字符叠加模块 EA30T

技术说明书

一、产品特性

1、可同时显示两行，每行最多可显示 14 字符，显示位置可灵活控制。

2、字符点阵可更新。

3、内含同步发生器，有无外部视频自动转换。

4、RS232 通讯控制，带地址（5 位，范围 0~31），可与上位机通讯。

5、白字暗背景，可叠加于彩色视频图像中。

二、 RS232 通讯控制

波特率 1200

刷新间隔时间>30ms

祥见附录：字符叠加器模块 EA30T 演示版 C 语言程序

1,位置及字符串长度指令：

[地址][地址校对][0][1F][第一行竖向位置][第一行横向位置][第一行字符串长度] [第二行竖向位置][第二行横向位置][第二行字符串长度][校验]；

说明：

地址：模块地址可设，全部悬空时地址为 31

地址校对：计算方式为：地址取反&0x1F+(地址<<3)&0xe0

[0][1F]:本指令特有字节

[竖向位置]：字符串的竖向位置，范围 0~245，上下两行间距〉20。

[横向位置]：字符串的横向位置，限制：横向位置+字符串长度*4<160

[字符串长度]：范围：0~14，限制：横向位置+字符串长度*4<160

[校验]：[第一行竖向位置]~ [第二行字符串长度]的逐个异或。

2，更新字符点阵指令

[地址][地址校对][字符编码][字符编码校对][字符点阵数据 16 字节][校验]；

说明：

地址：模块地址可设，全部悬空时地址为 31

地址校对：计算方式为：地址取反&0x1F+(地址<<3)&0xe0

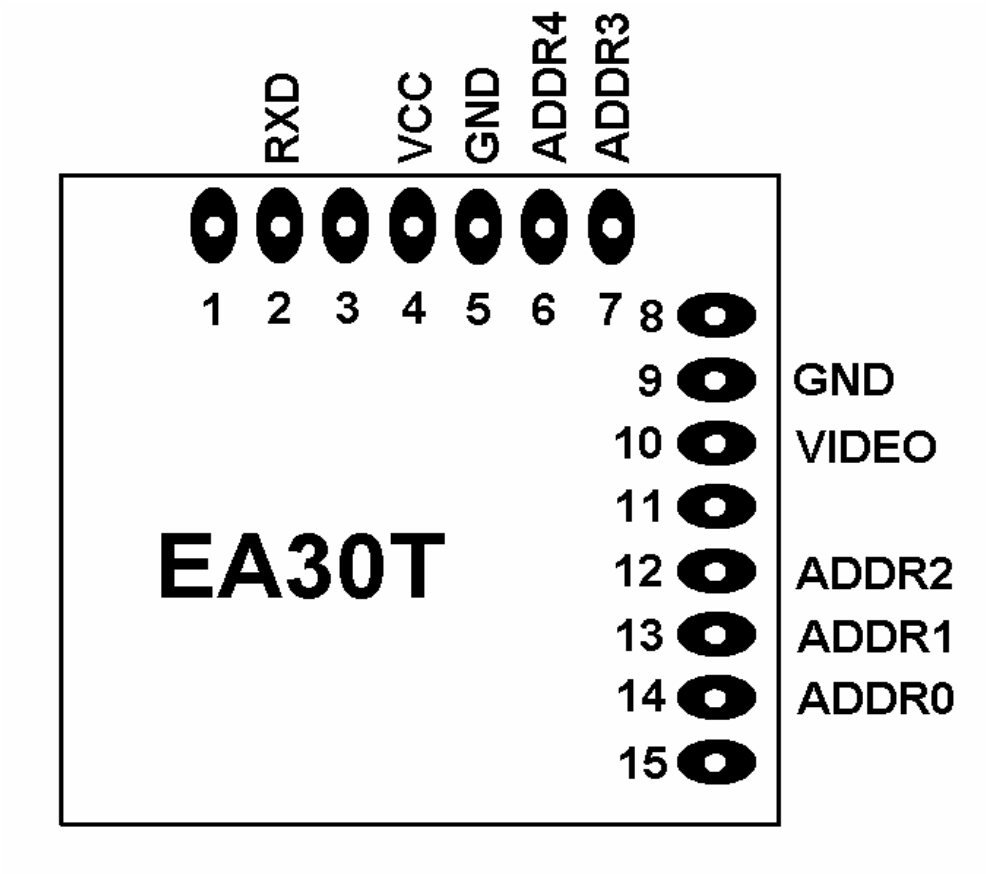
[字符编号]：每个字符的编号，编码规则：第一行依次为 1~14，第二行依次 15~28；

