

产品规格书

Product specifications

型号

model

QMDB19264Y01AKY

修订记录

编号	修订内容	日期	版本
1	新文件生成		V. 1

修订		审核	
日期		日期	

目录

contents

编号	内容	页
1	基本参数	
2	结构尺寸图纸	
3	原理图	
4	极限参数值	
5	光电特性	
6	接口定义及说明	
7	MPU 指令说明	
8	程序设计	
9	产品使用说明	
10	附件一	

1. 基本参数

类型	规格	说明
主要硬件	LCD 液晶屏+LED 背光+铁框+PCB 驱动电路 (COB)	
显示模式	STN 正向显示，黄绿底色黑字	
显示类型	半透型 LCD+黄绿色背光	
物理分辨率	192×64	
外形尺寸 (mm)	130.00×65.00×13.00	
可视区尺寸 (mm)	104.00×39.00	
DOT 点尺寸 (mm)	0.458×0.458	
DOT 点间距 (mm)	0.508×0.508	
视角	6 O'clock	12O'clock可选
占空比 (Duty)	1/64	
偏压比 (Bias)	1/9	
控制器	KS0107/KS0108 or Equivalent	
接口类型	8位并口，插针式	
模块电压	5.0V	
工作温度 (℃)	-20~+70	
存储温度 (℃)	-30~+80	

2. 结构尺寸图纸

2-1 产品实物图片



图 1.实物图片

2-2 结构尺寸图纸内容

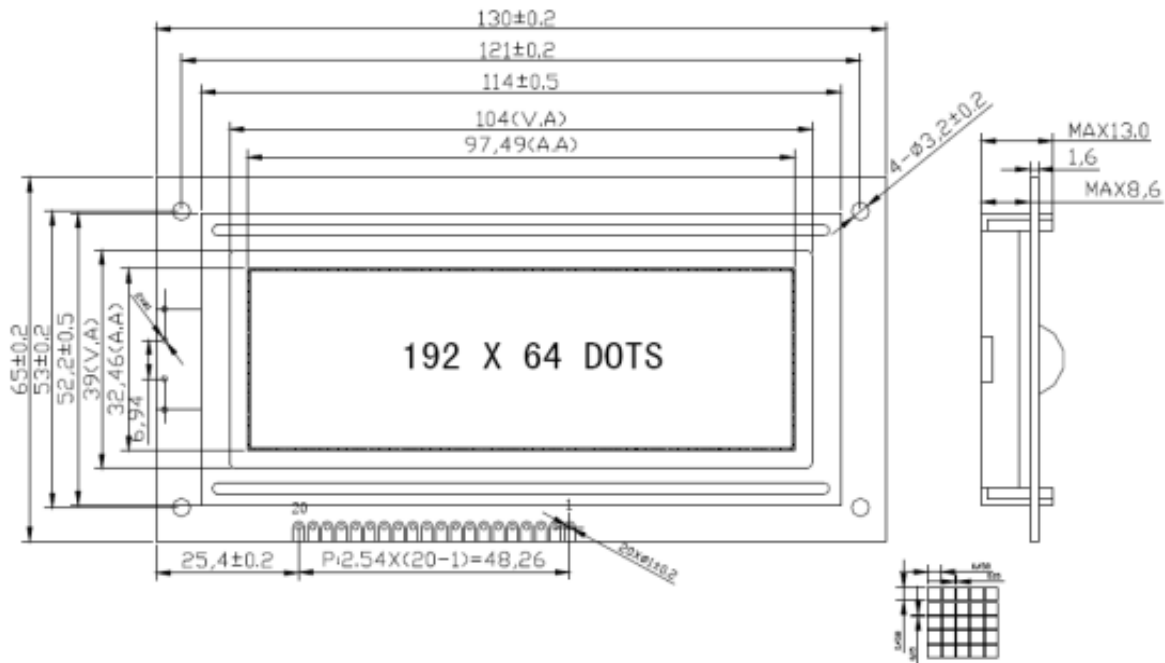


图 2.产品结构图纸

3. 原理图

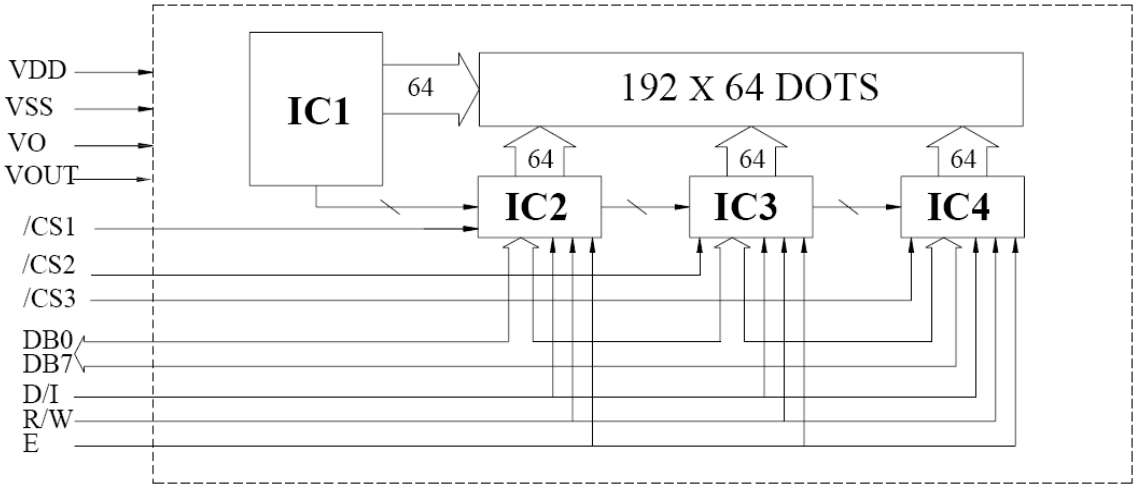


图 3.显示原理图

4. 极限参数值

4-1 极限电压参数

Item	Symbol	Min	Max	Unit	Note
电源电压	Vdd-Vss	-0.3	7.0	V	
LCD 驱动电压	Vdd-Vee	8.0	17.0	V	
I/O 口输入电压	-	Vss	Vdd	-	

