h: Color 6-bit format

fmt = 8Ch: Color 12-bit format

fmt = 98h: Color 24-bit format

fmt = F0h: BMP file format

Function: Display, at the cursor position, the packaged bit image defined in RAM bit image, bit image FROM1/ FROM2.

Cursor position does not change.

Select RAM bit image, bit image FROM1/ FROM2 by select bit image data display memory 'm'.

Displayed bit image is specified by bit Image Index address and bit Image number.

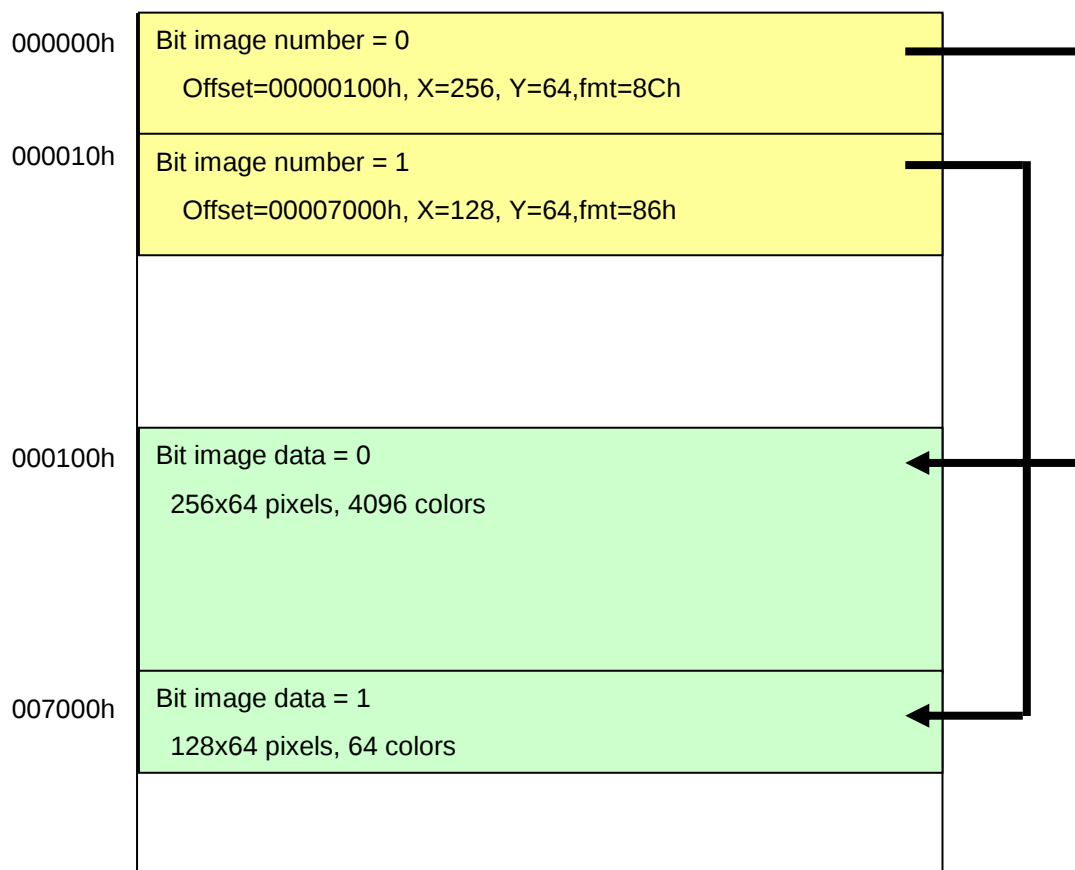
If the bit image extends beyond the currently-selected window, only the portion within the current window is displayed.

When an image in FROM is being written to the display memory, if the read memory area is exceeded, undefined data is displayed.

Note: For BMP file format, the image size specified in the file header is used. 1, 4, 8, 24, and 32 bit BMP files are supported. 16 bit BMP files are not supported.

Index format / 1 image data

Index count	Contents	
xxxxx0h	Bit image data offset (offset)	x 1000000h
xxxxx1h		x 10000h
xxxxx2h		x 100h
xxxxx3h		x 1h
xxxxx4h	Reserved	-
xxxxx5h	Reserved	-
xxxxx6h	Pixel size X (X)	x 100h
xxxxx7h		x 1h
xxxxx8h	Pixel size Y (Y)	x 100h
xxxxx9h		x 1h
xxxxxAh	Format (fmt)	81h, 86h, 8Ch, 98h, F0h
xxxxxBh	Reserved	-
xxxxxC	Reserved	-
xxxxxDh	Reserved	-
xxxxxEh	Reserved	-
xxxxxFh	Reserved	-

ExampleRAM bit image/ bit image FROM1/ bit image FROM2 Bit Image Memory

3.7.4.44 US r n (Reverse Display)**Code:** 1Fh 72h n

n: Reverse display ON/OFF

Definable area: 00h ≤ n ≤ 01h

n = 00h: Reverse OFF

n = 01h: Reverse ON

Default: n = 00h or Memory SW setting.**Function:** Reverse display ON/OFF for character and image display.

Changing this setting only affects subsequent data.

Content already displayed is not affected.

For text and monochrome bit images, background and foreground are interchanged. For color images (including 2-color monochrome BMP file format), the displayed color is inverted. Refer to 3.9 Color Combination Format for details.

3.7.4.45 US w n (Write Mixture Display Mode)**Code:** 1Fh 77h n

n: Display write mode

Definable area: n = 10h, 11h

n = 10h: Normal display write

(Background pixels are written to display memory)

n = 11h: Thru write

(Background pixels are not written to display memory)

Default: n = 10h or Memory SW setting.**Function:** Specifies write mixture mode.

For newly-written characters and images, background pixels can either overwrite the existing display memory content (normal display write), or not overwrite (thru write), which leaves the currently-displayed pixel unchanged. The effect of “thru write” is to make the background “transparent”. For color images, black pixels (if any) are treated as background. Refer to 3.9 Color Combination Format for details.

3.7.4.46 US (w 01h a (Current-window Select)**Code:** 1Fh 28h 77h 01h a

a: Window number

Definable area: 00h ≤ a ≤ 04h

a = 00h: Base-window

a = 01h – 04h: User-window

Function: Select current window

Command is ignored if window number is for a user-window that is not defined.

3.7.4.47 US (w 02h a b[xPL xPH yPL yPH xSL xSH ySL ySH] (User Window Define/ Cancel)**Code:** 1Fh 28h 77h 02h a b [xPL xPH yPL yPH xSL xSH ySL ySH]

a: Definable window number (No.1 – 4)
 b: Define or cancel
 xPL: Left position of window x, lower byte (by 1 pixel)
 xPH: Left position of window x, upper byte (by 1 pixel)
 yPL: Top position of window y, lower byte (by 1 pixel)
 yPH: Top position of window y, upper byte (by 1 pixel)
 xSL: X size of window, lower byte (by 1 pixel)
 xSH: X size of window, upper byte (by 1 pixel)
 ySL: Y size of window, lower byte (by 1 pixel)
 ySH: Y size of window, upper byte (by 1 pixel)

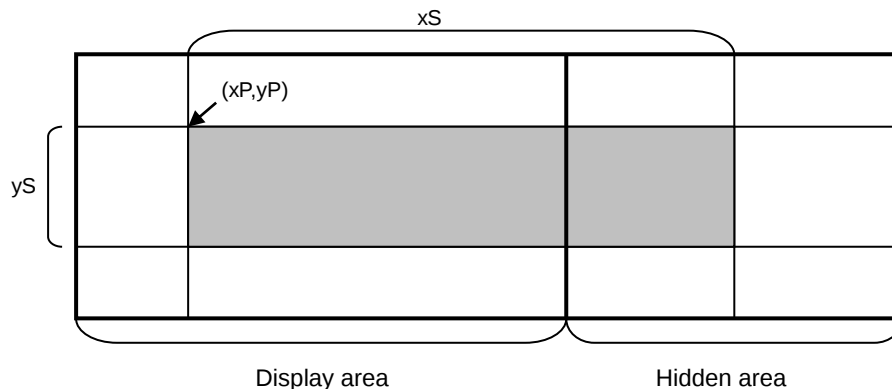
Definable area: $01h \leq a \leq 04h$
 $00h \leq b \leq 01h$
 b = 00h: Cancel
 b = 01h: Define
 $0000h \leq (xPL + xPH \times 100h) \leq \text{Max_Xdot}$
 $0000h \leq (yPL + yPH \times 100h) \leq \text{Max_Ydot}$
 $0001h \leq (xSL + xSH \times 100h) \leq (\text{Xdots} - (xPL + xPH \times 100h))$
 $0001h \leq (ySL + ySH \times 100h) \leq (\text{Ydots} - (yPL + yPH \times 100h))$

Function: Define or cancel user-window

Display contents are not changed by this command.

User-window define (b=01h)

Specify user-window number, window position, and window size.
 Window position and window size are specified in units of 1 pixel.



Up to 4 user-windows can be defined.

The cursor position for the window is initialized to top left (X=0, Y=0).

User-window cancel (b=00h)

For user-window cancel, window range parameters [xPL – ySH] are not used.

If the currently-selected window is cancelled, the base-window becomes the currently-selected window.

If any of 'a', 'b', 'xP', 'yP', 'xS', or 'yS' are outside the definable area, the command is cancelled at that point and the following data is treated as standard data.

3.7.4.48 ESC % n (Download Character ON/ OFF)**Code:** 1Bh 25h n

Definable area: n = 00h, 01h
 n = 01h: Enable
 (If download character is not defined, built-in character is displayed)
 n = 00h: Disable

Function: Enable or disable display of download characters
 (6×8, 8×16, 12×24, and 16×32 pixel).

Characters already displayed are not affected.

3.7.4.49 ESC & a c1 c2 [x1 d1...d(y×x1)] ... [xk d1...d(y×xk)] (Download Character Definition)**Code:** 1Bh 26h a c1 c2 [x1 d1...d(y×x1)] ... [xk d1...d(y×xk)]

a: Select character type
 c1: Start character code
 c2: End character code
 x: Number of pixels for X-direction
 d: Defined data (Refer to 3.10 Download Character Format)

Definable area: 01h ≤ a ≤ 04h
 a = 01h: Select 6×8 pixel character
 a = 02h: Select 8×16 pixel character
 a = 03h: Select 12×24 pixel character
 a = 04h: Select 16×32 pixel character

When 6×8 pixel character is selected;

00h ≤ x ≤ 06h

When 8×16 pixel character is selected;

00h ≤ x ≤ 08h

When 12×24 pixel character is selected;

00h ≤ x ≤ 0Ch

When 16×32 pixel character is selected;

00h ≤ x ≤ 10h

20h ≤ c1 ≤ c2 ≤ FFh

00h ≤ d ≤ FFh

k = c2 – c1 + 1

Function: Define download characters (1-byte characters) into RAM.

For each font size, a maximum of 16 download characters can be defined.

After the maximum number of download characters are defined, in order to define other character codes, space must first be obtained using the download character delete command.

Downloaded characters are valid until redefined, an initialize (ESC@) sequence is executed, or the power is turned off.

To display download character, the commands download character definition and download character ON/ OFF (set to ON) are required.

If 'x' is smaller than the character width, the remaining space on the right is filled with blank (non-display) pixels.

If a currently-displayed download character is re-defined, there is no affect on the currently-displayed character. It is effective only for newly input characters.

Download characters can be saved into FROM using the Save download character command.

3.7.4.50 ESC ? a c (Downloaded Character Delete)**Code:** 1Bh 3Fh a c

a: Select character type
 c: Delete Character code

Definable area: 01h ≤ a ≤ 04h
 a = 01h: Select 6×8 pixel character
 a = 02h: Select 8×16 pixel character
 a = 03h: Select 12×24 pixel character
 a = 04h: Select 16×32 pixel character

20h ≤ c ≤ FFh

Function: Delete defined download character (1-byte character).

Built-in character is displayed after download character is deleted.

Characters already displayed are not affected.

Command is ignored if download character is not defined for the given character code.

