

JHD202A 系列

■ 特性:

显示内容: 20 字符 x 2 行

字符点阵: 5 x 8 点

驱动方式: 1/16D

可供型号:

TN, STN(黄绿模、灰模、黑白模)

反射型、带 EL 或 LED 背光源

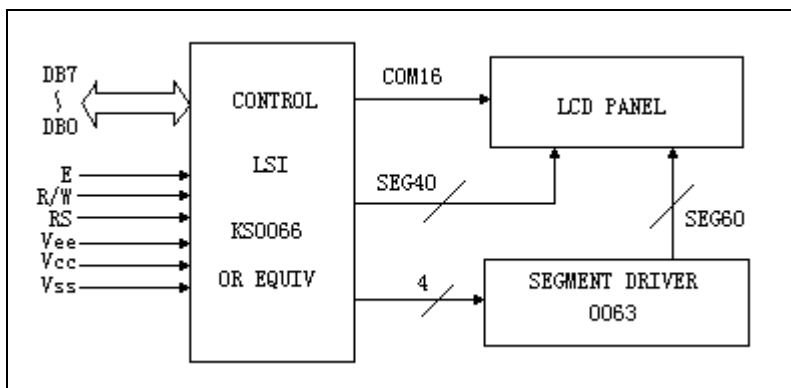
EL/100VAC, 400HZ

LED/4.2VDC

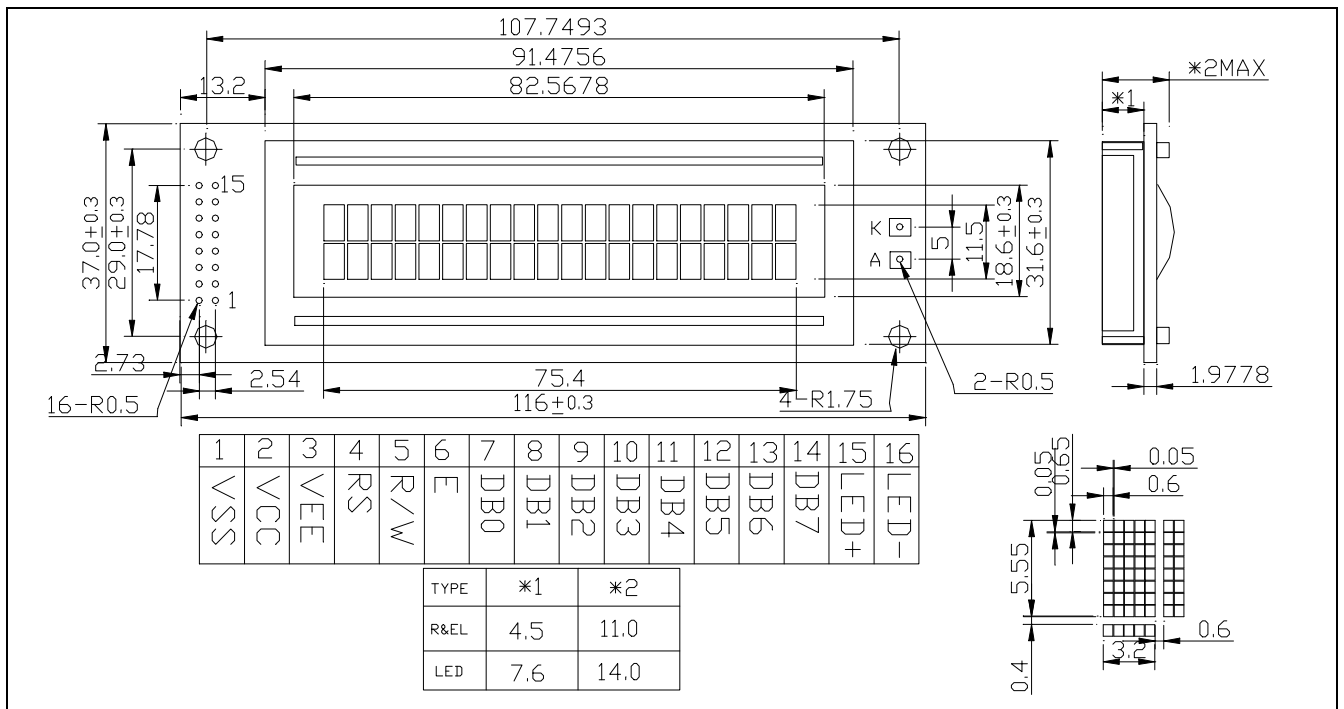
■ 电参数 (V_{DD}=5.0V ± 10%, V_{SS}=0V, T_a=25°C)