4-2 极限环境参数

Item	Min	Max	Unit	Note
工作温度	-20	70	°C	
存储温度	-30	80	°C	
环境湿度	-	90	%RH	

5. 光电特性

5-1 模组电特性参数

Item	Symbol	Condition	Min	Typ	Max	Unit	Note
逻辑电压	Vdd	----	4.5	5.0	5.5	V	
LCD 驱动电压	Vdd-Vo	----	9.8	10.5	11.2	V	
输入电压	VIH	----	2.0	---	Vdd	V	
	VIL	----	Vss	---	0.8	V	
频率	FELM	Vdd=5V	65	78	85	Hz	

5-2 背光电特性参数

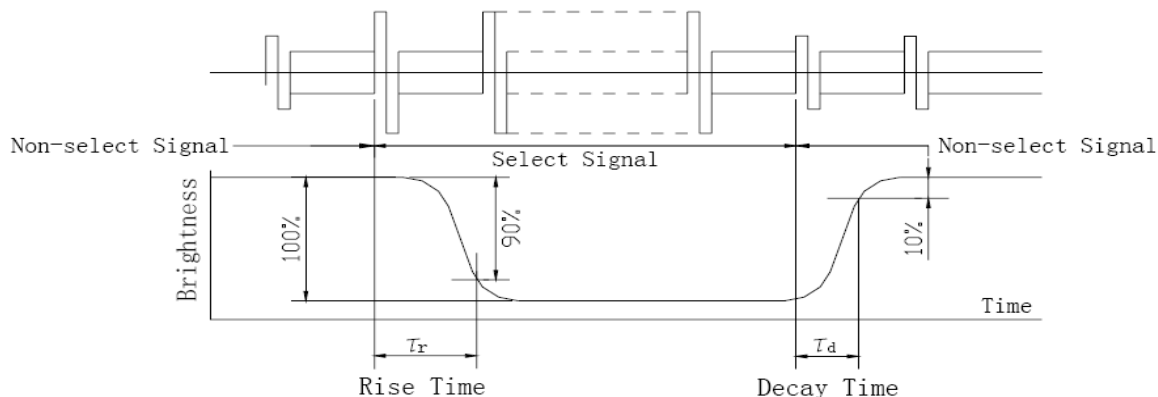
Item	Symbol	Min	Typ	Max	Unit	Condition	Note
正向电压	Vf	2.8	3.0	3.2	V	If=115mA	

5-3 光特性参数（测试环境25℃）

Item	Symbol	Condition	Min	Typ	Max	Unit	Note
Viewing angle	θ	$K \geq 2.0 \phi$	40	---	---	deg	
Contrast ration	K	$\theta = 50 \phi$	---	5	---	---	
Response	Tr	$\theta = 50 \phi$	---	100	120	ms	
Response time(fall)	Tf	$\theta = 50 \phi$	---	60	80	ms	