3.7.4.51 US (g 10h c1 c2 d1...dk (16×16 Download Character Definition)**Code:** 1Fh 28h 67h 10h c1 c2 d1...d32

c1: Character code, upper byte

c2: Character code, lower byte

d: Definition data (refer to 3.10 Download Character Format)

Definable area: c1, c2: Depends on currently-selected language:

Language	Encoding	c1	c2
Japanese	JIS X0208 (SHIFT-JIS)	c1 = ECh	40h ≤ c2 ≤ 4Fh
Korean	KSC5601-87	c1 = FEh	A1h ≤ c2 ≤ B0h
Simplified Chinese	GB2312-80	c1 = FEh	A1h ≤ c2 ≤ B0h
Traditional Chinese	Big-5	c1 = FEh	A1h ≤ c2 ≤ B0h

00h ≤ d ≤ FFh

Function: Defines a 16×16 pixel downloaded character (2-byte character) in character code specified by c1 and c2.

A maximum 16 download characters can be defined.

Definition data 'd' is processed as character pattern data in column format, and is stored sequentially from the left.

Download character is temporary stored in RAM, but can be stored in FROM using download character save command.

3.7.4.52 US (g 11h c1 c2 (16×16 Downloaded Character Delete)**Code:** 1Fh 28h 67h 11h c1 c2

c1: Character code, upper byte

c2: Character code, lower byte

Definable area: c1, c2: Depends on currently-selected language:

Language	Encoding	c1	c2
Japanese	JIS X0208 (SHIFT-JIS)	c1 = ECh	40h ≤ c2 ≤ 4Fh
Korean	KSC5601-87	c1 = FEh	A1h ≤ c2 ≤ B0h
Simplified Chinese	GB2312-80	c1 = FEh	A1h ≤ c2 ≤ B0h
Traditional Chinese	Big-5	c1 = FEh	A1h ≤ c2 ≤ B0h

Function: Delete defined 16×16 download character in code specified by c1 and c2.**3.7.4.53 US (g 14h c1 c2 d1...d128 (32×32 Download Character Definition)****Code:** 1Fh 28h 67h 14h c1 c2 d1...d128

c1: Character code, upper byte

c2: Character code, lower byte

d: Definition data (refer to 3.10 Download Character Format)

Definable area: c1, c2: Depends on currently-selected language:

Language	Encoding	c1	c2
Japanese	JIS X0208 (SHIFT-JIS)	c1 = ECh	40h ≤ c2 ≤ 4Fh

00h ≤ d ≤ FFh

Function: Defines a 32×32 pixel downloaded character (2-byte character) in character code specified by c1 and c2.

A maximum 16 download characters can be defined.

Definition data 'd' is processed as character pattern data in column format, and is stored sequentially from the left.

Download character is temporary stored in RAM, but can be stored in FROM using Download character save command.

This command is invalid if language selection is not set to Japanese.

3.7.4.54 US (g 15h c1 c2 (32×32 Downloaded Character Delete)**Code: 1Fh 28h 67h 15h c1 c2**

c1: Character code, upper byte

c2: Character code, lower byte

Definable area: c1, c2: Depends on currently-selected language:

Language	Encoding	c1	c2
Japanese	JIS X0208 (SHIFT-JIS)	c1 = ECh	40h ≤ c2 ≤ 4Fh

Function: Delete defined 32×32 download character in code specified by c1 and c2.

This command is invalid if language selection is not set to Japanese.

3.7.4.55 US (e 11h a (Download Character Save)**Code: 1Fh 28h 65h 11h a**

a: Font size

Definable area: 01h ≤ a ≤ 06h

a = 01h: 6×8 pixel

a = 02h: 8×16 pixel

a = 03h: 16×16 pixel

a = 04h: 16×32 pixel

a = 05h: 32×32 pixel

a = 06h: 12×24 pixel

Function: Save the download characters defined on RAM to FROM (RAM→FROM).

The saved content is re-enabled using the download character restore command.

This command is only valid in user setup mode.

Data is not committed to FROM until user setup mode end.

3.7.4.56 US (e 21h a (Download Character Restore)**Code: 1Fh 28h 65h 21h a**

a: Font size

Definable area: 01h ≤ a ≤ 06h

a = 01h: 6×8 pixel

a = 02h: 8×16 pixel

a = 03h: 16×16 pixel

a = 04h: 16×32 pixel

a = 05h: 32×32 pixel

a = 06h: 12×24 pixel

Function: Transfer the download characters saved in FROM to RAM.

Command is ignored if specified font size download characters are not registered in FROM.

Command is valid in both user setup mode and normal mode.

3.7.4.57 US (e 13h m P(80h-1) P(80h-2) ... P(FFh-n) (FROM User Font Definition)**Code: 1Fh 28h 65h 13h m P(80h-1) P(80h-2) ... P(FFh-n)**

m: User table

p: Definition data (refer to 3.9 Download Character Format)

Definable area: 01h ≤ m ≤ 04h

m = 01h: 6×8 pixel user table

m = 02h: 8×16 pixel user table

m = 03h: 12×24 pixel user table

m = 04h: 16×32 pixel user table

m=01h: P(80h-1) ... P(80h-6) ... P(FFh-6) 6 Bytes / font × 128 characters (768 bytes)

m=02h: P(80h-1) ... P(80h-16) ... P(FFh-16) 16 Bytes / font × 128 characters (2,048 bytes)

m=03h: P(80h-1) ... P(80h-36) ... P(FFh-36) 36 Bytes / font × 128 characters (4,608 bytes)

m=04h: P(80h-1) ... P(80h-64) ... P(FFh-64) 64 Bytes / font × 128 characters (8,192 bytes)

Function: Define the user font for each size of 1-byte code to the user table.

This command defines all 128 characters at once; it is not possible to only define a part of the character code space.

User font tables for each font size are set to blank (00h) when shipped.

This command is only valid in user setup mode.

Data is not committed to FROM until user setup mode end.

FROM user table display is selected using “character table type” command (refer to 3.7.4.6).

3.7.4.58 US (e 15h a b p(1) ... p(65536) (FROM Extension Font Definition)**Code:** 1Fh 28h 65h 15h a b p(1) ... p(65536)

a: Bank
 b: Define / Delete
 p: Definition data (if Define)

Definable area: a = 01h
 b = 00h, 01h
 b = 00h: FROM extension font is deleted.
 Definition data parameter is not used.
 b = 01h: FROM extension font is defined.
 Consult manufacturer for definition data format.
 FROM extension font is in deleted state when shipped.
 00h ≤ p ≤ FFh

Function: Define or delete FROM extension font.

FROM extension font is in deleted state when shipped.

This command is only valid in user setup mode.

Data is not committed to FROM until user setup mode end.

3.7.4.59 US (e 01h d1 d2 (User Setup Mode Start)**Code:** 1Fh 28h 65h 01h 49h 4Eh

Definable area: d1 = 49h (Character 'I')
 d2 = 4Eh (Character 'N')

Function: Start user setup mode.

The following data is transmitted from the currently-active interface:

Transmitted data	Hex	Data length
(1) Header	28h	1 byte
(2) Identifier 1	65h	1 byte
(3) Identifier 2	01h	1 byte
(4) NUL	00h	1 byte

This command is only valid in normal mode.

Display screen is blanked.

3.7.4.60 US (e 02h d1 d2 d3 (User Setup Mode End)**Code:** 1Fh 28h 65h 02h 4Fh 55h 54h

Definable area: d1 = 4Fh (Character 'O')
 d2 = 55h (Character 'U')
 d3 = 54h (Character 'T')

Function: End user setup mode, and software reset of display as follows:

- (1) Wait for any in-progress operations (memory control, information transmission, etc.) to complete.
- (2) Output display BUSY signal.
- (3) Software reset.

This command is only valid in user setup mode.

This command clears the receive buffer, and all settings (download character, macro settings, GPIO settings, etc.) are reset to power-on state.

The Initialize Display command (3.7.4.2) is executed. In addition, all memory switches and jumper settings are re-read, and the general-purpose ports are re-initialized.

3.7.4.61 US (p 01h n a (I/O Port Input/ Output Setting)**Code:** 1Fh 28h 70h 01h n a

n: I/O port number

a: Set input / output (bit-wise)

Definable area: $00h \leq n \leq 03h$

n = 00h: Port 0 (GPIO 0 – 7)

n = 01h: Port 1 (GPIO 8 – 15)

n = 02h: Port 2 (GPIO 16 – 23)

n = 03h: Port 3 (GPIO 24, 25)

 $00h \leq a \leq FFh$

Bit value = 0: Input

Bit value = 1: Output

Default: a = 00h (all ports set to input)**Function:** Set input or output for general-purpose I/O ports.

Port input/ output is set by value of 'a'.

Bit assignment is as follows:

I/O port number	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
n = 00h	GPIO 7	GPIO 6	GPIO 5	GPIO 4	GPIO 3	GPIO 2	GPIO 1	GPIO 0
n = 01h	GPIO 15	GPIO 14	GPIO 13	GPIO 12	GPIO 11	GPIO 10	GPIO 9	GPIO 8
n = 02h	GPIO 23	GPIO 22	GPIO 21	GPIO 20	GPIO 19	GPIO 18	GPIO 17	GPIO 16
n = 03h	-	-	-	-	-	-	GPIO 25	GPIO 24

Caution: I/O port is intended for simple peripheral switches and for controlling lights, etc., and should not be used for applications where high reliability is required.

3.7.4.62 US (p 10h n a (I/O Port Output)**Code:** 1Fh 28h 70h 10h n a

n: I/O port number

a: Output data value

Definable area: $00h \leq n \leq 03h$

n = 00h: Port 0 (GPIO 0 – 7)

n = 01h: Port 1 (GPIO 8 – 15)

n = 02h: Port 2 (GPIO 16 – 23)

n = 03h: Port 3 (GPIO 24, 25)

 $00h \leq a \leq FFh$ **Function:** Output data to general-purpose I/O port.

Output data is set by value of 'a'.

Bit assignment is as follows:

I/O port number	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
n = 00h	GPIO 7	GPIO 6	GPIO 5	GPIO 4	GPIO 3	GPIO 2	GPIO 1	GPIO 0
n = 01h	GPIO 15	GPIO 14	GPIO 13	GPIO 12	GPIO 11	GPIO 10	GPIO 9	GPIO 8
n = 02h	GPIO 23	GPIO 22	GPIO 21	GPIO 20	GPIO 19	GPIO 18	GPIO 17	GPIO 16
n = 03h	-	-	-	-	-	-	GPIO 25	GPIO 24

3.7.4.63 US (p 20h n (I/O Port Input)**Code:** 1Fh 28h 70h 20h n

n: I/O port number

Definable area: $00h \leq n \leq 01h$

n = 00h: Port 0 (GPIO 0 – 7)

n = 01h: Port 1 (GPIO 8 – 15)

n = 02h: Port 2 (GPIO 16 – 23)

n = 03h: Port 3 (GPIO 24 – 25)

Function: The state of a general-purpose I/O port at the time this command is processed is transmitted.

The following data is transmitted from the currently-active interface:

Transmitted data	Hex	Data length
(1) Header	28h	1 byte
(2) Identifier (1)	70h	1 byte
(3) Identifier (2)	20h	1 byte
(4) Data	00h–FFh	1 byte

Response time varies depending on the state of the receive buffer.

3.7.4.64 US : pL pH [d1 ... dk] (RAM Macro Define/ Delete)**Code: 1Fh 3Ah pL pH [d1 ... dk]**

pL: RAM macro data length, lower byte

pH: RAM macro data length, upper byte

d: RAM macro data

Definable area: $0000h \leq (pL + pH \times 100h) \leq 0400h$
 $(pL + pH \times 100h) > 0000h$: Supplied data 'd' is stored as macro.
 $(pL + pH \times 100h) = 0000h$: Macro is deleted.

Function: Define or delete RAM macro or RAM program macro.

If macro data length is outside the definable area, the command is cancelled, and the following data is treated as standard data.

Do not define any of the following commands in a macro:

Initialize, macro execution, RAM macro processing definition, user setup mode start, [US (e) group commands (Bit Image FROM1 Definition, Download Character Save, etc.), macro execution settings memory re-writing mode, General-purpose memory commands.

Program macro details: refer to specification DS-1940-0005-xx Program Macro.