名称	符号	测试条件	标准值			单位
			最小	典型	最大	
电源电压	V _{DD} -V _{SS}	-	4.5	5.0	5.5	V
输入高电平	V _{IH}	-	2.2	-	V _{DD}	V
输入低电平	V _{IL}	-	-0.3	-	0.6	V
输入高电压	V _{OH}	-I _{OH} =0.2mA	2.4	-	-	V
输入低电压	V _{OL}	I _{OL} =1.2mA	-	-	0.4	V
工作电流	I _{DD}	V _{DD} =5.0V	-	1.8	3.0	mA

■ 电路图



■ 外形尺寸/显示内容



■ 接口

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
VSS	VCC	VEE	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	LED+	LED-

■外观

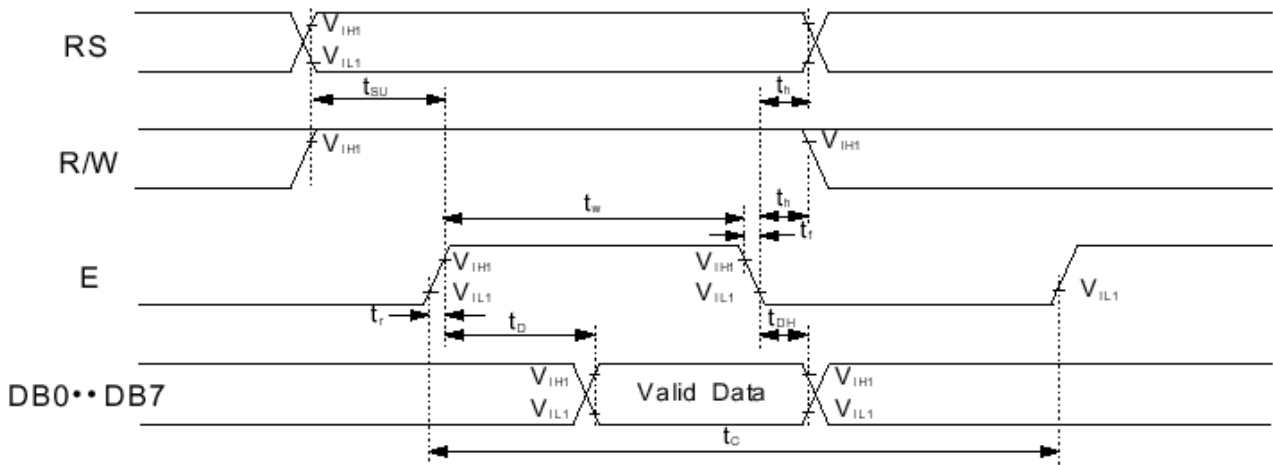
■AC Characteristics

Table 12. AC Characteristics ($V_{DD} = 4.5V \sim 5.5V$, $T_a = -30 \sim +85^{\circ}C$)