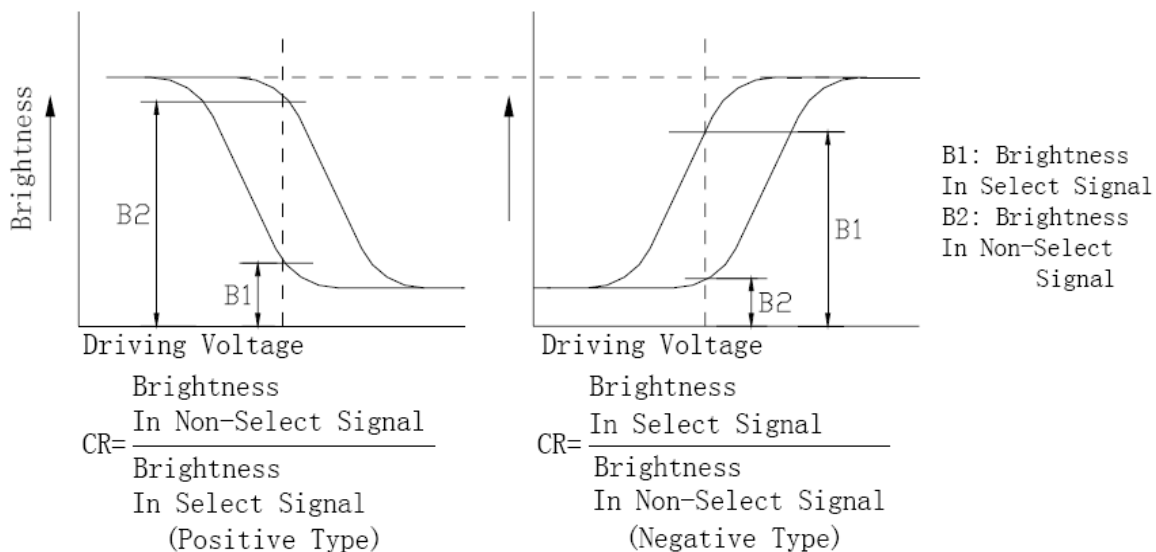
5-4 光学特性测定方法

● Definition of Optical Response Time

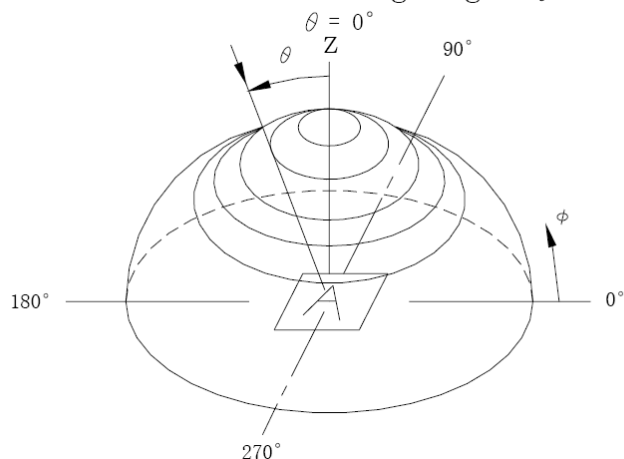


In case of Negative type,
wave form of changing brightness becomes reverse
(Non Select Signals:0%, Select Signals:100%)

● Definition of Contrast Ratio (CR)



● Definition of Viewing Angle θ and ϕ



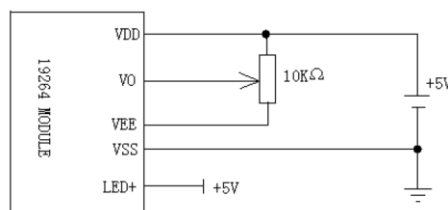
6.接口定义及说明

6-1 数据接口

Pin No.	Symbol	Level	Function	Note
1	VSS	0V	逻辑电源地	
2	VDD	5.0V	逻辑电源正	
3	VO	---	LCD 驱动电压输入端	
4	D/I	H/L	命令或数据, 高电平 数据;低电平: 命令	
5	R/W	H/L	高电平: 读模式 低电平: 写模式	
6	E	H → L	数据读或写的起始信号, 从高到低有效	
7	DB0	H/L	数据信号线 0	
8	DB1	H/L	数据信号线 1	
9	DB2	H/L	数据信号线 2	
10	DB3	H/L	数据信号线 3	
11	DB4	H/L	数据信号线 4	
12	DB5	H/L	数据信号线 5	
13	DB6	H/L	数据信号线 6	
14	DB7	H/L	数据信号线 7	
15	/CS	L	LCD 选通信号, 低有效	
16	/RST	L	复位信号, 低有效	
17	/CS2	L	中 1/3 屏选通端, 低有效	
18	/CS3	L	右 1/3 屏选通端, 低有效	
19	VOOUT	-15V	LCD 模块 DC/DC 电路输出端	
20	LED+	5.0V	背光电源正端	

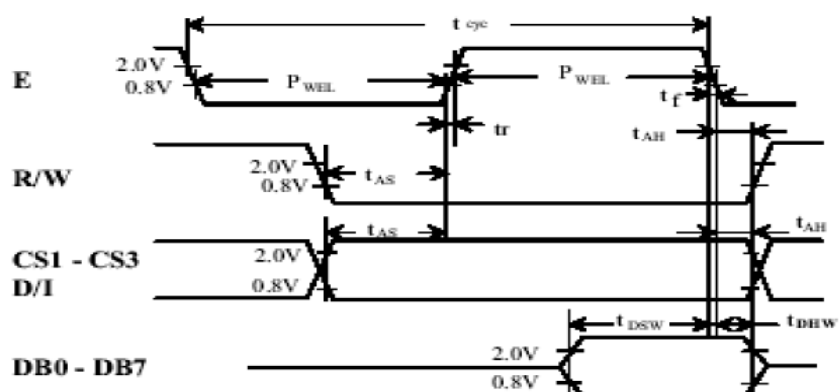
说明:

1. 如需调整参数请与技术联系;
2. 外部对 VCC 接可调电阻 (20K Ω), 调整对比度, 如下图;