3.7.4.65 US (e 12h a pL pH t1 t2 [d(1) ... d(p)] (FROM Macro Define/ Delete)**Code: 1Fh 28h 65h 12h a pL pH t1 t2 [d1 ... d(p)]**

a: FROM macro registration number

pL: FROM macro data length, lower byte

pH: FROM macro data length, upper byte

t1: Display time interval ($t1 \times \text{IntTime}$)t2: Idle time for macro repetition ($t2 \times \text{IntTime}$)

d: FROM macro data

Definable area: < Firmware version: F220, or later >
 $01h \leq a \leq 08h$: FROM macro number 1 – 8
 $0000h \leq (pL + pH \times 100h) \leq FFFFh$
 $(pL + pH \times 100h) = 0$: macro is deleted.
a = 01h:
 $0000h \leq (pL + pH \times 100h) \leq FFFFh$
a > 01h:
 $0000h \leq (pL + pH \times 100h) \leq E000h$
Note: $(a - 1) \times 2000h + (pL + pH \times 100h) \leq 10000h$
 $00h \leq t1 \leq FFh$
 $00h \leq t2 \leq FFh$
 $00h \leq d \leq FFh$
< Firmware version: less than F220 >
 $01h \leq a \leq 04h$: FROM macro number 1 – 4
 $0000h \leq (pL + pH \times 100h) \leq 2000h$ (if using 4 macros), 8000h (if using 1 macro)
 $(pL + pH \times 100h) > 0$: Supplied data 'd' is stored as macro.
 $(pL + pH \times 100h) = 0$: macro is deleted.
 $00h \leq t1 \leq FFh$
 $00h \leq t2 \leq FFh$
 $00h \leq d \leq FFh$

Function: Define or delete FROM macro or FROM program macro.

FROM macro storage capacity

< Firmware version: F220, or later >

FROM macro storage capacity is a total of 64KB, 8KB / macro when using 8 macros.

< Firmware version: less than F220 >

FROM macro storage capacity is a total of 32KB, 8KB / macro when using 4 macros.

For macros exceeding 8KB, multiple macro definition areas are used, which may result in some macro number areas being undefined.

If macro data length is outside the definable area, the command is cancelled, and the following data is treated as standard data.

Display time interval (t1) and idle time (t2) settings are used when FROM macro execution at power on is used.

Display time interval refers to the interval time between displaying characters, and does not affect the processing speed of command code.

Idle time refers to the time period from processing the last macro data until the macro is re-executed.

This command is only valid in user setup mode.

Do not define any of the following commands in a macro:

Initialize, macro execution, RAM macro define / delete, user setup mode start, [US (e] group commands (Bit Image FROM1 definition, Download Character Save, etc.), macro execution settings, memory re-writing mode, general-purpose memory commands.

Data is not committed to FROM until user setup mode end.

Program macro details: refer to specification DS-1940-0005-xx Program Macro.

Example:

FROM macro 1 area	Undefined
FROM macro 2 area	Undefined
FROM macro 3 area	Undefined
FROM macro 4 area	Undefined

↓Define 8KB macros in macro 1 – 4

FROM macro 1 area	FROM macro 1: 8KB
FROM macro 2 area	FROM macro 2: 8KB
FROM macro 3 area	FROM macro 3: 8KB
FROM macro 4 area	FROM macro 4: 8KB

↓Define 16KB macro in macro 2

FROM macro 1 area	FROM macro 1: 8KB
FROM macro 2 area	FROM macro 2: 16KB
FROM macro 3 area	
FROM macro 4 area	FROM macro 4: 8KB

↓Define 16KB macro in macro 3

FROM macro 1 area	FROM macro 1: 8KB
FROM macro 2 area	Undefined
FROM macro 3 area	FROM macro 3: 16KB
FROM macro 4 area	

↓Define 24KB macro in macro 1

FROM macro 1 area	FROM macro 1: 24KB
FROM macro 2 area	
FROM macro 3 area	
FROM macro 4 area	Undefined

3.7.4.66 US ^ a t1 t2 (Macro Execution)**Code:** 1Fh 5Eh a t1 t2

a: Macro processing definition number
 t1: Display time interval (t1 × *IntTime*)
 t2: Idle time for macro repetition (t2 × *IntTime*)

Definable area: < Firmware version: F220, or later >
 00h ≤ a ≤ 08h, 80h ≤ a ≤ 88h
 a = 00h: RAM macro 0
 01h ≤ a ≤ 08h: FROM macro 1 – 8
 a = 80h: RAM program macro 0
 81h ≤ a ≤ 88h: FROM program macro 1 – 8
 00h ≤ t1 ≤ FFh
 00h ≤ t2 ≤ FFh
 < Firmware version: less than F220 >
 00h ≤ a ≤ 04h, 80h ≤ a ≤ 84h
 a = 00h: RAM macro 0
 01h ≤ a ≤ 04h: FROM macro 1 – 4
 a = 80h: RAM program macro 0
 81h ≤ a ≤ 84h: FROM program macro 1 – 4
 00h ≤ t1 ≤ FFh
 00h ≤ t2 ≤ FFh

Function: Continuously execute contents of defined macro 'a'.

Display time interval refers to the interval time between displaying characters, and does not affect the processing speed of command code.

Idle time refers to the time period from processing the last macro data until the macro is re-executed.

If macro 'a' is not defined, or is outside the definable area, the entire command (up to t2) is ignored.

Macro execution is stopped when a command is input.

The current window (write screen mode area if base-window) is cleared and cursor moves to home position.

Display settings remain in the current state when the macro ended.

3.7.4.67 US (i 20h a b c (Macro End Condition)**Code:** 1Fh 28h 69h 20h a b c

a: Macro end code enable/ disable
 b: Macro end code
 c: Macro end clear screen setting

Definable area: a = 00h, 01h
 a = 00h: Macro end code disabled
 a = 01h: Macro end code enabled
 00h ≤ b ≤ FFh
 c = 00h, 01h
 c = 00h: Clear screen at macro end
 c = 01h: Do not clear screen at macro end

Default: a = 00h or Memory SW setting.
 b = 00h or Memory SW setting.
 c = 00h or Memory SW setting.

Function: Macro end condition set.

a = 00h: Macro will unconditionally end if data is received.

a = 01h: Macro will end if data byte 'b' is received. All other values are ignored.

c = 00h: Clear screen on macro end.

c = 01h: Do not clear screen on macro end.

The received byte code that ends the macro is processed as the first byte of the next command.

This setting is not applicable for program macro.

3.7.4.68 US (e 03h a b (Memory SW Setting)

Code: 1Fh 28h 65h 03h a b
 or 1Fh 28h 65h 03h a b c[1] d[1] [... c[b] d[b]]

a: Memory SW number

b: Setting data

Definable area:**Single Memory SW setting:**

$00h \leq a \leq 3Fh$

$00h \leq b \leq FFh$

Multiple Memory SW setting:

$a = FFh$

$01h \leq b \leq FFh$

$00h \leq c \leq 3Fh$

$00h \leq d \leq FFh$

Function: Set Memory SW.

A single Memory Switch can be set (a=00h–3Fh) or multiple Memory Switches can be set (a=FFh).

Single setting (a=00h–3Fh): a = Memory SW number, b = setting value.

Multiple setting (a=FFh): b = number of settings, c = Memory SW number, d = setting value.

This command is only valid in user setup mode.

Data is not committed to FROM until user setup mode end.

Memory SW details: refer to section 4.2 Memory SW.

3.7.4.69 US (e 04h a (Memory SW Data Send)

Code: 1Fh 28h 65h 04h a
 or 1Fh 28h 65h 04h a b c[1] [... c[b]]

a: Memory SW number

Definable area:

Single Memory SW read:

$00h \leq a \leq 3Fh$

Multiple Memory SW read:

$a = FFh$

$01h \leq b \leq FFh$

$00h \leq c \leq 3Fh$

Function: Send the contents of Memory SW data.

A single Memory Switch can be read (a=00h–3Fh) or multiple Memory Switches can be read (a=FFh).

Single read (a=00h–3Fh): a = Memory SW number.

Multiple read (a=FFh): b = number of reads, c = Memory SW number.

The following data is transmitted from the currently-active interface:

Transmitted data	Hex	Data length
(1) Header	28h	1 byte
(2) Identifier 1	65h	1 byte
(3) Identifier 2	04h	1 byte
(4) Data	00h–FFh	1 byte / b byte(s)

This command is valid in both user setup mode and normal mode.

Memory SW details: refer to section 4.2 Memory SW.

3.7.4.70 US (e 18h sL sH sE m1 a1L a1H a1E d[1] ... d[s] (General-purpose Memory Store/ Bit Image FROM2 Store)

Code: 1Fh 28h 65h 18h sL sH sE m1 a1L a1H a1E d[1] ... d[s]

sL: Data size, lower byte
 sH: Data size, upper byte
 sE: Data size, extension byte
 m1: Memory select
 a1L: Memory address, lower byte
 a1H: Memory address, upper byte
 a1E: Memory address, extension byte
 d: Data to store

Definable area: $10h \leq m1 \leq 1Fh$, $m1 = 30h, 31h$

$m1 = 10h$: General-purpose FROM2 base address 0000.0000h
 $m1 = 11h$: General-purpose FROM2 base address 0100.0000h
 $m1 = 12h$: General-purpose FROM2 base address 0200.0000h
 $m1 = 13h$: General-purpose FROM2 base address 0300.0000h
 $m1 = 14h$: General-purpose FROM2 base address 0400.0000h
 $m1 = 15h$: General-purpose FROM2 base address 0500.0000h
 $m1 = 16h$: General-purpose FROM2 base address 0600.0000h
 $m1 = 17h$: General-purpose FROM2 base address 0700.0000h
 $m1 = 18h$: General-purpose FROM2 base address 0800.0000h
 $m1 = 19h$: General-purpose FROM2 base address 0900.0000h
 $m1 = 1Ah$: General-purpose FROM2 base address 0A00.0000h
 $m1 = 1Bh$: General-purpose FROM2 base address 0B00.0000h
 $m1 = 1Ch$: General-purpose FROM2 base address 0C00.0000h
 $m1 = 1Dh$: General-purpose FROM2 base address 0D00.0000h
 $m1 = 1Eh$: General-purpose FROM2 base address 0E00.0000h
 $m1 = 1Fh$: General-purpose FROM2 base address 0F00.0000h
 $000001h \leq (sL + sH \times 100h + sE \times 10000h) \leq FFFFFFFh$
 $000000h \leq (a1L + a1H \times 100h + a1E \times 10000h) \leq FFFFFFFh$
 $m1 = 30h$ (General-purpose RAM):
 $000001h \leq (sL + sH \times 100h + sE \times 10000h) \leq 000400h$
 $000000h \leq (a1L + a1H \times 100h + a1E \times 10000h) \leq 0003FFh$
 $m1 = 31h$ (General-purpose FROM):
 $000001h \leq (sL + sH \times 100h + sE \times 10000h) \leq 001000h$
 $000000h \leq (a1L + a1H \times 100h + a1E \times 10000h) \leq 00FFFFh$
 $00h \leq d \leq FFh$

Function: Store the supplied data into general-purpose memory / FROM2.

Note: **General-purpose FROM2 valid area is 0000.0000h – 0FAF.FFFFh (257,024KB).**
General-purpose FROM is in units of 001000h, with a total of 16 areas.

Stored data can be read using general-purpose memory send command or by program macro.

Images stored in bit image FROM2 can be displayed using the commands “Downloaded Bit Image Display” and “Packaged Downloaded Bit Image Display”.

Data is stored into the specified address range, and all other memory locations in the same 128KB general-purpose FROM2 block area are set to FFh.

For General-purpose RAM, data is stored only into the specified address range.

For General-purpose FROM, data is stored into the specified address range, and all other memory locations in the same FROM area are set to FFh. Further, it is not possible to specify data storage that would exceed a general-purpose FROM area.

Storage that would exceed the address range cannot be set.

General-purpose FROM2 area from 0E00.0000h is reserved for the built-in outline fonts. To prevent accidental overwriting, writes to this area are, by default, not processed. In order to write to this area, it is necessary to initially issue this command with address to write set to 0E00.0000h and size set to 00.0000 bytes.

This command is valid in both user setup mode and normal mode.

