Main Features

- Operation Voltage Range: 4.5V to 5.5V
- Support 8-bit, 4-bit and serial bus MPU interface
- 64 x 16-bit display RAM (DDRAM)
- LCD display range 16 words x 2 lines
- 64 x 256-bit Graphic Display RAM (GDRAM)
- Built-in voltage booster (2 times)
- Built-in RC oscillator: Frequency is adjusted by an external resistor
- Automatic power on reset (POR)
- External reset pin (XRESET)
- Multiple instructions:
  - Display Clear
  - Return Home
  - Display ON/OFF
  - Cursor ON/OFF
  - Display Character Blink
  - Display Shift
  - Vertical Line Scroll
  - Reverse Display (by line)
  - Standby Mode
- Support 8192 Chinese words (16x16 dot matrix)
- Supports 16 words x 4 lines (Max)
- LCD display range 16 words x 2 lines
- 16K-bit half-width Character Generation ROM (HCGROM):
- Supports 126 characters (16x8 dot matrix)
- 2M-bits Character Generation ROM (CGROM):
- Mixed-mode display with both character and graphic data is possible
- ST7920 has built-in CGRAM and provide 4 sets software programmable 16x16 fonts.
- ST7920 has wide operating voltage range (2.7V to 5.5V). It also has low power consumption. So ST7920 is suitable for battery-powered portable device.
- ST7920 LCD driver consists of 32-common and 64-segment. Company with the extension segment driver (ST7921) ST7920 can support up to 32-common x 256-segment display.

Part Number | Font Code
-------------|-------------
ST7920-0A    | Big-5 Code Set (Traditional Chinese)
ST7920-0B    | GB Code Set (Simplified Chinese)
ST7920-0C    | Chinese (Traditional/Simplified) & Japanese
ST7920-0F    | Chinese (Traditional/Simplified), Japanese & Korean
System Block Diagram

- **Reset Circuit**
- **Instruction Register (IR)**
- **Instruction Decoder**
- **Display Data RAM (DDRAM) 64 x 16 bits**
- **33/49-bit shift register**
- **Common Signal Driver**
- **Timing Generator**
- **Segment Signal Driver**
- **64-bit latch circuit**
- **64-bit shift register**
- **LCD Drive**
- **Voltage Selector**
- **MPU Interface**
- **Instruction Register (IR)**
- **Address Counter**
- **Data Register (DR)**
- **Busy Flag**
- **Input/Output Buffer**
- **Data Register (DR)**
- **Parallel/Serial converter and Attribute Circuit**
- **Graphic RAM (GDRAM) 1024 x 16 bits**
- **Half size Character ROM (HCGROM) 1024x16 bits**
- **Character Generator RAM (CGRAM) 1024 bits**
- **Character Generator ROM (CGROM) 2M bits**
- **Cursor Blink Scroll Controller**
- **Character Generator ROM (CGROM) 1024x16 bits**
- **Half size Character ROM (HCGROM) 1024x16 bits**
- **Character Generator RAM (CGRAM) 1024 bits**

**Power Supply:**
- **VDD**
- **VSS**
- **XRESET**

**I/O Signals:**
- **RS**
- **RW**
- **E**
- **DB4 to DB7**
- **DB0 to DB3**
- **SEG1 to SEG64**
- **COM1 to COM32**
- **CL1 CL2 CL3**
- **CLK**
- **RESI RESO**
- **DOUT**
- **XOFF**
- **V0 V1 V2 V3 V4**
## Pin Description

<table>
<thead>
<tr>
<th>Name</th>
<th>No.</th>
<th>I/O</th>
<th>Connects to</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>RST</td>
<td>17</td>
<td>I</td>
<td>—</td>
<td>System reset input (low active).</td>
</tr>
<tr>
<td>PSB</td>
<td>15</td>
<td>I</td>
<td>—</td>
<td>Interface selection: 0: serial mode; 1: 8/4-bit parallel bus mode.</td>
</tr>
<tr>
<td>RS(CS*)</td>
<td>4</td>
<td>I</td>
<td>MPU</td>
<td><strong>Parallel Mode</strong>: Register select. 0: Select instruction register (write) or busy flag, address counter (read); 1: Select data register (write/read). <strong>Serial Mode</strong>: Chip select. 1: chip enabled; 0: chip disabled. When chip is disabled, SID and SCLK should be set as “H” or “L”. Transient of SID and SCLK is not allowed.</td>
</tr>
<tr>
<td>RW(SID*)</td>
<td>5</td>
<td>I</td>
<td>MPU</td>
<td><strong>Parallel Mode</strong>: Read/Write control. 0: Write; 1: Read. <strong>Serial Mode</strong>: Serial data input.</td>
</tr>
<tr>
<td>E(SCLK*)</td>
<td>6</td>
<td>I</td>
<td>MPU</td>
<td><strong>Parallel Mode</strong>: 1: Enable trigger. <strong>Serial Mode</strong>: Serial clock.</td>
</tr>
<tr>
<td>D4 to D7</td>
<td>11-14</td>
<td>I/O</td>
<td>MPU</td>
<td>Higher nibble data bus of 8-bit interface and data bus for 4-bit interface</td>
</tr>
<tr>
<td>D0 to D3</td>
<td>7-10</td>
<td>I/O</td>
<td>MPU</td>
<td>Lower nibble data bus of 8-bit interface.</td>
</tr>
<tr>
<td>VDD</td>
<td>2</td>
<td>I</td>
<td>Power</td>
<td>VDD : 4.5V to 5.5V.</td>
</tr>
<tr>
<td>Vss</td>
<td>1</td>
<td>I</td>
<td>Power</td>
<td>VSS: 0V.</td>
</tr>
<tr>
<td>VOUT</td>
<td>18</td>
<td>O</td>
<td>Resistors</td>
<td>LCD voltage doubler output. VOUT ≤ 7V.</td>
</tr>
</tbody>
</table>
Function Description

System interface
ST7920 supports 3 kinds of bus interface to communicate with MPU: 8-bit parallel, 4-bit parallel and clock synchronized serial interface. Parallel interface is selected by PSB=“1” and serial interface is by PSB=“0”. 8-bit / 4-bit interface is selected by function set instruction DL bit.

Two 8-bit registers (Data Register DR and Instruction Register IR) are used in ST7920 to access DRAM or Register. Data Register (DR) can access DDRAM, CGRAM and GDRAM through the address pointer implemented by Address Counter (AC). Instruction Register (IR) stores the instruction sent by MPU to ST7920.

4 kinds of parallel interface access mode can be selected through RS and RW:

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>L</td>
<td>MPU write instruction to instruction register (IR)</td>
</tr>
<tr>
<td>L</td>
<td>H</td>
<td>MPU read busy flag (BF) and address counter (AC)</td>
</tr>
<tr>
<td>H</td>
<td>L</td>
<td>MPU write data to data register (DR)</td>
</tr>
<tr>
<td>H</td>
<td>H</td>
<td>MPU read data from data register (DR)</td>
</tr>
</tbody>
</table>

* The serial interface access modes do not have Read operation.

Busy Flag (BF)
ST7920 needs a process time for any received instruction. Before finishing the received instruction, any further instruction is not accepted. The process time of each instruction is not equal and the internal process is finished or not can be determined by the BF. Internal operation is in progress while BF=“1”, that means ST7920 is in busy state. No further instructions will be accepted until BF=“0”. MPU must check BF to determine whether the internal operation is finished or not before issuing instruction.