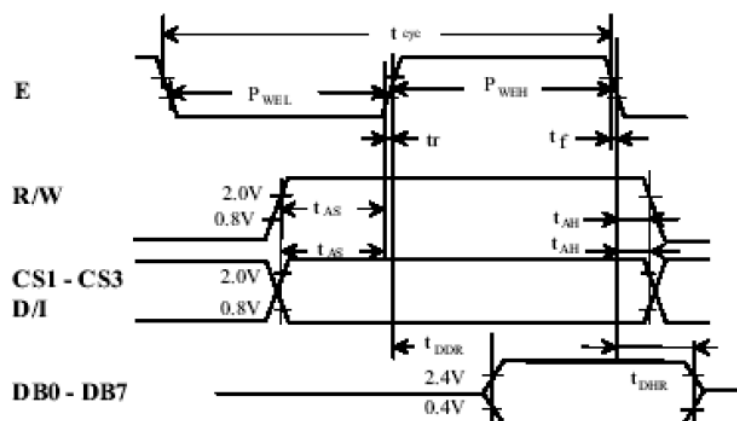
6-2 时序图

6-2-1MPU 写时序



MPU Write Timing

6-2-2MPU读时序



MPU Read Timing

6-2-3MPU 时序表

Item	Symbol	Min	Typ	Max	Unit
E cycle time	t_{CYC}	1,000	--	--	ns
E high level width	P_{WEH}	400	--	--	ns
E low level width	P_{WEL}	400	--	--	ns
E rise time	t_r	--	--	25	ns
E fall time	t_f	--	--	25	ns
Address setup time	t_{AS}	200	--	--	ns
Address hold time	t_{AH}	10	--	--	ns
Data setup time	t_{DSW}	200	--	--	ns
Data delay time	t_{DDR}	--	--	320	ns
Data hold time (Write)	t_{DHW}	10	--	--	ns
Data hold time (Read)	t_{DHR}	20	--	--	ns

7. MPU 指令说明

7-1 Display control instruction

The display control instructions control the internal state of the KS0108. Instruction is received from MPU to KS0108 for the display control. The following table shows various instructions.

Instruction	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Function
Display on/off	L	L	L	L	H	H	H	H	H	L/H	Controls the display on or off. Internal status and display RAM data is not affected. L: OFF, H: ON
Set address (Y address)	L	L	L	H	Y address (0 - 63)						Sets the Y address in the Y address counter.
Set page (X address)	L	L	H	L	H	H	H	Page (0 - 7)			Sets the X address at the X address register.
Display start line (Z address)	L	L	H	H	Display start line (0 - 63)						Indicates the display data RAM displayed at the top of the screen.
Status read	L	H	Busy	L	On/Off	Reset	L	L	L	L	Read status. BUSY L: Ready H: In operation ON/OFF L: Display ON H: Display OFF RESET L: Normal H: Reset
Write display data	H	L	Write data								Writes data (DB0:7) into display data RAM. After writing instruction, Y address is increased by 1 automatically.
Read display data	H	H	Read data								Reads data (DB0:7) from display data RAM to the data bus.

7-2 Functional Description

7-2-1 Display On/Off

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

The display data appears when D is 1 and disappears when D is 0. Though the data is not on the screen with D = 0, it remains in the display data RAM. Therefore, you can make it appear by changing D = 0 into D = 1.

7-2-2 Set Address (Y Address)

S	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	1	AC5	AC4	AC3	AC2	AC1	AC0

Y address (AC0 - AC5) of the display data RAM is set in the Y address counter. An address is set by instruction and increased by 1 automatically by read or write operations of display data.

7-2-3 Set Page (X Address)

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	0	1	1	1	AC2	AC1	AC0

X address(AC0 - AC2) of the display data RAM is set in the X address register. Writing or reading to or from MPU is executed in this specified page until the next page is set.

7-2-4 Display Start Line (Z Address)

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	1	AC5	AC4	AC3	AC2	AC1	AC0

Z address (AC0 - AC5) of the display data RAM is set in the display start line register and displayed at the top of the screen. When the display duty cycle is 1/64 or others(1/32 - 1/64), the data of total line number of LCD screen, from the line specified by display start line instruction, is displayed.

7-2-5 Status Read

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	1	BUSY	0	ON/OFF	RESET	0	0	0	0

- BUSY
When BUSY is 1, the Chip is executing internal operation and no instructions are accepted.
When BUSY is 0, the Chip is ready to accept any instructions.
- ON/OFF
When ON/OFF is 1, the display is off.
When ON/OFF is 0, the display is on.
- RESET
When RESET is 1, the system is being initialized.
In this condition, no instructions except status read can be accepted.
When RESET is 0, initializing has finished and the system is in the usual operation condition.

7-2-6 Write Display Data

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1	0	D7	D6	D5	D4	D3	D2	D1	D0

Writes data (D0 - D7) into the display data RAM. After writing instruction, Y address is increased by 1 automatically.

7-2-7 Read Display Data

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1	1	D7	D6	D5	D4	D3	D2	D1	D0

Reads data (D0 - D7) from the display data RAM. After reading instruction, Y address is increased by 1 automatically.

8. 驱动程序

如附件一

9. 产品使用说明

9-1. 产品出厂说明

产品在我公司出厂前经过 100% 的完全系统的检测, 请在收到产品后确认包装是否完好, 如有问题请直接和我公司联系, 以确保后续的正常使用的, 减少不必要的麻烦;

9-2. 产品说明

打开包装后请确认产品的型号是否正确, 方便正常使用;

9-3 产品使用说明

9-3-1. 静电防护

液晶显示模块中的液晶显示驱动芯片是大规模的 COMS 电路, 极易被静电击穿, 而静电击穿是不可修复, 所以在操作、装配以及使用中都应严防静电, 因此必须注意:

- ※ 不能用手随意接触液晶显示模块的驱动芯片与电路板上的金手指, 若必须接触时, 一定带上防静电手环, 防静电手指套/手套, 最好穿上防静电工作服, 使人体良好接地。
- ※ 工作台, 烙铁及工具良好接地。
- ※ 工作间保持湿度在 RH60% 以上。

9-3-2. 焊接使用说明

- ※ 烙铁头温度小于 $280 \pm 10^{\circ}\text{C}$ 。
- ※ 焊接时间小于 4S。
- ※ 不使用酸性助焊剂。
- ※ 重复焊接不超过 3 次, 且每次重复需间隔 5 分钟以上。
- ※ 焊接后需要清洁, 防止焊接短路和产生信号干扰, 不能把药水渗透到产品内。