3.7.4.71 US (e 19h sL sH sE m1 a1L a1H a1E m2 a2L a2H a2E (General-purpose Memory Transfer)**Code: 1Fh 28h 65h 19h sL sH sE m1 a1L a1H a1E m2 a2L a2H a2E**

sL: Transfer data size, lower byte
 sH: Transfer data size, upper byte
 sE: Transfer data size, extension byte
 m1: Destination memory select
 a1L: Destination address, lower byte
 a1H: Destination address, upper byte
 a1E: Destination address, extension byte
 m2: Source memory select
 a2L: Source address, lower byte
 a2H: Source address, upper byte
 a2E: Source address, extension byte

Definable area: $10h \leq m1, m2 \leq 1Fh, m1, m2 = 30h, 31h$

$m1, m2 = 10h$: General-purpose FROM2 base address 0000.0000h
 $m1, m2 = 11h$: General-purpose FROM2 base address 0100.0000h
 $m1, m2 = 12h$: General-purpose FROM2 base address 0200.0000h
 $m1, m2 = 13h$: General-purpose FROM2 base address 0300.0000h
 $m1, m2 = 14h$: General-purpose FROM2 base address 0400.0000h
 $m1, m2 = 15h$: General-purpose FROM2 base address 0500.0000h
 $m1, m2 = 16h$: General-purpose FROM2 base address 0600.0000h
 $m1, m2 = 17h$: General-purpose FROM2 base address 0700.0000h
 $m1, m2 = 18h$: General-purpose FROM2 base address 0800.0000h
 $m1, m2 = 19h$: General-purpose FROM2 base address 0900.0000h
 $m1, m2 = 1Ah$: General-purpose FROM2 base address 0A00.0000h
 $m1, m2 = 1Bh$: General-purpose FROM2 base address 0B00.0000h
 $m1, m2 = 1Ch$: General-purpose FROM2 base address 0C00.0000h
 $m1, m2 = 1Dh$: General-purpose FROM2 base address 0D00.0000h
 $m1, m2 = 1Eh$: General-purpose FROM2 base address 0E00.0000h
 $m1, m2 = 1Fh$: General-purpose FROM2 base address 0F00.0000h
 $000000h \leq (sL + sH \times 100h + sE \times 10000h) \leq FFFFFFFh$
 $000000h \leq (a1L + a1H \times 100h + a1E \times 10000h) \leq FFFFFFFh$
 $000000h \leq (a2L + a2H \times 100h + a2E \times 10000h) \leq FFFFFFFh$
 $m1, m2 = 30h$ (General-purpose RAM):
 $000001h \leq (sL + sH \times 100h + sE \times 10000h) \leq 000400h$
 $000000h \leq (a1L + a1H \times 100h + a1E \times 10000h) \leq 0003FFh$
 $000000h \leq (a2L + a2H \times 100h + a2E \times 10000h) \leq 0003FFh$
 $m1, m2 = 31h$ (General-purpose FROM):
 $000001h \leq (sL + sH \times 100h + sE \times 10000h) \leq 001000h$
 $000000h \leq (a1L + a1H \times 100h + a1E \times 10000h) \leq 00FFFFh$
 $000000h \leq (a2L + a2H \times 100h + a2E \times 10000h) \leq 00FFFFh$
 $00h \leq d \leq FFh$

Function: Transfer data between general-purpose memory/ general-purpose FROM2 areas.

Note: **General-purpose FROM2 valid area is 0000.0000h – 0FAF.FFFFh (257,024KB).**
General-purpose FROM is in units of 001000h, with a total of 16 areas.

For General-purpose RAM, data is transferred only into the specified address range.

For General-purpose FROM, data is transferred into the specified address range, and all other memory locations in the same FROM area are set to FFh. Further, it is not possible to specify a data transfer that would exceed a general-purpose FROM area.

For general-purpose FROM2, data is transferred into the specified address range, and all other memory locations in the same 128KB block area are set to FFh.

Storage that would exceed the address range cannot be set.

General-purpose FROM2 area from 0E00.0000h is reserved for the built-in outline fonts. To prevent accidental overwriting, writes to this area are, by default, not processed. In order to write to this area, please refer to 3.7.4.70 General-purpose Memory Store/ Bit Image FROM2 Store.

This command is valid in both user setup mode and normal mode.

If size is 000000h, no data is transferred.

3.7.4.72 US (e 28h sL sH sE m1 a1L a1H a1E (General-purpose Memory Send)**Code: 1Fh 28h 65h 28h sL sH sE m1 a1L a1H a1E**

n: Classify command
 sL: Data size, lower byte
 sH: Data size, upper byte
 sE: Data size, extension byte
 m1: Memory select
 a1L: Memory address, lower byte
 a1H: Memory address, upper byte
 a1E: Memory address, extension byte

Definable area: $10h \leq m1 \leq 1Fh$, $m1 = 30h, 31h$

$m1 = 10h$: General-purpose FROM2 base address 0000.0000h
 $m1 = 11h$: General-purpose FROM2 base address 0100.0000h
 $m1 = 12h$: General-purpose FROM2 base address 0200.0000h
 $m1 = 13h$: General-purpose FROM2 base address 0300.0000h
 $m1 = 14h$: General-purpose FROM2 base address 0400.0000h
 $m1 = 15h$: General-purpose FROM2 base address 0500.0000h
 $m1 = 16h$: General-purpose FROM2 base address 0600.0000h
 $m1 = 17h$: General-purpose FROM2 base address 0700.0000h
 $m1 = 18h$: General-purpose FROM2 base address 0800.0000h
 $m1 = 19h$: General-purpose FROM2 base address 0900.0000h
 $m1 = 1Ah$: General-purpose FROM2 base address 0A00.0000h
 $m1 = 1Bh$: General-purpose FROM2 base address 0B00.0000h
 $m1 = 1Ch$: General-purpose FROM2 base address 0C00.0000h
 $m1 = 1Dh$: General-purpose FROM2 base address 0D00.0000h
 $m1 = 1Eh$: General-purpose FROM2 base address 0E00.0000h
 $m1 = 1Fh$: General-purpose FROM2 base address 0F00.0000h
 $000000h \leq (a1L + a1H \times 100h + a1E \times 10000h) \leq FFFFFFFh$
 $000000h \leq (sL + sH \times 100h + sE \times 10000h) \leq FFFFFFFh$
 $m1 = 30h$ (General-purpose RAM):
 $000001h \leq (sL + sH \times 100h + sE \times 10000h) \leq 000400h$
 $000000h \leq (a1L + a1H \times 100h + a1E \times 10000h) \leq 0003FFh$
 $m1 = 31h$ (General-purpose FROM):
 $000001h \leq (sL + sH \times 100h + sE \times 10000h) \leq 001000h$
 $000000h \leq (a1L + a1H \times 100h + a1E \times 10000h) \leq 00FFFFFFh$

Function: Send data stored in general-purpose memory.

Note: **General-purpose FROM2 valid area is 0000.0000h – 0FAF.FFFFh (257,024KB).**
General-purpose FROM is in units of 001000h, with a total of 16 areas.

Data read that would exceed the address range cannot be set.
 This command is valid in both user setup mode and normal mode.

The following data is transmitted from the currently-active interface:

Transmitted data	Hex	Data length
(1) Header	28h	1 byte
(2) Identifier 1	65h	1 byte
(3) Identifier 2	28h	1 byte
(4) Data	00h–FFh	s byte(s)

3.7.4.73 US (e 40h a [b c] (Display Status Send)**Code:** 1Fh 28h 65h 40h a [b c]**Code:** 1Fh 28h 65h 40h a st ad3 ad2 ad1 ad0 sz3 sz2 sz1 sz0**Definable area:** a = 01h, 02h, 10h, 11h, 20h, 30h, 40h, 41h

a = 01h: Boot version information (b, c not used)

a = 02h: Firmware version information (b, c not used)

a = 10h: 2-byte character code information (b, c not used)

a = 11h: Language type information (b, c not used)

a = 20h: Memory checksum information

00h ≤ b ≤ FFh: Start address (effective address = b×10000h)

01h ≤ c ≤ FFh: Data length (effective data length = c×10000h)

a = 21h: General-purpose FROM2 memory checksum information

st: SUM type (00h fixed)

ad: General-purpose FROM2 start address (32bit)

00000000h ≤ (ad3*1000000h + ad2*10000h + ad1*100h + ad0) ≤ FFFFFFFFh

sz: size (32bit)

00000000h ≤ (sz3*1000000h + sz2*10000h + sz1*100h + sz0) ≤ FFFFFFFFh

a = 30h: Product type information (b, c not used)

a = 40h: Display x pixel information (b, c not used)

a = 41h: Display y pixel information (b, c not used)

a = 70h: Touch setting package name (b, c not used)

a = 71h: Touch setting package ID (b, c not used)

a = 72h: Touch sensitivity (current gain) setting value (b, c not used)

a = 73h: Touch sensitivity (current threshold) setting value (b, c not used)

Function: Send display status information.

The following data is transmitted from the currently-active interface:

Transmitted data	Hex	Data length	Information data example
(1) Header	28h	1 byte	"("
(2) Identifier 1	65h	1 byte	"e"
(3) Identifier 2	40h	1 byte	"@"
(4) Data	00h–FFh	a = 01h: 4 bytes a = 02h: 4 bytes a = 10h: 15 bytes a = 11h: 15 bytes a = 20h: 4 bytes a = 21h: 4 bytes a = 30h: 15 bytes a = 40h: 3 bytes a = 41h: 3 bytes a = 70h: 15 bytes a = 71h: 4 bytes a = 72h: 1 bytes a = 73h: 1 bytes	"0.00" "1.62" "JIS X208 L 100" "MULTI LANGUAGE" 00h 6Dh A6h 97h 00h 00h 10h EFh "GT-C900PA " "800" "480" "Noritake " "0003" 06h 50h

This command is valid in both user setup mode and normal mode.

The relationship between the firmware version information readout value and the actual firmware version is shown as following. Reading value "1.25" = "F125"

For the firmware version, refer to 4 Lot number / firmware version specification.

3.7.4.74 FS | M m d1 ... d6 (Memory Re-writing Mode Start)**Code:** 1Ch 7Ch 4Dh m d1 ... d6**Definable area:** m = D0h

d1 ... d6 = "MODEIN"

Function: Shift to "memory re-writing mode" from "normal mode".

Memory re-writing mode is used for changing the firmware and fonts, etc. in FROM that cannot be changed in user setup mode.

Changing this FROM requires special commands and tools.

Do not use this command.

3.7.4.75 US (a 48h m w (Power Saving Mode)**Code:** 1Fh 28h 61h 48h m w

m: Mode

w: Wakeup method

Definable area:

m = 01h: Sleep mode

w = b7(upper bit), b6, b5, b4, b3, b2, b1, b0(lower bit)

bit	0	1
b7	Reserved	
b6	Refer to the following list	
b5		
b4		
b3	Reserved	
b2	Reserved	
b1	Reserved	
b0	No wakeup on touch	Wakeup on touch

* If w = 00h (no wakeup methods are set), return to normal operation is only possible by external reset or VCC power cycling.

b4	b5	b6	GPIO15 wakeup condition
0	-	-	No wakeup on GPIO15 input
1	0	0	Wakeup on GPIO15 LOW level
1	0	1	Wakeup on GPIO15 falling edge
1	1	0	Wakeup on GPIO15 HIGH level
1	1	1	Wakeup on GPIO15 rising edge

Function: Transition to power saving mode and set the wakeup method

The communication interfaces (USB/ UART/ I²C/ SPI) do not function during power saving mode. Current consumption can be reduced if wakeup by touch is not required (b0 = 0) (GPIO15 only). The current value of volatile settings (e.g., Brightness Level Setting, GPIO input / output setting, etc.) are retained in sleep mode. For any GPIO pins set to output, the output level is maintained during sleep mode.

3.8 Bit Image Data Format

3.8.1 Normal Bit Image Data Format

Monochrome (1-bit) format (1 byte per 8 pixels or part thereof for each horizontal line):

D7	D6	D5	D4	D3	D2	D1	D0
F.C.	F.C.	F.C.	F.C.	F.C.	F.C.	F.C.	F.C.

F.C.: Fore Color (color of character)

F.C. = 1: Fore Color (color of character)

F.C. = 0: Background color

*Note: D7 represents leftmost pixel.

If horizontal width is not a multiple of 8, the remaining lower-order bits are ignored.

Color 6-bit format (1 byte per pixel):

D7	D6	D5	D4	D3	D2	D1	D0
X	X	B bit1	B bit0	G bit1	G bit0	R bit1	R bit0

Color 12-bit format (2 bytes per pixel):

1st byte								2nd byte							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
X	X	X	X	B bit3	B bit2	B bit1	B bit0	G bit3	G bit2	G bit1	G bit0	R bit3	R bit2	R bit1	R bit0

Color 16-bit format (2 bytes per pixel):

1st byte								2nd byte							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
I bit0	B bit5	B bit4	B bit3	B bit2	B bit1	G bit5	G bit4	G bit3	G bit2	G bit1	R bit5	R bit4	R bit3	R bit2	R bit1

I: Intensity bit. Used as common least-significant bit for each of R, G, B.