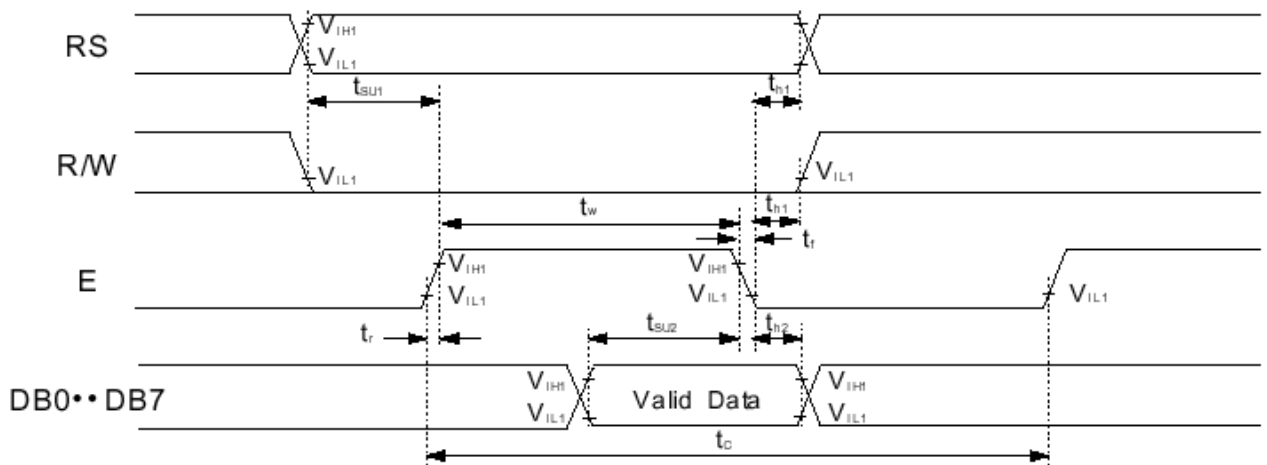
Mode	Characteristic	Symbol	Min.	Typ.	Max.	Unit
Write Mode (Refer to Fig-6)	E Cycle Time	t_c	500	-	-	ns
	E Rise / Fall Time	t_{R,t_F}	-	-	20	
	E Pulse Width (High, Low)	t_w	230	-	-	
	R/W and RS Setup Time	t_{su1}	40	-	-	
	R/W and RS Hold Time	t_{H1}	10	-	-	
	Data Setup Time	t_{su2}	80	-	-	
	Data Hold Time	t_{H2}	10	-	-	
Read Mode (Refer to Fig-7)	E Cycle Time	t_c	500	-	-	ns
	E Rise / Fall Time	t_{R,t_F}	-	-	20	
	E Pulse Width (High, Low)	t_w	230	-	-	
	R/W and RS Setup Time	t_{su}	40	-	-	
	R/W and RS Hold Time	t_H	10	-	-	
	Data Output Delay Time	t_D	-	-	120	
	Data Hold Time	t_{DH}	5	-	-	

Table 13. AC Characteristics ($V_{DD} = 2.7V \sim 4.5V$, $T_a = -30 \sim +85^{\circ}C$)

Mode	Characteristic	Symbol	Min.	Typ.	Max.	Unit
Write Mode (Refer to Fig-6)	E Cycle Time	t_c	1000	-	-	ns
	E Rise / Fall Time	t_{R,t_F}	-	-	25	
	E Pulse Width (High, Low)	t_w	450	-	-	
	R/W and RS Setup Time	t_{su1}	60	-	-	
	R/W and RS Hold Time	t_{H1}	20	-	-	
	Data Setup Time	t_{su2}	195	-	-	
	Data Hold Time	t_{H2}	10	-	-	
Read Mode (Refer to Fig-7)	E Cycle Time	t_c	1000	-	-	ns
	E Rise / Fall Time	t_{R,t_F}	-	-	25	
	E Pulse Width (High, Low)	t_w	450	-	-	
	R/W and RS Setup Time	t_{su}	60	-	-	
	R/W and RS Hold Time	t_H	20	-	-	
	Data Output Delay Time	t_D	-	-	360	
	Data Hold Time	t_{DH}	5	-	-	



