

JLX1602A-1 使用说明书

目 录

| 序号 | 内 容 标 题 | 页码 |
|----|-----------|------|
| 1 | 概述 | 2 |
| 2 | 字符型模块的特点 | 2 |
| 3 | 外形及接口引脚功能 | 3~4 |
| 4 | 基本原理 | 4~5 |
| 5 | 技术参数 | 6 |
| 6 | 时序特性 | 7~9 |
| 7 | 指令功能 | 9~18 |

1. 概述

晶联讯电子专注于液晶屏及液晶模块的研发、制造。所生产 JLX1602A 型液晶模块由于使用方便、显示清晰，广泛应用于各种人机交流面板。JLX1602A 可以显示 2 行，每行 16 个英文、数字、符号，并可选择日文、俄文、以色列文、希腊文等文字（注俄文、以色列文、希腊文需订制）。并可以自编字符（每显示一个界面最多可以达到 8 个 5*8 点阵或 4 个 5*11 自编字符）。

2. 字符型模块的性能

重量轻: ≤30g;

体积小: ≤11mm 厚;

功耗低: 10 - 100mw (不带背光 10Mw, 带背光不大于 100Mw) ;

显示内容: 192 种字符 (5×8 点字型);

32 种字符 (5×11 点字型);

可自编 8 种 (5×8) 或 4 种 (5×11) 种字符, (注每显示一个界面最多可以达到 8 个自编字符, 但更换显示界面后可再编);

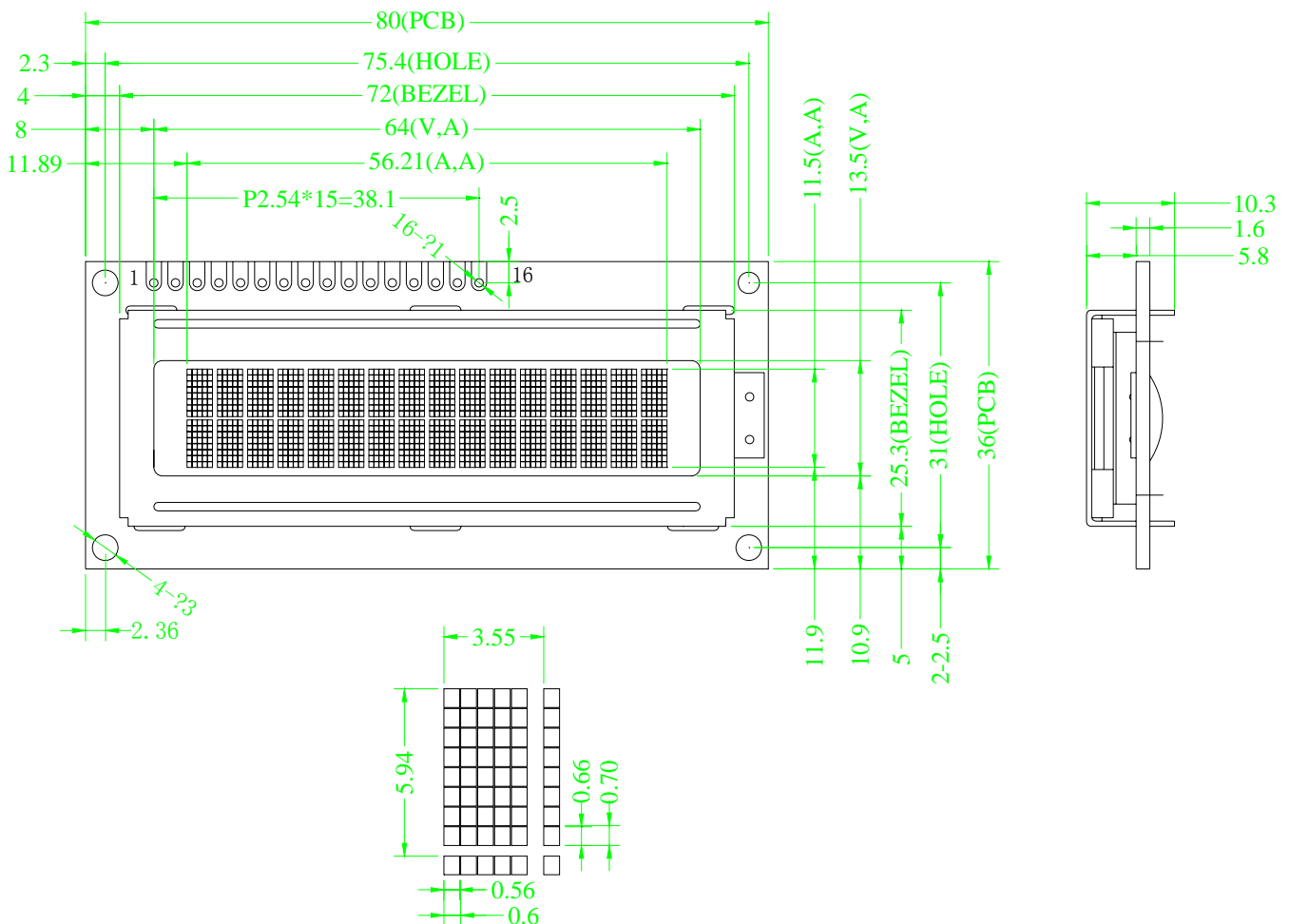
指令功能强: 可组合成各种输入、显示、移位方式以满足不同的要求;

接口简单方便: 可与 8 位微处理器或微控制器相联; 并可选择 4 位或 8 位并行接口

工作温度宽: -20℃ - 70℃;

可靠性高: 寿命为 50,000 小时 (25℃)。

3. 外形尺寸及接口引脚功能



模块的接口引脚功能

| 引线号 | 符号 | 名称 | 功能 |
|--------------|-----------------|---------|---------------------|
| 1 | VSS | 接地 | 0V |
| 2 | VDD | 电路电源 | 5V±10% |
| 3 | V0 | 液晶驱动电压 | 保证 VDD-V0=4.5V±0.3V |
| 4 | RS | 寄存器选择信号 | H:数据寄存器 0:指令寄存器 |
| 5 | R/W | 读/写信号 | H:读 0:写 |
| 6 | E | 使能信号 | 下降沿触发,锁存数据 |
| 7 14 | DB0 DB7 | 数据线 | 数据传输 |
| 15 | A | 背光电源线 | 正极背光电源线 |
| 16 | K | 背光电源线 | 负极背光电源线 |