Color 16-bit high-speed format (2 bytes per pixel):

1st byte								2nd byte							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
G bit3	G bit2	G bit1	R bit5	R bit4	R bit3	R bit2	R bit1	I bit0	B bit5	B bit4	B bit3	B bit2	B bit1	G bit5	G bit4

I: Intensity bit. Used as common least-significant bit for each of R, G, B.

Note: 16-bit high-speed format is only supported for downloaded bit image display command.

Color 24-bit format (3 bytes per pixel):

1st byte								2nd byte							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
B bit7	B bit6	B bit5	B bit4	B bit3	B bit2	B bit1	B bit0	G bit7	G bit6	G bit5	G bit4	G bit3	G bit2	G bit1	G bit0

3rd byte							
D7	D6	D5	D4	D3	D2	D1	D0
R bit7	R bit6	R bit5	R bit4	R bit3	R bit2	R bit1	R bit0

3.8.2 Packaged Bit Image Data Format

Color 6-bit format (1 byte per pixel):

D7	D6	D5	D4	D3	D2	D1	D0
LEN bit1	LEN bit0	B bit1	B bit0	G bit1	G bit0	R bit1	R bit0

LEN: Continued pixel length

LEN = 0: until right end

LEN = 1: 1 pixel

LEN = 2: 2 pixels

LEN = 3: 3 pixels

Color 12-bit format (2 bytes per pixel):

1st byte								2nd byte							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
LEN Bit3	LEN bit2	LEN bit1	LEN bit0	B bit3	B bit2	B bit1	B bit0	G bit3	G bit2	G bit1	G bit0	R bit3	R bit2	R bit1	R bit0

LEN: Continued pixel length

LEN = 0: until right end

LEN = 1: 1 pixel

LEN = 2: 2 pixels

LEN = 3: 3 pixels

:

LEN = 15: 15 pixels

Color 24-bit format (4 bytes per pixel):

1st byte								2nd byte							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
LEN bit7	LEN bit6	LEN bit5	LEN bit4	LEN bit3	LEN bit2	LEN bit1	LEN bit0	B bit7	B bit6	B bit5	B bit4	B bit3	B bit2	B bit1	B bit0

3rd byte								4th byte							
D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0
G bit7	G bit6	G bit5	G bit4	G bit3	G bit2	G bit1	G bit0	R bit7	R bit6	R bit5	R bit4	R bit3	R bit2	R bit1	R bit0

LEN: Continued pixel length

LEN = 0: until right end

LEN = 1: 1 pixel

LEN = 2: 2 pixels

LEN = 3: 3 pixels

:

LEN = 255: 255 pixels

Example

8×8 pixels bit Image, color 12-bit format

0000h	0FFFh	0FFFh	0FFFh	0FFFh	0FFFh	0000h	0000h
0000h	0FFFh	0555h	0555h	0555h	0555h	0555h	0000h
0000h	0FFFh	0555h	0000h	0000h	0000h	0000h	0000h
0000h	0FFFh	0FFFh	0FFFh	0FFFh	0FFFh	0000h	0000h
0000h	0000h	0555h	0555h	0555h	0FFFh	0555h	0000h
0000h	0000h	0000h	0000h	0000h	0FFFh	0555h	0000h
0000h	0FFFh	0FFFh	0FFFh	0FFFh	0FFFh	0555h	0000h
0000h	0000h	0555h	0555h	0555h	0555h	0555h	0000h

d(1, 2)=1000h d(3, 4)=5FFFh d(5, 6)=0000h
d(7, 8)=1000h d(9,10)=1FFFh d(11,12)=5555h d(13,14)=0000h
d(15,16)=1000h d(17,18)=1FFFh d(19,20)=1555h d(21,22)=0000h
d(23,24)=1000h d(25,26)=5FFFh d(27,28)=0000h
d(29,30)=2000h d(31,32)=3555h d(33,34)=1FFFh d(35,36)=1555h d(37,38)=0000h
d(39,40)=5000h d(41,42)=1FFFh d(43,44)=1555h d(45,46)=0000h
d(47,48)=1000h d(49,50)=5FFFh d(51,52)=1555h d(53,54)=0000h
d(55,56)=2000h d(57,58)=5555h d(59,60)=0000h

Data length comparison

Normal bit image data format: 2×8×8 = 128 bytes

Packaged bit image data format: 60 bytes

3.9 Color Combination Format

This section explains how the combination of reverse display and write mixture display mode affect the display of text and images.

3.9.1 Character Display Color Combination Format

Reverse display setting interchanges the character color (fore color) and background, and write mixture display mode determines whether the background is transparent or opaque, as summarized in the following table. Pixels for shadow or bordering character style (if applicable) are not affected. Pixels for bold character style (if applicable) are treated the same as for the character.

C 1: character color, C 2: background color, C 3: shadow and bordering color

Write mixture display mode	Reverse display	Character style	Character	Background	Character style
Normal display write	OFF	OFF	C 1	C 2	
		Bold	C 1	C 2	C 1
		Shadow	C 1	C 2	C 3
		Bordering	C 1	C 2	C 3
	ON	OFF	C 2	C 1	
		Bold	C 2	C 1	C 2
		Shadow	C 2	C 1	C 3
		Bordering	C 2	C 1	C 3
Thru write	OFF	OFF	C 1	Transparent	
		Bold	C 1	Transparent	C 1
		Shadow	C 1	Transparent	C 3
		Bordering	C 1	Transparent	C 3
	ON	OFF	Transparent	C 1	
		Bold	Transparent	C 1	Transparent
		Shadow	Transparent	C 1	C 3
		Bordering	Transparent	C 1	C 3

3.9.2 Monochrome Bit Image Color Combination Format

Monochrome bit images are handled similar to character display. Reverse display setting interchanges the character color (fore color) and background, and write mixture display mode determines whether the background is transparent or opaque, as summarized in the following table:

C 1: character color, C 2: background color

Write mixture display mode	Reverse display	Bit = 1	Bit = 0
Normal display write	OFF	C 1	C 2
	ON	C 2	C 1
Thru write	OFF	C 1	Transparent
	ON	Transparent	C 1

3.9.3 Color Image / BMP File Color Combination Format

For color images (including 2-color monochrome BMP files), the reverse display setting inverts the color data for each pixel that is displayed, and write mixture display mode determines whether the black pixels in the image (if any) are transparent or opaque, as summarized in the following table:

C 1: Pixel color

Write mixture display mode	Reverse display	Pixel Color	
		C 1 = black	C 1 = other
Normal display write	OFF	C 1	C 1
	ON	~C 1	~C 1
Thru write	OFF	Transparent	C 1
	ON	Transparent	~C 1

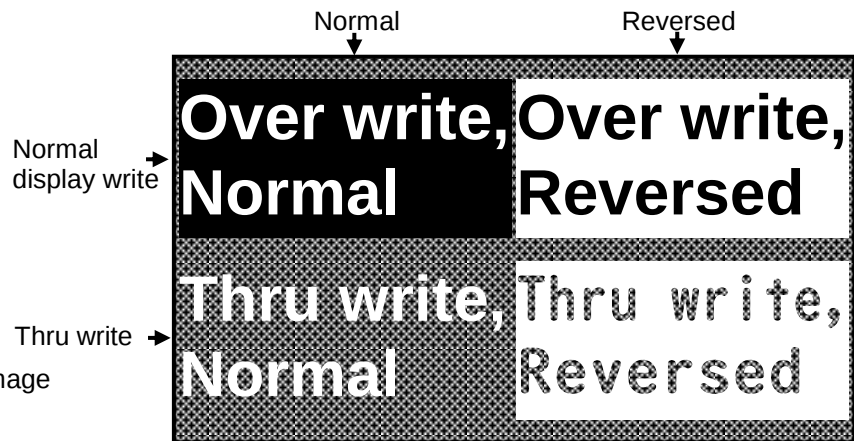
3.9.4 Color Combination Examples

Character display example

 Character color

 Background color

 Existing display pattern / image



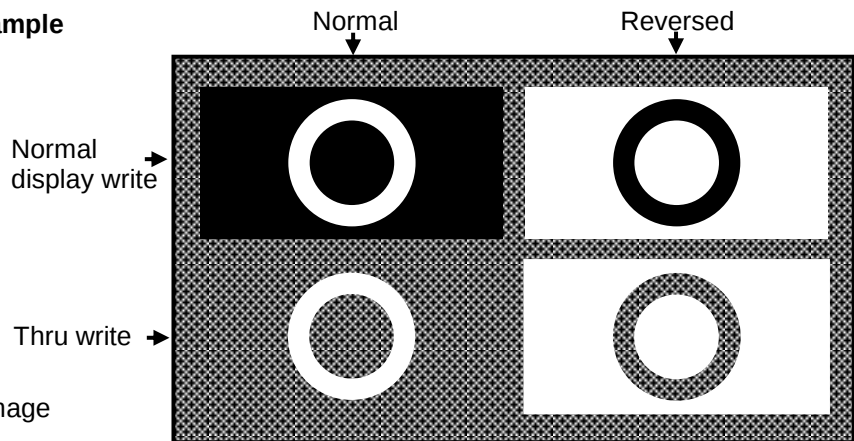
Monochrome bit image example

 Character color

 Background color

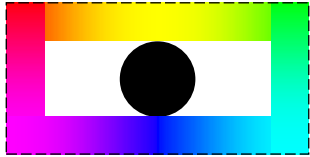
 Existing display pattern / image

(Character color = Fore Color)



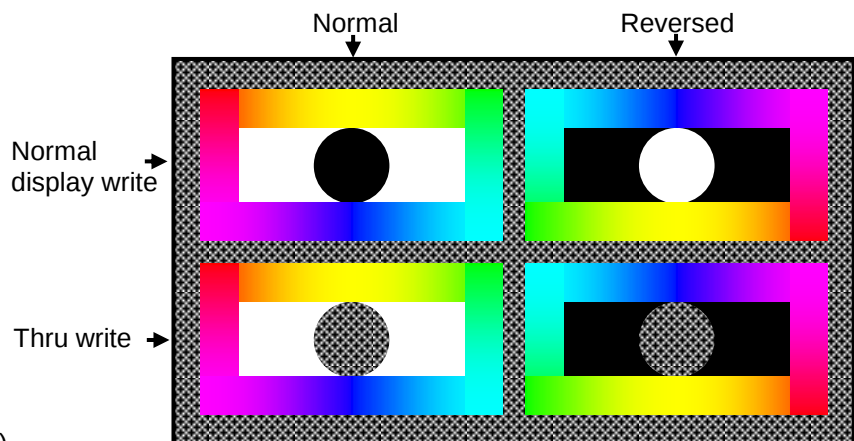
Color image example

Color image:

Red Yellow Green

 Violet Blue Cyan

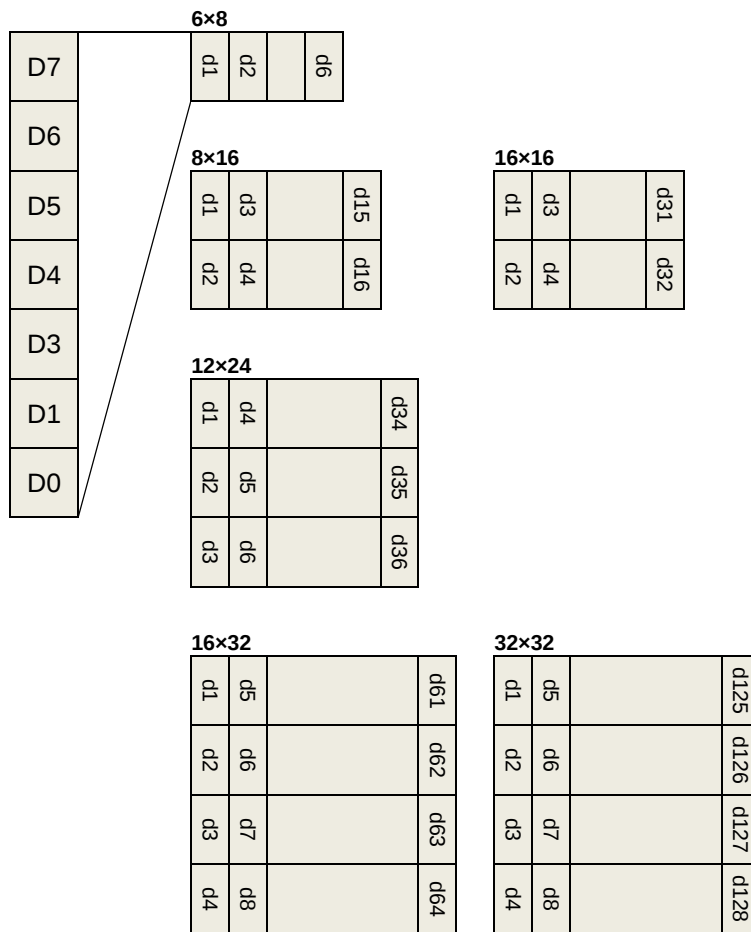
(Solid black circle in center)

 Existing display pattern/ image



3.10 Download Character Format

Download character format is shown below.