9-3-3. 测试使用说明

- ※ 按接口定义连接产品各接口, 注意正、负电源的极性不能搞错, 否则造成过流、过压、烧毁电路上的芯片。
- ※ 正确接线后, 开始测试液晶显示模块时, 先打开逻辑电源, 再打开液晶显示模块的驱动电源; 断电时, 先关掉液晶显示模块的驱动电源, 再关掉逻辑电源。
- ※ 液晶显示模块的显示效果会随着温度的变化而相应变化。

9-4 产品存储说明

- ※ 建议装入聚乙稀口袋(最好有防静电涂层)并将口封住。
- ※ 在 $-10 \sim +35^{\circ}\text{C}$ 之间存储。放暗处, 避强光。不能在表面压放任何物品。
- ※ 严格避免在极限温 / 湿度条件下存放。特殊条件下必须存放时, 也可在 40°C 、85%RH 时, 或 60°C , 小于 60%RH 条件下存放, 但不宜超过 168 小时。
- ※ 在运输途中不能剧烈震动或跌落, 不能有外力压迫, 并且无水、无尘也无日光直射。

10.附件一.驱动程序

```
#include <reg52.h>
#include <stdlib.h>
#include <intrins.h>
#include <stdio.h>

/*****引脚定义*****/
sbit RS =P2^7;    //数据指令
sbit RW =P2^6;    //读写
sbit E =P2^5;     //使能
sbit CSL=P2^4;    //左片选
sbit CSM=P2^3;    //中片选
sbit CSR=P2^2;    //右片选

unsigned char Page;    //页 地址
unsigned char Col;     //列 地址
unsigned int  textsize=192; //屏幕尺寸
unsigned char code BMP1[];
unsigned char code BMP2[];
void Delay(unsigned int MS);
void wtcom(void);
/*检查 Busy          */
void BusyL(void)
{
    CSL= 0;
    CSM= 1;
    CSR= 1;
    wtcom();
}
void BusyM(void)
{
    CSL= 1;
    CSM= 0;
    CSR= 1;
    wtcom();
}
void BusyR(void)
{
    CSL= 1;
```

```

        CSM= 1;
        CSR= 0;
        wtcom();
    }
void wtcom(void)
{
    RS = 0;          //指令
    RW = 1;
    PO = 0xFF;       //输出 0xff 以便读取正确
    E  = 1;
    _nop_();
    while(PO & 0x80); //Status Read Bit7 = BUSY
    E  = 0;
    _nop_();
}
/*根据设定的坐标数据，定位 LCM 上的下一个操作单元位置 */
void Locatexy(void)
{
    unsigned char x,y;
    switch (Col&0xc0)
    {
        case 0:      {BusyL();break;}/*左区 */
        case 0x40:    {BusyM();break;}/*中区 */
        case 0x80:    {BusyR();break;}/*右区 */
    }
    x = Col&0x3F|0x40; // Set Y Address*/
    y = Page&0x07|0xB8; // set Page */
    wtcom();
    RS = 0;
    RW = 0;
    PO = y;           //设置页面地址
    E  = 1;
    _nop_();
    E  = 0;
    _nop_();
    wtcom();
    RS = 0;
    RW = 0;

```

```

        P0 = x;                //设置列地址
        E  = 1;
        _nop_();
        E  = 0;
        _nop_();
    }
    /*写指令                      */
    void WriteCommandL( unsigned char CommandByte )
    {
        BusyL();
        P0 = CommandByte;
        RS = 0;
        RW = 0;
        E  = 1;
        _nop_();
        E  = 0;
        _nop_();
    }
    void WriteCommandM( unsigned char CommandByte )
    {
        BusyM();
        P0 = CommandByte;
        RS = 0;
        RW = 0;
        E  = 1;
        _nop_();
        E  = 0;
        _nop_();
    }
    void WriteCommandR( unsigned char CommandByte )
    {
        BusyR();
        P0 = CommandByte;
        RS = 0;
        RW = 0;
        E  = 1;
        _nop_();
        E  = 0;
    }

```

```

        _nop_();
    }
    /*读数据                                */
    unsigned char ReadData( void )
    {
        unsigned char DataByte;
        Locatexy();    /*坐标定位，返回时保留分区状态不变    */
        RS = 1;
        RW = 1;
        P0 = 0xFF;
        E   = 1;
        _nop_();
        DataByte = P0; /*数据读出到数据口 P1 */
        E   = 0;
        _nop_();
        return DataByte;
    }

    /*****
    /*写数据                                */
    /*****
    void WriteData( unsigned char DataByte )
    {
        Locatexy();
        RS = 1;
        RW = 0;
        P0 = DataByte;
        E   = 1;
        _nop_();
        E   = 0;
        _nop_();
    }

    void LcmClear( void )
    {
        Page = 0;
        Col   = 0;
        for(Page=0;Page<8;Page++)

