第十七章　OLED 显示

本章我们将介绍 STM32F103 如何使用 OLED 显示屏来实现显示数字、英文以及汉字的功能，同时我们也将介绍如何使用取模软件。本章分为如下几个部分：

17.1. OLED 简介

17.2. 硬件设计

17.3. 取模软件的使用

17.4. 软件设计

17.5. 显示现象

17.1. OLED 简介

OLED（Organic Light-Emitting Diode）是一种有机发光二极管，具有自发光性，不需要背光源，对比度高，响应速度快，使用温度范围广，功耗低等特点，性能优异。

本章 OLED 学习我们使用的为 0.96 寸 OLED 显示屏，其特点如下：

分辨率：128*64；

模块尺寸：26*26mm；

电源电压：3.3V~5V；

我们使用的是 4 针脚的 OLED 模块，其接线说明如下：

序号	引脚	引脚说明
1	GND	接地引脚
2	VCC	电源引脚，用于提供模块的电源
3	SCL	时钟引脚，用于传输数据时的时钟信号
4	SDA	数据引脚，用于传输数据

17.2. 硬件电路

本章我们需要用到的硬件资源有：

1）STM32F103C8T6 开发板

2）4 针脚 OLED 显示屏

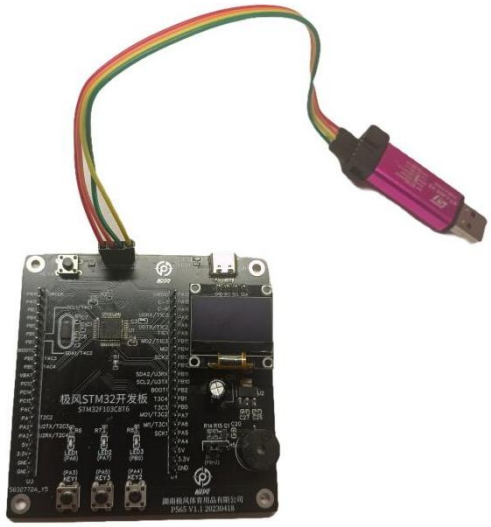
OLED 显示屏的接线电路如下：

21	**PB10**	I/O	FT	PB10	I2C2_SCL/USART3_TX
22	**PB11**	I/O	FT	PB11	I2C2_SDA/USART3_RX

其中,OLED 模块的 GND 脚接单片机的 GND,VCC 脚接单片机的 3.3V，SCL 接单片机的 PB10 脚，SDA 接单片机的 PB11 脚。

实物接线图如下：

将 OLED 模块的 4 个针脚对准开发板的 H4，摁入即可。



17.3. 取模软件的使用

本章学习我们将用到取字模软件，以便后续查找需要的字符。

我们将使用 PCtoLCD2002 完美版对想要显示的数据进行代码提取，具体操作如下：

1．选择模式：字符模式。

2．打开选项，按图 17.3-2 勾选选项：点阵格式选择阴码，取模方式选择列行式，取模走向选择逆向，输出选项全选，自定义格式改为 C51 格式（STM32 和 51 均适用），选完后勾选确认。

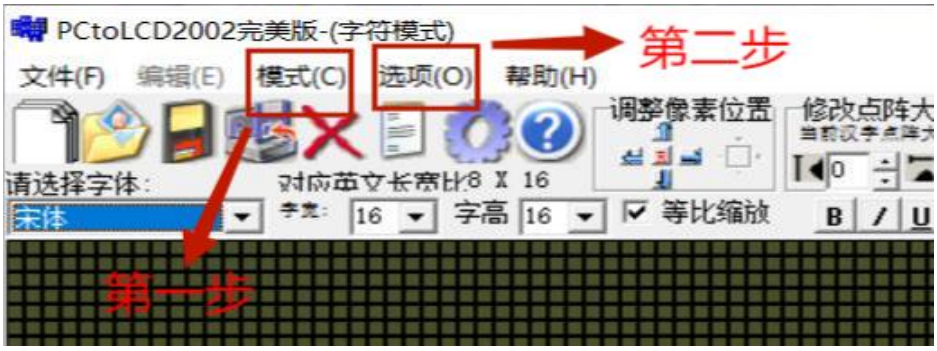


图 17.3-1

图 17.3-2

选项设置好后，我们就可以在下方的文本输入框中输入想要得到的字符，点击生成字模后保存字模即可。以下演示是对汉字进行代码提取，一般选择 16*16 像素大小，而英文像素大小一般为 8*16。