Address Counter (AC)
Address Counter (AC) is used as the address pointer of DDRAM, CGRAM and GDRAM. (AC) can be set by instruction. After that, accesses (Read/Write operations) to the memories, such as DDRAM, CGRAM or GDRAM, (AC) will be increased or decreased by 1 (according to the setting in “Entry Mode Set” Register). When RS=“0”, RW=“1” and E=“1” the value of (AC) will be output to DB6~DB0.

Character Generation ROM (CGROM) and Half-width Character Generation ROM (HCGROM)
ST7920 is built in a Character Generation ROM (CGROM) to provide 8192 16x16 character fonts and a Half-width Character Generation ROM to provide 126 8x16 alphanumeric characters. It is easy to support multi-language applications such as Chinese and English. Two consecutive bytes are used to specify one 16x16 character or two 8x16 half-width characters. Character codes are written into DDRAM and the corresponding fonts are mapped from CGROM or HCGROM to the display drivers.

Character Generation RAM (CGRAM)
ST7920 is built in a Character Generation RAM (CGRAM) to support user-defined fonts. Four sets of 16x16 bit-maped RAM spaces are available. These user-defined fonts are displayed the same ways as CGROM fonts by writing the related character code into the DDRAM.
Display Data RAM (DDRAM)

There are 64x2 bytes RAM spaces for the Display Data RAM. It can store display data such as 16 characters (16x16) by 4 lines or 32 characters (8x16) by 4 lines. However, only 2 character-lines (maximum 32 common outputs) can be displayed at one time. Character codes stored in DDRAM will refer to the fonts specified by CGROM, HCGROM and CGRAM.

ST7920 can display half-width HCGROM fonts, user-defined CGRAM fonts and full 16x16 CGROM fonts. The character codes in 0000H~0006H will use user-defined fonts in CGRAM. The character codes in 02H~7FH will use half-width alpha numeric fonts. The character code larger than A1H will be treated as 16x16 fonts and will be combined with the next byte automatically. The 16x16 BIG5 fonts are stored in A140H~D75FH while the 16x16 GB fonts are stored in A1A0H~F7FFH. In short:

1. To display HCGROM fonts:
   - Write 2 bytes of data into DDRAM to display two 8x16 fonts. Each byte represents 1 character.
   - The data is among 02H~7FH.
2. To display CGRAM fonts:
   - Write 2 bytes of data into DDRAM to display one 16x16 font.
   - Only 0000H, 0002H, 0004H and 0006H are acceptable.
3. To display CGROM fonts:
   - Write 2 bytes of data into DDRAM to display one 16x16 font.
   - A140H~D75FH are BIG5 code, A1A0H~F7FFH are GB code.

The higher byte (D15~D8) is written first and the lower byte (D7~D0) is the next.

Please refer to Table 5 for the relationship between DDRAM and the address/data of CGRAM.

CGRAM fonts and CGROM fonts can only be displayed in the start position of each address. (Refer to Table 4)

<table>
<thead>
<tr>
<th>80</th>
<th>81</th>
<th>82</th>
<th>83</th>
<th>84</th>
<th>85</th>
<th>86</th>
<th>87</th>
<th>88</th>
<th>89</th>
<th>8A</th>
<th>8B</th>
<th>8C</th>
<th>8D</th>
<th>8E</th>
<th>8F</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
<td>H</td>
<td>L</td>
</tr>
<tr>
<td>Sitronix</td>
<td>ST7920</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>矽</td>
<td>創</td>
<td>電</td>
<td>子</td>
<td>.</td>
<td>.</td>
<td>中</td>
<td>文</td>
<td>編</td>
<td>碼</td>
<td>(</td>
<td>正</td>
<td>確</td>
<td>)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Incorrect start position

Table 4
Graphic RAM (GDRAM)

Graphic Display RAM has 64x256 bits bit-mapped memory space. GDRAM address is set by writing 2 consecutive bytes of vertical address and horizontal address. Two-byte data (16 bits) configures one GDRAM horizontal address. The Address Counter (AC) will be increased by one automatically after receiving the 16-bit data for the next operation. After the horizontal address reaching 0FH, the horizontal address will be set to 00H and the vertical address will not change. The procedure is summarized below:

1. Set vertical address (Y) for GDRAM
2. Set horizontal address (X) for GDRAM
3. Write D15~D8 to GDRAM (first byte)
4. Write D7~D0 to GDRAM (second byte)

Please refer to Table 7 for Graphic Display RAM mapping.

LCD driver

ST7920 embedded LCD driver has 33 commons and 64 segments to drive the LCD panel. Segment data from CGRAM, CGROM and HCGROM are shifted into the 64 bits segment latche to display. Extended segment driver (ST7921) can be used to extend the segment outputs upto 256 segments.
To select the CGRAM font, the bit4 through bit15 of DDRAM data must be 0. CGRAM data for each address is 16 bits. DDRAM data (character code) bit1 and bit2 are identical with CGRAM address bit4 and bit5.

<table>
<thead>
<tr>
<th>DDRAM data (char. code)</th>
<th>CGRAM Addr.</th>
<th>CGRAM data (higher byte)</th>
<th>CGRAM data (lower byte)</th>
</tr>
</thead>
<tbody>
<tr>
<td>B15~ B4</td>
<td>B</td>
<td>B</td>
<td>B</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 5: DDRAM data (character code) vs. CGRAM data/address map

Note:
1. DDRAM data (character code) bit1 and bit2 are identical with CGRAM address bit4 and bit5.
2. CGRAM address bit0 to bit3 specify total 16 rows. Row-16 is for cursor display. The data in Row-16 will be logically OR to the cursor.
3. CGRAM data for each address is 16 bits.
4. To select the CGRAM font, the bit4 through bit15 of DDRAM data must be "0" while bit0 and bit3 are "don't care".
<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
<tr>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
<td>☑</td>
</tr>
</tbody>
</table>

Table 6  16x8 half-width characters
Table 7   GDRAM display coordinates and corresponding address
### Instructions

ST7920 offers basic instruction set and extended instruction set:

**Instruction Set 1: (RE=0: Basic Instruction)**

<table>
<thead>
<tr>
<th>Inst.</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Clear</td>
<td>0 0 0 0 0 0 0 0 0 1</td>
<td>Fill DDRAM with &quot;20H&quot; and set DDRAM address counter (AC) to &quot;00H&quot;. 1.6 ms</td>
</tr>
<tr>
<td>Return Home</td>
<td>0 0 0 0 0 0 0 0 0 1 X</td>
<td>Set DDRAM address counter (AC) to &quot;00H&quot;, and put cursor to origin. The content of DDRAM are not changed 72 us</td>
</tr>
<tr>
<td>Entry Mode Set</td>
<td>0 0 0 0 0 0 0 0 0 1 I/D S</td>
<td>Set cursor position and display shift when doing write or read operation 72 us</td>
</tr>
<tr>
<td>Display Control</td>
<td>0 0 0 0 0 0 0 1 D C B</td>
<td>D=1: Display ON C=1: Cursor ON B=1: Character Blink ON 72 us</td>
</tr>
<tr>
<td>Cursor Display Control</td>
<td>0 0 0 0 0 0 0 1 S/C R/L X X</td>
<td>Cursor position and display shift control; the content of DDRAM are not changed 72 us</td>
</tr>
<tr>
<td>Function Set</td>
<td>0 0 0 0 0 0 1 DL 0 X X</td>
<td>DL=1: 8-bit interface DL=0: 4-bit interface RE=1: extended instruction RE=0: basic instruction 72 us</td>
</tr>
<tr>
<td>Set CGRAM Address</td>
<td>0 0 0 0 0 1 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Set CGRAM address to address counter (AC) Make sure that in extended instruction SR=0 (scroll or RAM address select) 72 us</td>
</tr>
<tr>
<td>Set DDRAM Address</td>
<td>0 0 0 0 1 AC6 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Set DDRAM address to address counter (AC) AC6 is fixed to 0 72 us</td>
</tr>
<tr>
<td>Read Busy Flag (BF) &amp; AC</td>
<td>0 1 BF AC6 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Read busy flag (BF) for completion of internal operation, also Read out the value of address counter (AC) 0 us</td>
</tr>
<tr>
<td>Write RAM</td>
<td>1 0 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>Write data to internal RAM (DDRAM/CGRAM/GDRAM) 72 us</td>
</tr>
<tr>
<td>Read RAM</td>
<td>1 1 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>Read data from internal RAM (DDRAM/CGRAM/GDRAM) 72 us</td>
</tr>
</tbody>
</table>
### Instruction set 2: (RE=1: extended instruction)

<table>
<thead>
<tr>
<th>Inst.</th>
<th>Code</th>
<th>Description</th>
<th>Exec time (540KHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standby</td>
<td>0 0 0 0 0 0 0 1</td>
<td>Enter standby mode, any other instruction can terminate. COM1...32 are halted.</td>
<td>72 us</td>
</tr>
<tr>
<td>Scroll or RAM Address Select</td>
<td>0 0 0 0 0 0 0 1</td>
<td>SR=1: enable vertical scroll position SR=0: enable CGRAM address (basic instruction)</td>
<td>72 us</td>
</tr>
<tr>
<td>Reverse (by line)</td>
<td>0 0 0 0 0 0 0 1</td>
<td>Select 1 out of 4 line (in DDRAM) and decide whether to reverse the display by toggling this instruction. ( R1,R0 ) initial value is 0,0.</td>
<td>72 us</td>
</tr>
<tr>
<td>Extended Function Set</td>
<td>0 0 0 0 1 DL X 1 RE G 0</td>
<td>DL=1 : 8-bit interface DL=0 : 4-bit interface ( RE=1: \text{extended instruction set} ) ( RE=0: \text{basic instruction set} ) G=1 : graphic display ON G=0 : graphic display OFF</td>
<td>72 us</td>
</tr>
<tr>
<td>Set Scroll Address</td>
<td>0 0 0 1 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>SR=1: AC5~AC0 the address of vertical scroll</td>
<td>72 us</td>
</tr>
<tr>
<td>Set Graphic Display RAM Address</td>
<td>0 0 1 0 0 0 0 AC5 AC4 AC3 AC2 AC1 AC0</td>
<td>Set GDRAM address to address counter (AC) Set the vertical address first and followed the horizontal address by consecutive writings Vertical address range: AC5...AC0 Horizontal address range: AC3...AC0</td>
<td>72 us</td>
</tr>
</tbody>
</table>

**Note:**

1. Make sure that ST7920 is not in busy state by reading the busy flag before sending instruction or data. If using delay loop instead, please make sure the delay time is enough. Please refer to the instruction execution time.

2. "RE" is the selection bit of basic and extended instruction set. After setting the RE bit, the value will be kept. So that the software doesn’t have to set RE every time when using the same instruction set.
### Initial Setting (Register flag) (RE=0: basic instruction)

<table>
<thead>
<tr>
<th>Inst.</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entry Mode Set</td>
<td></td>
<td>Cursor move to right, DDRAM address counter (AC) plus 1</td>
</tr>
<tr>
<td>Display Control</td>
<td></td>
<td>Display, cursor and blink are ALL OFF</td>
</tr>
<tr>
<td>CURSOR DISPLAY SHIFT</td>
<td></td>
<td>No cursor or display shift operation</td>
</tr>
<tr>
<td>FUNCTION SET</td>
<td></td>
<td>8-bit MPU interface, basic instruction set</td>
</tr>
</tbody>
</table>

### Initial Setting (Register flag) (RE=1: extended instruction set)

<table>
<thead>
<tr>
<th>Inst.</th>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCROLL OR RAM ADDR. SELECT</td>
<td>0 0 0 0 0 0 0 0 1</td>
<td>Allow vertical scroll or set CGRAM address</td>
</tr>
<tr>
<td>REVERSE</td>
<td>0 0 0 0 0 0 0 1</td>
<td>Begin with normal and toggle to reverse</td>
</tr>
<tr>
<td>EXTENDED FUNCTION SET</td>
<td>0 0 0 0 1 DL X 1</td>
<td>Graphic display OFF</td>
</tr>
</tbody>
</table>
Description of basic instruction set

### Display Clear

![Code Table]

This instruction will change the following items:
1. Fill DDRAM with "20H" (space code).
2. Set DDRAM address counter (AC) to "00H".
3. Set Entry Mode I/D bit to be "1". Cursor moves right and AC adds 1 after write or read operation.

### Return Home

![Code Table]

Set address counter (AC) to "00H". Cursor moves to origin. Then content of DDRAM is not changed.

### Entry Mode Set

![Code Table]

Set the cursor movement and display shift direction when doing write or read operation.

#### I/D: Address Counter Control: (Increase/Decrease)

When I/D = "1", cursor moves right, address counter (AC) is increased by 1.
When I/D = "0", cursor moves left, address counter (AC) is decreased by 1.

#### S: Display Shift Control: (Shift Left/Right)

<table>
<thead>
<tr>
<th>S</th>
<th>I/D</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>H</td>
<td>Entire display shift left by 1</td>
</tr>
<tr>
<td>H</td>
<td>L</td>
<td>Entire display shift right by 1</td>
</tr>
</tbody>
</table>
Display Control

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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>1</td>
<td>D</td>
<td>C</td>
<td>B</td>
</tr>
</tbody>
</table>

Controls display, cursor and blink ON/OFF.

**D: Display ON/OFF control bit**
- When D = "1", display ON
- When D = "0", display OFF, the content of DDRAM is not changed

**C: Cursor ON/OFF control bit**
- When C = "1", cursor ON.
- When C = "0", cursor OFF.

**B: Character Blink ON/OFF control bit**
- When B = "1", cursor position blink ON. Then display data (character) in cursor position will blink.
- When B = "0", cursor position blink OFF

Cursor/Display Shift Control

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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>S/C</td>
<td>R/L</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

This instruction configures the cursor moving direction or the display shifting direction. The content of DDRAM is not changed.