3.11 Touch Panel

3.11.1 Outline

Detection method: projected-capacitive touch (multi-touch (multiple-point input) supported)

3.11.1.1 Touch Detection

The touch sensor of this product quantifies the strength of the electric field between the electrodes and determines whether the touch is ON/ OFF according to the change.

<Touch detection flow>

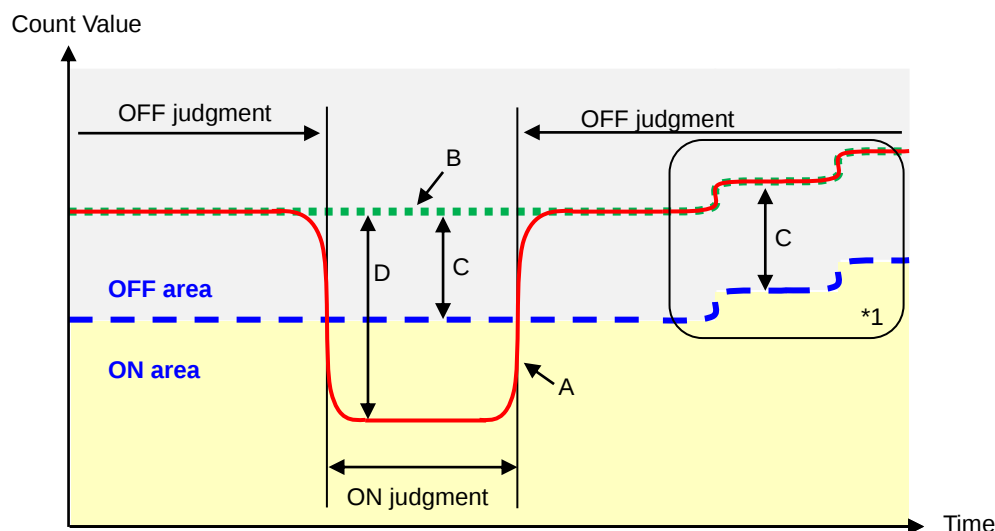
1. If there is no human finger (or equivalent conductor) near the FLETAS touch panel, the OFF judgment is maintained.
2. Count Value decreases as a finger approaches the FLETAS touch panel.
3. When Count Value falls below the threshold, the judgment is ON.
4. Count Value increases when a finger leaves the FLETAS touch panel.
5. When Count Value exceeds the threshold, the judgment is OFF.

A. Count Value: A numerical value of the strength of the electric field between the electrodes

B. Touch Reference: Count Value when not touching

C. Threshold: Threshold for judging ON / OFF (a constant value from the touch reference)

D. Touch Level: Touch Reference - Count Value



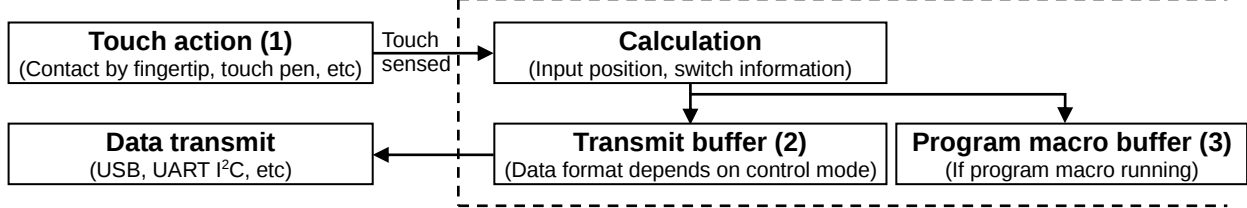
*1: Count Value at the time of non-touch (OFF state) may change due to changes in the surrounding environment.

Since the calibration function is provided, Count Value is used as the average value of Count Value, so that stable touch detection can be obtained without being affected by changes in the surrounding environment. It is also possible to turn off the calibration function. Please contact our sales representative for details.

*2: If the ON judgment period continues for 10 seconds, Count Value is reset as Touch Reference and automatically switches from ON judgment to OFF judgment to prevent foreign objects or unexpected conductors from adhering and continuing the ON judgment. It is also possible to turn off the function. Please contact our sales representative for details.

3.11.2 Basic Operation

The display module features a touch panel for handling input by fingertip or touch pen, etc. The touch panel function sends data for the input position or operation of user-defined switch buttons.



Notes:

- (1) The number of simultaneous touches recognized depends on the touch mode.
- (2) Touch information is queued when touch panel data transmit is ON and sufficient space is available in the transmit buffer (buffer capacity: 128 bytes). When there is insufficient space, touch actions are not queued, so the queued data should be periodically read.
- (3) Touch information is queued when program macro is running and space is available in the buffer (buffer capacity: 1 touch). Refer to program macro specification for usage details.

3.11.3 Touch Modes

There are two touch modes. Single-touch mode (default) recognizes only one touch at a time, generating continuous touch reports while the touch continues, stopping the reports when touch is released. Multi-touch mode recognizes a maximum of 1 to 10 (configurable) touches, generating touch reports only when changes (touch / release / touch position change) occur.

Note: In multi-touch mode, if the Touch Panel Control Mode is changed while a touch is present, the touch reports generated depend on the Touch Panel Control Data Transmit Mode selection.

Single Touch Mode Report Generation

Touch reports generated in each condition are shown below.

Condition			Touch Panel Control Data Transmit Mode	
Touch Panel Data Transmit ON/OFF	Touch Panel Control Mode	Touch state		Data transmit mode #1
OFF	-	Touch state		No report
		Release state		No report
ON	Switch Matrix Mode Custom Switch Mode	Non-switch area touched		No report
		Switch area touched		Touch reports generated continuously
		Released after touching switch		No report
	Coordinates Mode	Touched		Touch reports generated continuously
		Released		No report
	Change to same mode, different configuration e.g.: Custom Switch mode → Custom Switch mode	Release state		No report
		Touch state	Same Switch No. defined at touch position	Touch reports (for new mode) generated continuously
			Different Switch No. defined at touch position	Touch reports (for new mode) generated continuously
			Switch not defined at touch position	No report
	Change to different mode e.g.: Custom Switch Mode → Coordinates Mode	Release state		No report
		Touch state		Touch reports (for new mode) generated continuously

Multi-Touch Mode Report Generation

Touch / Release reports generated in each condition are shown below.

Condition			Touch Panel Control Data Transmit Mode	
Touch Panel Data Transmit ON/OFF	Touch Panel Control Mode	Touch Panel Data Transmit ON/OFF	Touch Panel Control Mode	Touch Panel Data Transmit ON/OFF
OFF	-	Touched	No report	
		Released	No report	
ON	Switch Matrix Mode Custom Switch Mode	Non-switch area touched	No report	
		Switch area touched	Touch Report	
		Released after touching switch	Release Report	
	Coordinates mode	Touched	Touch Report	
		Released	Release Report	
	Change to same mode, different configuration e.g.: Custom Switch mode → Custom Switch mode	Release state	No report	
		Touch state	Same Switch No. defined at touch position	Touch Report (new mode)
			Different Switch No. defined at touch position	Touch Report (new mode)
			Switch not defined at touch position	No report
	Change to different mode e.g.: Custom Switch Mode → Coordinates Mode	Release state	No report	
		Touch state	No report	Touch Report (new mode)

3.11.4 Control Modes

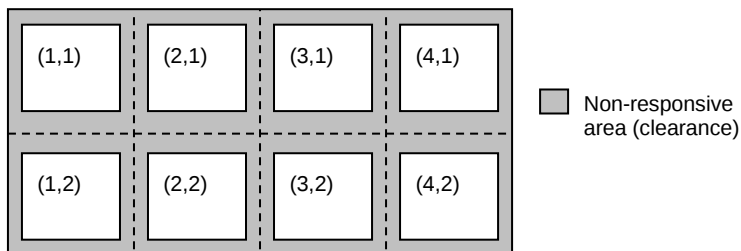
There are three control modes. The control mode can be set independently for up to four separate channels. By selecting a channel, the control mode can be easily changed during use. The control modes are explained below.

3.11.4.1 Coordinates Mode

In this mode, the touch panel is configured as a display area. The (x,y) coordinate values of the input position (in 1-pixel units) are reported.

3.11.4.2 Matrix Switch Mode

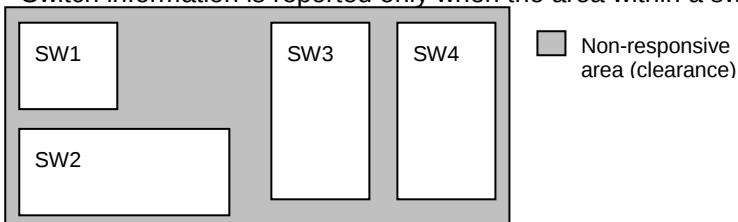
The touch panel is configured as a matrix of switch buttons, of x columns and y rows. The non-responsive (clearance) margin around each switch can also be specified. Switch information is reported only when the area within a switch is touched.



Example: x: 4 switches, y: 2 switches

3.11.4.3 Custom Switch Mode

Touch panel is configured with switch button areas of user-specified position and size. Switch information is reported only when the area within a switch is touched.



3.12 Touch Panel Commands

3.12.1 US P 01h n (Touch Mode Selection: Single-Touch Mode/ Multi-Touch Mode)

Code: 1Fh 50h 01h n

n: Select touch mode and maximum simultaneous touch detection (for Multi-Touch mode)

00h: Single-touch mode

01h ≤ n ≤ 0Ah: Multi-Touch mode (n = maximum simultaneous touches)

Default = 00h

Definable area: 00h ≤ n ≤ 0Ah

Function: Touch mode select (Single touch mode / Multi-Touch mode). For Multi-Touch mode, also specifies the maximum number of simultaneous touches that are handled.

3.12.2 US P 10h ch md (Coordinates Mode)

Code: 1Fh 50h 10h ch md

ch: Channel

md: Mode

Definable area: 00h ≤ ch ≤ 03h
md = 00h

Function: Set the specified channel to coordinates mode.

In coordinates mode, (x, y) coordinate values corresponding to the touch position are transmitted.

Transmitted data format (single-touch mode):

Transmitted data	Hex	Data length
(1) Header	10h	1 byte
(2) Identifier	00h	1 byte
(3) Data	00h–FFh	4 bytes tXL: x-coordinate, lower byte tXH: x-coordinate, upper byte tYL: y-coordinate, lower byte tYH: y-coordinate, upper byte

Touch data is transmitted when touch panel is touched.

Transmitted data format (multi-touch mode):

Transmitted data	Hex	Data length
(1) Header	10h	1 byte
(2) Identifier	10h, 11h	1 byte 10h: Released 11h: Touched
(3) Touch number	01h–0Ah	1 byte
(4) Data	00h–FFh	4 bytes tXL: x-coordinate, lower byte tXH: x-coordinate, upper byte tYL: y-coordinate, lower byte tYH: y-coordinate, upper byte

Touch data is transmitted, for each detected touch, when touch panel is touched or released, or a touch position changes.

