INTRODUCTION
KS0078 is a dot matrix LCD driver & controller LSI which is fabricated by low power CMOS technology. It can display 1, 2, or 4 lines with 5 x 8 or 6 x 8 dots format.

FUNCTIONS
- Character type dot matrix LCD driver & controller
- Internal driver: 34 common and 120 segment signal output
- Easy interface with 4-bit or 8-bit MPU
- Clock synchronized serial Interface
- 5 x 8 dot matrix possible
- 6 x 8 dot matrix possible
- Bidirectional shift function
- All character reverse display
- Display shift per line
- Voltage converter for LCD drive voltage: 13 V max (2 times / 3 times)
- Various instruction functions
- Automatic power on reset
FEATURES

- Internal Memory
  - Character Generator ROM (CGROM) : 9,600 bits (240 characters x 5 x 8 dot)
  - Character Generator RAM (CGRAM) : 64 x 8 bits (8 characters x 5 x 8 dot)
  - Segment Icon RAM (SEGRAM) : 16 x 8 bits (96 icons max.)
  - Display Data RAM (DDRAM) : 96 x 8 bits (96 characters max.)
- Low power operation
  - Power supply voltage range : 2.7 ~ 5.5 V (VDD)
  - LCD Drive voltage range : 3.0 ~ 13.0 V (VDD - V5)
- CMOS process
- Programmable duty cycle : 1/17, 1/33 (refer to Table 1.)
- Internal oscillator with an external resistor
- Bare chip available

Table 1. Programmable duty cycles

<table>
<thead>
<tr>
<th>Display Line Numbers</th>
<th>Duty Ratio</th>
<th>Single-chip Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Displayable characters</td>
</tr>
<tr>
<td>1</td>
<td>1/17</td>
<td>1 line of 48 characters</td>
</tr>
<tr>
<td>2</td>
<td>1/33</td>
<td>2 lines of 48 characters</td>
</tr>
<tr>
<td>4</td>
<td>1/33</td>
<td>4 lines of 24 characters</td>
</tr>
</tbody>
</table>

6-dot font width

<table>
<thead>
<tr>
<th>Display Line Numbers</th>
<th>Duty Ratio</th>
<th>Single-chip Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Displayable characters</td>
</tr>
<tr>
<td>1</td>
<td>1/17</td>
<td>1 line of 40 characters</td>
</tr>
<tr>
<td>2</td>
<td>1/33</td>
<td>2 lines of 40 characters</td>
</tr>
<tr>
<td>4</td>
<td>1/33</td>
<td>4 lines of 20 characters</td>
</tr>
</tbody>
</table>
PAD CONFIGURATION

COM31 57
COM26 52
COM25 51
COM15 49
COM14 48
COM13 47
COM11 45
COM10 44
COM9 43
SEG120 42
SEG118 40
SEG117 39
SEG116 38
SEG115 37
SEG114 36
SEG113 35
SEG112 34
SEG111 33
SEG110 32
SEG109 31
SEG108 30
SEG107 29
SEG106 28
SEG105 27
SEG104 26
SEG103 25
SEG102 24
SEG101 23
SEG100 22
SEG99 21
SEG98 20
SEG97 19
SEG96 18
SEG95 17
SEG94 16
SEG93 15
SEG92 14
SEG91 13
SEG90 12
SEG89 11
SEG88 10
SEG87 9
SEG86 8
SEG85 7
SEG84 6
SEG83 5
SEG82 4
SEG81 3
SEG80 2
SEG79 1
SEG78 0
SEG77 9
SEG76 8
SEG75 7
SEG74 6
SEG73 5
SEG72 4
SEG71 3
SEG70 2
SEG69 1
SEG68 0
SEG67 9
SEG66 8
SEG65 7
SEG64 6
SEG63 5
SEG62 4
SEG61 3
SEG60 2
SEG59 1
SEG58 0
SEG57 9
SEG56 8
SEG55 7
SEG54 6
SEG53 5
SEG52 4
SEG51 3
SEG50 2
SEG49 1
SEG48 0
SEG47 9
SEG46 8
SEG45 7
SEG44 6
SEG43 5
SEG42 4
SEG41 3
SEG40 2
SEG39 1
SEG38 0
SEG37 9
SEG36 8
SEG35 7
SEG34 6
SEG33 5
SEG32 4
SEG31 3
SEG30 2
SEG29 1
SEG28 0
SEG27 9
SEG26 8
SEG25 7
SEG24 6
SEG23 5
SEG22 4
SEG21 3
SEG20 2
SEG19 1
SEG18 0
SEG17 9
SEG16 8
SEG15 7
SEG14 6
SEG13 5
SEG12 4
SEG11 3
SEG10 2
SEG9 1
SEG8 0
SEG7 9
SEG6 8
SEG5 7
SEG4 6
SEG3 5
SEG2 4
SEG1 3
SEG0 2
SEG-1 1
SEG-2 0

KS0078
34COM/120SEG LCD CONTROLLER
# PAD DESCRIPTION

<table>
<thead>
<tr>
<th>PAD (NO)</th>
<th>INPUT/OUTPUT</th>
<th>NAME</th>
<th>DESCRIPTION</th>
<th>INTERFACE</th>
</tr>
</thead>
<tbody>
<tr>
<td>VDD (60)</td>
<td>-</td>
<td>Power supply</td>
<td>for logical circuit (+3V, +5V)</td>
<td>Power supply</td>
</tr>
<tr>
<td>VSS1, VSS2 (66, 81)</td>
<td>-</td>
<td>Bias voltage level for LCD driving.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>V1~V5 (88, 84)</td>
<td>Input</td>
<td>Input voltage to the voltage converter to generate LCD drive voltage (Vci = 2.5V, 4.5V).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vci (78)</td>
<td>Input</td>
<td>Input voltage to the voltage converter to generate LCD drive voltage (Vci = 2.5V, 4.5V).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEG1<del>SEG80 (106</del>42)</td>
<td>Output</td>
<td>Segment output</td>
<td>Segment signal output for LCD drive.</td>
<td>LCD</td>
</tr>
<tr>
<td>COM0<del>COM33 (105</del>89, 43~59)</td>
<td>Output</td>
<td>Common output</td>
<td>Common signal output for LCD drive.</td>
<td>LCD</td>
</tr>
<tr>
<td>OSC1, OSC2 (61, 62)</td>
<td>Input (OSC1), Output (OSC2)</td>
<td>Oscillator</td>
<td>When use internal oscillator, connect external R resistor. If external clock is used, connect it to OSC1.</td>
<td>External resistor/oscillator (OSC1)</td>
</tr>
<tr>
<td>C1, C2 (80, 79)</td>
<td>Input</td>
<td>External capacitance input</td>
<td>To use the voltage converter (2 times, 3 times), these pins must be connected to the external capacitance.</td>
<td>External capacitance</td>
</tr>
<tr>
<td>RESET (63)</td>
<td>Input</td>
<td>Reset pin</td>
<td>Initialized to Low</td>
<td>-</td>
</tr>
<tr>
<td>IE (65)</td>
<td>Input</td>
<td>Select pin of instruction set</td>
<td>When IE = &quot;High&quot;, Instruction set is selected as Table 6. When IE = &quot;Low&quot;, Instruction set is selected as Table 10.</td>
<td>-</td>
</tr>
<tr>
<td>V5OUT2 (82)</td>
<td>Output</td>
<td>Two times converter output</td>
<td>The value of Vci is converted two times. To use three times converter, the same capacitance as that of C1-C2 should be connected here.</td>
<td>V5 capacitance</td>
</tr>
<tr>
<td>V5OUT3 (83)</td>
<td>Output</td>
<td>Three times converter output</td>
<td>The value of Vci is converted three times.</td>
<td>V5</td>
</tr>
</tbody>
</table>
### PAD DESCRIPTION (continued)