表 1：模块的接口引脚功能

4. 基本原理

4.1 液晶屏 (LCD)

在液晶板上排列着若干5×7或5×10点阵的字符显示位,每个显示位可显示1个字符,本产品每行16个显示位,共两行。若要每行8、20、24、40位,1行、2行或4行请选用本厂的:0802,1601,1604,2002,2004,4004等产品。

4.2 工作电路图:

图1是JLX1602A字符型模块的电路框图,它由ST7066,ST7065及几个电阻电容组成。ST7065是扩展显示字符用的(例如:16字符×1行模块就不用ST7065,16字符×2行模块就要用1片ST7065)。

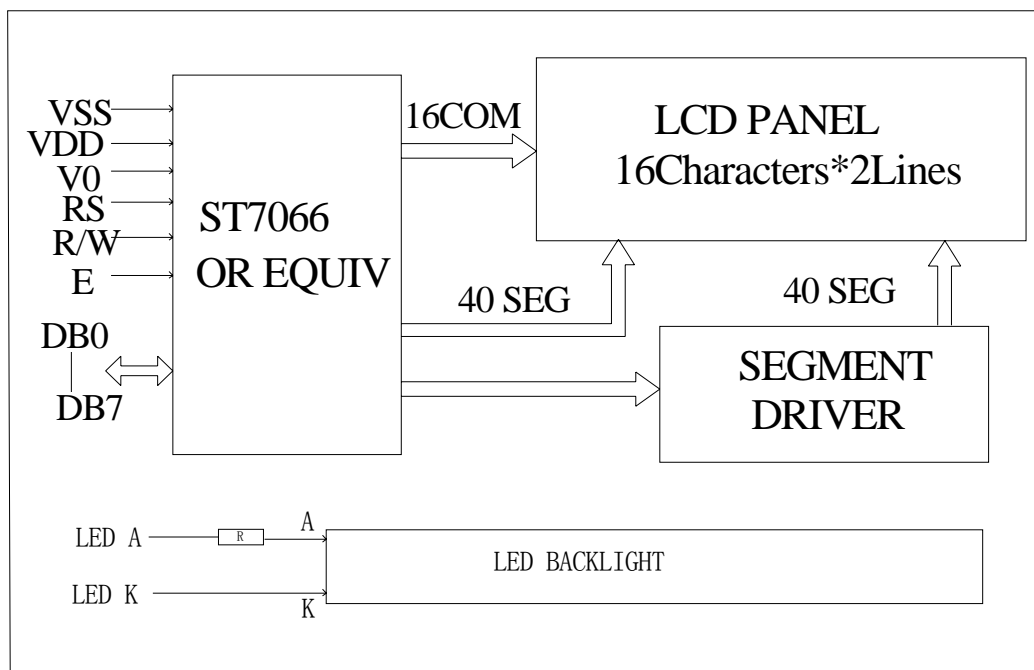


图 1:JLX1602A 字符型模块的电路框图

接口方面,有 8 条数据线,三条控制线。可与微处理器或微控制器相连,通过送入数据和指令,就可使模块正常工作。也可选择 4 条数据线,三条控制线。

4. 3 LCD 驱动器和控制器(LCD driver and controller):ST7066

见图 2, ST7066 是用低功耗 CMOS 技术制造的大规模点阵 LCD 控制器(兼带驱动器),和 4Bit/8Bit 微处理器相连,它能使点阵 LCD 显示大小英文字母,数字和符号。应用 ST7066,用户能用少量元件可组成一个完整点阵 LCD 系统。

特性:

- a. 容易和 4Bit/8Bit MPU 相连;
- b. 可选择 5×7 或 5×10 点阵字符;
- c. 显示数据 RAM 容量: $80 \times 8\text{Bit}$ (80 字符);
- d. 字符发生器 ROM 能提供用户所需字符库或标准库;
字库容量: 192 个字符(5×7 点字型);
 32 个字符(5×10 点字型);
- e. DDRAM 和 CGRAM 都能从 MPU 读取数据;(DDRAM 为显示缓冲区;CGRAM 为可自编数据区)
- f. 输出信号: 16 个行扫描信号(common signal),
 40 个列扫描信号(segment signal),本产品通过增加 ST7065 扩展至 80 个列扫描数量。
- g. 电源复位电路;
- h. 显示占空比:
 $1/16\text{duty}(2 \text{ line}, 5 \times 7\text{dots} + \text{Cursor})$;
- i. 振荡电路;
- j. 指令: 11 种;