<table>
<thead>
<tr>
<th>S/C</th>
<th>R/L</th>
<th>Description</th>
<th>AC Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>L</td>
<td>Cursor moves left by 1 position</td>
<td>AC=AC-1</td>
</tr>
<tr>
<td>L</td>
<td>H</td>
<td>Cursor moves right by 1 position</td>
<td>AC=AC+1</td>
</tr>
<tr>
<td>H</td>
<td>L</td>
<td>Display shift left by 1, cursor also follows to shift.</td>
<td>AC=AC</td>
</tr>
<tr>
<td>H</td>
<td>H</td>
<td>Display shift right by 1, cursor also follows to shift.</td>
<td>AC=AC</td>
</tr>
</tbody>
</table>
Digole 12864ZW Module

Function Set

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>DL</td>
<td>X</td>
<td>RE</td>
</tr>
</tbody>
</table>

**DL: 4/8-bit interface control bit**

- When DL = "1", 8-bit MPU bus interface
- When DL = "0", 4-bit MPU bus interface

**RE: extended instruction set control bit**

- When RE = "1", extended instruction set
- When RE = "0", basic instruction set

In same instruction cannot alter DL and RE at once. Make sure that change DL first then RE.

Set CGRAM Address

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>AC5</td>
<td>AC4</td>
</tr>
</tbody>
</table>

Set CGRAM address into address counter (AC)

AC range is 00H…3FH

Make sure that in extended instruction SR=0 (scroll address or RAM address select)

Set DDRAM Address

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>AC6</td>
<td>AC5</td>
</tr>
</tbody>
</table>

Set DDRAM address into address counter (AC).

- First line AC range is 80H…8FH
- Second line AC range is 90H…9FH
- Third line AC range is A0H…AFH
- Fourth line AC range is B0H…BFH

Please note that only 2 lines can be display with one ST7920.

Read Busy Flag (BF) and Address

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>BF</td>
<td>AC6</td>
<td>AC5</td>
<td>AC4</td>
</tr>
</tbody>
</table>

Read busy flag (BF) can check whether the internal operation is finished or not. At the same time, the value of address counter (AC) is also read. When BF = "1", further instruction(s) will not be accepted until BF = "0".
### Write Data to RAM

<table>
<thead>
<tr>
<th>Code</th>
<th>RS</th>
<th>RW</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>
<td></td>
</tr>
</tbody>
</table>

Write data to the internal RAM and increase/decrease the (AC) by 1
Each RAM address (CGRAM, DDRAM and GDRAM...) must write 2 consecutive bytes for 16-bit data. After receiving the second byte, the address counter will increase or decrease by 1 according to the entry mode set control bit.

### Read RAM Data

<table>
<thead>
<tr>
<th>Code</th>
<th>RS</th>
<th>RW</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>1</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>
<td></td>
</tr>
</tbody>
</table>

Read data from the internal RAM and increase/decrease the (AC) by 1
After the operation mode changed to Read (CGRAM, DDRAM and GDRAM...), a "Dummy Read" is required. There is no need to add a "Dummy Read" for the following bytes unless a new address set instruction is issued.
Description of extended instruction set

**1. Standby**

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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>1</td>
<td></td>
</tr>
</tbody>
</table>

This Instruction will set ST7920 entering the standby mode. Any other instruction follows this instruction will terminate the standby mode. The content of DDRAM remains the same.

**2. Vertical Scroll or RAM Address Select**

When SR = “1”, the Vertical Scroll mode is enabled. When SR = “0”, “Set CGRAM Address” instruction (basic instruction) is enabled.

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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>1</td>
<td></td>
</tr>
</tbody>
</table>

**3. Reverse**

Select 1 out of 4 lines to reverse the display and to toggle the reverse condition by repeating this instruction. R1, R0 initial value is 00. The first time issuing this instruction, the display will be reversed while the second time will return the display become normal.

<table>
<thead>
<tr>
<th>R1</th>
<th>R0</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>L</td>
<td>First line normal or reverse</td>
</tr>
<tr>
<td>L</td>
<td>H</td>
<td>Second line normal or reverse</td>
</tr>
<tr>
<td>H</td>
<td>L</td>
<td>Third line normal or reverse</td>
</tr>
<tr>
<td>H</td>
<td>H</td>
<td>Fourth line normal or reverse</td>
</tr>
</tbody>
</table>

Please note that only 2 lines out of 4 lines of display data can be displayed with one ST7920.
### Extended Function Set

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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>X</td>
<td>RE</td>
<td>G</td>
</tr>
</tbody>
</table>

**DL**: 4/8-bit interface control bit  
- When DL = "1", 8-bit MPU interface.  
- When DL = "0", 4-bit MPU interface.

**RE**: extended instruction set control bit  
- When RE = "1", extended instruction set  
- When RE = "0", basic instruction set

**G**: Graphic display control bit  
- When G = "1", Graphic Display ON  
- When G = "0", Graphic Display OFF

**In same instruction cannot alter DL, RE and G at once. Make sure that change DL or G first and then RE.**

### Set Scroll Address

<table>
<thead>
<tr>
<th>RS</th>
<th>RW</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>AC5</td>
<td>AC4</td>
<td>AC3</td>
<td>AC2</td>
</tr>
</tbody>
</table>

SR=1: AC5~AC0 is vertical scroll displacement address

### Set Graphic RAM Address

Set GDRAM address into address counter (AC). This is a 2-byte instruction. The first instruction sets the vertical address while the second one sets the horizontal address (write 2 consecutive bytes to complete the vertical and horizontal address setting).  
- Vertical address range is AC5...AC0  
- Horizontal address range is AC3...AC0

The address counter (AC) of graphic RAM (GRAM) will be increased automatically after the vertical and horizontal addresses are set. After horizontal address is increased upto 0FH, it will automatically return to 00H. However, the vertical address will not increase as the result of the same action.
Parallel interface:

ST7920 is in parallel mode by pulling up PSB pin. ST7920 can select 8-bit or 4-bit bus interface by setting the DL control bit in “Function Set” instruction. MPU can control RS, RW, E and DB0…DB7 pins to complete the data transmission.

In 4-bit transfer mode, every 8-bit data or instruction is separated into 2 parts. The higher 4 bits (bit-7~bit-4) data will be transfered first through data pins (DB7~DB4). The lower 4 bits (bit-3~bit-0) data will be transfered second through data pins (DB7~DB4). The (DB3~DB0) data pins are not used during 4-bit transfer mode.
Serial interface:

ST7920 is in serial interface mode when pulling down PSB pin. Two pins (SCLK and SID) are used to complete the data transfer. Only write data is available in the serial interface mode.

When chip select (CS) is low, ST7920 serial clock counter and serial data will be reset. Serial transfer counter is set to the first bit and data register is cleared. After CS is “L”, any further change on SID or SCLK is not allowed. It is recommended to keep SCLK at “L” and SID at the last status before set CS to “L”. For a minimal system with only one ST7920 and one MPU, only SCLK and SID pins are necessary. CS pin should pull to high.

ST7920’s serial clock (SCLK) is asynchronous to the internal clock and is generated by MPU. When multiple instruction/data is transferred, the instruction execution time must be considered. MPU must wait till the previous instruction is finished and then send the next instruction. ST7920 has no internal instruction buffer area.

When starting a transmission, a start byte is required. It consists of 5 consecutive “1” (sync character). Serial transfer counter will be reset and synchronized. Followed by 2-bit flag that indicates: read/write (RW) and register/data selected (RS) operation. Last 4 bits are filled by “0”.

After receiving the sync character, RW and RS bits, every 8 bits instruction/data will be separated into 2 groups. Higher 4 bits (DB7~DB4) will be placed in the first section followed by 4 “0”s. And lower 4 bits (DB3~DB0) will be placed in the second section followed by 4 “0”s.