<table>
<thead>
<tr>
<th>PAD (NO)</th>
<th>INPUT/OUTPUT</th>
<th>NAME</th>
<th>DESCRIPTION</th>
<th>INTERFACE</th>
</tr>
</thead>
<tbody>
<tr>
<td>IM (64)</td>
<td>Input</td>
<td>Interface mode selection</td>
<td>Select Interface mode with the MPU. When IM = “Low”: Serial mode, When IM = “High”: 4-bit/8-bit bus mode.</td>
<td>-</td>
</tr>
<tr>
<td>RS/CS (67)</td>
<td>Input</td>
<td>Register select/Chip select</td>
<td>When bus mode, used as register selection input. When RS/CS = “High”, Data register is selected. When RS/CS = “Low”, Instruction register is selected. When serial mode, used as chip selection input. When RS/CS = “Low”, selected. When RS/CS = “High”, not selected. (Low access enable)</td>
<td>MPU</td>
</tr>
<tr>
<td>RW/SID (68)</td>
<td>Input</td>
<td>Read/write/Serial input data</td>
<td>When bus mode, used as read/write selection input. When RW/SID = “High”, read operation. When RW/SID = “Low”, write operation. When serial mode, used for data input pin.</td>
<td>MPU</td>
</tr>
<tr>
<td>E/SCLK (69)</td>
<td>Input</td>
<td>Read/write enable/Serial clock</td>
<td>When bus mode, used as read/write enable signal. When serial mode, used as serial clock input pin.</td>
<td>MPU</td>
</tr>
<tr>
<td>DB0/SOD (70)</td>
<td>Input/Output</td>
<td>Data bus 0 bit/Serial output data</td>
<td>When 8-bit bus mode, used as lowest bidirectional data bit. During 4-bit bus mode, open this pin. When serial mode, used as serial data output pin. If not in read operation, open this pin.</td>
<td>MPU</td>
</tr>
<tr>
<td>DB1<del>DB3 (71</del>73)</td>
<td>Input/Output</td>
<td>Data bus 1~7</td>
<td>When 8-bit bus mode, used as low order bidirectional data bus. During 4-bit bus mode or serial mode, open these pins.</td>
<td>MPU</td>
</tr>
<tr>
<td>DB4<del>DB7 (74</del>77)</td>
<td>Output</td>
<td></td>
<td>When 8-bit bus mode, used as high order bidirectional data bus. In case of 4-bit bus mode, used as both high and low order. DB7 used for Busy Flag output. During serial mode, open these pins.</td>
<td>MPU</td>
</tr>
</tbody>
</table>
FUNCTION DESCRIPTION

System Interface
This chip has all three kinds interface type with MPU : serial, 4-bit bus and 8-bit bus. Serial and bus(4-bit/8-bit) is selected by IM input, and 4-bit bus and 8-bit bus is selected by DL bit in the instruction register.

During read or write operation, two 8-bit registers are used. one is data register (DR), the other is instruction register(IR).

The data register(DR) is used as temporary data storage place for being written into or read from DDRAM/CGRAM/SEGRAM, target RAM is selected by RAM address setting instruction. Each internal operation, reading from or writing into RAM, is done automatically.

So to speak, after MPU reads DR data, the data in the next DDRAM/CGRAM/SEGRAM address is transferred into DR automatically. Also after MPU writes data to DR, the data in DR is transferred into DDRAM/CGRAM/SEGRAM automatically.

The Instruction register(IR) is used only to store instruction code transferred from MPU. MPU cannot use it to read instruction data.

To select register, use RS/CS input pin in 4-bit/8-bit bus mode(IM = "High") or RS bit in serial mode(IM = "Low").

Table 2. Various kinds of operations according to RS and R/W bits.

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Instruction Write operation (MPU writes Instruction code into IR)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Read Busy flag(DB7) and address counter (DB0 – DB6)</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Data Write operation (MPU writes data into DR)</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Data Read operation (MPU reads data from DR)</td>
</tr>
</tbody>
</table>

Busy Flag (BF)
When BF = "High", it indicates that the internal operation is being processed. So during this time the next instruction cannot be accepted. BF can be read, when RS = Low and R/W = High(Read Instruction Operation), through DB7
Before executing the next instruction, be sure that BF is not High.
PRELIMINARY
KS0078 34COM/120SEG DRIVER & CONTROLLER FOR DOT MATRIX LCD

Display Data RAM (DDRAM)

DDRAM stores display data of maximum 96 x 8 bits (96 characters). DDRAM address is set in the address counter (AC) as a hexadecimal number. (refer to Fig-1.)

Fig-1. DDRAM Address

1) Display of 5-dot font width character

(1) 5-dot 1 line display
In case of 1 line display with 5-dot font, the address range of DDRAM is 00H - 5FH. (Refer to Fig-2)

Fig-2. 1-line X 48ch. display
(2) 5-dot 2 line display
In case of 2 line display with 5-dot font, the address range of DDRAM is 00H~2FH, 40H~6FH. (refer to Fig-3)

Fig-3. 2-line X 48ch. display (5-dot font width)
(3) 5-dot 4 line display

In case of 4 line display with 5-dot font, the address range of DDRAM is 00H - 17H, 20H - 37H, 40H - 57H, 60H - 77H. (refer to Fig-4)

Fig-4. 4-line X 24ch. display (5-dot font width)
2) Display of 6-dot font width character

(1) 6-dot 1 line display
In case of 1 line display with 6-dot font, the address range of DDRAM is 00H~5FH. (refer to Fig-5)

Fig-5. 1-line X 40ch. display
(2) 6-dot 2 line display

In case of 2 line display with 6-dot font, the address range of DDRAM is 00H - 2FH, 40H - 6FH. (refer to Fig-6)

Fig-6. 2-line X 40ch. display (6-dot font width)
(3) 6-dot 4 line display
In case of 4 line display with 6-dot font, the address range of DDARM is 00H ~ 17H, 20H ~ 37H, 40H ~ 57H, 60H ~ 77H. (refer to Fig-7)

Fig-7. 4-line X 20ch. display (6-dot font width)
Timing Generation Circuit

Timing generation circuit generates clock signals for the internal operations.

Address Counter (AC)

Address Counter (AC) stores DDRAM/CGRAM/SEGRAM address, transferred from IR. After writing into (reading from) DDRAM/CGRAM/SEGRAM, AC is automatically increased (decreased) by 1. When RS = “Low” and R/W = “High”, AC can be read through DB0–DB6.

Cursor/Blink Control Circuit

It controls cursor/blink ON/OFF and black/white inversion at cursor position.

LCD Driver Circuit

LCD Driver circuit has 34 common and 120 segment signals for LCD driver. Data from SEGRAM/CGRAM/CGROM is transferred to 120-bit segment latch serially, and then it is stored to 120-bit shift latch. When each com is selected by 34-bit common register, segment data also output through segment driver from 100-bit segment latch.

In case of 1-line display mode, COM0–COM17 have 1/17 duty, and in 2-line or 4-line mode, COM0–COM33 have 1/33 duty ratio.
CGROM (Character Generator ROM)

CGROM has 5 X 8-dot 240 character pattern. (refer to Table 3)
Table 3. CGROM Character Code Table
CGRAM (Character Generator RAM)
CGRAM has up to 5 X 8-dot 8 characters. By writing font data to CGRAM, user defined character can be used.
(Refer to Table 4)

Table 4. Relationship between Character Code(DDRAM) and Character Pattern(CGRAM)

1) 5x8 dot Character pattern

<table>
<thead>
<tr>
<th>Character Code (DDRAM data)</th>
<th>CGRAM address</th>
<th>CGRAM data</th>
<th>Pattern number</th>
</tr>
</thead>
<tbody>
<tr>
<td>D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>A5 A4 A3 A2 A1 A0</td>
<td>B1 B0 X 0</td>
<td>1</td>
</tr>
<tr>
<td>0 0 0 0 X 0 0 0</td>
<td>0 0 0 0</td>
<td>0 0 1</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>0 0 1 1</td>
<td>0 1 0</td>
<td>0 1 1</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>1 0 1 1</td>
<td>1 0 0</td>
<td>1 1 1</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
</tbody>
</table>

(pattern 1)