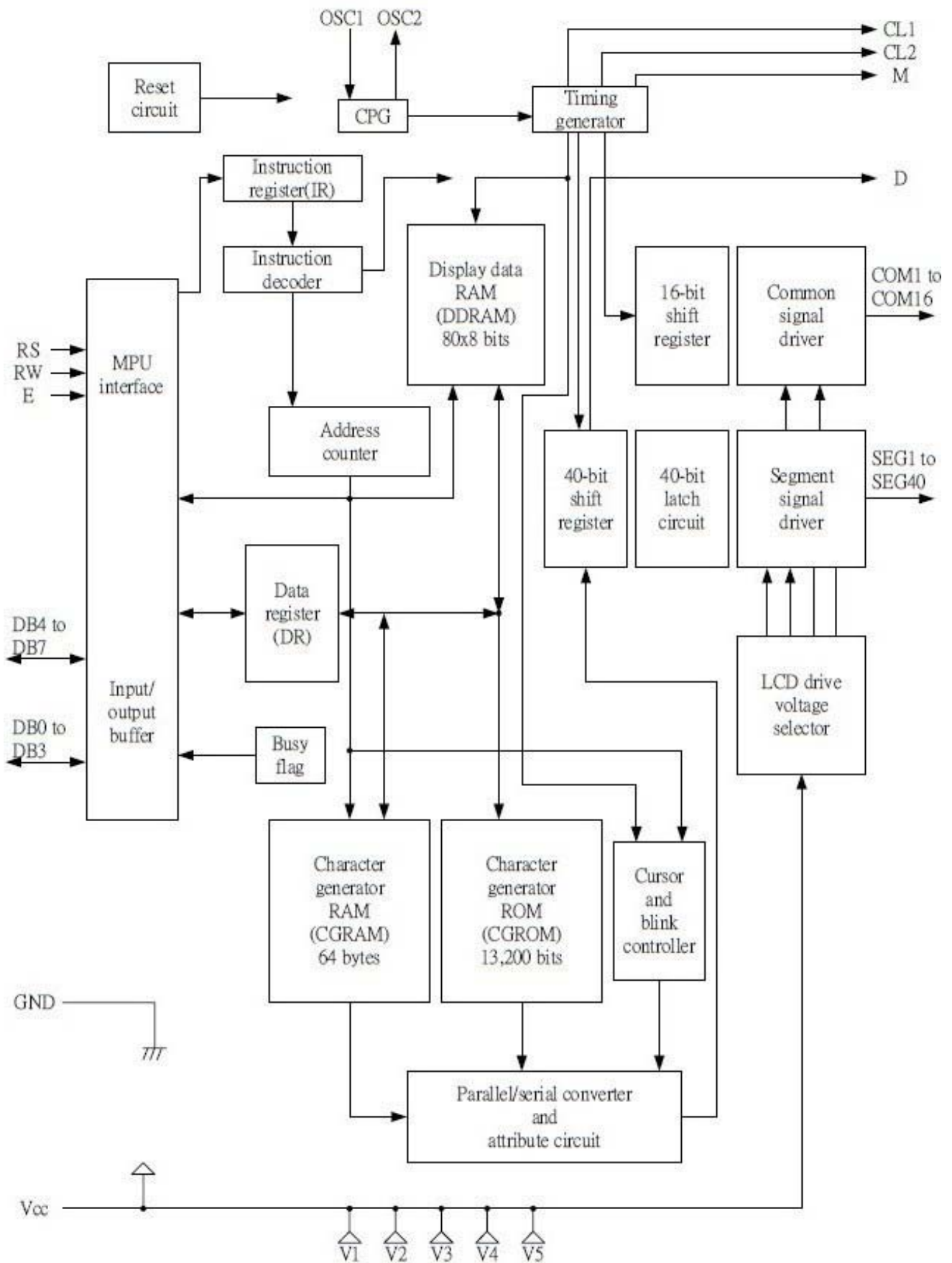


图 2: IC ST7066 内部电路框图

4.4 背光参数

字符模块通常带 LED 背光板。它的性能参数如下:

工作温度: $-20\sim+70^{\circ}\text{C}$;

存储温度: $-30\sim+80^{\circ}\text{C}$;

背光板可显示绿色, 黄绿色, 兰色和白色。背光一般为绿色, 也可为客户设计为其他颜色, 但价格较绿色贵一点。

正常工作电流为: $10\sim 20\text{mA}$ (若 LED 灯数不止一颗, 则乘以相应数量);

工作电压: $4.2\sim 5\text{V}$;

正常工作条件下, LED 可连续点亮 5 万小时;

5. 技术参数

5.1 最大极限参数 (超过极限参数则会损坏液晶模块)

| 名称 | 符号 | 标准值 | | | 单位 |
|----------|-----------|------------|----|-----------|--------------------|
| | | 最小 | 典型 | 最大 | |
| 电路电源 | VDD - VSS | -0.3 | | 7.0 | V |
| LCD 驱动电压 | VDD - V0 | VDD - 13.5 | | VDD + 0.3 | V |
| 静电电压 | | - | - | 100 | V |
| 工作温度 | | -20 | | +70 | $^{\circ}\text{C}$ |
| 储存温度 | | -30 | | +80 | $^{\circ}\text{C}$ |

表 2: 最大极限参数

5.2 直流 (DC) 参数

| 名称 | 符号 | 测试条件 | 标准值 | | | 单位 |
|--------|----------|---------------------------|------|------|-----|----|
| | | | MIN | TYPE | MAX | |
| 输入高电平 | VIH | - | 2.2 | | VDD | V |
| 输入低电平 | VIO | - | -0.3 | | 0.6 | V |
| 输出高电平 | VOH | IOH = 0.2mA | 2.4 | | - | V |
| 输出低电平 | VOO | IOO = 1.2mA | - | | 0.4 | V |
| 工作电流 | IDD | VDD = 5.0V | | 2.0 | | mA |
| 液晶驱动电压 | VDD - V0 | Ta = 0°C | | 4.8 | | V |
| | | Ta = 25°C | | 4.5 | | |
| | | Ta = 50°C | | 4.2 | | |

表 3: 直流 (DC) 参数

6. 读写时序特性

6.1 从 CPU 写到 ST7066 (Writing Data from CPU to ST7066)