![Timing Diagram of Serial Mode Data Transfer](image-url)
8051 demo program for serial interface

WRINS:

```assembly
SETB CS
SETB SID ; SID = 1
CLR SCLK
SETB SCLK ; READ DATA FROM SID
CLR SCLK
SETB SCLK ; READ DATA FROM SID
CLR SCLK
SETB SCLK ; READ DATA FROM SID
CLR SCLK
CLR SID ; SID = 0
SETB SCLK ; READ DATA FROM SID
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
MOVBIT SID, A.7 ; SID = A.7
SETB SCLK ; READ DATA FROM SID
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
MOVBITS PID, A.6 ; PID = A.6
SETB SCLK ; READ DATA FROM SID
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
MOVBIT PID, A.5 ; PID = A.5
SETB SCLK ; READ DATA FROM SID
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
MOVBIT PID, A.4 ; PID = A.4
SETB SCLK ; READ DATA FROM SID
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCLK
CLR SCL
**Application circuit for testing CGROM and HCGROM:**

We can use the function of “CHECK SUM” to check the CGROM is right or error.  
See the following notes: Useing IC Pad (Pin4: CLK, Pin5: TT1, Pin6: TT2) to do the “CHECK SUM” function.  
The application circuit is at Page49.

**Timing Diagram for checking CGROM (TT1=0, TT2=1)**

The ST7920 check sum process: (DDRAM must be cleared by 0x00 before this process)  
In the first place: Resetting the internal counter (set TT1 and TT2 to Height)  
In the second place: Setting CGROM mode (set TT1 to Low, TT2 to Height).  
In the third place: CLK starts to count 655362 times.  
In the final place: Finishing the counting, read the last four bytes to CHECK SUM (reading only when the CLK is Height).  
ST7920 check sum circuit: Data is available when CLK is height; if CLK is low then the data is always FFH. The last four bytes are Y0, Y1, Y2, and Y3.

<table>
<thead>
<tr>
<th>Version (Font)</th>
<th>CGROM Last four bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Y0</td>
</tr>
<tr>
<td>1 Big5 (0A)</td>
<td>38</td>
</tr>
<tr>
<td>2 GB (0B)</td>
<td>9D</td>
</tr>
<tr>
<td>3 0C</td>
<td>FD</td>
</tr>
</tbody>
</table>
Timing Diagram for checking HCGROM (TT1=1, TT2=0)

The ST7920 check sum process: (DDRAM must be cleared by 0x00 before this process)
In the first place: Resetting the internal counter (set TT1 and TT2 to Height)
In the second place: Setting CGROM mode (set TT1 to Height, TT2 to Low).
In the third place: CLK starts to count 10242 times.
In the final place: Finishing the counting, read the last four bytes to CHECK SUM (reading only when the CLK is Height).

ST7920 check sum circuit: Data is available when CLK is height; if CLK is low then the data is always FFH. The last four bytes are Y0, Y1, Y2, and Y3.

The fastest execution time is: tCYC=2us (0.5MHz at 5V).

The table below is a comparing table of HCGROM for different versions.

<table>
<thead>
<tr>
<th>Version (Font)</th>
<th>HCGROM last four bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Y0</td>
</tr>
<tr>
<td>1  Big5 (0A)</td>
<td>B5</td>
</tr>
<tr>
<td>2  GB (0B)</td>
<td>B5</td>
</tr>
<tr>
<td>3  0C</td>
<td>B5</td>
</tr>
</tbody>
</table>
**Digole 12864ZW Module**

**Testing Step:**
1. Clear whole DDRAM area by writing data 0x00.
2. Composing TT1 and TT2 to make the ‘Reset’ action, and clear the internal counter.
3. Selecting the test mode by setting TT1 and TT2 (CGROM or HCGROM).
4. After setp1 and setp2, entering some impulse signals through Pin4 (CLK).
5. Reading the CHECK SUM data through D0 to D7.
6. Comparing CHECK SUM with the Code Table (upper table) to check if the data is correct or not.

<table>
<thead>
<tr>
<th>TT1</th>
<th>TT2</th>
<th>No. of counts</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>--</td>
<td>RESET</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>655362</td>
<td>CGROM</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>10242</td>
<td>HGROM</td>
</tr>
</tbody>
</table>

**Test process flow:**

1. **Clear whole DDRAM** (by writing data 0x00)
2. **Set TT1= TT2=HIGH** (reset internal counter)
3. **Select ROM mode** (HCGROM or CGROM)
4. **Start CLK to count**
5. **Check the last 4 bytes**
8051 CGROM、HCGROM illustrative test program

;********************************************
;*      CHECK_ROM                           *
;********************************************
;* Definition of outside Pin                *
;********************************************
CLK   REG   P3.5                    
TT1   REG   P3.0                    
TT2   REG   P3.1                    
TT3   REG   P3.2  ;CHECK CGROM FLAG   
TT4   REG   P3.3  ;CHECK HCGROM FLAG   
TT5   REG   P3.4  ;ERROR FLAG         
;********************************************
;* Definition of internal RAM               *
;********************************************
STACK  EQU   6FH             
FUNC   EQU   20H             
; Interrupt set                            
;********************************************
ORG   00H                     
AJMP   RESET                   
;********************************************
* PROGRAM START                        
;********************************************
RESET:  MOV     SP,#STACK               
        MOV     P1,#FFH                 
        MOV     P3,#FFH                 
;********************************************
*      CHECK_CGROM                      
;********************************************
; Initial DDRAM                          
CALL    WR0x00                  ;Write 0x00 to whole DDRAM  
;********************************************
*      Initial setting                   
;********************************************
CGROM:  SETB    TT1                     
        SETB    TT2                     
CALL    DELAY_100US             ;Wait Reset 100us  
CLR     TT1                     
        SETB    CLK                     
        DJNZ    R1,CN2                  
        DJNZ    R2,CN3                  
        DJNZ    R3,CN4                  
        MOV     R3,#0                   
        CN5:  MOV     R2,#0                 
CN4:    MOV     R1,#0                 
        CN2:  CLR     CLK                         
        DJNZ    R1,CN2                 
        DJNZ    R2,CN3                 
        DJNZ    R3,CN4                 
        MOV     R3,#0                   