<table>
<thead>
<tr>
<th>Character Code (DDRAM data)</th>
<th>CGRAM address</th>
<th>CGRAM data</th>
<th>Pattern number</th>
</tr>
</thead>
<tbody>
<tr>
<td>D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>A5 A4 A3 A2 A1 A0</td>
<td>B1 B0 X 1</td>
<td>8</td>
</tr>
<tr>
<td>0 0 0 0 X 1 1 1</td>
<td>0 0 0 0</td>
<td>0 0 1</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>0 1 1 1</td>
<td>0 1 0</td>
<td>0 1 1</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>1 0 1 1</td>
<td>1 0 0</td>
<td>1 1 1</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>( \cdot 0 0 0 0 0 \cdot )</td>
</tr>
</tbody>
</table>

(pattern 8)
2) 6x8 dot Character pattern

<table>
<thead>
<tr>
<th>Character Code (CGRAM data)</th>
<th>CGRAM address</th>
<th>CGRAM data</th>
<th>Pattern number</th>
</tr>
</thead>
<tbody>
<tr>
<td>D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>A5 A4 A3 A2 A1 A0</td>
<td>P7 P6 P5 P4 P3 P2 P1 P0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 X 0 0 0</td>
<td>0 0 0 0</td>
<td>B1 B0 0 0</td>
<td>pattern 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0 0 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0 1 0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 0 0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 0 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 1 0</td>
<td>0 0 0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 1 1</td>
<td>0 0 0</td>
<td></td>
</tr>
</tbody>
</table>

In case of 5-dot font width, when B1 = "1", enabled dots of P0...P4 will blink, and when B1 = "0" and B0 = "1", enabled dots in P4 will blink, when B1 = "0" and B0 = "0", blink will not happen.

| 0 0 0 0 X 1 1 1             | 1 1 1       | 1 1 0     | pattern 8       |
|                             | 1 1 1       | 0 0 0     |
|                             | 1 1 1       | 0 0 0     |
|                             | 1 1 1       | 0 0 0     |

In case of 6-dot font width, when B1 = "1", enabled dots of P0...P5 will blink, and when B1 = "0" and B0 = "1", enabled dots of P5 will blink, when B1 = "0" and B0 = "0", blink will not happen.

2. "X" : Don't care
SEGGRAM (Segment Icon RAM)
SEGGRAM has segment control data and segment pattern data. During 1-line display mode, COM0(COM17) makes the data of SEGGRAM enable to display icons. When used in 2/4-line display mode COM0(COM33) does that.
Its higher 2-bits are blinking control data, and lower 6-bits are pattern data. (refer to Table 5 and Fig-8)

Table 5. Relationship between SEGGRAM address and display pattern

<table>
<thead>
<tr>
<th>SEGGRAM address</th>
<th>SEGGRAM data display pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>A3 A2 A1 A0</td>
<td>5-dot font width</td>
</tr>
<tr>
<td></td>
<td>6-dot font width</td>
</tr>
<tr>
<td>0 0 0 0</td>
<td>D7 D6 D5 D4 D3 D2 D1 D0</td>
</tr>
<tr>
<td>0 0 0 1</td>
<td>B1 B0 X S1 S2 S3 S4 S5</td>
</tr>
<tr>
<td>0 0 1 0</td>
<td>B1 B0 X S11 S12 S13 S14 S15</td>
</tr>
<tr>
<td>0 0 1 1</td>
<td>B1 B0 X S16 S17 S18 S19 S20</td>
</tr>
<tr>
<td>0 1 0 0</td>
<td>B1 B0 X S21 S22 S23 S24</td>
</tr>
<tr>
<td>0 1 0 1</td>
<td>B1 B0 X S26 S27 S28 S29 S30</td>
</tr>
<tr>
<td>0 1 1 0</td>
<td>B1 B0 X S31 S32 S33 S34 S35</td>
</tr>
<tr>
<td>0 1 1 1</td>
<td>B1 B0 X S36 S37 S38 S39 S40</td>
</tr>
<tr>
<td>1 0 0 0</td>
<td>B1 B0 X S41 S42 S43 S44 S45</td>
</tr>
<tr>
<td>1 0 0 1</td>
<td>B1 B0 X S46 S47 S48 S49 S50</td>
</tr>
<tr>
<td>1 0 1 0</td>
<td>B1 B0 X S51 S52 S53 S54 S55</td>
</tr>
<tr>
<td>1 0 1 1</td>
<td>B1 B0 X S56 S57 S58 S59 S60</td>
</tr>
<tr>
<td>1 1 0 0</td>
<td>B1 B0 X S61 S62 S63 S64 S65</td>
</tr>
<tr>
<td>1 1 0 1</td>
<td>B1 B0 X S66 S67 S68 S69 S70</td>
</tr>
<tr>
<td>1 1 1 0</td>
<td>B1 B0 X S71 S72 S73 S74 S75</td>
</tr>
<tr>
<td>1 1 1 1</td>
<td>B1 B0 X S76 S77 S78 S79 S80</td>
</tr>
</tbody>
</table>

* 1. B1, B0 : Blinking control bit

<table>
<thead>
<tr>
<th>Control Bit</th>
<th>Blanking Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>BE B1 B0</td>
<td>5-dot font width</td>
</tr>
<tr>
<td>0 X X</td>
<td>No blink</td>
</tr>
<tr>
<td>1 0 0</td>
<td>D4</td>
</tr>
<tr>
<td>1 1 X</td>
<td>D4 ~ D0</td>
</tr>
</tbody>
</table>

2. S1~S80 : Icon pattern ON/OFF in 5-dot font width
S1~S96 : Icon pattern ON/OFF in 6-dot font width
3. “X” : Don’t care

SAMSUNG ELECTRONICS
### INSTRUCTION DESCRIPTION

1) 5-dot font width (FW = 0)

![Diagram showing the relationship between SEGRAM and segment display for 5-dot font width.]

2) 6-dot font width (FW = 1)

![Diagram showing the relationship between SEGRAM and segment display for 6-dot font width.]

---

**Fig-8. Relationship between SEGRAM and segment display**
OUTLINE
To overcome the speed difference between internal clock of KS0078 and MPU clock, KS0078 performs internal operation by storing control information to IR or DR. The internal operation is determined according to the signal from MPU, composed of read/write and data bus. (refer to Table 6/10) Instruction can be divided largely four kinds,
(1) KS0078 function set instructions (set display methods, set data length, etc.)
(2) address set instructions to internal RAM
(3) data transfer instructions with internal RAM
(4) others.
The address of internal RAM is automatically increased or decreased by 1.
When IE = "High", KS0078 is operated according to Instruction Set 1(Table 6) and when IE = "Low", KS0078 is operated according to Instruction Set 2(Table 10).

* Note: During internal operation, Busy Flag (DB7) is read High. Busy Flag check must precede the next instruction.
### Table 6. Instruction Set 1