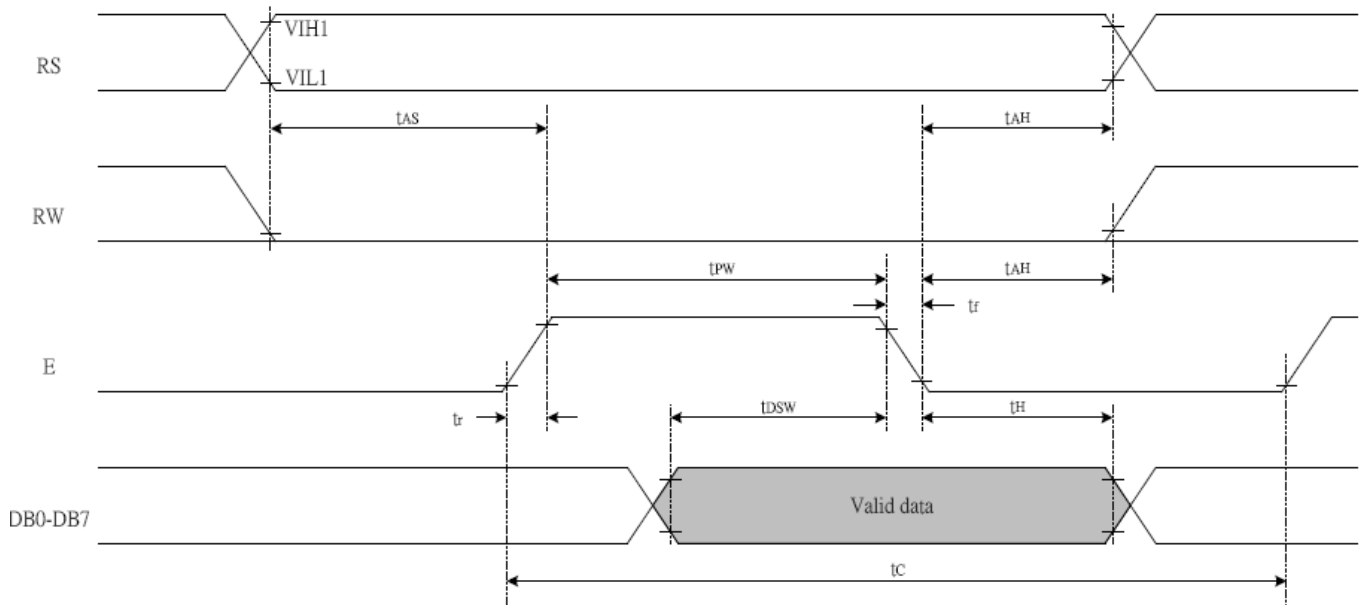


图 3. 从 CPU 写到 ST7066 (Writing Data from CPU to ST7066)

6.2 从 ST7066 读到 CPU (Reading Data from ST7066 to CPU)

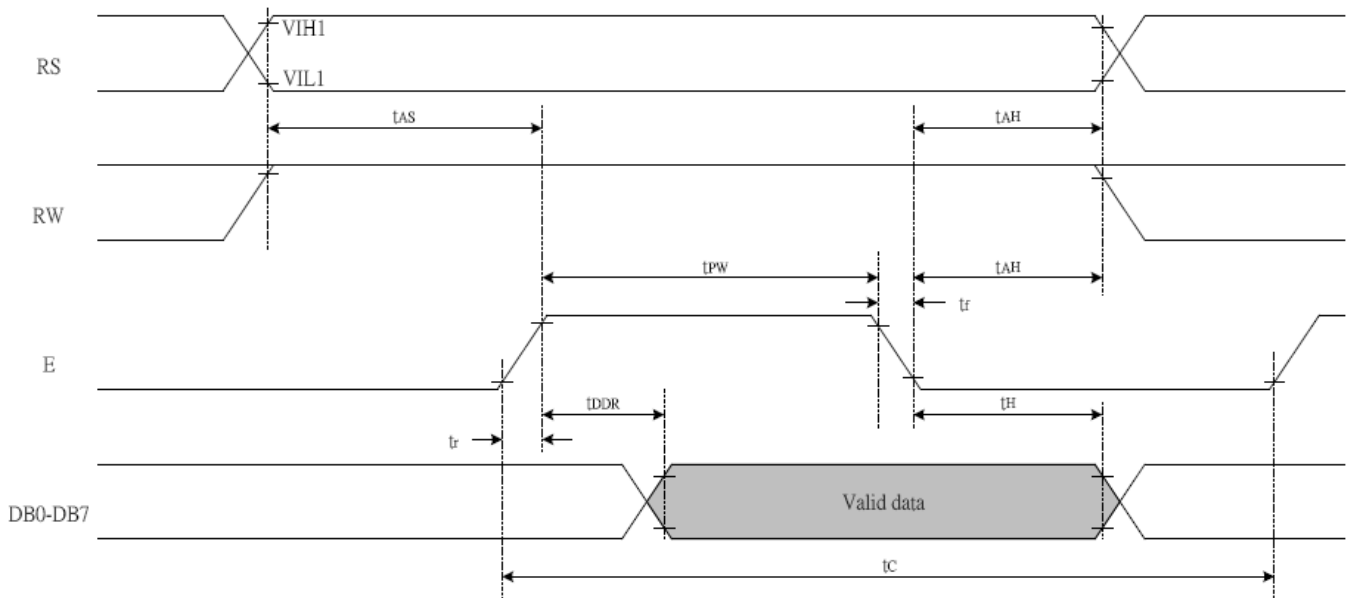


图 4: 从 ST7066 读到 CPU (Reading Data from ST7066 to CPU)

6.3 时序要求 (AC 参数):

写数据到 ST7066 的时序要求:

表 4.

| 项目 | 符号 | 测试条件 | 极限值 | | | 单位 |
|----------|------------|--------------|------|------|-----|----|
| | | | MIN | TYPE | MAX | |
| E上升和下降时间 | $t_{R、tF}$ | 引脚: E | -- | -- | 25 | ns |
| E信号周期 | t_C | 引脚: E | 1200 | | | ns |
| E脉冲宽度 | t_{PW} | 引脚: E | 460 | | | ns |
| 地址建立时间 | t_{AS} | 引脚: E、RS, RW | 0 | -- | -- | ns |
| 地址保持时间 | t_{AH} | 引脚: E、RS, RW | 10 | -- | -- | ns |
| 数据建立时间 | t_{DSW} | | 80 | -- | -- | ns |
| 数据保持时间 | t_H | | 10 | -- | -- | ns |

VDD = 5.0V ± 5%, Ta = 25°C

读数据到 ST7066 的时序要求:

表 5.

| 项目 | 符号 | 测试条件 | 极限值 | | | 单位 |
|----------|------------|--------------|------|------|-----|----|
| | | | MIN | TYPE | MAX | |
| E上升和下降时间 | $t_{R、tF}$ | 引脚: E | -- | -- | 25 | ns |
| E信号周期 | t_C | 引脚: E | 1200 | | | ns |
| E脉冲宽度 | t_{PW} | 引脚: E | 480 | | | ns |
| 地址建立时间 | t_{AS} | 引脚: E、RS, RW | 0 | -- | -- | ns |
| 地址保持时间 | t_{AH} | 引脚: E、RS, RW | 10 | -- | -- | ns |
| 数据建立时间 | t_{DSW} | | -- | -- | 320 | ns |
| 数据保持时间 | t_H | | 10 | -- | -- | ns |

Vcc = 5.0V ± 5%, Ta = 25°C

6.4 电源启动时序要求 (POWER SUPPLY CONDITION):