[字符编号校对]：字符编号取反&0x1F+(字符编号<<3)&0xe0

[字符点阵]：共 16 字节数据

[校验]：[字符点阵]的逐个异或。

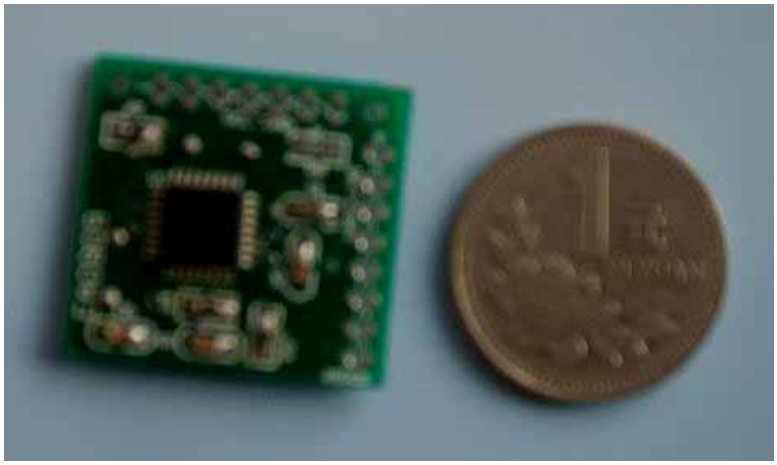
三、EA30T 模块引脚图



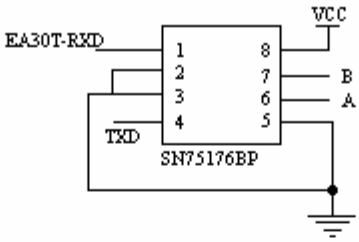
四、尺寸

模块：27.5mm*26mm

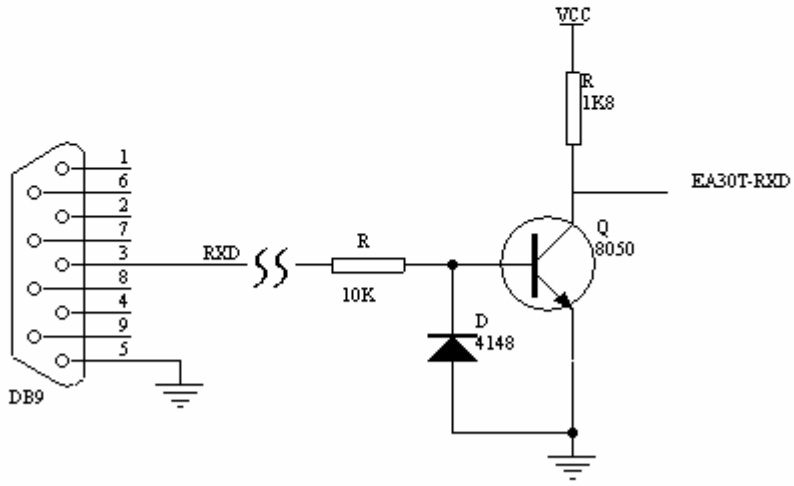
引脚间距：2.54mm



五、与上位机通讯电路



RS485通讯



RS232通讯

六、上位机控制软件

使用说明见软件自带《使用说明》。

七、电器特性

工作电压：5V
耗电电流：10mA（实测 3.5mA）

济南怡安电子有限公司
网址：www.zifuqi.com www.osd-china.com
2011.4.11

附一：字符叠加器模块 EA30T 演示版 C 语言程序

```
/******EA30 控制演示程序*********

    控制芯片型号:89s52
    指令速度：2MHz
    晶振速度:24MHz
    日期：20091110
    编写者：济南怡安电子有限公司

*******************************/
#include "C:\Keil\C51\INC\Atmel\at89X51.h"
#define PinLED P3_5
unsigned int n;
unsigned char i,j,m,Drr,Veri,Typedata[11];
unsigned char code eaType[25][16]={                    //字库
{0x00,0x00,0x1c,0x36,0x63,0x63,0x63,0x63,0x63,0x63,0x63,0x36,0x1c,0x00,0x00,0x00},    //0      0
{0x00,0x00,0x0c,0x0f,0x0c,0x0c,0x0c,0x0c,0x0c,0x0c,0x0c,0x0c,0x3f,0x00,0x00,0x00},    //1      1
{0x00,0x00,0x3e,0x63,0x63,0x63,0x30,0x30,0x18,0x0c,0x06,0x63,0x7f,0x00,0x00,0x00},    //2      2
{0x00,0x00,0x3e,0x63,0x63,0x30,0x1c,0x30,0x60,0x60,0x63,0x33,0x1e,0x00,0x00,0x00},    //3      3
{0x00,0x00,0x30,0x38,0x3c,0x36,0x36,0x33,0x33,0x7f,0x30,0x30,0x7c,0x00,0x00,0x00},    //4      4
{0x00,0x00,0x7f,0x03,0x03,0x03,0x1f,0x37,0x60,0x60,0x63,0x33,0x1e,0x00,0x00,0x00},    //5      5
{0x00,0x00,0x3c,0x36,0x03,0x03,0x1f,0x37,0x63,0x63,0x63,0x36,0x1c,0x00,0x00,0x00},    //6      6
{0x00,0x00,0x7f,0x33,0x33,0x18,0x18,0x0c,0x0c,0x0c,0x0c,0x0c,0x0c,0x00,0x00,0x00},    //7      7
{0x00,0x00,0x3e,0x63,0x63,0x63,0x36,0x1c,0x36,0x63,0x63,0x63,0x3e,0x00,0x00,0x00},    //8      8
{0x00,0x00,0x1c,0x36,0x63,0x63,0x63,0x76,0x7c,0x60,0x60,0x36,0x1e,0x00,0x00,0x00},    //9      9
{0x00,0x00,0x00,0x38,0x38,0x6c,0x6c,0x6c,0xc6,0xc6,0xfe,0xc6,0x83,0x83,0x00,0x00},    //A      10
{0x00,0x00,0x00,0x3f,0x63,0xc3,0xc3,0x63,0x3f,0x63,0xc3,0xc3,0x63,0x3f,0x00,0x00},    //B      11
{0x00,0x00,0x00,0x3c,0x66,0xc3,0xc3,0x03,0x03,0x03,0xc3,0xc3,0x66,0x3c,0x00,0x00},    //C      12
{0x00,0x00,0x00,0x3f,0x63,0xc3,0xc3,0xc3,0xc3,0xc3,0xc3,0xc3,0x63,0x3f,0x00,0x00},    //D      13
{0x00,0x00,0x00,0xfe,0x06,0x06,0x06,0x06,0x7e,0x06,0x06,0x06,0x06,0xfe,0x00,0x00},    //E      14
{0x00,0x00,0x00,0xff,0x03,0x03,0x03,0x03,0x7f,0x03,0x03,0x03,0x03,0x03,0x00,0x00},    //F      15
{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00},    //空格      16
{0x08,0x08,0x08,0x18,0xaa,0x4a,0xe9,0x08,0xc8,0x48,0x48,0x48,0x48,0xc8,0x48,0x00},    //怡      17
{0x02,0x02,0x01,0x11,0x20,0x7c,0x43,0x00,0x3f,0x20,0x20,0x20,0x20,0x3f,0x20,0x00},
{0x40,0x80,0xfc,0x04,0x22,0x60,0x20,0xff,0x10,0x10,0x10,0x60,0x80,0x60,0x18,0x07},    //安      19
{0x00,0x00,0x7f,0x20,0x10,0x00,0x00,0x7f,0x04,0x04,0x02,0x01,0x00,0x03,0x1c,0x08},
{0x00,0x20,0x60,0x20,0x10,0x10,0x08,0x44,0xc2,0x40,0x20,0x10,0x88,0xfc,0x08,0x00},    //公      21
{0x00,0x02,0x02,0x02,0x04,0x08,0x18,0x70,0x20,0x00,0x04,0x08,0x1f,0x18,0x08,0x00},