```

```

        for(Col=0;Col<textsize;Col++)
            WriteData(0);
    }

void LcmInit( void )
{
    WriteCommandL(0x3f);    //开显示
    WriteCommandM(0x3f);
    WriteCommandR(0x3f);
    WriteCommandL(0xc0);    //设置起始地址=0
    WriteCommandM(0xc0);
    WriteCommandR(0xc0);
    WriteCommandL(0x3f);    //开显示
    WriteCommandM(0x3f);
    WriteCommandR(0x3f);
    LcmClear();
    Col = 0;
    Page= 0;
    Locatexy();
}

void LcmPutDots( unsigned char DotByte )
{
    Page = 0;
    Col  = 0;
    for(Page=0;Page<8;Page++)
    {
        for(Col=0;Col<textsize;Col++)
        {
            WriteData( DotByte );
            // DotByte = ~DotByte;
        }
    }
}

void LcmPutDots1( unsigned char DotByte )
{
    Page = 0;
    Col  = 0;
    for(Page=0;Page<8;Page++)
    {

```

```

        for(Col=0;Col<textsize;Col++)
        {
            WriteData( DotByte );
            DotByte = ~DotByte;
        }
    }
}

void LcmPutBMP( unsigned char *puts )
{
    unsigned int X=0;
    Page = 0;
    Col  = 0;
    for(Page=0;Page<8;Page++)
    {
        for(Col=0;Col<textsize;Col++)
        {
            WriteData( puts[X] );
            X++;
        }
    }
}

void Delay(unsigned int MS)
{
    unsigned char us,usn;
    while(MS!=0)
    {
        usn = 2;        //for 12M
        while(usn!=0)
        {
            us=0xf6;
            while (us!=0){us--};
            usn--;
        }
        MS--;
    }
}

void Main( void )
{

```

```

Delay(200);      //等待复位
LcmInit();
LcmClear();
while(1)
{
    LcmPutBMP(BMP1);
    Delay(2000);
    LcmPutBMP(BMP2);
    Delay(2000);
    LcmPutDots( 0xff );Delay(2000);
    LcmPutDots( 0xaa );Delay(2000);
    LcmPutDots( 0x55 );Delay(2000);
    LcmPutDots1( 0xff );Delay(2000);
    LcmPutDots1( 0xaa );Delay(2000);
    LcmPutDots1( 0x55 );Delay(2000);      }
}

unsigned char code BMP1[]={
/*-- 宽度 x 高度=192x64 纵向取模下高位,数据排列:从左到右从上到下 --*/
0x00,0x04,0x04,0x04,0xE4,0xE4,0xAF,0xA4,0xA4,0xA4,0xA4,0xA4,0xA4,0xA4,0xA4,0xAF,
    0xF4,0xF4,0x26,0x04,0x04,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x04,0x04,0x04,0x04,
    0xF4,0xF4,0x04,0x8C,0x8C,0x55,0x26,0x26,0x54,0x8C,0x8C,0x04,0xF4,0xF4,0x04,0x06,
    0x06,0x04,0x00,0x00,0x04,0x04,0x04,0x04,0xF4,0xF4,0x04,0x8C,0x8C,0x55,0x26,0x26,
    0x54,0x8C,0x8C,0x04,0xF4,0xF4,0x04,0x06,0x06,0x04,0x00,0x00,0x00,0x00,0x00,0xFE,
    0x02,0x02,0xE2,0xA2,0xA2,0xB2,0xAE,0xAE,0xA2,0xA2,0xA2,0xA2,0xA2,0xA2,0xF3,0x22,
    0x22,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xFF,
    0x20,0x20,0x20,0x20,0x30,0x30,0x20,0x00,0x00,0x00,0x00,0x00,0x00,0x04,0x04,0x04,
    0xE4,0xE4,0xAF,0xA4,0xA4,0xA4,0xA4,0xA4,0xA4,0xA4,0xAF,0xF4,0xF4,0x26,0x04,
    0x04,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x08,0x08,0x08,0x08,0x0B,0x0B,0x0A,0x0A,0x0A,0x0A,0xFE,0xFE,0x0A,0x0A,0x0A,0x0A,
    0x0B,0x0B,0x08,0x0C,0x0C,0x08,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0xFC,0x04,
    0x05,0x05,0x25,0x35,0x35,0x2D,0x27,0x27,0x25,0x35,0x35,0x25,0x45,0x45,0x84,0x7E,
    0x7E,0x04,0x00,0x00,0x00,0xFC,0xFC,0x04,0x05,0x05,0x25,0x35,0x35,0x2D,0x27,0x27,
    0x25,0x35,0x35,0x25,0x45,0x45,0x84,0x7E,0x7E,0x04,0x00,0x00,0x40,0x30,0x30,0x0F,