Read Mode Timing Diagram

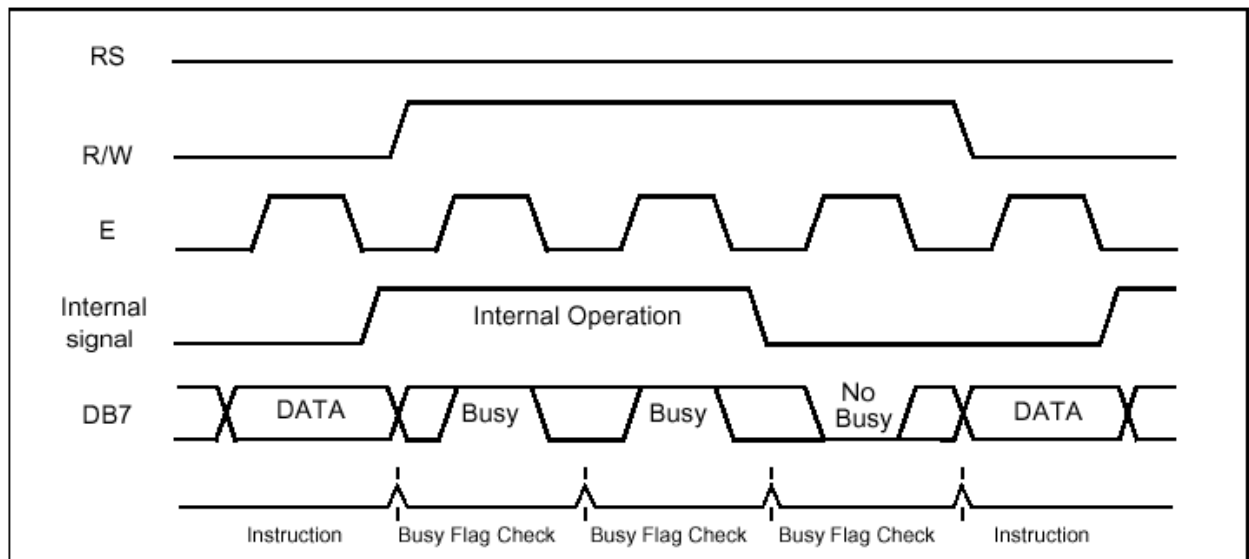


Write Mode Timing Diagram

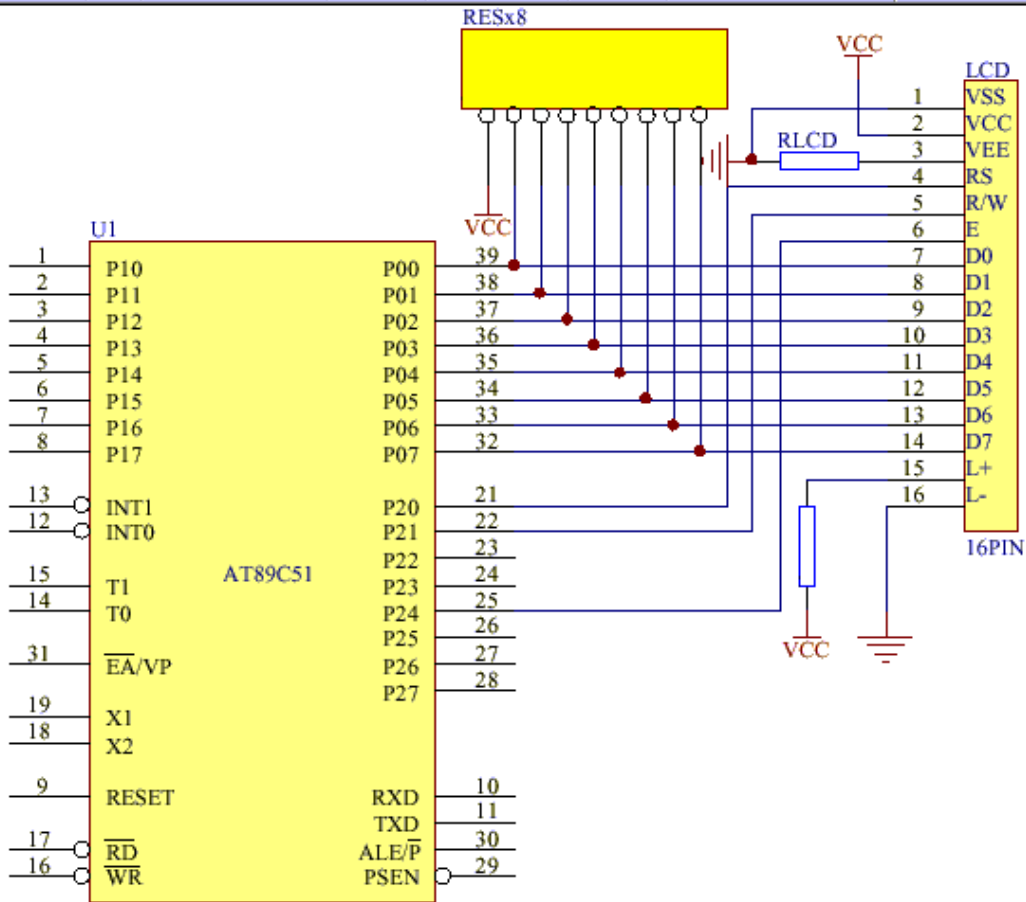
■使用时序

1) Interface with 8-bit MPU

When interfacing data length are 8-bit, transfer is performed at a time through 8 ports, from DB0 to DB7. Example of timing sequence is shown below.



■使用连接



■ CGROM

Table 5. Relationship between Character Code (DDRAM) and Character Pattern (CGRAM)

Character Code (DDRAM data)								CGRAM Address						CGRAM Data								Pattern number
D7	D6	D5	D4	D3	D2	D1	D0	A5	A4	A3	A2	A1	A0	P7	P6	P5	P4	P3	P2	P1	P0	
0	0	0	0	x	0	0	0	0	0	0	0	0	0	x	x	x	0	1	1	1	0	pattern 1
											0	0	1				1	0	0	0	1	
											0	1	0				1	0	0	0	1	
											0	1	1				1	1	1	1	1	
											1	0	0				1	0	0	0	1	
											1	0	1				1	0	0	0	1	
											1	1	0				1	0	0	0	1	
											1	1	1				0	0	0	0	0	
0	0	0	0	x	1	1	1	0	0	0	0	0	0	x	x	x	1	0	0	0	1	pattern 8
											0	0	1				1	0	0	0	1	
											0	1	0				1	0	0	0	1	
											0	1	1				1	1	1	1	1	
											1	0	0				1	0	0	0	1	
											1	0	1				1	0	0	0	1	
											1	1	0				1	0	0	0	1	
											1	1	1				0	0	0	0	0	

■使用举例(仅供参考)

```

#include <reg51.h>
#include <intrins.h>
sbit dc=0xa0;          /*P2.0 LCD 的 RS  21*/
sbit rw=0xa1;          /*P2.1 LCD 的 R/W 22*/