{0x00,0xfc,0x00,0x00,0xfe,0x00,0x00,0xfc,0x04,0x04,0xfc,0x04,0x04,0x00,0x00,0x00},    //司      22
{0x00,0x3f,0x20,0x20,0x27,0x20,0x20,0x23,0x22,0x22,0x23,0x22,0x20,0x28,0x10,0x00}
};
```

```
void init_serialcom( void ) //串口通信初始设定
{
    SCON = 0xc0 ;          //UART 为模式 3，9 位数据，不接收
    TMOD |= 0x20 ;         //定时器 1 为模式 2,8 位自动重装
    TH1 = 0xcc ;           //Baud:1200 fosc=24MHz
    TL1 = 0xcc ;           //Baud:1200 fosc=24MHz
    IE |= 0x90 ;           //Enable Serial Interrupt
    TR1 = 1 ;              // timer 1 run
    TI=1;
}

void serial () interrupt 4 using 3

{
    if (TI)
    {
        TI = 0 ;
        if(m==0)
        {
            if(i<11)
            {
                TB8=0;
                SBUF=Typedata[i];
            }
            else
            {
                ES=0;
            }
            i++;
        }else
        {
            if(i<4)
            {
                TB8=0;
                SBUF=Typedata[i];
            }
            else if(i<20)
            {
                SBUF=eaType[Typedata[4]][i-4];;
            }else if(i==20)
            {
                SBUF=Veri;
            }else
            {
```

```
                    ES=0;
                }
                i++;
            }
        }
    }

int main (void)
{
  Drr=31;
  init_serialcom();
  EA=1;//打开所有中断
//  ES=1;//打开串口中断

  while(1)
  {
      for(j=0;j<5;j++)                    //字符串向右下方移动，长度增加
      {
          ES=0;
          Typedata[0]=Drr;
          Typedata[1]=((Drr^0x1f)&0x1f) | ((Drr<<3)&0xe0);
          m=0;
          Typedata[2]=m;
          Typedata[3]=((m^0x1f)&0x1f) | ((m<<3)&0xe0);

          Typedata[4]=j*10;          //第一行竖向起始位置
          Veri=Typedata[4];
          Typedata[5]=j*10;          //第一行横向起始位置
          Veri^=Typedata[5];
          Typedata[6]=j*2;           //第一行长度
          Veri^=Typedata[6];

          Typedata[7]=j*20+20;       //第 2 行竖向起始位置
          Veri^=Typedata[7];
          Typedata[8]=j*20+20;       //第 2 行横向起始位置
          Veri^=Typedata[8];
          Typedata[9]=10+j;          //第 2 行长度
          Veri^=Typedata[9];
          Typedata[10]=Veri;

          TB8=1;
          SBUF=Typedata[0];
          i=1;
          ES=1;                      //打开串口中断
```

```
                while(i<12)
                {
                        n++;
                        if(n>255*256)
                        {
                                PinLED =~PinLED;
                                n=0;
                        }
                }
                for(n=0;n<255*256;n++);
                PinLED =~PinLED;
                for(n=0;n<255*256;n++);
                PinLED =~PinLED;
                for(n=0;n<255*256;n++);
                PinLED=~PinLED;
                for(n=0;n<255*256;n++);
                PinLED =~PinLED;
        }


//刷新字符数据
        Typedata[0]=Drr;
        Typedata[1]=((Drr^0x1f)&0x1f) | ((Drr<<3)&0xe0);
        for(m=1;m<25;m++)