<table>
<thead>
<tr>
<th>Instruction</th>
<th>RE</th>
<th>Instruction Code</th>
<th>Description</th>
<th>Execution Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear Display</td>
<td>X</td>
<td>0 0 0 0 0 0 0 0 1</td>
<td>Write &quot;20H&quot; to DDRAM, and set DDRAM address to &quot;00H&quot; from AC.</td>
<td>1.52ms</td>
</tr>
<tr>
<td>Return Home</td>
<td>0</td>
<td>0 0 0 0 0 0 0 1 X</td>
<td>Set DDRAM address to &quot;00H&quot; from AC and return cursor to its original position if shifted. The contents of DDRAM are not changed.</td>
<td>1.52ms</td>
</tr>
<tr>
<td>Power Down Mode</td>
<td>1</td>
<td>0 0 0 0 0 0 0 0 1 PD</td>
<td>Set power down mode bit. PD = &quot;1&quot;: power down mode set, PD = &quot;0&quot;: power down mode disable</td>
<td>37 µs</td>
</tr>
<tr>
<td>Entry Mode Set</td>
<td>0</td>
<td>0 0 0 0 0 0 0 0 1 I/D S</td>
<td>Assign cursor moving direction. I/D = &quot;1&quot;: increment, I/D = &quot;0&quot;: decrement and display shift enable bit. S = &quot;1&quot;: make display shift of the enabled lines by the DS4-DS1 bits in the Shift Enable instruction. S = &quot;0&quot;: display shift disable</td>
<td>37 µs</td>
</tr>
<tr>
<td>Display ON/OFF Control</td>
<td>0</td>
<td>0 0 0 0 0 0 1 D C B</td>
<td>Set display/cursor/blink on/off D = &quot;1&quot;: display on, D = &quot;0&quot;: display off, C = &quot;1&quot;: cursor on, C = &quot;0&quot;: cursor off, B = &quot;1&quot;: blink on, B = &quot;0&quot;: blink off.</td>
<td>37 µs</td>
</tr>
<tr>
<td>Extended function set</td>
<td>1</td>
<td>0 0 0 0 0 0 0 1 FW B/W NW</td>
<td>Assign font width, black/white inverting of cursor, and 4-line display mode control bit. FW = &quot;1&quot;: 6-dot font width, FW = &quot;0&quot;: 5-dot font width, B/W = &quot;1&quot;: black/white inverting of cursor enable, B/W = &quot;0&quot;: black/white inverting of cursor disable NW = &quot;1&quot;: 4-line display mode, NW = &quot;0&quot;: 1-line or 2-line display mode.</td>
<td>37 µs</td>
</tr>
<tr>
<td>Instruction Description</td>
<td>Instruction Code</td>
<td>Execution Time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------------</td>
<td>------------------</td>
<td>----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cursor or Display Shift</td>
<td>RS R/W DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0</td>
<td>37 µs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shift Enable</td>
<td>1 0 0 0 0 0 0 1 DS4 DS3 DS2 DS1</td>
<td>37 µs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scroll Enable</td>
<td>1 0 0 0 0 0 0 1 HS4 HS3 HS2 HS1</td>
<td>37 µs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Function Set</td>
<td>0 0 0 0 0 1 DL N RE(0) DH REV</td>
<td>37 µs</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 0 0 0 0 0 1 DL N RE(1) BE 0</td>
<td>37 µs</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### (Table 6, continued)

<table>
<thead>
<tr>
<th>Instruction</th>
<th>RE</th>
<th>Instruction Code</th>
<th>Description</th>
<th>Execution Time (fosc = 270 KHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set CGRAM</td>
<td>0 0 0 0 1 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Set CGRAM address in address counter.</td>
<td>37 µs</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Set SEGRAM</td>
<td>1 0 0 0 1 X X AC3 AC2 AC1 AC0</td>
<td>Set SEGRAM address in address counter.</td>
<td>37 µs</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Set DDRAM</td>
<td>0 0 0 1 AC6 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Set DDRAM address in address counter.</td>
<td>37 µs</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Set Scroll</td>
<td>1 0 0 1 X SQ5 SQ4 SQ3 SQ2 SQ1 SQ0</td>
<td>Set the quantity of horizontal dot scroll.</td>
<td>37 µs</td>
<td></td>
</tr>
<tr>
<td>Quantity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Read Busy</td>
<td>X 0 1 BF AC6 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Can be known whether during internal operation or not by reading BF. The</td>
<td>0 µs</td>
<td></td>
</tr>
<tr>
<td>flag and Address</td>
<td></td>
<td></td>
<td>contents of address counter can also be read.</td>
<td></td>
</tr>
<tr>
<td>Write Data</td>
<td>X 1 0 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>Write data into internal RAM (DDRAM / CGRAM / SEGRAM).</td>
<td>43 µs</td>
<td></td>
</tr>
<tr>
<td>Read Data</td>
<td>X 1 1 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>Read data from internal RAM (DDRAM / CGRAM / SEGRAM).</td>
<td>43 µs</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* ‘X’ : don’t care
1) Display Clear

Clear all the display data by writing \( \text{"20H"} \) (space code) to all DDRAM address, and set DDRAM address to \( \text{"00H"} \) into AC (address counter). Return cursor to the original status, namely, bring the cursor to the left edge on first line of the display. Make entry mode increment (I/D = "1").

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>DB7</th>
<th>DB6</th>
<th>DB5</th>
<th>DB4</th>
<th>DB3</th>
<th>DB2</th>
<th>DB1</th>
<th>DB0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

2) Return Home : (RE = 0)

Return Home is cursor return home instruction. Set DDRAM address to "00H" into the address counter. Return cursor to its original site and return display to its original status, if shifted. Contents of DDRAM does not change.

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>DB7</th>
<th>DB6</th>
<th>DB5</th>
<th>DB4</th>
<th>DB3</th>
<th>DB2</th>
<th>DB1</th>
<th>DB0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

3) Power Down Mode Set : (RE = 1)

Power down mode enable bit set instruction.
When PD = "High", it makes KS0078 suppress current consumption except the current needed for data storage by executing next three functions.
1. make the output value of all the COM/SEG ports VDD
2. make the COM/SEG output value of extension driver VDD by setting D output to "High" and M output to "Low"
3. disable voltage converter to remove the current through the divide resistor of power supply.
You can use this instruction as power sleep mode.
When PD = "Low", power down mode becomes disabled.

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>DB7</th>
<th>DB6</th>
<th>DB5</th>
<th>DB4</th>
<th>DB3</th>
<th>DB2</th>
<th>DB1</th>
<th>DB0</th>
<th>PD</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
4) Entry Mode Set

(1) (RE = 0)

Set the moving direction of cursor and display.
I/D : Increment / decrement of DDRAM address (cursor or blink)
When I/D = "High", cursor/blink moves to right and DDRAM address is increased by 1.
When I/D = "Low", cursor/blink moves to left and DDRAM address is decreased by 1.
*CGRAM/SEGRAM operates the same as DDRAM, when read from or write to CGRAM/SEGRAM.

When S = "High", after DDRAM write, the display of enabled line by DS1 - DS4 bits in the Shift Enable instruction is shifted to the right (I/D = "0") or to the left (I/D = "1"). But it will seem as if the cursor does not move.
When S = "Low", or DDRAM read, or CGRAM/SEGRAM read/write operation, shift of display like this function is not performed.

(2) (RE = 1)

Set the data shift direction of segment in the application set.
BID : Data Shift Direction of Segment
When BID = "Low", segment data shift direction is set to normal order from SEG1 to SEG120.
When BID = "High", segment data shift direction is set to reversely from SEG80 to SEG1.
By using this instruction, you can raise the efficiency of application board area.
* The BID setting instruction is recommended to be set at the same time level of function set instruction.
* DB1 bit must be set to "1".
5) Display ON/OFF Control ( RE = 0 )

```
RS  R/W  DB7  DB6  DB5  DB4  DB3  DB2  DB1  DB0
   0   0   0   0   0   0   1   0   1   1
```

Control display/cursor/blink ON/OFF 1 bit register.
D : Display ON/OFF control bit
When D = "High", display is turned on.
When D = "Low", display is turned off, but display data is remained in DDRAM.

C : Cursor ON/OFF control bit
When C = "High", cursor is turned on.
When C = "Low", cursor is disappeared in current display, but I/D register remains its data.

B : Cursor Blink ON/OFF control bit
When B = "High", cursor blink is on, that performs alternate between all the high data and display character at the cursor position. If fosc has 270 Khz frequency, blinking has 370 ms interval.
When B = "Low", blink is off.

6) Extended Function Set ( RE = 1 )

```
RS  R/W  DB7  DB6  DB5  DB4  DB3  DB2  DB1  DB0
   0   0   0   0   0   0   0   0   0   1
```

FW : Font Width control
When FW = "High", display character font width is assigned to 6-dot and execution time becomes 6/5 times than that of 5-dot font width.
The user font, specified in CGRAM, is displayed into 6-dot font width, bit-5 to bit-0,including the leftmost space bit of CGRAM.(refer to Fig-9)
When FW = "Low", 5-dot font width is set.

B/W : Black/White Inversion enable bit
When B/W = "High", black/white inversion at the cursor position is set. In this case C/B bit of display ON/OFF control instruction becomes don't care condition. If fosc has frequency of 270 Khz, inversion has 370 ms intervals.