        CN5:  MOV     R2,#255              
CN6:    CLR     CLK                         
        SETB    CLK                     
        DJNZ    R2,CN6                 
        DJNZ    R3,CN5
MOV R3,#63  ; Counter 655356
          ;---------------------
CLR CLK   ;Counter 655357
SETB CLK
MOV A,P1  ;A=Y0
CJNE A,#FDH,ERRORC  ;COMPARE Y0 DATA
CLR CLK  
SETB CLK
MOV A,P1  ;A=Y1
CJNE A,#6FH,ERRORC  ;COMPARE Y1 DATA
CLR CLK
CLRTT3   ;IF OK CLR TT3
CALL HCGROM
          ;---------------------------------------;
ERRORC:                                                  
CLR TT5   ;IF CGROM CHECK ERROR CLR TT5
          ;************************************
;* CHECK_HCGROM*                                  
;************************************
;* Initial setting*                               
;************************************
HCGROM: SETB TT1  ;TT1,TT2 SET HIGH (RESET)
SETB TT2  ;TT1,TT2 SET HIGH (RESET)
CALL DELAY_100US    ;Wait Reset 100us
CLRTT2   ;TT2=LOW TT1=HIGH (CHECK HCGROM)
SETB CLK
CALL DELAY_100US
          ;************************************
;* start counter*                                
;************************************
MOV R3,#9
          ;<-----
N4: MOV R2,#32
N3: MOV R1,#32
N2: CLR CLK
SETB CLK
DJNZ R1,N2
DJNZ R2,N3
DJNZ R3,N4
MOV R3,#32
N5: MOV R2,#31
N6: CLR CLK
SETB CLK
DJNZ R2,N6
DJNZ R3,N5
MOV R2,#30
          ; Digole 12864ZW Module
Digole 12864ZW Module
N7: CLR CLK ; Counter 10236
SETB CLK ;
DJNZ R2,N7 ;

N7: CLR CLK ; Counter 10237
SETB CLK ;
MOV A,P1 ; A=Y0
CJNE A,#85H,ERROR ; COMPARE Y0 DATA
CLR CLK ; Counter 10238
SETB CLK ;
MOV A,P1 ; A=Y1
CJNE A,#11H,ERROR ; COMPARE Y1 DATA
CLR CLK ; Counter 10239
SETB CLK ;
MOV A,P1 ; A=Y2
CJNE A,#85H,ERROR ; COMPARE Y2 DATA
CLR CLK ; Counter 10240
MOV A,P1 ; A=Y3
CJNE A,#11H,ERROR ; COMPARE Y3 DATA
CLR TT4 ; IF HCGROM CHECK OK THEN CLR TT4
AJMP $ ;

ERROR:
CLR TT5 ; IF HCGROM CHECK ERROR THEN CLR TT5
AJMP $ ;

; *******************************
; * DELAY TIME 100US       *
; *******************************
DELAY_100US ;
DEL_10 MOV R6,#5 ;
DEL_9 MOV R7,#3 ;
DJNZ R7,$ ;
DJNZ R6,DEL_9 ;
RET ;

END ;
8-bit interface:

- **POWER ON**
  - Wait time >40ms
  - XRESET LOW → HIGH

- **Function set**
  - RS | RW | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0
  - 0 0 0 0 1 1 X 0 X X
  - Wait time >100μS

- **Function set**
  - RS | RW | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0
  - 0 0 0 0 1 1 X 0 X X
  - Wait time >37μS

- **Display ON/OFF control**
  - RS | RW | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0
  - 0 0 0 0 0 0 1 D C B
  - Wait time >100μS

- **Display clear**
  - RS | RW | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0
  - 0 0 0 0 0 0 0 0 0 1
  - Wait time >10mS

- **Entry mode set**
  - RS | RW | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0
  - 0 0 0 0 0 0 0 1 I/D S
  - Initialization end
4-bit interface:

**POWER ON**

Wait time > 40mS
(for VDD stable)
XRESET: LOW → HIGH

**Function set**

<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>1</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Wait time > 100 μS

**Function set**

<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>1</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
<td>0</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Wait time > 100 μS

**Display ON/OFF Control**

<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>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>D</td>
<td>C</td>
<td>B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Wait time > 100 μS

**Display Clear**

<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>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Wait time > 10mS

**Entry Mode Set**

<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>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>I/D</td>
<td>S</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

**INITIALIZATION END**
**Digole 12864ZW Module**

**Built in voltage booster**

*Diagram showing components and connections.*

**External reset timing**

*Diagram showing reset timing with pulse widths.*

<table>
<thead>
<tr>
<th>XRESET pulse width</th>
<th>Trw</th>
<th>10us</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESET start time</td>
<td>Tres</td>
<td>50ns</td>
</tr>
</tbody>
</table>
LCD driving wave form (1/33 duty, 1/5 bias)

When oscillation frequency is 540KHZ, 1 clock cycle time = 1.85us
1 frame = 1.85us x 300 x 33 = 18315us=18.3ms

![LCD driving wave form diagram](image-url)
## Absolute Maximum Ratings

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power Supply Voltage</td>
<td>$V_{DD}$</td>
<td>-0.3V to +6.0V</td>
</tr>
<tr>
<td>LCD Driver Voltage</td>
<td>$V_{LCD}$ or $V_0$</td>
<td>-0.3V to +7.0V</td>
</tr>
<tr>
<td>Voltage Doubler Output</td>
<td>$V_{OUT}$</td>
<td>-0.3V to +7.0V</td>
</tr>
<tr>
<td>Input Voltage</td>
<td>$V_{IN}$</td>
<td>-0.3V to $V_{DD}$+0.3V</td>
</tr>
<tr>
<td>Operating Temperature</td>
<td>$T_A$</td>
<td>-30°C to +85°C</td>
</tr>
<tr>
<td>Storage Temperature</td>
<td>$T_{STO}$</td>
<td>-65°C to +150°C</td>
</tr>
</tbody>
</table>

## DC Characteristics ($T_A = -30°C ~ 85°C$, $V_{DD} = 2.7$ V - 4.5 V)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Characteristics</th>
<th>Test Condition</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{DD}$</td>
<td>Operating Voltage</td>
<td></td>
<td>2.7</td>
<td>-</td>
<td>5.5</td>
<td>V</td>
</tr>
<tr>
<td>$V_{LCD}$</td>
<td>LCD Voltage</td>
<td>$V_0$-$V_{SS}$</td>
<td>3.0</td>
<td>-</td>
<td>7</td>
<td>V</td>
</tr>
<tr>
<td>$I_{CC}$</td>
<td>Power Supply Current $I_{OSC} = 530KHz, V_{DD}=3.0V$</td>
<td>$R_f=18K\Omega$</td>
<td>-</td>
<td>0.20</td>
<td>0.45</td>
<td>mA</td>
</tr>
<tr>
<td>$V_{IH1}$</td>
<td>Input High Voltage (Except OSC1)</td>
<td></td>
<td>0.7$V_{DD}$</td>
<td>-</td>
<td>$V_{DD}$</td>
<td>V</td>
</tr>
<tr>
<td>$V_{IL1}$</td>
<td>Input Low Voltage (Except OSC1)</td>
<td></td>
<td>- 0.3</td>
<td>-</td>
<td>0.6</td>
<td>V</td>
</tr>
<tr>
<td>$V_{IH2}$</td>
<td>Input High Voltage (OSC1)</td>
<td></td>
<td>$V_{DD}$ - 1</td>
<td>-</td>
<td>$V_{DD}$</td>
<td>V</td>
</tr>
<tr>
<td>$V_{IL2}$</td>
<td>Input Low Voltage (OSC1)</td>
<td></td>
<td>-</td>
<td>-</td>
<td>1.0</td>
<td>V</td>
</tr>