        {
                Typedata[2]=m;
                Typedata[3]=((m^0x1f)&0x1f) | ((m<<3)&0xe0);

                Typedata[4]=m-1;//字符编号
                Veri=0;
                for(n=0;n<16;n++)
                {
                        Veri^=eaType[Typedata[4]][n];
                }
                TB8=1;
                SBUF=Typedata[0];
                i=1;
                ES=1;                   //打开串口中断
                while(i<21)
                {
                        n++;
                        if(n>255*256)
                        {
                                PinLED =~PinLED;
```

```
                                n=0;
                        }
                }
                for(n=0;n<255*256;n++);
                PinLED =~PinLED;
}
for(n=0;n<255*256;n++);
PinLED =~PinLED;

Typedata[0]=Drr;
Typedata[1]=((Drr^0x1f)&0x1f) | ((Drr<<3)&0xe0);
m=0;
Typedata[2]=m;
Typedata[3]=((m^0x1f)&0x1f) | ((m<<3)&0xe0);
ES=0;
Typedata[4]=220;          //第一行竖向起始位置
Veri=Typedata[4];
Typedata[5]=80;           //第一行横向起始位置
Veri^=Typedata[5];
Typedata[6]=14;           //第一行长度
Veri^=Typedata[6];

Typedata[7]=240;          //第 2 行竖向起始位置
Veri^=Typedata[7];
Typedata[8]=100;          //第 2 行横向起始位置
Veri^=Typedata[8];
Typedata[9]=14;           //第 2 行长度
Veri^=Typedata[9];
Typedata[10]=Veri;

TB8=1;
SBUF=Typedata[0];
i=1;
ES=1;//打开串口中断
while(i<12)
{
        n++;
        if(n>255*256)
        {
                PinLED =~PinLED;
                n=0;
        }
}
for(n=0;n<255*256;n++);
```

```
            PinLED =~PinLED;
            for(n=0;n<255*256;n++);
            PinLED =~PinLED;
            for(n=0;n<255*256;n++);
            PinLED =~PinLED;
            for(n=0;n<255*256;n++);
            PinLED =~PinLED;

//刷新字符数据
            Typedata[0]=Drr;
            Typedata[1]=((Drr^0x1f)&0x1f) | ((Drr<<3)&0xe0);
            for(m=3;m<27;m++)
            {
                        Typedata[2]=m;
                        Typedata[3]=((m^0x1f)&0x1f) | ((m<<3)&0xe0);

                        Typedata[4]=m-3;//字符编号
                        Veri=0;
                        for(n=0;n<16;n++)
                        {
                                    Veri^=eaType[Typedata[4]][n];
                        }
                        TB8=1;
                        SBUF=Typedata[0];
                        i=1;
                        ES=1;                 //打开串口中断
                        while(i<21)
                        {
                                    n++;
                                    if(n>255*256)
                                    {
                                                PinLED =~PinLED;
                                                n=0;
                                    }
                        }
                        for(n=0;n<255*256;n++);
                        PinLED =~PinLED;
            }
            for(n=0;n<255*256;n++);
            PinLED =~PinLED;
   }
}
```