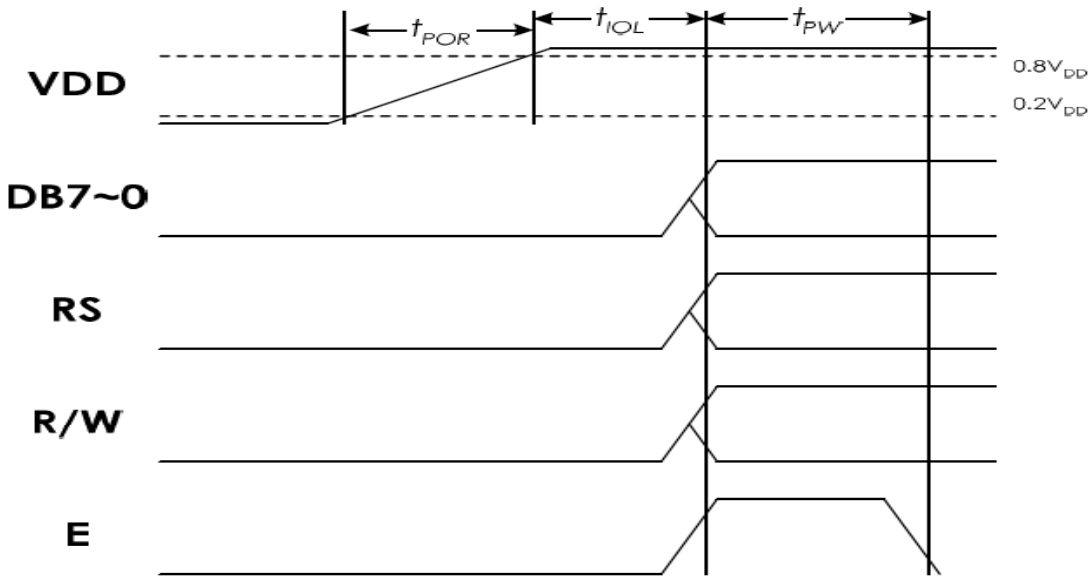


图 5：电源启动时序

表 6：电源启动时序要求

| 项目 | 符号 | 测试条件 | 极限值 | | | 单位 |
|--------------|-----------|---------------|-----|------|-----|----|
| | | | MIN | TYPE | MAX | |
| 电源启动上升时间 | t_{POR} | 引脚：VDD | 0.1 | -- | 100 | ms |
| I/O 口保持低电平时间 | t_{IOL} | I/O 口保持低电平的时间 | 40 | -- | -- | ms |
| 使能信号时间 | t_{PW} | 请参考上述 AC 参数 | | | | |

7. 指令功能:

7.1 寄存器选择功能

表 7.

| RS | R/W | 操作 |
|----|-----|---------------|
| 0 | 0 | 指令寄存器 (IR) 写入 |
| 0 | 1 | 忙标志和地址计数器读出 |
| 1 | 0 | 数据寄存器 (DR) 写入 |
| 1 | 1 | 数据寄存器读出 |

备注: 忙标志为“1”时, 表明正在进行内部操作, 此时不能输入指令或数据, 要等内部操作结束, 忙标志为“0”时才能进行内部操作。

7.2 指令表

格式:

| | | | | | | | | | |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| RS | R/W | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0 |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|

共 11 种指令: 1. 清除, 2. 返回, 3. 输入方式设置, 4. 显示开关, 5. 控制, 移位, 6. 功能设置, 7. CGRAM 地址设置, 8. DDRAM 地址设置, 9. 读忙标志, 10. 写数据到 CG/DDRAM, 11. 读数据由 CG/DDRAM。

指令表

表 8.

| 指令名称 | 指令码 | | | | | | | | | | 说明 | 执行周期 FCP=270KHZ | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|---------------------------------|---------|
| | RS | R/W | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0 | | | |
| 清 屏 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 清除屏幕,置AC为零 | 1.52ms |
| 返 回 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | X | 设 DDRAM 地址为零,显示回原位,DDRAM 内容不变 | 1.52 ms |
| 输入方式设置 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | I/D | S | | 设光标移动方向并指定整体显示是否移动 | 37us |
| 显示开关控制 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | D | C | B | | 设整体显示开关(D),光标开关(C),及光标位的字符闪耀(B) | 37us |
| 移 位 | 0 | 0 | 0 | 0 | 0 | 1 | S/C | R/0 | X | X | | 移动光标或整体显示,同时不改变 DDRAM 内容 | 37us |
| 功能设置 | 0 | 0 | 0 | 0 | 0 | D0 | N | F | X | X | | 设接口数据位数(D0),显示行数(0),及字型(F) | 37us |
| CGRAM 地址设置 | 0 | 0 | 0 | 1 | AC5 | AC4 | AC3 | AC2 | AC1 | AC0 | | 设 CGRAM 地址,设置后 CGRAM 数据被发送和接收 | 37us |
| DDRAM 地址设置 | 0 | 0 | 1 | AC6 | AC5 | AC4 | AC3 | AC2 | AC1 | AC0 | | 设 DDRAM 地址,设置后 DDRAM | 37us |
| 读忙信号(BF)及地址计数器 | 0 | 1 | BF | AC6 | AC5 | AC4 | AC3 | AC2 | AC1 | AC0 | | 读忙信号位(BF)判断内部操作正在执行并读地址计数器内容 | 0us |
| 写数据 CG/DD RAM | 1 | 0 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | | 写数据到 CG 或 DDRAM | 37us |
| 读数据由 CG/DD RAM | 1 | 1 | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 | | 读数据由 CG 或 DDRAM | 37us |

| | | |
|--|---|---|
| <p>I/D 1:增量方式, 0:减量方式</p> <p>S 1:移位</p> <p>S/C 1:显示移位, 0:光标移位</p> <p>R/0 1:右移, 0:左移</p> <p>D0 1:8位, 0:4位</p> <p>N 1:2行, 0:1行</p> <p>F 1:5×10, 0: 5×7</p> <p>BF 1:内部操作, 0:接收指令</p> <p>RS :寄存器选择</p> <p>R/W :读/写</p> | <p>DDRAM:显示数据 RAM</p> <p>CGRAM:字符生成 RAM</p> <p>AC0~AC6:用于 DD 和 CGRAM 地址的地址计数器</p> | <p>执行周期随主频率改变而改变</p> <p>例如:当 Fosc 或 fcp 为 250KHZ</p> <p>37us*270K/250K= 40 us</p> |
|--|---|---|

7.3 字符库及对应关系

7.3.1 显示位与 DD RAM 地址的对应关系

表 9.