sbit cs=0xa4;          /*P2.4 LCD 的 E   25*/
sfr  lcdbus=0x80;      /*p0LCD 数据 D0=P0.0*/
unsigned int sys10mscounter;
unsigned char syslimitcounter;
char path1[8]={0x00,0x1f,0x00,0x1f,0x00,0x1f,0x00,0x1f};/*自定义符号:横 1*/
char path2[8]={0x1f,0x00,0x1f,0x00,0x1f,0x00,0x1f,0x00};/*自定义符号:横 2*/
char pats1[8]={0x15,0x15,0x15,0x15,0x15,0x15,0x15,0x15};/*自定义符号:竖 1*/
char pats2[8]={0x0a,0x0a,0x0a,0x0a,0x0a,0x0a,0x0a,0x0a};/*自定义符号:竖 2*/
void soft_nop() {}
void soft_10ms()/******12MHZ 提供 10MS 软件延时*****/
{
    register int i;
    for(i=0;i<711;i++);
}
void soft_20ms()/******12MHZ 提供 20MS 软件延时*****/
{
    soft_10ms();
    soft_10ms();
}
void hard_10ms(unsigned int delaytime) /*基于 10MS 的硬件延时*/
{
    sys10mscounter=delaytime;
    while(sys10mscounter);
}
unsigned char data lcdcounter;
bit lcdusing1,lcdusing2;
bit lcd_checkbusy()/*检查 LCD 忙*/
{
    register lcdstate;
    dc=0;          /*dc=1 为数据,=0 为命令.*/
    rw=1;          /*rw=1 为读,=0 为写.*/
    cs=1;          /*cs=1 选通.*/
    soft_nop();
    lcdstate=lcdbus;
    cs=0;
    return((bit)(lcdstate&0x80));
}
void lcd_wrcmd(unsigned char lcdcmd) /*写 LCD 命令*/
{
    lcdusing1=1;
    while(lcd_checkbusy());
    lcdbus=lcdcmd;
    dc=0;          /*dc=1 为数据,=0 为命令.*/
    rw=0;          /*rw=1 为读,=0 为写.*/
    cs=1;          /*cs=1 选通.*/
    soft_nop();
}