NW : 4 Line mode enable bit
When NW = "High", 4 line display mode is set. In this case N bit of function set instruction becomes don't care condition.
7) Cursor or Display Shift (RE = 0)

Without writing or reading of display data, shift right/left cursor position or display.
This instruction is used to correct or search display data.(Refer to Table 7)
During 2-line mode display, cursor moves to the 2nd line after 48th digit of 1st line.
When 4-line mode, cursor moves to the next line, only after every 24th digit of the current line.
Note that display shift is performed simultaneously in all the line enabled by DS1 - DS4 in the Shift Enable instruction.
When displayed data is shifted repeatedly, each line shifted individually.
When display shift is performed, the contents of address counter are not changed.
During low power consumption mode, display shift may not be performed normally.

Table 7. Shift patterns according to S/C and R/L bits

<table>
<thead>
<tr>
<th>S/C</th>
<th>R/L</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Shift cursor to the left, ADDRESS COUNTER is decreased by 1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Shift cursor to the right, ADDRESS COUNTER is increased by 1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Shift all the display to the left, cursor moves according to the display</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Shift all the display to the right, cursor moves according to the display</td>
</tr>
</tbody>
</table>
8) Shift/Scroll Enable (RE = 1)

(1) (DH = 0)

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>DB7</th>
<th>DB6</th>
<th>DB5</th>
<th>DB4</th>
<th>DB3</th>
<th>DB2</th>
<th>DB1</th>
<th>DB0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>HS4</td>
<td>HS3</td>
<td>HS2</td>
<td>HS1</td>
</tr>
</tbody>
</table>

HS : Horizontal Scroll per Line Enable
This instruction makes valid dot shift by a display line unit.
HS1, HS2, HS3 and HS4 indicate each line to be dot scrolled, and each scroll is performed individually in each line.
If you want to scroll the line in 1-line display mode or the 1st line in 2-line display mode, set HS1 and HS2 to "High". If the 2nd line scroll is needed in 2-line mode, set HS3 and HS4 to "High". (refer to Table 8)

(2) (DH = 1)

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>DB7</th>
<th>DB6</th>
<th>DB5</th>
<th>DB4</th>
<th>DB3</th>
<th>DB2</th>
<th>DB1</th>
<th>DB0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>DS4</td>
<td>DS3</td>
<td>DS2</td>
<td>DS1</td>
</tr>
</tbody>
</table>

DS : Display Shift per Line Enable
This instruction selects shifting line to be shifted according to each line mode in display shift right/left instruction.
DS1, DS2, DS3 and DS4 indicate each line to be shifted, and each shift is performed individually in each line.
If you set DS1 and DS2 to "High" (enable) in 2 line mode, only the 1st line is shifted and the 2nd line is not shifted. When only DS1 = "High", only the half of the 1st line is shifted. If all the DS bits (DS1 to DS4) are set to "Low" (disable), no display is shifted.

Table 8. Relationship between DS and COM signal

<table>
<thead>
<tr>
<th>Enable bit</th>
<th>Enabled common signals during shift</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HS1/DS1</td>
<td>COM1, COM8</td>
<td>The part of display line that corresponds to enabled common signal can be shifted.</td>
</tr>
<tr>
<td>HS2/DS2</td>
<td>COM9, COM16</td>
<td></td>
</tr>
<tr>
<td>HS3/DS3</td>
<td>COM17, COM24</td>
<td></td>
</tr>
<tr>
<td>HS4/DS4</td>
<td>COM25, COM32</td>
<td></td>
</tr>
</tbody>
</table>
9) Function Set

(1) (RE = 0)

<table>
<thead>
<tr>
<th>RS</th>
<th>S/W</th>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
<th>RE</th>
<th>DH</th>
<th>REV</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>DL</td>
<td>N</td>
<td>RE</td>
<td>(0)</td>
<td>D4</td>
<td>D6</td>
<td>D8</td>
<td>D0</td>
</tr>
</tbody>
</table>

**DL**: Interface data length control bit
- When DL = “High”, it means 8-bit bus mode with MPU.
- When DL = “Low”, it means 4-bit bus mode with MPU. So to speak, DL is a signal to select 8-bit or 4-bit bus mode.
- When 4-bit bus mode, it needs to transfer 4-bit data by two times.

**N**: Display line number control bit
- It is variable only when NW bit of extended function set instruction is Low.
- When N = “Low”, it means 1-line display mode.
- When N = “High”, 2-line display mode is set.
- When NW = “High”, N bit is invalid, it means 4-line mode independent of N bit.

**RE**: Extended function registers enable bit
- At this instruction, RE must be “Low”.

**DH**: Display shift enable selection bit.
- When DH = “High”, display shift per line becomes enable.
- When DH = “Low”, smooth dot scroll becomes enable.
- This bit can be accessed only when IE pin input is “High”.

**REV**: Reverse enable bit
- When REV = “High”, all the display datas are reversed. Namely, all the white dots become black and black dots become white.
- When REV = “Low”, the display mode set normal display.
(2) (RE = 1)

DL : Interface data length control bit
When DL = "High", it means 8-bit bus mode with MPU.
When DL = "Low", it means 4-bit bus mode with MPU. So to speak, DL is a signal to select
8-bit or 4-bit bus mode.
When 4-bit bus mode, it needs to transfer 4-bit data by two times.

N : Display line number control bit
It is variable only when NW bit of extended function set instruction is Low.
When N = "Low", it means 1-line display mode.
When N = "High", 2-line display mode is set.
When NW = "High", N bit is invalid, it means 4-line mode independent of N bit.

RE : Extended function registers enable bit
When RE = "High", extended function set registers, SEGRAM address set registers, BID bit, HS/DS bits of
shift/scroll enable instruction and BE bits of function set register can be accessed.

BE : CGRAM/SEGRAM data blink enable bit
If BE is "High", it makes user font of CGRAM and segment of SEGRAM blinkable. The quantity of blink is
assigned the highest 2 bit of CGRAM/SEGRAM.

10) Set CGRAM Address (RE = 0)

Set CGRAM address to AC.
This instruction makes CGRAM data available from MPU.

11) Set SEGRAM Address (RE = 1)

Set CGRAM address to AC.
This instruction makes CGRAM data available from MPU.
12) Set DDRAM Address (RE = 0)

Set DDRAM address to AC.
This instruction makes DDRAM data available from MPU.
When 1-line display mode (N = 0, NW = 0), DDRAM address is from "00H" to "5FH".
In 2-line display mode (N = 1, NW = 0), DDRAM address in the 1st line is from "00H" to "2FH",
and DDRAM address in the 2nd line is from "40H" to "6FH".
In 4-line display mode (NW = 1), DDRAM address is from "00H" to "13H" in the 1st line, from
"20H" to "37H" in the 2nd line, from "40H" to "57H" in the 3rd line and from "60H" to "77H"
in the 4th line.

13) Set Scroll Quantity (RE = 1)

As set SQ5 to SQ0, horizontal scroll quantity can be controlled in dot units. (Refer to Table 9).
In this case KS0078 can show hidden areas of DDRAM by executing smooth scroll
from 1 to 48 dots.

<table>
<thead>
<tr>
<th>SQ5</th>
<th>SQ4</th>
<th>SQ3</th>
<th>SQ2</th>
<th>SQ1</th>
<th>SQ0</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>No shift</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>shift left by 1-dot</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>shift left by 2-dot</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>shift left by 3-dot</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>shift left by 47-dot</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>shift left by 48-dot</td>
</tr>
</tbody>
</table>
14) Read Busy Flag & Address

This instruction shows whether KS0078 is in internal operation or not. If the resultant BF is High, it means the internal operation is in progress and you have to wait until BF to be Low, and then the next instruction can be performed. In this instruction you can read also the value of address counter.

15) Write data to RAM

Write binary 8-bit data to DDRAM/CGRAM/SEGRAM.
The selection of RAM from DDRAM, CGRAM, or SEGARM, is set by the previous address set instruction: DDRAM address set, CGRAM address set, SEGARM address set.
RAM set instruction can also determines the AC direction to RAM.
After write operation, the address is automatically increased/decreased by 1, according to the entry mode.