```

0x40,0x40,0x23,0x1A,0x1A,0x42,0x82,0x82,0x7E,0x02,0x02,0x02,0x0A,0x0A,0x13,0x60,
0x60,0x00,0x00,0x00,0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x7F,0x7F,
0x40,0x40,0x40,0x40,0x40,0x40,0x60,0x60,0x40,0x00,0x00,0x08,0x08,0x08,0x08,
0x0B,0x0B,0x0A,0x0A,0x0A,0x0A,0xFE,0xFE,0x0A,0x0A,0x0A,0x0A,0x0B,0x0B,0x08,0x0C,
0x0C,0x08,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0x46,0x2F,0x1F,0x1F,0x06,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x0A,0x09,0x08,0x08,0x08,0x08,0x08,0x08,0x08,0x08,0xFC,0x08,0x08,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x80,0x80,0x80,
0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0xC0,
0xC0,0x80,0x00,0x00,0x00,0x00,0x00,0x3E,0x10,0x10,0x90,0xF0,0xF0,0x90,0x9F,0x9F,
0x90,0x90,0x90,0x90,0x90,0x90,0xBE,0x80,0x80,0x00,0x00,0x00,0x80,0x80,0x80,0x80,
0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,0xC0,
0xC0,0x80,0x00,0x00,0x10,0x10,0x10,0xD0,0xFF,0xFF,0x50,0x90,0x90,0x10,0x10,0x10,
0x10,0x10,0x10,0xFF,0x10,0x10,0x10,0x98,0x98,0x10,0x00,0x00,0x84,0x64,0x64,0x24,
0x24,0x24,0x24,0x2F,0x2F,0x24,0xE4,0xE4,0x24,0x2F,0x2F,0x24,0x24,0x24,0x24,0xA6,
0xA6,0x64,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x7F,0x21,0x21,0x21,0x21,
0x21,0x21,0x21,0x21,0x21,0x21,0x21,0x21,0x21,0x21,0x7F,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x80,0x84,0x84,0x82,0x41,0x41,0x40,0x42,0x42,0x24,0x38,0x38,
0x10,0x08,0x08,0x04,0x02,0x02,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x04,0x03,0x03,0x00,0xFF,0xFF,0x00,0x01,0x01,0x00,0xFF,0xFF,
0x41,0x41,0x41,0x41,0x41,0x41,0x41,0xFF,0xFF,0x01,0x00,0x00,0x00,0x42,0x42,0x42,
0x22,0x22,0x12,0x0A,0x0A,0x06,0xFF,0xFF,0x06,0x0A,0x0A,0x12,0x22,0x22,0x62,0x22,
0x22,0x00,0x00,0x00,0x00,0x38,0x38,0x7C,0x44,0x44,0x44,0x7C,0x7C,0x38,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0x92,0x92,0x92,0x92,
0x92,0x92,0x92,0x92,0xF2,0x92,0x92,0x92,0x92,0x92,0x9F,0xC2,0xC2,0x80,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFE,0xFE,0x92,
0x92,0x92,0xFE,0x92,0x92,0x92,0xFE,0xFE,0x40,0x42,0x42,0x4A,0xF2,0xF2,0x4A,0x46,
0x46,0xC0,0x00,0x00,0x00,0x00,0x00,0x80,0x70,0x70,0x00,0x00,0x00,0x00,0xFF,0xFF,
0x00,0x80,0x80,0x40,0x20,0x20,0x30,0x00,0x00,0x00,0x00,0x00,0x00,0x80,0x70,0x70,0x00,
0xFF,0xFF,0x10,0x18,0x18,0x10,0x10,0x10,0x9F,0xA8,0xA8,0x48,0xA8,0xA8,0x0C,0x08,
0x08,0xC0,0x00,0x00,0x02,0x02,0x02,0x02,0x02,0x02,0x02,0x82,0x82,0x42,0xF2,0xF2,

0x0E,0x42,0x42,0x82,0x02,0x02,0x02,0x03,0x03,0x02,0x00,0x00,0x00,0x00,0x00,0x00,
0xFE,0xFE,0x12,0x12,0x12,0x12,0x32,0x32,0xD2,0x12,0x12,0x12,0x12,0x12,0x3F,0x02,
0x02,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x40,0x30,0x30,0x0F,0x00,0x00,0xFC,0x44,
0x44,0x44,0x44,0x44,0x47,0x44,0x44,0x44,0x44,0x44,0xFE,0x04,0x04,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x40,0xC4,0xC4,0x44,
0x44,0x44,0x7F,0x24,0x24,0x26,0x24,0x24,0x00,0x40,0x40,0x80,0x7F,0x7F,0x00,0x01,
0x01,0x00,0x00,0x00,0x00,0x81,0x81,0x40,0x20,0x20,0x10,0x0C,0x0C,0x03,0x00,0x00,
0x03,0x0C,0x0C,0x10,0x20,0x20,0x40,0xC0,0xC0,0x40,0x00,0x00,0x40,0x20,0x20,0x18,
0x07,0x07,0x88,0xB5,0xB5,0x45,0x3D,0x3D,0x04,0x04,0x04,0x7C,0x84,0x84,0x87,0x85,
0x85,0xE1,0x00,0x00,0x00,0x08,0x08,0x04,0x02,0x02,0x01,0x00,0x00,0x00,0xFF,0xFF,
0x00,0x00,0x00,0x00,0x01,0x01,0x03,0x06,0x06,0x00,0x00,0x00,0x10,0x08,0x08,0x04,
0x03,0x03,0x00,0x12,0x12,0x12,0x24,0x24,0x64,0xC1,0xC1,0x02,0x04,0x04,0x08,0x18,
0x18,0x08,0x00,0x00,0x00,0x00,0x00,0x00,0x46,0x46,0x2F,0x1F,0x1F,0x06,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x7F,0x7F,0xC9,0x49,
0x49,0x49,0x49,0x49,0x49,0x49,0x49,0x49,0x7F,0x7F,0x00,0x80,0x80,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x20,0x22,0x22,0x2A,
0x2A,0x2A,0xAA,0x6A,0x6A,0x3A,0x2F,0x2F,0x2A,0x6A,0x6A,0xAA,0x2A,0x2A,0x2A,0x22,
0x22,0x20,0x00,0x00,0x00,0x00,0x00,0xFE,0x02,0x02,0x0A,0x12,0x12,0x62,0x82,0x82,
0x62,0x1A,0x1A,0x02,0xFF,0xFF,0x02,0x00,0x00,0x00,0x00,0x00,0x00,0xFC,0xFC,0x04,
0x04,0x04,0xFE,0x04,0x04,0x20,0x10,0x10,0x0F,0xE8,0xE8,0x08,0x08,0x08,0x28,0x18,
0x18,0x00,0x00,0x00,0x00,0x02,0x02,0x02,0x1E,0x1E,0x62,0x82,0x82,0x02,0x02,0x02,
0x02,0xC2,0xC2,0x32,0x0E,0x0E,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xC0,0xC0,0x30,
0x1E,0x1E,0x10,0x10,0x10,0x10,0xFF,0xFF,0x10,0x10,0x10,0x10,0x10,0x10,0x18,0x10,
0x10,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x08,0x04,0x04,0x22,0x11,0x11,0x09,0x87,
0x87,0x41,0x21,0x21,0x19,0x07,0x07,0x41,0x81,0x81,0x41,0x3F,0x3F,0x01,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x04,0x04,0x04,0x02,
0x01,0x01,0xFF,0x49,0x49,0x49,0x49,0x49,0x49,0x49,0x49,0xFF,0x01,0x01,0x02,0x06,
0x06,0x02,0x00,0x00,0x80,0x40,0x40,0x3F,0x20,0x20,0x10,0x08,0x08,0x06,0x01,0x01,
0x06,0x18,0x18,0x00,0x3F,0x3F,0x40,0x80,0x80,0xE0,0x00,0x00,0x00,0x1F,0x1F,0x08,
0x08,0x08,0x9F,0x40,0x40,0x20,0x10,0x10,0x0C,0x03,0x03,0x0C,0x10,0x10,0x60,0xC0,
0xC0,0x40,0x00,0x00,0x80,0x80,0x80,0x40,0x40,0x40,0x20,0x11,0x11,0x0A,0x04,0x04,
0x0B,0x10,0x10,0x20,0x20,0x20,0x40,0xC0,0xC0,0x40,0x00,0x00,0x41,0x40,0x40,0x42,
0x42,0x42,0x42,0x42,0x42,0x7F,0x7F,0x42,0x42,0x42,0x42,0x43,0x43,0x42,0x60,