```

```

    cs=0;
    lcdbus=0xff;
    lcdusing1=0;
}
void lcd_moveto(char position) /*移动光标到指定位.0-79*/
{
    register cmd=0x80;
    lcdcounter=position;
    if (position > 59)
        position += 0x18;
    else
    {
        if (position > 39) position -= 0x14;
        else
        {
            if (position > 19) position += 0x2c;
        }
    }
    cmd=cmd|position;
    lcd_wrcmd(cmd);
}
void lcd_wrdata(char lcddata) /*在当前显示位置显示数据*/
{
    char i;
    lcdusing2=1;
    while(lcd_checkbusy());
    if(lcdcounter==20) {
        lcd_moveto(20);
        while(lcd_checkbusy());
    }
    if(lcdcounter==40) {
        lcd_moveto(40);
        while(lcd_checkbusy());
    }
    if(lcdcounter==60) {
        lcd_moveto(60);
        while(lcd_checkbusy());
    }
    if(lcdcounter==80) {
        lcd_moveto(0);
        while(lcd_checkbusy());
        lcdcounter=0;
    }
    /*为通用而如此*/
    lcdcounter++;
    lcdbus=lcddata;
    dc=1;          /*dc=1 为数据,=0 为命令.*/
    rw=0;         /*rw=1 为读,=0 为写.*/
    cs=1;         /*cs=1 选通.*/
    soft_nop();
    cs=0;
}

```

```

    lcdbus=0xff;
    lcdusing2=0;
}
void lcd_string(char *strpoint) /*在当前显示位置显示 LCD 字符串*/
{
    register i=0;
    while(strpoint[i]!=0) {
        lcd_wrddata(strpoint[i]);
        i++;
    }
}
void lcd_init() /*初始化*/
{
    lcd_wrcmd(0x38); /*设置 8 位格式, 2 行, 5*7*/
    lcd_wrcmd(0x0c); /*整体显示, 关光标, 不闪烁*/
    lcd_wrcmd(0x06); /*设定输入方式, 增量不移位*/
    lcd_wrcmd(0x01); /*清除显示*/
    lcdcounter=0;
}
void lcd_cls() /*清除显示*/
{
    lcd_wrcmd(0x01);
    lcdcounter=0;
}
void timer0(void) interrupt 1 /*T0 中断*/
{
    TH0=0xd8; /*12M, 10ms*/
    TL0=0xf6;
    TR0=1;
    if(sys10mscounter!=0) sys10mscounter--; /*定时器 10ms*/
    if(syslimitcounter!=0) syslimitcounter--; /*定时器 10ms*/
}
main()
{
    unsigned char j;
    IE=0; P0=0xff; P1=0xff; P2=0xff; P3=0xff; /*初始化 T*/
    lcd_init(); soft_20ms();
    TMOD=0x51;
    TH0=0xd8; /*12M, 10ms*/
    TL0=0xf6;
    TR0=1; ET0=1; EA=1;
    while(1)
    {
        /*全黑、横一、横二、竖一、竖二、U、Q、ABCD... , */
        lcd_init(); /*全黑*/
        for(j=0; j<80; j++) {lcd_wrddata(0xff);}
        hard_10ms(50);
        lcd_init(); /*横一, 可参考自行设计符号*/
        lcd_wrcmd(0x40);
        for(j=0; j<8; j++) lcd_wrddata(path1[j]);
    }
}