16) Read data from RAM

Read binary 8-bit data from DDRAM/CGRAM/SEGRAM.
If address set instruction of RAM is not performed before this instruction, the data that read first is invalid, because the direction of AC is not determined. If you read RAM data several times without RAM address set instruction before read operation, you can get correct RAM data from the second, but the first data would be incorrect, because there is no time margin to transfer RAM data. In case of DDRAM read operation, cursor shift instruction plays the same role as DDRAM address set instruction: it also transfer RAM data to output data register.
After read operation address counter is automatically increased/decreased by 1 according to the entry mode. After CGRAM/SEGRAM read operation, display shift may not be executed correctly.

* In case of RAM write operation, after this AC is increased/decreased by 1 like read operation. In this time, AC indicates the next address position, but you can read only the previous data by read instruction.
## Table 10. Instruction Set 2

<table>
<thead>
<tr>
<th>Instruction</th>
<th>RE</th>
<th>Instruction Code</th>
<th>Description</th>
<th>Execution Time (fosc = 270KHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear Display</td>
<td>X</td>
<td>0 0 0 0 0 0 0 0 1</td>
<td>Write &quot;20H&quot; to DDRAM and set DDRAM address to &quot;00H&quot; from AC.</td>
<td>1.52ms</td>
</tr>
<tr>
<td>Return Home</td>
<td>X</td>
<td>0 0 0 0 0 0 0 0 1 X</td>
<td>Set DDRAM address to &quot;00H&quot; from AC and return cursor to its original position if shifted. The contents of DDRAM are not changed.</td>
<td>1.52ms</td>
</tr>
<tr>
<td>Entry Mode Set</td>
<td>X</td>
<td>0 0 0 0 0 0 0 0 1 I/D S</td>
<td>Assign cursor moving direction. I/D = &quot;1&quot; : increment, I/D = &quot;0&quot; : decrement, and display shift enable bit. S = &quot;1&quot; : make entire display shift of all lines during DDRAM write, S = &quot;0&quot; : display shift disable</td>
<td>37 µs</td>
</tr>
<tr>
<td>Display ON/OFF Control</td>
<td>0 0 0 0 0 0 1 D C B</td>
<td>Set display/cursor/blink on/off D = &quot;1&quot; : display on, D = &quot;0&quot; : display off, C = &quot;1&quot; : cursor on, C = &quot;0&quot; : cursor off, B = &quot;1&quot; : blink on, B = &quot;0&quot; : blink off.</td>
<td>37 µs</td>
<td></td>
</tr>
<tr>
<td>Extended function set</td>
<td>1 0 0 0 0 0 0 0 1 FW B/W NW</td>
<td>Assign font width, black/white inverting of cursor, and 4-line display mode control bit. FW = &quot;1&quot; : 6-dot font width, FW = &quot;0&quot; : 5-dot font width, B/W = &quot;1&quot; : black/white inverting of cursor enable, B/W = &quot;0&quot; : black/white inverting of cursor disable NW = &quot;1&quot; : 4-line display mode, NW = &quot;0&quot; : 1-line or 2-line display mode</td>
<td>37 µs</td>
<td></td>
</tr>
<tr>
<td>Cursor or Display Shift</td>
<td>0 0 0 0 0 0 1 S/C R/L X X</td>
<td>Cursor or display shift. S/C = &quot;1&quot; : display shift, S/C = &quot;0&quot; : cursor shift, R/L = &quot;1&quot; : shift to right, R/L = &quot;0&quot; : shift to left</td>
<td>37 µs</td>
<td></td>
</tr>
</tbody>
</table>
(Table 10. continued)

<table>
<thead>
<tr>
<th>Instruction</th>
<th>RE</th>
<th>Instruction Code</th>
<th>Description</th>
<th>Execution Time (fosc = 270 kHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Scroll Enable</strong></td>
<td>1</td>
<td>0 0 0 0 0 1 HS4 HS3 HS2 HS1</td>
<td>Determine the line for horizontal smooth scroll. ( \text{HS1} = &quot;1/0&quot; : 1\text{st line dot scroll enable/disble}) ( \text{HS2} = &quot;1/0&quot; : 2\text{nd line dot scroll enable/disble}) ( \text{HS3} = &quot;1/0&quot; : 3\text{rd line dot scroll enable/disble}) ( \text{HS4} = &quot;1/0&quot; : 4\text{th line dot scroll enable/disble})</td>
<td>37 (\mu s)</td>
</tr>
<tr>
<td><strong>Function Set</strong></td>
<td>0</td>
<td>0 0 0 0 1 DL N RE(0) X X</td>
<td>Set interface data length. ( \text{DL} = &quot;1&quot; : 8\text{-bit}) ( \text{DL} = &quot;0&quot; : 4\text{-bit}) numbers of display line when NW = &quot;0&quot;, ( N = &quot;1&quot; : 2\text{-line}) ( N = &quot;0&quot; : 1\text{-line}) extension register, RE(&quot;0&quot;)</td>
<td>37 (\mu s)</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0 0 0 0 1 DL N RE(1) BE 0</td>
<td>Set DL, N, RE(&quot;1&quot;) and CGRAM/SEGAM blink enable (BE) ( \text{BE} = &quot;1/0&quot; : \text{CGRAM/SEGAM blink enable/disble})</td>
<td>37 (\mu s)</td>
</tr>
<tr>
<td><strong>Set CGRAM Address</strong></td>
<td>0</td>
<td>0 0 0 1 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Set CGRAM address in address counter.</td>
<td>37 (\mu s)</td>
</tr>
<tr>
<td><strong>Set SEGRAM Address</strong></td>
<td>1</td>
<td>0 0 0 1 X X AC3 AC2 AC1 AC0</td>
<td>Set SEGRAM address in address counter.</td>
<td>37 (\mu s)</td>
</tr>
<tr>
<td><strong>Set DDRAM Address</strong></td>
<td>0</td>
<td>0 0 1 AC6 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Set DDRAM address in address counter.</td>
<td>37 (\mu s)</td>
</tr>
<tr>
<td><strong>Set Scroll Quantity</strong></td>
<td>1</td>
<td>0 0 1 X QC5 QC4 QC3 QC2 QC1 QC0</td>
<td>Set the quantity of horizontal dot scroll.</td>
<td>37 (\mu s)</td>
</tr>
<tr>
<td><strong>Read Busy flag and Address</strong></td>
<td>X</td>
<td>0 1 BF AC6 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Can be known whether during internal operation or not by reading BF. The contents of address counter can also be read. ( BF = &quot;1&quot; : \text{busy state}) ( BF = &quot;0&quot; : \text{ready state})</td>
<td>0 (\mu s)</td>
</tr>
<tr>
<td><strong>Write Data</strong></td>
<td>X</td>
<td>1 0 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>Write data into internal RAM (DDRAM / CGRAM / SEGRAM).</td>
<td>43 (\mu s)</td>
</tr>
<tr>
<td><strong>Read Data</strong></td>
<td>X</td>
<td>1 1 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>Read data from internal RAM (DDRAM / CGRAM / SEGRAM).</td>
<td>43 (\mu s)</td>
</tr>
</tbody>
</table>

\* "X" : don't care
1) Display Clear

```
RS  R/W  DB7  DB6  DB5  DB4  DB3  DB2  DB1  DB0
0   0    0    0    0    0    0    0    1    
```

Clear all the display data by writing "20H" (space code) to all DDRAM address, and set DDRAM address to "00H" into AC (address counter). Return cursor to the original status, namely, bring the cursor to the left edge on first line of the display. And entry mode is set to increment mode (I/D = "1").

2) Return Home

```
RS  R/W  DB7  DB6  DB5  DB4  DB3  DB2  DB1  DB0
0   0    0    0    0    0    0    1    0    x
```

Return Home is cursor return home instruction. Set DDRAM address to "00H" into the address counter. Return cursor to its original site and return display to its original status, if shifted. Contents of DDRAM does not change.

3) Entry Mode Set

```
RS  R/W  DB7  DB6  DB5  DB4  DB3  DB2  DB1  DB0
0   0    0    0    0    0    1    1/D  1    
```

Set the moving direction of cursor and display. I/D : Increment / decrement of DDRAM address (cursor or blink)

When I/D = "High", cursor/blink moves to right and DDRAM address is increased by 1.
When I/D = "Low", cursor/blink moves to left and DDRAM address is decreased by 1.

* CGRAM/SEGGRAM operates the same as DDRAM, when read from or write to CGRAM/SEGGRAM.