3.12.3 US P 10h ch md nx cx ny cy (Switch Matrix Mode)

Code: 1Fh 50h 10h ch md nx cx ny cy

ch: Channel

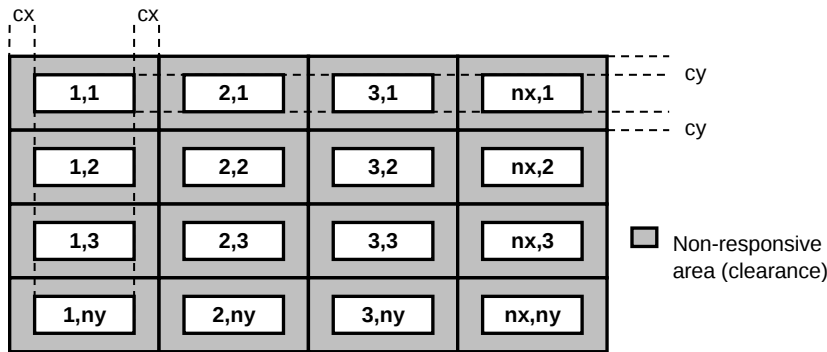
md: Mode

Definable area: $00h \leq ch \leq 03h$
 $md = 01h$
 $01h \leq nx \leq 10h$
 $01h \leq cx \leq 10h$
 $01h \leq ny \leq 10h$
 $01h \leq cy \leq 10h$

Function: Sets the specified channel to switch matrix mode.

In switch matrix mode, the touch panel is configured as a matrix of switch buttons, specified by nx, ny. The size of the non-responsive (clearance) area in-between switch buttons is specified by parameters cx, cy.

Configuration details are shown below.



Transmitted data format (single-touch mode):

Transmitted data	Hex	Data length
(1) Header	10h	1 byte
(2) Identifier	01h	1 byte
(3) Data	01h–10h	2 bytes tX: x-direction switch number tY: y-direction switch number

Touch data is transmitted when touch panel is touched.

Transmitted data format (multi-touch mode):

Transmitted data	Hex	Data length
(1) Header	10h	1 byte
(2) Identifier	20h, 21h	1 byte 20h: Released 21h: Touched
(3) Data	01h–10h	2 bytes tX: x-direction switch number tY: y-direction switch number

Touch data is transmitted, for each switch, when its touch status (touched / released) changes.

Switch button size is calculated from the supplied parameters.

Depending on the set values, the calculated size may not be an integer number of pixels.

In this case, the size of each switch button is rounded down to an integer number of pixels, and the remaining pixel area forms an extra non-responsive (clearance) area on the bottom and/ or right edge (shown as (2), below).

Switch button pixel-size calculation (fractions discarded):

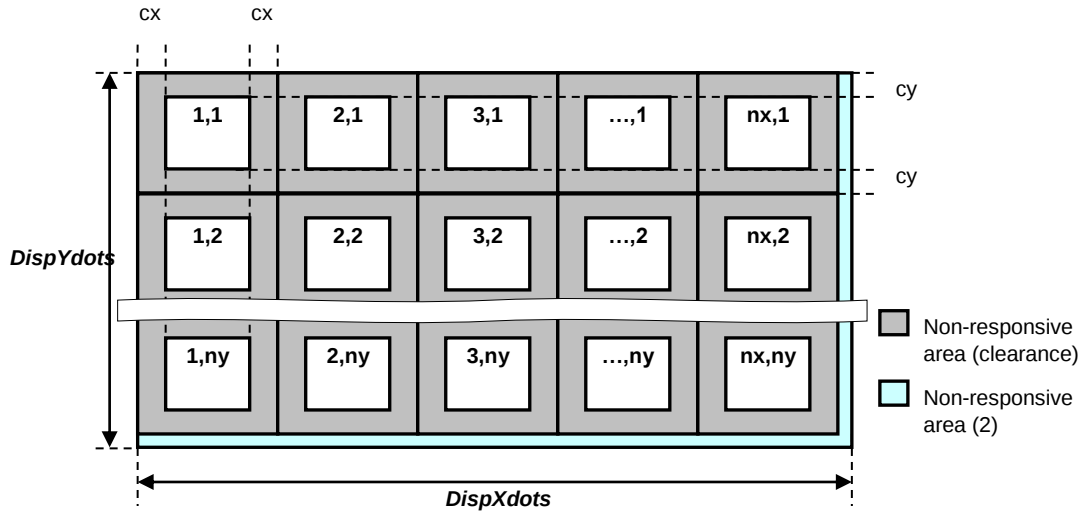
X (horizontal) size: $(DispXdots - (cx \times 2 \times nx)) / nx$

Y (vertical) size: $(DispYdots - (cy \times 2 \times ny)) / ny$

Example: $DispXdots = 800$, $DispYdots = 480$, $cx = 02h$, $nx = 06h$, $cy = 04h$, $ny = 07h$

Switch button X-size: $(800 - (2 \times 2 \times 6)) / 6 = 776 / 6 = 129.3 \rightarrow 129$ pixels

Switch button Y-size: $(480 - (4 \times 2 \times 7)) / 7 = 424 / 7 = 60.6 \rightarrow 60$ pixels



3.12.4 US P 10h ch md sn px1L px1H py1L py1H sx1L sx1H sy1L sy1H [... px(sn)L px(sn)H py(sn)L py(sn)H sx(sn)L sx(sn)H sy(sn)L sy(sn)H] (Custom Switch Mode)

Code: 1Fh 50h 10h ch md sn px1L px1H py1L py1H sx1L sx1H sy1L sy1H [... px(sn)L px(sn)H py(sn)L py(sn)H sx(sn)L sx(sn)H sy(sn)L sy(sn)H]

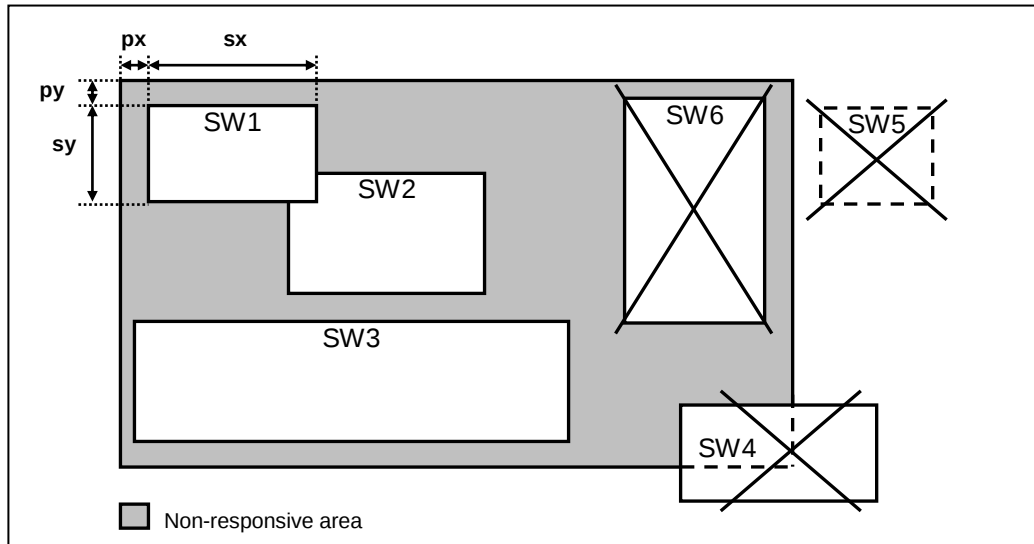
ch: Channel
md: Mode
sn: Number of switch buttons
pxL: Switch button left horizontal position, lower byte (by 1 pixel)
pxH: Switch button left horizontal position, upper byte (by 1 pixel)
pyL: Switch button top vertical position, lower byte (by 1 pixel)
pyH: Switch button top vertical position, lower byte (by 1 pixel)
sxL: Switch button x-size, lower byte (by 1 pixel)
sxH: Switch button x-size, upper byte (by 1 pixel)
syL: Switch button y-size, lower byte (by 1 pixel)
syH: Switch button y-size, upper byte (by 1 pixel)

Definable area: $00h \leq ch \leq 03h$
 $md = 02h$
 $00h \leq sn \leq 20h$
 $0000h \leq (pxH \times 100h + pxL) \leq DispXdots - 1$
 $0000h \leq (pyH \times 100h + pyL) \leq DispYdots - 1$
 $0001h \leq (sxH \times 100h + sxL) \leq (DispXdots - px)$
 $0001h \leq (syH \times 100h + syL) \leq (DispYdots - py)$

Function: Sets the specified channel to custom switch mode.

In custom switch mode, the touch panel is configured with a user-defined number of switch buttons, each of user-defined size and position. Any remaining area is non-responsive.

Configuration details are shown below.



Switch button areas that overlap may be defined, however only the lowest switch button number is responsive in any overlapped area.

If the start position (px, py) and/or size (sx, sy) for a switch button is outside the defined area, that switch button number and any subsequent switch button numbers are invalid.

Transmitted data format (single-touch mode):

Transmitted data	Hex	Data length
(1) Header	10h	1 byte
(2) Identifier	02h	1 byte
(3) Data	01h–20h	1 byte tn: Switch button number

Touch data is transmitted when touch panel is touched.

Transmitted data format (multi-touch mode):

Transmitted data	Hex	Data length
(1) Header	10h	1 byte
(2) Identifier	30h, 31h	1 byte 30h: Released 31h: Touched
(3) Data	01h–20h	1 bytes tn: Switch button number

Touch data is transmitted, for each switch, when its touch status (touched / released) changes.

3.12.5 US P 20h m (Touch Panel Data Transmit ON/ OFF)

Code: 1Fh 50h 20h m

m: Transmit ON/OFF

Definable area: m = 00h, 01h
m = 00h: Transmit OFF
m = 01h: Transmit ON

Default: m = 00h (Transmit OFF)

Function: Sets whether or not touch operation data is transmitted to the host.

When OFF, touch operation data is not placed in the transmit buffer.
Setting is not applicable for reading touch data within a program macro.

3.12.6 US P 21h ch (Touch Panel Channel Select)**Code:** 1Fh 50h 21h ch

ch: Channel number

Definable area: 00h ≤ ch ≤ 03h**Default:** ch = 00h**Function:** Selects the currently-active touch panel control channel.

If channel is undefined, coordinates mode is selected.

3.12.7 US K 70h a [b [c]] (Touch Parameter Setting)**Code:** 1Fh 4Bh 70h a [b [c]]

a : parameter selection/ operation designation

b, c : value

Definable area:

- a = 00h : Threshold setting (c is unnecessary)
00h ≤ b ≤ FFh : threshold value
- a = 04h : Gain setting (c is unnecessary)
00h ≤ b ≤ 0Fh : gain value
- a = 06h : Touch standard reference setting
00h ≤ b ≤ FFh : maximum noise value
(If something channel over this value, this command will be failed.)
00h ≤ c ≤ FFh : designation of measurement numbers
- a = 07h : Touch standard reference use (valid/ invalid) (c is unnecessary)
00h ≤ b ≤ 01h : 00h (invalid), 01h (valid)
- a = 08h : Touch standard reference status read (b, c are unnecessary)

Function: Touch parameter setting.**3.12.7.1 Threshold and Gain (a= 00h/ a = 04h)**

These commands are used for adjusting touch sensitivity.

Decreasing the threshold value increases sensitivity.

Increasing the threshold value reduces sensitivity.

Optimum gain value depends on the touch sensor construction. This should be left at the factory default value.

Settings take effect immediately, but they are not stored in non-volatile memory.

The initial value is set by the memory SW setting value (threshold = MSW59, gain = MSW58).

For the default memory SW setting value, refer to the hardware specifications of each model.

3.12.7.2 Touch Standard Reference Related Command (a= 06h/ 07h/ 08h)

The touch standard references function is an optional function that can be used to help improve the reliability of touch detection with changing environmental conditions, such as water on the screen. In order to improve the reliability of touch detection using this function, it is necessary to execute "touch standard references setting procedure" once (see below) for each module and confirm the result is "success". To ensure that accurate reference values are measured, this command must be run in a controlled environment (for example, in the final stage of product assembly) with the product in its final form (mounting case, a cover, etc.), with no touches, moisture, or other foreign matter. If touch standard references have been successfully set as described above, the function can then be enabled with the "Touch Standard References Usage" command. If standard reference values are not set, or if "Touch Standard Reference Usage" has not been set to "on", this function is not used. (In this situation touch detection operates with base-level performance).

a = 06h: Touch Standard Reference Setting Procedure

In order to measure accurate reference values, this procedure must be run in a controlled environment (no touches, moisture, foreign objects, or excessive noise) with the product in its actual usage configuration (casing, a cover, etc.). When the Touch Standard References Setting Procedure command is executed, FLETAS touch panel is measured two (or more) times, and if the differences between the measurements (noise level) for all measurement points (channels) is less than b, the measurement is "successful". If exceeded for any channel, measurement is "failed". The noise level of the channel with the most noise and the x, y sensor position of that channel are provided in the response data.