| 显示位序号 | | DD RAM 地址 | | | | | | | | | | |
|-------------------|-----|-----------|----|----|----|----|-------|----|--|--|--|--|
| DD RAM 地址(HEX) | 第一行 | 00 | 01 | 02 | 03 | 04 | | 15 | | | | |
| | 第二行 | 40 | 41 | 42 | 43 | 44 | | 55 | | | | |

7.3.2 标准字符库

下表所示的是字符库的内容, 字符码和字形的对应关系。例如“A”的字符码为 41 (HEX), “B”的字符码为 42 (HEX)。

NO.7066-0A

| b7-b4 b3-b0 | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 | 1010 | 1011 | 1100 | 1101 | 1110 | 1111 | |
|----------------|------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|---|
| 0000 | CG RAM (1) | | | 0 | 1 | A | Q | a | q | | | | — | 夕 | 三 | α | p |
| 0001 | (2) | | ! | 1 | A | Q | a | q | | | | 。 | ア | チ | △ | ä | g |
| 0010 | (3) | | " | 2 | B | R | b | r | | | | 「 | イ | ウ | × | β | θ |
| 0011 | (4) | | # | 3 | C | S | c | s | | | | 」 | ウ | テ | ε | ε | ∞ |
| 0100 | (5) | | \$ | 4 | D | T | d | t | | | | 、 | エ | ト | ト | μ | Ω |
| 0101 | (6) | | % | 5 | E | U | e | u | | | | 、 | オ | ナ | 1 | σ | Ü |
| 0110 | (7) | | & | 6 | F | V | f | v | | | | ヲ | カ | ニ | ヨ | ρ | Σ |
| 0111 | (8) | | ' | 7 | G | W | g | w | | | | ア | キ | ヌ | フ | g | π |
| 1000 | (1) | | (| 8 | H | X | h | x | | | | イ | ウ | ホ | リ | フ | 又 |
| 1001 | (2) | |) | 9 | I | Y | i | y | | | | ウ | ウ | ル | リ | フ | 又 |
| 1010 | (3) | | * | : | J | Z | j | z | | | | エ | コ | ロ | レ | j | キ |
| 1011 | (4) | | + | ; | K | [| k | [| | | | オ | サ | ヒ | ロ | × | 又 |
| 1100 | (5) | | , | < | L | ¥ | l | l | | | | ト | シ | フ | ワ | φ | 又 |
| 1101 | (6) | | - | = | M |]n |]n |]n | | | | ユ | ズ | ハ | フ | ト | ÷ |
| 1110 | (7) | | . | > | N | ^n | ^n | ^n | | | | ヨ | セ | ホ | ハ | ん | |
| 1111 | (8) | | / | ? | O | _o | _o | _o | | | | ウ | リ | ヌ | ハ | ö | ■ |

表 10. ST7066-0A 字库表

NO.7066-0B

| b7-b4 b3-b0 | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 | 1010 | 1011 | 1100 | 1101 | 1110 | 1111 |
|----------------|------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 0000 | CG RAM (1) | 士 | 0 | 0 | P | * | P | G | E | A | 1 | r | M | B | C | |
| 0001 | (2) | ≡ | ! | 1 | A | a | 9 | 0 | a | i | 7 | J | + | y | U | |
| 0010 | (3) | 7 | " | 2 | B | b | r | e | E | o | ° | e | S | o | X | |
| 0011 | (4) | L | # | 3 | C | c | s | a | o | U | ' | P | 7 | e | ψ | |
| 0100 | (5) | 7 | \$ | 4 | D | d | t | a | o | o | ' | 4 | 7 | z | o | |
| 0101 | (6) | 7 | % | 5 | E | e | u | a | o | E | b | † | Δ | n | 7 | |
| 0110 | (7) | 7 | & | 6 | F | f | v | a | o | * | W | ↓ | θ | θ | ≡ | |
| 0111 | (8) | 7 | ' | 7 | G | g | w | S | U | R | X | ÷ | A | L | 4 | |
| 1000 | (1) | 7 | (| 8 | H | h | x | e | o | * | ÷ | ÷ | E | K | ≡ | |
| 1001 | (2) | 7 |) | 9 | I | i | w | e | o | i | Σ | Γ | Π | Λ | Δ | |
| 1010 | (3) | * | * | : | J | j | z | e | o | A | Σ | Γ | Σ | μ | P | |
| 1011 | (4) | 7 | + | : | K | k | c | i | R | a | * | L | 7 | U | 4 | |
| 1100 | (5) | ≡ | , | < | L | \ | l | i | R | o | * | U | o | z | U | |
| 1101 | (6) | * | - | = | M | m | > | i | a | o | † | - | 7 | π | - | |
| 1110 | (7) | ≡ | . | > | N | ^ | n | ~ | A | o | † | θ | o | p | o | |
| 1111 | (8) | ≡ | / | ? | O | _ | o | Δ | A | o | o | ≡ | θ | o | o | o |

表 11. ST7066-0B 字库表

7.5.3 自编字库 (CGRAM)

字符码 (DDRAM DATA), CGRAM ADDRESS 与自编字形 (CGRAM DATA) 之间关系如下面 2 个表 所示:

表 12. 5 × 7 点阵字符模式 (注明的 “SST”)

| DD RAM Data (字符代码) | | CG RAM (地址) | | CG RAM Data (字符模式) |
|--------------------------|--|--|--|--|
| 6 5 4 3 2 1 0 MSB OSB | | 4 3 2 1 0 MSB OSB | | 6 5 4 3 2 1 0 MSB OSB |
| 0 0 0 0 X 0 0 0 | | 0 0 0 0 0 1 0 1 0 0 0 0 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 | | X X X 0 1 1 1 1 1 0 0 0 0 1 0 0 0 0 ↓ 0 1 1 1 0 0 0 0 0 1 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 |
| 0 0 0 0 X 0 0 1 | | 0 0 0 0 0 1 0 1 0 0 0 1 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 | | X X X 0 1 1 1 1 1 0 0 0 0 1 0 0 0 0 ↓ 0 1 1 1 0 0 0 0 0 1 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 |
| 0 0 0 0 X 0 1 0 | | 0 0 0 0 0 1 0 1 0 0 1 0 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 | | X X X 1 1 1 1 1 0 0 1 0 0 0 0 1 0 0 ↓ 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1 0 0 |
| : | | : | | : |
| : | | : | | : |
| : | | : | | : |
| 0 0 0 0 X 1 1 1 | | 0 0 0 0 0 1 0 1 0 1 1 1 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 | | X X X ↓ |