When S = "High", after DDRAM write, the entire display of all lines is shifted to the right (I/D = "0") or to the left (I/D = "1"). But it will seem as if the cursor does not move.
When S = "Low", or DDRAM read, or CGRAM/SEGGRAM read/write operation, shift of entire display is not performed.
4) Display ON/OFF Control (RE = 0)

Control display/cursor/blink ON/OFF 1 bit register.

D: Display ON/OFF control bit
   When D = "High", entire display is turned on.
   When D = "Low", display is turned off, but display data is remained in DDRAM.

C: Cursor ON/OFF control bit
   When C = "High", cursor is turned on.
   When C = "Low", cursor is disappeared in current display, but I/D register remains its data.

B: Cursor Blink ON/OFF control bit
   When B = "High", cursor blink is on, that performs alternate between all the high data and display character at the cursor position. If fosc has 270 kHz frequency, blinking has 370 ms interval.
   When B = "Low", blink is off.

5) Extended Function Set (RE = 1)

FW: Font Width control
   When FW = "High", display character font width is assigned to 6-dot and execution time becomes 6/5 times than that of 5-dot font width.
   The user font, specified in CGRAM, is displayed into 6-dot font width, bit-5 to bit-0, including the leftmost space bit of CGRAM. (Refer to Fig-10)
   When FW = "Low", 5-dot font width is set.

B/W: Black/White Inversion enable bit
   When B/W = "High", black/white inversion at the cursor position is set. In this case C/B bit of display ON/OFF control instruction becomes don't care condition. If fosc has frequency of 270 kHz, inversion has 370 ms intervals.

NW: 4 Line mode enable bit
   When NW = "High", 4 line display mode is set. In this case N bit of function set instruction becomes don't care condition.
6) Cursor or Display Shift (RE = 0)

Without writing or reading of display data, shift right/left cursor position or display. 
This instruction is used to correct or search display data. (Refer to Table 7)
During 2-line mode display, cursor moves to the 2nd line after 48th digit of 1st line.
When 4-line mode, cursor moves to the next line, only after every 24th digit of the current line.
Note that display shift is performed simultaneously in all the line.
When displayed data is shifted repeatedly, each line shifted individually.
When display shift is performed, the contents of address counter are not changed.

Table 11. Shift patterns according to S/C and R/L bits

<table>
<thead>
<tr>
<th>S/C</th>
<th>R/L</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>Shift cursor to the left, ADDRESS COUNTER is decreased by 1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>Shift cursor to the right, ADDRESS COUNTER is increased by 1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>Shift all the display to the left, cursor moves according to the display</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>Shift all the display to the right, cursor moves according to the display</td>
</tr>
</tbody>
</table>
7) Scroll Enable (RE = 1)

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>DB7</th>
<th>DB6</th>
<th>DB5</th>
<th>DB4</th>
<th>DB3</th>
<th>DB2</th>
<th>DB1</th>
<th>DB0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>HS1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HS2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HS3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>HS4</td>
</tr>
</tbody>
</table>

HS : Horizontal Scroll per Line Enable
This instruction makes valid dot shift by a display line unit.
HS1, HS2, HS3 and HS4 indicate each line to be dot scrolled, and each scroll is performed individually in each line.
If you want to scroll the line in 1-line display mode or the 1st line in 2-line display mode, set HS1 and HS2 to "High". If the 2nd line scroll is needed in 2-line mode, set HS3 and HS4 to "High". (refer to Table 8)

8) Function Set
(1) (RE = 0)

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>DB7</th>
<th>DB6</th>
<th>DB5</th>
<th>DB4</th>
<th>DB3</th>
<th>DB2</th>
<th>DB1</th>
<th>DB0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>DL</td>
<td>N</td>
<td>RE</td>
<td>0</td>
</tr>
</tbody>
</table>

DL : Interface data length control bit
When DL = "High", it means 8-bit bus mode with MPU.
When DL = "Low", it means 4-bit bus mode with MPU.
So to speak, DL is a signal to select 8-bit or 4-bit bus mode.
When 4-bit bus mode, it needs to transfer 4-bit data by two times.

N : Display line number control bit
It is variable only when NW bit of extended function set instruction is Low.
When N = "Low", it means 1-line display mode.
When N = "High", 2-line display mode is set.
When NW = "High", N bit is invalid, it means 4-line mode independent of N bit.

RE : Extended function registers enable bit
At this instruction, RE must be "Low".
DL : Interface data length control bit
   When DL = "High", it means 8-bit bus mode with MPU.
   When DL = "Low", it means 4-bit bus mode with MPU.
   So to speak, DL is a signal to select 8-bit or 4-bit bus mode.
   When 4-bit bus mode, it needs to transfer 4-bit data by two times.

N : Display line number control bit
   It is variable only when NW bit of extended function set instruction is Low.
   When N = "Low", it means 1-line display mode.
   When N = "High", 2-line display mode is set.
   When NW = "High", N bit is invalid, it means 4-line mode independent of N bit.

RE : Extended function registers enable bit
   When RE = "High", extended function set registers, SEGRAM address set registers, HS bits of scroll enable instruction and BE bits of function set register can be accessed.

BE : CGRAM/SEGRAM data blink enable bit
   If BE is "High", it makes user font of CGRAM and segment of SEGRAM blinkable. The quantity of blink is assigned the highest 2 bit of CGRAM/SEGRAM.

9) Set CGRAM Address (RE = 0)
   
   RS R/W DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0
   0 0 0 0 1 AC5 AC4 AC3 AC2 AC1 AC0

   Set CGRAM address to AC.
   This instruction makes CGRAM data available from MPU.

10) Set SEGRAM Address (RE = 1)
   
   RS R/W DB7 DB6 DB5 DB4 DB3 DB2 DB1 DB0
   0 0 0 1 - - AC3 AC2 AC1 AC0

   Set SEGRAM address to AC.
   This instruction makes SEGRAM data available from MPU.
11) Set DDRAM Address (RE = 0)

Set DDRAM address to AC.
This instruction makes DDRAM data available from MPU.
When 1-line display mode (N = 0, NW = 0), DDRAM address is from "00H" to "5FH".
In 2-line display mode (N = 1, NW = 0), DDRAM address in the 1st line is from "00H" to "2FH",
and DDRAM address in the 2nd line is from "40H" to "6FH".
In 4-line display mode (NW = 1), DDRAM address is from "00H" to "17H" in the 1st line, from "20H" to "37H" in the 2nd line, from "40H" to "57H" in the 3rd line and from "60H" to "77H"
in the 4th line.

12) Set Scroll Quantity (RE = 1)

As set SQ5 to SQ0, horizontal scroll quantity can be controlled in dot units. (Refer to Table 12). In this case KS0078 execute dot smooth scroll from 1 to 48 dots.

Table 12. Scroll quantity according to HDS bits

<table>
<thead>
<tr>
<th>SQ5</th>
<th>SQ4</th>
<th>SQ3</th>
<th>SQ2</th>
<th>SQ1</th>
<th>SQ0</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>No shift</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>shift left by 1-dot</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>shift left by 2-dot</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>shift left by 3-dot</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>shift left by 47-dot</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>shift left by 48-dot</td>
</tr>
</tbody>
</table>
13) Read Busy Flag & Address

This instruction shows whether KS0078 is in internal operation or not. If the resultant BF is High, it means the internal operation is in progress and you have to wait until BF to be Low, and then the next instruction can be performed. In this instruction you can read also the value of address counter.

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>BF</td>
<td>AC6</td>
<td>AC5</td>
<td>AC4</td>
<td>AC3</td>
<td>AC2</td>
<td>AC1</td>
<td>AC0</td>
</tr>
</tbody>
</table>

14) Write data to RAM

Write binary 8-bit data to DDRAM/CGRAM/SEGRAM.
The selection of RAM from DDRAM, CGRAM, or SEGRAM, is set by the previous address set instruction : DDRAM address set, CGRAM address set, SEGRAM address set. RAM set instruction can also determines the AC direction to RAM.
After write operation, the address is automatically increased/decreased by 1, according to the entry mode.