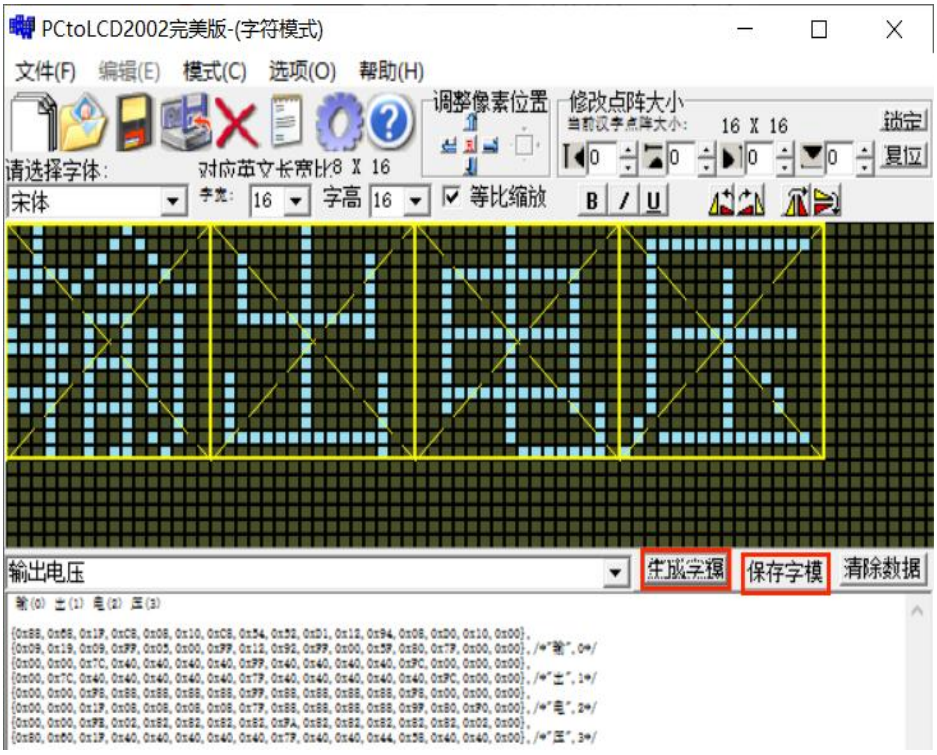
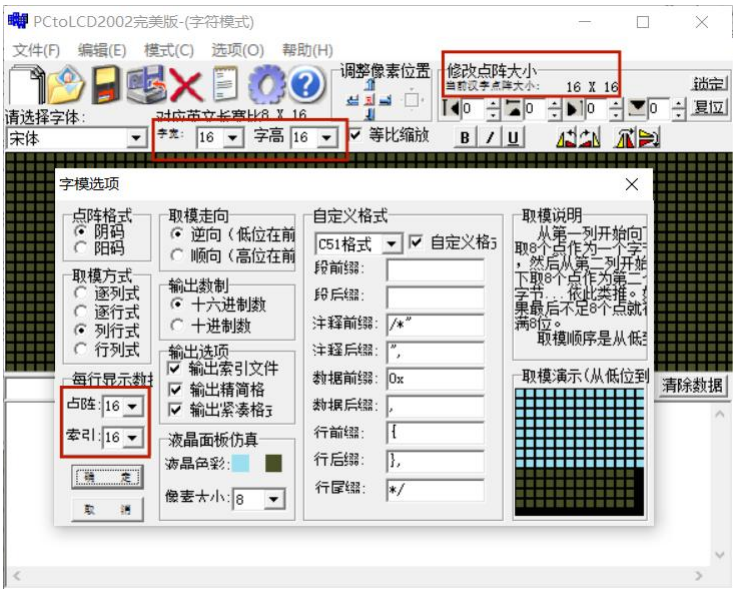


图 17.3-3

若想要更改汉字字体大小，需要同时更改如下三处设置，若设置 16*16 字体大小，则将三处都改为 16*16，若设置 8*8 的大小则更改相应位置，注意合理设置字体大小，否则可能会产生显示不完全的现象。



17.4. 软件设计

打开 OLED 显示工程，我们可以看到工程中拥有 4 个源文件，分别是 OLED.c、main.c 和 OLED.h、OLED_Font.h。OLED.c 文件存放配置 OLED 屏幕以及显示字符的代码，OLED_Font.h 存放 OLED 字模库代码，main.c 文件存放应用代码。