X:表示不用理会

备注:1. 字符码的高 4 位为 0000 时, 它的低 3 位对应于第 1-8 个(000 - 111)自编字形;

2. 字符码的 0 - 2 位对应于 CGRAM 地址的 3 - 5 位;
3. 自编字形的列位置对应于 CGRAM DATA 的 0 - 4 位, 行位置对应于 CGRAM ADDRESS 的 0 - 2 位;
4. X 代表无效位;
5. H 代表显示位;

表 13. 5x10 点字符模式(注明" K11")

| DD RAM Data (字符代码) | | CG RAM (地址) | | CG RAM Data (字符模式) | |
|-----------------------|---------|----------------|-----------|-----------------------|-----|
| 6 5 4 3 2 1 0 | | 4 3 2 1 0 | | 7 6 5 4 3 2 1 0 | |
| MSB | | MSB | OSB | MSB | OSB |
| OSB | | | | | |
| 0 0 0 0 X 0 0 X | | 0 0 0 0 | | X X X 0 0 0 0 0 | |
| | | 0 0 0 H | | 0 0 0 0 0 | |
| | | 0 0 H 0 | | H 0 0 0 H | |
| | | 0 0 0 0 H H | | ↓ H 0 0 H 0 | |
| | | 0 H 0 0 | | H 0 H 0 0 | |
| | | 0 H 0 H | | H H 0 0 H | |
| | | 0 H H 0 | | H 0 H 0 0 | |
| | | 0 H H H | | 0 0 0 H 0 | |
| | | H 0 0 0 | | 0 0 0 0 H | |
| | | H 0 0 H | | 0 0 0 0 0 | |
| | H 0 H 0 | | 0 0 0 0 0 | | |
| 0 0 0 0 X 0 H 0 | | 0 0 0 0 | | X X X 0 0 0 0 0 | |
| | | 0 0 0 H | | H H H H H | |
| | | 0 0 H 0 | | 0 H 0 H 0 | |
| | | 0 H 0 0 H H | | ↓ 0 H 0 H 0 | |
| | | 0 H 0 0 | | 0 H 0 H 0 | |
| | | 0 H 0 H | | 0 H 0 H 0 | |
| | | 0 H H 0 | | 0 H 0 H 0 | |
| | | 0 H H H | | 0 H 0 H 0 | |
| | | H 0 0 0 | | H H H H H | |
| | | H 0 0 H | | 0 0 0 0 0 | |
| | H 0 H 0 | | 0 0 0 0 0 | | |
| | | 0 H H 0 H H | | X X X X X | |
| | | H H 0 0 | | | |
| | | H H 0 H | | | |
| | | H H H 0 | | | |
| | | H H H H | | | |

| | | | | |
|------------------------------|--|--|--|------------------------------|
| <pre> : : : </pre> | | <pre> : : : </pre> | | <pre> : : : </pre> |
| <pre> 0 0 0 0 X H H X </pre> | | <pre> 0 0 0 0 0 0 0 H 0 0 H 0 H H 0 0 H H 0 H 0 0 0 H 0 H 0 H H 0 0 H H H H 0 0 0 H 0 0 H H 0 H 0 </pre> | | <pre> X X X X X X X X </pre> |
| | | <pre> H H H 0 H H H H 0 0 H H 0 H H H H 0 H H H H </pre> | | |

X:表示不用理会

7.4 初始化方法

用户所编的显示程序, 开始必须进行初始化, 否则模块无法正常显示, 下面介绍两种初始化方法;

7.4.1 用内部复位电路进行初始化

如果电路电源能满足图所示的条件的话, 就可实行初始化, 下面指令是在初始化过程中执行的。

清屏 (DISPOAY COEAR);

功能设置 (FUNCTION SET);

D0 = 1: 8Bit 接口数据;

N = 0: 1行显示; F = 0:5×7dot 字形;

显示开/关控制 (DISPOAY ON/OFF CONTR00)

D = 0: 显示关; C = 0: 光标关; B = 0: 消隐关

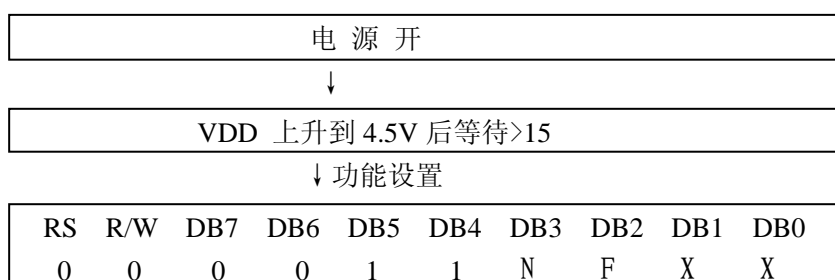
输入方式设置 (ENTRY MODE SET)

I/D = 1: (增量); S = 0: 无移位;

7.4.2 软件复位

如果电路电源不能满足复位电路的要求的话, 那么初始化就要用软件来实现, 过程如下:

8 位接口初始化流程图



↓ 等待>37uS (显示开关控制)

| | | | | | | | | | |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| RS | R/W | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0 |
| 0 | 0 | 0 | 0 | 1 | 1 | × | × | × | × |

↓ 等待>37uS (清除显示)

| | | | | | | | | | |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| RS | R/W | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | D | C | B |