<tr>
<td>$V_{OH1}$</td>
<td>Output High Voltage (DB0 - DB7)</td>
<td>$I_{OH} = -0.1mA$</td>
<td>0.8$V_{DD}$</td>
<td>-</td>
<td>$V_{DD}$</td>
<td>V</td>
</tr>
<tr>
<td>$V_{OL1}$</td>
<td>Output Low Voltage (DB0 - DB7)</td>
<td>$I_{OL} = 0.1mA$</td>
<td>-</td>
<td>-</td>
<td>0.1</td>
<td>V</td>
</tr>
<tr>
<td>$V_{OH2}$</td>
<td>Output High Voltage (Except DB0 - DB7)</td>
<td>$I_{OH} = -0.04mA$</td>
<td>0.8$V_{DD}$</td>
<td>-</td>
<td>$V_{DD}$</td>
<td>V</td>
</tr>
<tr>
<td>$V_{OL2}$</td>
<td>Output Low Voltage (Except DB0 - DB7)</td>
<td>$I_{OL} = 0.04mA$</td>
<td>-</td>
<td>-</td>
<td>0.1$V_{DD}$</td>
<td>V</td>
</tr>
<tr>
<td>$I_{LEAK}$</td>
<td>Input Leakage Current</td>
<td>$V_{IN} = 0V$ to $V_{DD}$</td>
<td>-1</td>
<td>-</td>
<td>1</td>
<td>$\mu$A</td>
</tr>
<tr>
<td>$I_{UP}$</td>
<td>Pull Up MOS Current</td>
<td>$V_{DD} = 3V$</td>
<td>22</td>
<td>27</td>
<td>32</td>
<td>$\mu$A</td>
</tr>
</tbody>
</table>
## DC Characteristics (\(T_A = -30^\circ C \sim 85^\circ C\), \(V_{DD} = 4.5\, V - 5.5\, V\))

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Characteristics</th>
<th>Test Condition</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>(V_{DD})</td>
<td>Operating Voltage</td>
<td>-</td>
<td>4.5</td>
<td>-</td>
<td>5.5</td>
<td>V</td>
</tr>
<tr>
<td>(V_{LCD})</td>
<td>LCD Voltage</td>
<td>(V_{0-SS})</td>
<td>3.0</td>
<td>-</td>
<td>7</td>
<td>V</td>
</tr>
<tr>
<td>(I_{CC})</td>
<td>Power Supply Current</td>
<td>(f_{OSC} = 540, \text{KHz},, V_{DD}=5, V,, R_f=33, \Omega)</td>
<td>-</td>
<td>0.45</td>
<td>0.75</td>
<td>mA</td>
</tr>
<tr>
<td>(V_{IH1})</td>
<td>Input High Voltage (Except OSC1)</td>
<td>-</td>
<td>0.7(V_{DD})</td>
<td>-</td>
<td>(V_{DD})</td>
<td>V</td>
</tr>
<tr>
<td>(V_{IL1})</td>
<td>Input Low Voltage (Except OSC1)</td>
<td>-</td>
<td>-0.3</td>
<td>-</td>
<td>0.6</td>
<td>V</td>
</tr>
<tr>
<td>(V_{IH2})</td>
<td>Input High Voltage (OSC1)</td>
<td>-</td>
<td>(V_{DD}-1)</td>
<td>-</td>
<td>(V_{DD})</td>
<td>V</td>
</tr>
<tr>
<td>(V_{IL2})</td>
<td>Input Low Voltage (OSC1)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1.0</td>
<td>V</td>
</tr>
<tr>
<td>(V_{OH1})</td>
<td>Output High Voltage (DB0 - DB7)</td>
<td>(I_{OH} = -0.1, \text{mA})</td>
<td>0.8(V_{DD})</td>
<td>-</td>
<td>(V_{DD})</td>
<td>V</td>
</tr>
<tr>
<td>(V_{OL1})</td>
<td>Output Low Voltage (DB0 - DB7)</td>
<td>(I_{OL} = 0.1, \text{mA})</td>
<td>-</td>
<td>-</td>
<td>0.4</td>
<td>V</td>
</tr>
<tr>
<td>(V_{OH2})</td>
<td>Output High Voltage (Except DB0 - DB7)</td>
<td>(I_{OH} = -0.04, \text{mA})</td>
<td>0.8(V_{DD})</td>
<td>-</td>
<td>(V_{DD})</td>
<td>V</td>
</tr>
<tr>
<td>(V_{OL2})</td>
<td>Output Low Voltage (Except DB0 - DB7)</td>
<td>(I_{OL} = 0.04, \text{mA})</td>
<td>-</td>
<td>-</td>
<td>0.1(V_{DD})</td>
<td>V</td>
</tr>
<tr>
<td>(I_{LEAK})</td>
<td>Input Leakage Current</td>
<td>(V_{IN} = 0, V) to (V_{DD})</td>
<td>-1</td>
<td>-</td>
<td>1</td>
<td>(\mu A)</td>
</tr>
<tr>
<td>(I_{PUP})</td>
<td>Pull Up MOS Current</td>
<td>(V_{DD} = 5, V)</td>
<td>75</td>
<td>80</td>
<td>85</td>
<td>(\mu A)</td>
</tr>
</tbody>
</table>
Digole 12864ZW Module

AC Characteristics (T_A = -30°C ~ 85°C, V_DD = 4.5V) Parallel Mode Interface

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Characteristics</th>
<th>Test Condition</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Internal Clock Operation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f OSC</td>
<td>OSC Frequency</td>
<td>R = 33KΩ</td>
<td>480</td>
<td>540</td>
<td>600</td>
<td>KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>External Clock Operation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f EX</td>
<td>External Frequency</td>
<td></td>
<td>-</td>
<td>480</td>
<td>540</td>
<td>600</td>
</tr>
<tr>
<td></td>
<td>Duty Cycle</td>
<td></td>
<td>-</td>
<td>45</td>
<td>50</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>Rise/Fall Time</td>
<td></td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0.2</td>
</tr>
</tbody>
</table>

Write Mode (Writing data from MPU to ST7920)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Characteristics</th>
<th>Test Condition</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>T C</td>
<td>Enable Cycle Time</td>
<td>Pin E</td>
<td>1200</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TPW</td>
<td>Enable Pulse Width</td>
<td>Pin E</td>
<td>140</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TR,TF</td>
<td>Enable Rise/Fall Time</td>
<td>Pin E</td>
<td>-</td>
<td>-</td>
<td>25</td>
<td>ns</td>
</tr>
<tr>
<td>TAS</td>
<td>Address Setup Time</td>
<td>Pins: RS,RW,E</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TAH</td>
<td>Address Hold Time</td>
<td>Pins: RS,RW,E</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TDSW</td>
<td>Data Setup Time</td>
<td>Pins: DB0 - DB7</td>
<td>40</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TH</td>
<td>Data Hold Time</td>
<td>Pins: DB0 - DB7</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
</tbody>
</table>

Read Mode (Reading Data from ST7920 to MPU)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Characteristics</th>
<th>Test Condition</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>T C</td>
<td>Enable Cycle Time</td>
<td>Pin E</td>
<td>1200</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TPW</td>
<td>Enable Pulse Width</td>
<td>Pin E</td>
<td>140</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TR,TF</td>
<td>Enable Rise/Fall Time</td>
<td>Pin E</td>
<td>-</td>
<td>-</td>
<td>25</td>
<td>ns</td>
</tr>
<tr>
<td>TAS</td>
<td>Address Setup Time</td>
<td>Pins: RS,RW,E</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TAH</td>
<td>Address Hold Time</td>
<td>Pins: RS,RW,E</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TDDR</td>
<td>Data Delay Time</td>
<td>Pins: DB0 - DB7</td>
<td>-</td>
<td>-</td>
<td>100</td>
<td>ns</td>
</tr>
<tr>
<td>TH</td>
<td>Data Hold Time</td>
<td>Pins: DB0 - DB7</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
</tbody>
</table>

Interface Mode with LCD Driver(ST7921)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Characteristics</th>
<th>Test Condition</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCWH</td>
<td>Clock Pulse with High</td>