<table>
<thead>
<tr>
<th>RS</th>
<th>R/W</th>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>D7</td>
<td>D6</td>
<td>D5</td>
<td>D4</td>
<td>D3</td>
<td>D2</td>
<td>D1</td>
<td>D0</td>
</tr>
</tbody>
</table>

15) Read data from RAM

Read binary 8-bit data from DDRAM/CGRAM/SEGRAM.
The selection of RAM is set by the previous address set instruction. If address set instruction of RAM is not performed before this instruction, the data that read first is invalid, because the direction of AC is not determined. If you read RAM data several times without RAM address set instruction before read operation, you can get correct RAM data from the second, but the first data would be incorrect, because there is no time margin to transfer RAM data. In case of DDRAM read operation, cursor shift instruction plays the same role as DDRAM address set instruction : it also transfer RAM data to output data register. After read operation address counter is automatically increased/decreased by 1 according to the entry mode.
After CGRAM/SEGRAM read operation, display shift may not be executed correctly.

* In case of RAM write operation, after this AC is increased/decreased by 1 like read operation. In this time, AC indicates the next address position, but you can read only the previous data by read instruction.
KS0078 can transfer data in bus mode (4-bit or 8-bit) or serial mode with MPU. So you can use any type 4 or 8-bit MPU. In case of 4-bit bus mode, data transfer is performed by two times to transfer 1 byte data.

(1) When interfacing data length are 4-bit, only 4 ports, from DB4 to DB7, are used as data bus. At first higher 4-bit (in case of 8-bit bus mode, the contents of DB4 - DB7) are transferred, and then lower 4-bit (in case of 8-bit bus mode, the contents of DB0 - DB3) are transferred. So transfer is performed by two times. Busy Flag outputs "High" after the second transfer are ended.

(2) When interfacing data length are 8-bit, transfer is performed at a time through 8 ports, from DB0 to DB7.

(3) If IM is set to "Low", serial transfer mode is set.
Interface with MPU in Bus Mode

1) Interface with 8-bits MPU
If 8-bits MPU is used, KS0078 can connect directly with that.
In this case, port E, RS, R/W and DB0 to DB7 need to interface each other.
Example of timing sequence is shown below.

Fig 11. Example of 8-bit Bus Mode Timing Sequence

2) Interface with 4-bits MPU
If 4-bits MPU is used, KS0078 can connect directly with this.
In this case, port E, RS, R/W and DB4 to DB7 need to interface each other. The transfer is performed by two times.
Example of timing sequence is shown below.

Fig 12. Example of 4-bit Bus Mode Timing Sequence
Interface with MPU in Serial Mode

When IM port input is “Low”, serial interface mode is started. At this time, all three ports, SCLK (synchronizing transfer clock), SID (serial input data), and SOD (serial output data), are used. If you want to use KS0078 with other chips, chip select port (CS) can be used. By setting CS to “Low”, KS0078 can receive SCLK input. If CS is set to “High”, KS0078 reset the internal transfer counter.

Before transfer real data, start byte has to be transferred. It is composed of succeeding 5 “High” bits, register selection bit (RS), read write control bit (R/W), and end bit that indicates the end of start byte. Whenever succeeding 5 “High” bits are detected by KS0078, it makes serial transfer counter reset and ready to receive next informations.

The next input data are register selection bit that determine which register will be used, and read write control bit that determine the direction of data. Then end bit is transferred, which must have “Low” value to show the end of start byte. (Refer to Fig 13, Fig 14)

(1) Write Operation (R/W = 0)
After start byte is transferred from MPU to KS0078, 8-bit data is transferred which is divided into 2 bytes, each byte has 4 bit's real data and 4 bit's partition token data. For example, if real data is “10110001” (D0 – D7), then serially transferred data becomes “1011 0000 0001 0000” where 2nd and 4th 4 bits must be “0000” for safe transfer.
To transfer several bytes continuously without changing RS bit and RW bit, start byte transfer is needed only at first starting time. Namely, after first start byte is transferred, real data can be transferred successively.

(2) Read Operation (R/W = 1)
After start byte is transferred to KS0078, MPU can receive 8-bit data through the SOD port at a time from the LSB. Wait time is needed to insert between start byte and data reading, because internal reading from RAM requires some delay. Continuous data reading is possible like serial write operation. It also needs only one start bytes, only if you insert some delay between reading operations of each byte. During the reading operation, KS0078 observes succeeding 5 “High” from MPU. If it is detected, KS0078 restarts serial operation at once and ready to receive RS bit. So in continuous reading operation, SID port must be “0”.
Fig 13. Timing Diagram of Serial Data Transfer

(1) Serial Write Operation

CS (Input)  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
SCLK (Input)  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
SID (Input)  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Starting Byte  Synchronizing
Bit string  Instruction
Lower Data  Upper Data

(2) Serial Read Operation

CS (Input)  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
SCLK (Input)  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
SID (Input)  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
SOD (Output)  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Starting Byte  Synchronizing
Bit string
Busy Flag/
Read Data
Lower Data  Upper Data

D0 D1 D2 D3 D4 D5 D6 D7
Invalid Data

S0 S1 S2 S3 S4 S5 S6 S7
Starting Byte

S0 S1 S2 S3 S4 S5 S6 S7
Read Data

S0 S1 S2 S3 S4 S5 S6 S7
Busy Flag/
Read Data

S0 S1 S2 S3 S4 S5 S6 S7
Lower Data  Upper Data

S0 S1 S2 S3 S4 S5 S6 S7
Starting Byte

S0 S1 S2 S3 S4 S5 S6 S7
Synchronizing
Bit string

S0 S1 S2 S3 S4 S5 S6 S7
Synchronizing
Bit string
Fig 14. Timing Diagram of Continuous Data Transfer

(1) Continuous Write Operation

SCLK: Start
SID: Instruction1

Wait
Start byte
1st byte
2nd byte
Instruction1
execution time

Wait
1st byte
2nd byte
Instruction2
execution time

Wait
1st byte
2nd byte
Instruction3
execution time

(2) Continuous Read Operation

SCLK: Start
SID: Instruction1

Wait
Start byte

Wait
Data read1

Wait
Data read2

Wait
Data read3

Instruction1
execution time

Instruction2
execution time

Instruction3
execution time
APPLICATION INFORMATION ACCORDING TO LCD PANEL

1) LCD Panel : 48 character x 1 line format (5-dot font, 1/17 duty)

2) LCD Panel : 48 character x 2 line format (5-dot font, 1/33 duty)
3) LCD Panel: 24 character x 4 line format (5-dot font, 1/33 bias)
4) LCD Panel: 20 character x 4 line format (6-dot font, 1/33 bias)
INITIALIZING

1) Initializing by Internal Reset Circuit
When the power is turned on, KS0078 is initialized automatically by power on reset circuit.
During the initialization, the following instructions are executed, and BF(Busy Flag) is kept "High"(busy state) to the end of initialization.
   (1) Display Clear instruction
   Write "20H" to all DDRAM
   (2) Set Functions instruction
       DL = 1 : 8-bit bus mode
       N = 1 : 2-line display mode
       RE = 0 : Extension register disable
       BE = 0 : CGRAM/SEGRAM blink OFF
       DH = 0 : Horizontal scroll enable
       REV = 0 : Normal display (Not reversed display)
   (3) Control Display ON/OFF instruction
       D = 0 : Display OFF
       C = 0 : Cursor OFF
       B = 0 : Blink OFF
   (4) Set Entry Mode instruction
       I/D = 1 : Increment by 1
       S = 0 : No entire display shift
       BID = 0 : Normal direction segment port
   (5) Set Extension Function instruction
       FW = 0 : 5-dot font width character display
       B/W = 0 : Normal cursor (8th line)
       NW = 0 : Not 4-line display mode, 2-line mode is set because of N("1")
   (6) Enable Shift instruction
       HS = 0000 : Scroll per line disable
       DS = 0000 : Shift per line disable
   (7) Set scroll Quantity instruction
       SQ = 000000 : Not scroll

2) Initializing by Hardware RESET input
When RESET pin = "Low", KS0078 can be initialized like the case of power on reset.
During the power on reset operation, this pin is ignored.