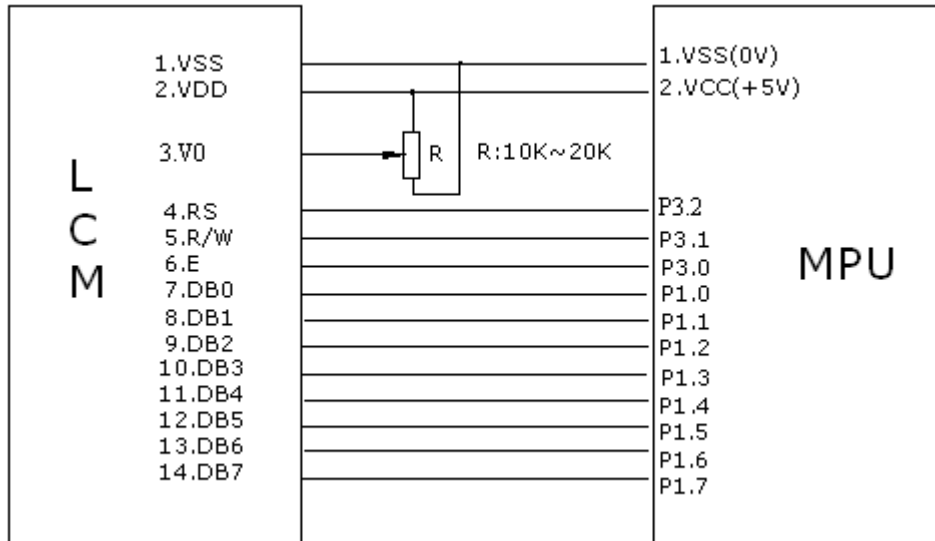
↓ 等待>37uS (进入模式设置)

| | | | | | | | | | |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| RS | R/W | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | I/D | S |

↓
初始化结束

7.4.3 程序举例:

液晶模块与 MPU(以 8051 系列单片机为例)接口图如下:




```

;-----
INITIAL_START:
    CALL DELAY40mS

    MOV A,#38H ;FUNCTION SET
    CALL WRINS_NOCHK ;8 bit,N=1,5*7dot
    CALL DELAY37uS

    MOV A,#38H ;FUNCTION SET
    CALL WRINS_NOCHK ;8 bit,N=1,5*7dot
    CALL DELAY37uS

    MOV A,#0FH ;DISPLAY ON
    CALL WRINS_CHK
    CALL DELAY37uS

    MOV A,#01H ;CLEAR DISPLAY
    CALL WRINS_CHK
    CALL DELAY1.52mS

    MOV A,#06H ;ENTRY MODE SET
    CALL WRINS_CHK ;CURSOR MOVES TO RIGHT
    CALL DELAY37uS
    
```

```

;-----
MAIN_START:
    
```

```

XXXX
XXXX
XXXX
XXXX
.
.
.
.
    
```

```

;-----
WRINS_CHK:
    
```

```

    CALL CHK_BUSY
    
```

```

WRINS_NOCHK:
    
```

```

    CLR RS ;EX:Port 3.0
    CLR RW ;EX:Port 3.1
    SETB E ;EX:Port 3.2
    MOV P1,A ;EX:Port 1=Data Bus
    CLR E
    MOV P1,#FFH ;For Check Busy Flag
    RET
    
```

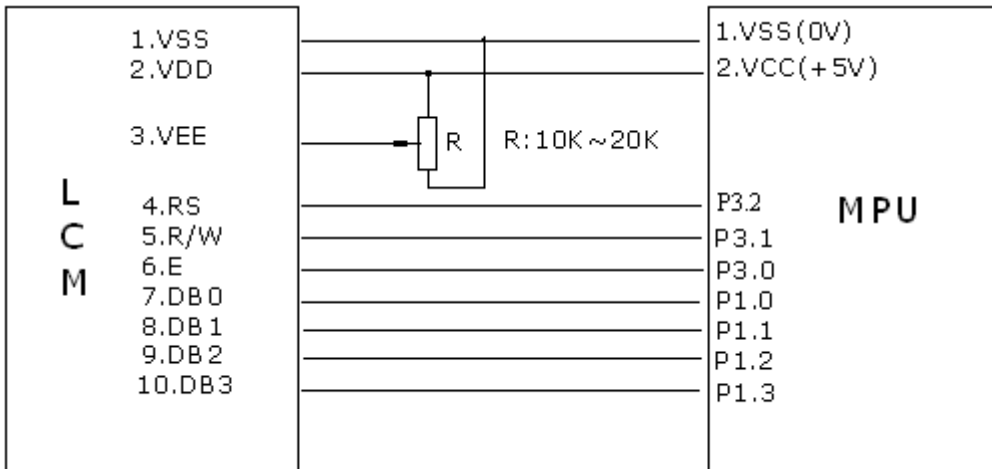
```

;-----
CHK_BUSY: ;Check Busy Flag
    
```

```

    CLR RS
    SETB RW
    SETB E
    JB P1.7,$
    CLR E
    RET
    
```

以下程序为 P3 口控制, 4 BIT 数据线传输;端口连接;
接口图如下:



```

;-----
INITIAL_START:
    CALL DELAY40mS

    MOV A,#38H ;FUNCTION SET
    CALL WRINS_ONCE ;8 bit,N=1,5*7dot
    CALL DELAY37uS

    MOV A,#28H ;FUNCTION SET
    CALL WRINS_NOCHK ;4 bit,N=1,5*7dot
    CALL DELAY37uS

    MOV A,#28H ;FUNCTION SET
    CALL WRINS_NOCHK ;4 bit,N=1,5*7dot
    CALL DELAY37uS

    MOV A,#0FH ;DISPLAY ON
    CALL WRINS_CHK
    CALL DELAY37uS

    MOV A,#01H ;CLEAR DISPLAY
    CALL WRINS_CHK
    CALL DELAY1.52mS

    MOV A,#06H ;ENTRY MODE SET
    CALL WRINS_CHK
    CALL DELAY37uS

;-----
MAIN_START:
    XXXX
    XXXX
    XXXX
    XXXX
    .
    .
    .
    .
    .
    .
    .
    .
    .
    .

;-----
WRINS_CHK:
    CALL CHK_BUSY
WRINS_NOCHK:
    PUSH A
    ANL A,#F0H
    CLR RS ;EX:Port 3.0
    CLR RW ;EX:Port 3.1
    SETB E ;EX:Port 3.2
    MOV P1,A ;EX:Port1=Data Bus
    CLR E
    POP A
    SWAP A
WRINS_ONCE:
    ANL A,#F0H
    CLR RS
    CLR RW
    SETB E
    MOV P1,A
    CLR E
    MOV P1,#FFH ;For Check Bus Flag
    RET

;-----
CHK_BUSY: ;Check Busy Flag
    PUSH A
    MOV P1,#FFH
$1
    CLR RS
    SETB RW
    SETB E
    MOV A,P1
    CLR E
    MOV P1,#FFH
    CLR RS
    SETB RW
    SETB E
    NOP
    CLR E
    JB A.7,$1
    POP A
    RET
    
```