<td>Pins: CL1, CL2</td>
<td>800</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TCWL</td>
<td>Clock Pulse with Low</td>
<td>Pins: CL1, CL2</td>
<td>800</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TCSR</td>
<td>Clock Setup Time</td>
<td>Pins: CL1, CL2</td>
<td>500</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TSD</td>
<td>Data Setup Time</td>
<td>Pin: D</td>
<td>300</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TDH</td>
<td>Data Hold Time</td>
<td>Pin: D</td>
<td>300</td>
<td>-</td>
<td>-</td>
<td>ns</td>
</tr>
<tr>
<td>TDM</td>
<td>M Delay Time</td>
<td>Pin: M</td>
<td>-1000</td>
<td>-</td>
<td>1000</td>
<td>ns</td>
</tr>
</tbody>
</table>
### AC Characteristics (T_A = -30°C ~ 85°C, V_DD = 2.7V) Parallel Mode Interface

#### Symbol | Characteristics | Test Condition | Min. | Typ. | Max. | Unit
--- | --- | --- | --- | --- | --- | ---
- | Internal Clock Operation |
- | Osc Frequency |
| fOSC | R = 18KΩ | 470 | 530 | 590 | KHz |
- | External Clock Operation |
- | External Frequency |
| fEX | Duty Cycle | - | 45 | 50 | 55 | % |
| TR,TF | Rise/Fall Time | - | - | - | 0.2 | µs |
- | Write Mode (Writing data from MPU to ST7920) |
- | Enable Cycle Time |
| TC | Pin E | 1800 | - | - | ns |
| TPW | Enable Pulse Width |
| Pin E | 160 | - | - | ns |
| TR,TF | Enable Rise/Fall Time |
| Pin E | - | - | 25 | ns |
| TAS | Address Setup Time |
| Pins: RS,RW,E | 10 | - | - | ns |
| TAH | Address Hold Time |
| Pins: RS,RW,E | 20 | - | - | ns |
| TDSW | Data Setup Time |
| Pins: DB0 - DB7 | 40 | - | - | ns |
| TH | Data Hold Time |
| Pins: DB0 - DB7 | 20 | - | - | ns |
- | Read Mode (Reading Data from ST7920 to MPU) |
- | Enable Cycle Time |
| TC | Pin E | 1800 | - | - | ns |
| TPW | Enable Pulse Width |
| Pin E | 320 | - | - | ns |
| TR,TF | Enable Rise/Fall Time |
| Pin E | - | - | 25 | ns |
| TAS | Address Setup Time |
| Pins: RS,RW,E | 10 | - | - | ns |
| TAH | Address Hold Time |
| Pins: RS,RW,E | 20 | - | - | ns |
| TDDR | Data Delay Time |
| Pins: DB0 - DB7 | - | - | 260 | ns |
| TH | Data Hold Time |
| Pins: DB0 - DB7 | 20 | - | - | ns |
- | Interface Mode with LCD Driver(ST7921) |
- | Clock Pulse with High |
| TCWH | Pins: CL1, CL2 | 800 | - | - | ns |
| TCWL | Clock Pulse with Low |
| Pins: CL1, CL2 | 800 | - | - | ns |
| TCSST | Clock Setup Time |
| Pins: CL1, CL2 | 500 | - | - | ns |
| TSU | Data Setup Time |
| Pin: D | 300 | - | - | ns |
| TDDH | Data Hold Time |
| Pin: D | 300 | - | - | ns |
| TDM | M Delay Time |
| Pin: M | -1000 | - | 1000 | ns |
8-bit interface timing diagram

1. MPU write data to ST7920

2. MPU read data from ST7920
**AC Characteristics (T_A = -30°C ~ 85°C, V_{DD} = 4.5V) Serial Mode Interface**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Characteristics</th>
<th>Test Condition</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Internal Clock Operation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f_{OSC}</td>
<td>OSC Frequency</td>
<td>R = 33KΩ</td>
<td>470</td>
<td>530</td>
<td>590</td>
<td>KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>External Clock Operation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f_{EX}</td>
<td>External Frequency</td>
<td></td>
<td>470</td>
<td>530</td>
<td>590</td>
<td>KHz</td>
</tr>
<tr>
<td></td>
<td>Duty Cycle</td>
<td></td>
<td>45</td>
<td>50</td>
<td>55</td>
<td>%</td>
</tr>
<tr>
<td>T_{R,T_F}</td>
<td>Rise/Fall Time</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>μs</td>
</tr>
<tr>
<td>T_{SCYC}</td>
<td>Serial clock cycle</td>
<td>Pin E</td>
<td>400</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{SHW}</td>
<td>SCLK high pulse width</td>
<td>Pin E</td>
<td>200</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{SLW}</td>
<td>SCLK low pulse width</td>
<td>Pin E</td>
<td>200</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{SDS}</td>
<td>SID data setup time</td>
<td>Pins RW</td>
<td>40</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{SDH}</td>
<td>SID data hold time</td>
<td>Pins RW</td>
<td>40</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{CSS}</td>
<td>CS setup time</td>
<td>Pins RS</td>
<td>60</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{CSH}</td>
<td>CS hold time</td>
<td>Pins RS</td>
<td>60</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
</tbody>
</table>

**AC Characteristics (T_A = -30°C ~ 85°C, V_{DD} = 2.7V) Serial Mode Interface**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Characteristics</th>
<th>Test Condition</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Internal Clock Operation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f_{OSC}</td>
<td>OSC Frequency</td>
<td>R = 18KΩ</td>
<td>470</td>
<td>530</td>
<td>590</td>
<td>KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>External Clock Operation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f_{EX}</td>
<td>External Frequency</td>
<td></td>
<td>470</td>
<td>530</td>
<td>590</td>
<td>KHz</td>
</tr>
<tr>
<td></td>
<td>Duty Cycle</td>
<td></td>
<td>45</td>
<td>50</td>
<td>55</td>
<td>%</td>
</tr>
<tr>
<td>T_{R,T_F}</td>
<td>Rise/Fall Time</td>
<td></td>
<td></td>
<td></td>
<td>0.2</td>
<td>μs</td>
</tr>
<tr>
<td>T_{SCYC}</td>
<td>Serial clock cycle</td>
<td>Pin E</td>
<td>600</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{SHW}</td>
<td>SCLK high pulse width</td>
<td>Pin E</td>
<td>300</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{SLW}</td>
<td>SCLK low pulse width</td>
<td>Pin E</td>
<td>300</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{SDS}</td>
<td>SID data setup time</td>
<td>Pins RW</td>
<td>40</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{SDH}</td>
<td>SID data hold time</td>
<td>Pins RW</td>
<td>40</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{CSS}</td>
<td>CS setup time</td>
<td>Pins RS</td>
<td>60</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>T_{CSH}</td>
<td>CS hold time</td>
<td>Pins RS</td>
<td>60</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
</tbody>
</table>
Serial interface timing diagram

- MPU write data to ST7920
I/O pin diagram

Input PAD: E (No Pull-up)

Input PAD: RS, RW (with Pull-up)

Output PAD: CL1, CL2, M, D

I/O PAD: DB0 – DB7

Enable

DATA
Digole 12864ZW Module