[illegible]

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xE1,0x10,0xC0,0x20,0x10,0x00,0xC0,0x60,0x30,
0x18,0x1E,0x1F,0x1F,0x0F,0x07,0x0F,0x0F,0x0F,0x0F,0x1E,0x1E,0x1E,0x1E,0x3C,
0x3F,0x3F,0x7F,0xFD,0x7D,0x1F,0x1F,0x0F,0xFF,0x17,0x36,0xE6,0x34,0xD8,0x30,0xE0,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x20,0x20,0x20,0x20,0xFF,0x00,0x00,0xFF,0x20,0x10,0x08,0x0C,0x00,0x00,0x00,
0x08,0x08,0x08,0xE8,0x28,0x28,0x29,0x2E,0x28,0x28,0x28,0xF8,0x28,0x0C,0x08,0x00,
0x00,0x00,0xF8,0x00,0x7F,0x55,0x55,0x55,0x55,0x7F,0x00,0xF8,0x00,0x00,0x00,
0x40,0x42,0x44,0xCC,0x00,0x20,0x22,0xE2,0x22,0x22,0xE2,0x23,0x32,0x20,0x00,0x00,
0x24,0x24,0x24,0xA4,0xFE,0xA3,0x22,0x00,0x24,0x48,0x00,0xFF,0x00,0x80,0x00,0x00,
0x10,0x10,0x10,0xFF,0x10,0x10,0x88,0x88,0x88,0xFF,0x88,0x88,0x8C,0x08,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x07,0x1F,0x10,0x17,0x0C,0x06,0x03,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0xC0,0xF0,0x0F,0x00,0x00,0x00,0x00,0x01,0xC2,0x41,0x60,0x20,0x11,0x9F,
0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x20,0x60,0x20,0x10,0x10,0x7F,0x00,0x00,0x3F,0x40,0x40,0x40,0x40,0x40,0x70,0x00,
0x00,0x00,0x40,0x23,0x1A,0x42,0x82,0x7E,0x02,0x0A,0x12,0x33,0x60,0x00,0x00,0x00,
0x44,0x44,0x25,0x1D,0x05,0x05,0xFF,0x00,0xFF,0x05,0x05,0x05,0x05,0x7C,0x00,0x00,
0x00,0x40,0x20,0x1F,0x20,0x50,0x4C,0x43,0x40,0x40,0x47,0x48,0x48,0x6E,0x20,0x00,
0x10,0x08,0x06,0x01,0xFF,0x00,0x01,0x02,0x02,0x02,0x02,0xFF,0x01,0x01,0x01,0x00,
0x04,0x44,0x82,0x7F,0x01,0x80,0x81,0x46,0x28,0x10,0x28,0x26,0x41,0xC0,0x40,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x30,0x20,
0x40,0x40,0x40,0x00,0x00,0x00,0x40,0x40,0x40,0x40,0xA0,0xA0,0x90,0x50,0x28,0xEC,
0xFC,0xFE,0x7F,0x7F,0xDF,0x66,0xF2,0xFA,0xFE,0x7E,0x3F,0xCE,0x42,0x21,0x21,0x20,
0x30,0x10,0x20,0x20,0x60,0xC0,0x80,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

[illegible] $\}.$