```


指令	指令码										说明	指令周期 f _{osc} =250kHz	
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0			
清屏	0	0	0	0	0	0	0	0	0	1	清除屏幕，置 AC 为 0，光标回位。	1.64ms	
光标返回	0	0	0	0	0	0	0	0	1	*	DDRAM 地址为 0，显示回原位，DDRAM 内容不变。	1.64ms	
设置输入方式	0	0	0	0	0	0	0	1	I/D	S	设置光标移动方向并指定显示是否移动。	40μs	
显示开关	0	0	0	0	0	0	1	D	C	B	设置显示开或关 D、光标开关 C、光标所在字符闪烁 B。	40μs	
移位	0	0	0	0	0	1	S/C	R/L	*	*	移动光标及整体显示，同时不改变 DDRAM 内容。	40μs	
功能设置	0	0	0	0	1	DL	N	F	*	*	设置接口数据位数 DL、显示行数 L、字符字体 F。	40μs	
CGRAM 地址设置	0	0	0	1	ACG							设置 CGRAM 地址。设置后发送接收数据。	40μs
DDRAM 地址设置	0	0	1	ADD							设置 DDRAM 地址。设置后发送接收数据。	40μs	
忙标志/读地址计数器	0	1	BF	AC							读忙标志 BF 标志正在执行内部操作并读地址计数器内容。	0μs	
CGRAM/DDRAM 数据写	1	0	写数据							从 CGRAM 或 DDRAM 写数据。		40μs	
CGRAM/DDRAM 数据读	1	1	读数据							从 CGRAM 或 DDRAM 读数据。		40μs	
I/D=1: 增量方式; I/D=0: 减量方式 S=1: 移位 S/C=1: 显示移位; S/C=0: 光标移位 R/L=1: 右移; R/L=0: 左移 DL=1: 8 位; DL=0: 4 位 N=1: 2 行; N=0: 1 行 F=1: 5 x 10 字体 F=0: 5 x 7 字体 BF=1: 执行内部操作; BF=0 可接收指令											DDRAM: 显示数据 RAM CGRAM: 字符发生器 RAM ACG: CGRAM 地址 ADD: DDRAM 地址及光标地址 AC: 地址计数器, 用于 DDRAM 和 CGRAM	执行周期主频改变而改变。 例如当 fcp 或 fosc=270KHZ 时: 40μs x 250/270=37μs	

High 4BIT	MSB	0000	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
LOW 4BIT	RAM (1)		0	@	P	`	p				一	夕	ミ	α	p	
xxxx0000	(2)	!	1	A	Q	a	q				。	ア	チ	ム	ä	q
xxxx0010	(3)	“	2	B	R	b	r				┌	イ	ツ	メ	β	θ
xxxx0011	(4)	#	3	C	S	c	s				└	ウ	テ	モ	ε	∞
xxxx0100	(5)	\$	4	D	T	d	t				、	エ	ト	ヤ	μ	Ω
xxxx0101	(6)	%	5	E	U	e	u				•	オ	ナ	ユ	σ	ü
xxxx0110	(7)	&	6	F	V	f	v				、	カ	ニ	ヨ	ρ	Σ
xxxx0111	(8)	'	7	G	W	g	w				ア	キ	ヌ	ラ	g	π
xxxx1000	(1)	(8	H	X	h	x				イ	ク	ネ	リ	√	ˉ _x
xxxx1001	(2))	9	I	Y	i	y				ウ	ケ	ノ	ル	¨	y
xxxx1010	(3)	*	:	J	Z	j	z				エ	コ	ハ	レ	j	千
xxxx1011	(4)	+	;	K	[k	{				オ	サ	ヒ	ロ	`	万
xxxx1100	(5)	,	<	L	¥	l					ヤ	シ	フ	ワ	φ	円
xxxx1101	(6)	-	=	M]	m	}				ユ	ス	へ	ン	キ	÷
xxxx1110	(7)	.	>	N	^	n	→				ヨ	セ	ホ	、	ˉ _n	
xxxx1111	(8)	/	?	O	_	o	←				ツ	ソ	マ	。	Ö	■