If the measurement is successful, Touch Standard References Usage is set to "on". If it fails, it is set to "off".

The reference values are saved in the touch controller, but the touch standard reference usage setting is not saved, so it is necessary to issue the touch standard reference usage (on) command after a reset or restart.

Response data (4 bytes)

00h NNh NXh NYh = Success (noise is within the limit)

01h NNh NXh NYh = Failure (noise limit exceed)

02h XXh XXh XXh = Failure (other problems / defects)

NNh = Noise value of noisiest channel

NXh = Noisiest channel X

NYh = Noisiest channel Y

Transmit data	Hex	Data length
Status	00h~02h	1 byte
NNh / XXh	00h~FFh	1 byte
NXh / XXh	00h~FFh	1 byte
NYh / XXh	00h~FFh	1 byte

a = 07h: Touch Standard Reference Usage (on / off) ('c' not used)

b = 00h: OFF (initial value)

b = 01h: ON

Note: "ON" setting has no effect if a valid reference values have not been stored in the touch controller using the above "Touch Standard References Setting Procedure" command.

a = 08h: Touch Standard Reference Usage Status Read ('b', 'c' not used)

Response data (1 byte): 00h = OFF 01h = ON

Note: The following data will be transmitted from the interface that is currently enabled.

Transmit data	Hex	Data length
Data	00h/ 01h	1 byte

3.12.8 Touch Setting Package Data Store US (e 1Ch a d[1] ... d[1024]**Code:** 1Fh 28h 65h 1Ch a d[1] ... d[1024]

a: Touch setting package data storing destination

d: Package data

Definable area: 01h ≤ a ≤ 04h
00h ≤ d ≤ FFh**Function:** Touch setting package data is stored.

For validating the setting value of touch setting package, it is necessary to set the Memory Switch (MSW63).

To start with the sensitivity setting value of touch setting package at power on, it is necessary to set the Memory Switch (MSW62).

Please contact our sales consultant for data provision of touch setting package.

3.12.9 Touch Setting Package Selection US K 70h 10h a**Code:** 1Fh 4Bh 70h 10h a

a: Touch setting package

Definable area: a = 00h : Factory setting
01h ≤ a ≤ 04h : Touch setting package 1 to 4.**Default:** Memory Switch (MSW63) setting
For the default memory SW setting value, refer to the hardware specifications of each model.**Function:** Select touch setting package to use.

After executing this command, the touch control will start with selected touch setting package value. Touch Setting Package Data Store command is necessary in advance.

3.12.10 Touch Level Read US K 70h a**Code:** 1Fh 4Bh 70h a

a: Touch level read mode

Function: Send touch level information of FLETAS touch panel.

This command is used to adjust the sensitivity with the product tool "GTOMP".

The details of the transmission contents are not disclosed.

3.12.11 Touch Panel Control Data Transmit Mode**Code:** 1Fh 50h 22h a

a: Data transmit mode

Definable area: a = 00h: Data transmit mode #1
a = 01h: Data transmit mode #2**Default:** Memory Switch (MSW60) setting
For the default memory SW setting value, refer to the hardware specifications of each model.**Function:** Select transmit mode of the touch / release reports for the FLETAS touch panel.
For details on report generation conditions, see 3.11.3 Touch Modes.**3.12.12 Touch Scan Period Setting for Power Saving Mode****Code:** 1Fh 28h 61h 49h p

p: Touch Scan Period

Definable area: 05h (5ms) ≤ p ≤ FEh (254ms)**Default:** Memory Switch (MSW61) setting
For the default memory SW setting value, refer to the hardware specifications of each model.**Function:** Set the touch scan period for power saving mode
Higher values result in lower power consumption during power saving mode at the expense of longer response time for touch detection.
This setting is not used if wakeup on touch is not set when entering power saving mode (b0 = 0).

4 Setup

4.1 Jumper

JP No.	Function	Default
J0	I ² C slave address	OPEN
J1		OPEN
J2	Reserved	OPEN
J3	Baud rate	OPEN
J4		OPEN
J5	Serial interface mode	OPEN
J6		OPEN
J7	Program macro start	OPEN
J8	Reserved	OPEN
J9	Reserved	OPEN
J10	Touch panel calibration mode	OPEN

4.1.1 Display Address (I²C Interface)

J0	J1	Function
OPEN	OPEN	I ² C slave address = 50h
SHORT	OPEN	I ² C slave address = 51h
OPEN	SHORT	I ² C slave address = 52h
SHORT	SHORT	I ² C slave address = 53h

4.1.2 Baud Rate (UART Interface)

J3	J4	Function
OPEN	OPEN	38,400bps
SHORT	OPEN	19,200bps
OPEN	SHORT	Apply Memory SW setting (MSW48)
SHORT	SHORT	115,200bps

For Memory SW setting details, refer to 4.2 Memory SW.

4.1.1 Serial Interface Type

J5	J6	Function
OPEN	OPEN	SPI mode
OPEN	SHORT	Reserved
SHORT	OPEN	I ² C mode
SHORT	SHORT	UART mode

4.1.2 Program Macro Start

J7	Function
OPEN	Apply Memory SW setting (MSW19)
SHORT	No start

4.1.3 Operating Mode

J10	TEST pin	Function
-	L	Test mode
OPEN	H (NC)	Normal command mode
SHORT	H (NC)	Touch panel calibration mode

4.2 Memory SW

MSW No.	Function	Valid range	Default
0	International font set	00h–0Dh	00h
1	Character table type	00h–05h, 10h–13h, FEh, FFh	00h
2	Horizontal scroll speed	00h–1Fh	00h
3	Reverse display	00h, 01h	00h
4	Write mixture display mode	10h, 11h	10h
5	Brightness level setting	00h–FFh	FFh
6	Display orientation 00h: 0° (Default) 01h: 90° 02h: 180° 03h: 270°	00h–03h	00h
7	Write screen mode	00h, 01h	00h
8	Font size	01h, 02h, 03h, 04h	01h
9	2-byte character	00h, 01h	00h
10	Font magnification X	01h–04h	01h
11	Font magnification Y	01h–04h	01h
12	Character style	00h–03h	00h
13	2-byte character type	00h–03h	00h
14, 15	Reserved	-	00h
16	Download character restore at power-on (FROM→RAM) 6×8 pixel (00h = Don't restore)	00h, 01h	00h
17	Download character restore at power-on (FROM→RAM) 8×16 pixel (00h = Don't restore) *1	00h, 01h	00h
18	Download character restore at power-on (FROM→RAM) 16×16 pixel (00h = Don't restore) *2	00h, 01h	00h
19	FROM macro execution at power-on (00h = Not execute)	< F220, or later > 00h–08h, 81h–88h < less than F220 > 00h–04h, 81h–84h	00h
20–47	Reserved	-	00h
48	UART Baud rate setting 00h: 19,200bps (default) 01h: 4,800bps 02h: 9,600bps 03h: 19,200bps 04h: 38,400bps 05h: 57,600bps 06h: 115,200bps	00h–06h	00h
49	UART Parity 00h: None 01h: Even 02h: Odd	00h–02h	00h
50, 51	Reserved	-	00h
52	Macro end code enable/ disable	00h, 01h	00h
53	Macro end code	00h–FFh	00h
54	Macro end clear screen setting	00h, 01h	00h
55–57	Reserved	-	00h
58	Touch sensitivity (signal gain) setting	00h–0Fh	*3
59	Touch sensitivity (threshold) setting	00h–FFh	*3
60	Touch Panel Control Data Transmit Mode 00h: Data transmit mode #1 01h: Data transmit mode #2	00h, 01h	*3
61	Touch scan period setting at power saving mode (ms)	05h(5ms)-FEh(254ms)	*3
62	Touch sensitivity setting selection at startup 00h: Apply the setting values of MSW58 and 59 01h: Apply touch setting package value	00h, 01h	*3

MSW No.	Function	Valid range	Default
63	Touch setting package selection at startup 00h: Factory setting 01h: Touch setting package 1 02h: Touch setting package 2 03h: Touch setting package 3 04h: Touch setting package 4	00h – 04h	*3

Note: Module operates with default value if Memory SW value is outside the valid range.

*1: If setting is 01h, download characters for 12×24 pixel and 16×32 pixel are also restored.

*2: If setting is 01h, download characters for 32×32 pixel are also restored.

*3: For the default memory SW setting value, refer to the hardware specifications of each model.

5 Lot No. / Firmware Version Specification

Lot No. and firmware version are displayed on this product. Lot No. and firmware version are determined based on the following specifications.

Lot No.	Firmware Version
<u>1</u> 8 <u>C</u> <u>2</u> <u>2</u> EH	<u>F</u> <u>1</u> <u>0</u> <u>0</u>
↑ ↑ ↑	↑
A B C	D

A: Year of manufacture
B: Month of manufacture

C: Date of manufacture
D: Firmware version

Indicated the last two digits of the Christian era.

Indicated by one letter of English character.

January, February, ... December = A, B, ... L

Indicated two digits of the date.

Indicated firmware version

The above lot number example is the product of firmware version F100 on March 22, 2018.

6 Firmware Version Revision History

Firmware version	Contents
F120b or later	Applicable TTF and OTF format of outline font (Previous version OTF format only) Applicable following commands 3.11.8 Touch Setting Package Data Store 3.11.9 Touch Setting Package Selection
F131b or later	Applicable following commands (The following commands are only supported on GTWV050C3A00PA.) 3.6.4.76 Power Saving Mode 3.6.4.76 Touch Scan Period at Power Saving Mode
F150b or later	3.11.10 Touch Level Read Added new command The following commands are only supported on GTWV050C3A00PA. 3.6.4.75 Power Saving Mode 3.6.4.76 Touch Scan Period at Power Saving Mode
F162b or later	3.6.4.73 Display Status Send Added a = 21h: General-purpose FROM2 memory check sum information The following commands are only supported on GTWV050C3A00PA. 3.6.4.75 Power Saving Mode 3.6.4.76 Touch Scan Period at Power Saving Mode
F200b or later	4.2 Memory SW Added SW No.6: display orientation. The following commands are supported on GTWV050C3A00PA and GTWV070C3A00PA. 3.6.4.75 Power Saving Mode 3.6.4.76 Touch Scan Period at Power Saving Mode
F210b or later	4.2 Memory SW Added SW No.60: Touch Panel Control Data Transmit Mode. The following commands are supported. 3.6.4.75 Power Saving Mode 3.6.4.76 Touch Scan Period at Power Saving Mode 3.7.4.35 US (a 40h p (Display Power ON/ OFF) “p=80h: Auto-ON” 3.12.11 Touch Panel Control Data Transmit Mode
F220b or later	FROM macro storage capacity has been increased from 32K bytes to 64K bytes. 3.7.4.65 FROM Macro Define/ Delete 3.7.4.66 Macro Execution 4.2 Memory SW

Revision Note

Rev.	Date	Revision
00	Jul. 21, 2021	Initial release The specifications have been completely revised so that the default setting values of the memory SW are referred to the hardware specifications of each model. * The changes from DS-2007-0003-11 are as follows. 1.2 Related specifications The "related specifications" of "1.1 Scope of application" has been added as a new section. 3.7.3.12 Touch Panel Control Commands Added following command. Touch Parameter Setting Added following comment. *1: For the default memory SW ... of each model. 3.12.4 Custom Switch Mode Function Corrected configuration details "If the start position ... button number is invalid." → "If the start position ... switch button numbers are invalid." 3.12.7.1 Threshold and Gain (a= 00h/ a = 04h) Added following comment. The initial value is ... (threshold = MSW59, gain = MSW58). 4.2 Memory SW Added following comment. *3: For the default memory SW ... of each model. Touch Panel Control Data Transmit Mode Move from "3.7.3.3 Display action command" to "3.7.3.12 Move to touch panel control command". Move from "3.7.4.76 Touch Scan Period Setting for Power Saving Mode" to "3.12.12 Touch Scan Period Setting for Power Saving Mode". Overall change Added following comment. "For the default memory SW setting value, refer to the hardware specifications of each model." 3.12.7.1 Threshold and Gain 3.12.9 Touch Setting Package Selection 3.12.11 Touch Panel Control Data Transmit Mode 3.12.12 Touch Scan Period Setting for Power Saving Mode
01	Aug. 11, 2021	3.1.2 Character Display Link error has been corrected. https://github.com/adobe-fonts/source-han-sans/