打开 **OLED.c 文件**，代码如下：

```
#include "stm32f10x.h"
#include "OLED_Font.h"

/*引脚配置*/
//PB10~SCL;PB11~SDA;
//x 参数用于指定 SCL（时钟线）和 SDA（数据线）的状态；SCL 线用于发出每个数据位的开始和结束信号，置 1 为开始信号，置 0 为结束信号；
#define OLED_W_SCL(x)		GPIO_WriteBit(GPIOB, GPIO_Pin_10, (BitAction)(x))
#define OLED_W_SDA(x)		GPIO_WriteBit(GPIOB, GPIO_Pin_11, (BitAction)(x))

/*引脚初始化*/
void OLED_I2C_Init(void)
{
	RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOB,ENABLE);
//IO 时钟使能
	RCC_APB1PeriphClockCmd(RCC_APB1Periph_I2C2,ENABLE);		//外设时钟使能

	GPIO_InitTypeDef GPIO_InitStructure;
	GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_OD;		//将端口设置为开漏输出模式
	GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
	GPIO_InitStructure.GPIO_Pin = GPIO_Pin_10;
	GPIO_Init(GPIOB, &GPIO_InitStructure);
	GPIO_InitStructure.GPIO_Pin = GPIO_Pin_11;
	GPIO_Init(GPIOB, &GPIO_InitStructure);

	OLED_W_SCL(1);
```

```
    OLED_W_SDA(1);                  //将 SCL 和 SDA 置高，确保线路在发生任何通信之前以空闲状态启动。可确保线路在发生任何通信之前以空闲状态启动。
}

/*配置 IIC*/
//IIC 开始
void OLED_I2C_Start(void)
{
    OLED_W_SDA(1);
    OLED_W_SCL(1);
    OLED_W_SDA(0);
    OLED_W_SCL(0);
}

//I2C 停止
void OLED_I2C_Stop(void)
{
    OLED_W_SDA(0);
    OLED_W_SCL(1);
    OLED_W_SDA(1);
}

/**
    *函数作用：IIC 写入字节
    *参数 Byte：写入的字节
*/
void OLED_I2C_SendByte(uint8_t Byte)
{
    uint8_t i;
    for (i = 0; i < 8; i++)
    {
        OLED_W_SDA(Byte & (0x80 >> i));
        OLED_W_SCL(1);
        OLED_W_SCL(0);
    }
    OLED_W_SCL(1);  //额外的一个时钟，不处理应答信号
    OLED_W_SCL(0);
}

/**
    *函数作用：OLED 写命令
    *参数 Command：写入的命令
*/
void OLED_WriteCommand(uint8_t Command)
```

```
{
    OLED_I2C_Start();
    OLED_I2C_SendByte(0x78);      //从机地址
    OLED_I2C_SendByte(0x00);      //写命令
    OLED_I2C_SendByte(Command);
    OLED_I2C_Stop();
}

/**
    *函数作用：OLED 写数据
    *参数 Data：写入的数据
*/
void OLED_WriteData(uint8_t Data)
{
    OLED_I2C_Start();
    OLED_I2C_SendByte(0x78);      //从机地址
    OLED_I2C_SendByte(0x40);      //写数据
    OLED_I2C_SendByte(Data);
    OLED_I2C_Stop();
}

/**
    *函数作用：OLED 设置光标位置
    *参数 Y：  以左上角为原点，向下方向的坐标，范围：0~7
    *参数 X：  以左上角为原点，向右方向的坐标，范围：0~127

*/
void OLED_SetCursor(uint8_t Y, uint8_t X)
{
    OLED_WriteCommand(0xB0 | Y);                    //设置 Y 位置
    OLED_WriteCommand(0x10 | ((X & 0xF0) >> 4)); //设置 X 位置高 4 位
    OLED_WriteCommand(0x00 | (X & 0x0F));           //设置 X 位置低 4
位
}

/**
    *函数作用：OLED 清屏
*/
void OLED_Clear(void)
{
    uint8_t i, j;
    for (j = 0; j < 8; j++)
    {
        OLED_SetCursor(j, 0);
```

```
            for(i = 0; i < 128; i++)
            {
                OLED_WriteData(0x00);
            }
        }
    }

    /**
        *函数作用：OLED 初始化
    */
    void OLED_Init(void)
    {
        uint32_t i, j;

        for (i = 0; i < 1000; i++)              //上电延时
        {
            for (j = 0; j < 1000; j++);
        }

        OLED_I2C_Init();                //端口初始化
        OLED_WriteCommand(0xAE);    //关闭显示

        OLED_WriteCommand(0xD5);    //设置显示时钟分频比/振荡器频率
        OLED_WriteCommand(0x80);

        OLED_WriteCommand(0xA8);    //设置多路复用率
        OLED_WriteCommand(0x3F);

        OLED_WriteCommand(0xD3);    //设置显示偏移
        OLED_WriteCommand(0x00);

        OLED_WriteCommand(0x40);    //设置显示开始行
        OLED_WriteCommand(0xA1);    //设置左右方向，0xA1 正常  0xA0 左右
反置
        OLED_WriteCommand(0xC8);    //设置上下方向，0xC8 正常  0xC0 上下
反置

        OLED_WriteCommand(0xDA);    //设置 COM 引脚硬件配置
        OLED_WriteCommand(0x12);

        OLED_WriteCommand(0x81);    //设置对比度控制
        OLED_WriteCommand(0xCF);

        OLED_WriteCommand(0xD9);    //设置预充电周期
```

```
    OLED_WriteCommand(0xF1);

    OLED_WriteCommand(0xDB);    //设置 VCOMH 取消选择级别
    OLED_WriteCommand(0x30);

    OLED_WriteCommand(0xA4);    //设置整个显示打开/关闭
    OLED_WriteCommand(0xA6);    //设置正常/倒转显示

    OLED_WriteCommand(0x8D);    //设置充电泵
    OLED_WriteCommand(0x14);

    OLED_WriteCommand(0xAF);    //开启显示
    OLED_Clear();               //OLED 清屏
}

/**
    *函数作用：    OLED 显示一个字符
    *参数 Line：    起始行位置，范围：1~4
    *参数 Column：起始列位置，范围：1~16
    *参数 Char：    要显示的一个字符，范围：ASCII 可见字符
*/
void OLED_ShowChar(uint8_t Line, uint8_t Column, char Char)
{
    uint8_t i;
    OLED_SetCursor((Line - 1) * 2, (Column - 1) * 8);      //设置光标位置在上半部分
    for (i = 0; i < 8; i++)
    {
        OLED_WriteData(OLED_F8x16[Char - ' '][i]);          //显示上半部分内容
    }
    OLED_SetCursor((Line - 1) * 2 + 1, (Column - 1) * 8); //设置光标位置在下半部分
    for (i = 0; i < 8; i++)
    {
        OLED_WriteData(OLED_F8x16[Char - ' '][i + 8]);      //显示下半部分内容
    }
}

/**
    *函数作用：    OLED 显示字符串
    *参数 Line：    起始行位置，范围：1~4
    *参数 Column：起始列位置，范围：1~16
```

```
    *参数 String：要显示的字符串，范围：ASCII 可见字符
*/
void OLED_ShowString(uint8_t Line, uint8_t Column, char *String)
{
    uint8_t i;
    for (i = 0; String[i] != '\0'; i++)
    {
        OLED_ShowChar(Line, Column + i, String[i]);
    }
}

/**
    *函数作用：    OLED 显示汉字
    *参数 Line：    起始行位置，范围：1~4
    *参数 Column：起始列位置，范围：1~8
    *参数 Chi：     要显示的汉字序列号
*/
void OLED_ShowChinese(uint8_t Line , uint8_t Column, uint8_t Chi)
{
    uint8_t i;
    OLED_SetCursor( (Line - 1) * 2    , (Column - 1) * 16 + 1 );      //设置光标
位置在上半部分
    for (i = 0; i < 16; i++)
    {
        OLED_WriteData(OLED_F16x16[Chi*2][i]);
//显示上半部分内容
    }
    OLED_SetCursor( (Line - 1) * 2 + 1    , (Column - 1) * 16 + 1 ); //设置光标
位置在下半部分
    for (i = 0; i < 16; i++)
    {
        OLED_WriteData(OLED_F16x16[Chi*2+1][i]);
//显示下半部分内容
    }
}

/**
    *函数作用：OLED 次方函数
    *返回值：    X 的 Y 次方
*/
uint32_t OLED_Pow(uint32_t X, uint32_t Y)
{
    uint32_t Result = 1;
    while (Y--)
```

```
    {
        Result *= X;
    }
    return Result;
}

/**
    *函数作用：    OLED 显示数字（十进制，正数）
    *参数 Line：    起始行位置，范围：1~4
    *参数 Column：起始列位置，范围：1~16
    *参数 Number：要显示的数字，范围：0~4294967295
    *参数 Length：要显示数字的长度，范围：1~10
*/
void OLED_ShowNum(uint8_t Line, uint8_t Column, uint32_t Number, uint8_t Length)
{
    uint8_t i;
    for (i = 0; i < Length; i++)
    {
        OLED_ShowChar(Line, Column + i, Number / OLED_Pow(10, Length - i - 1) % 10 + '0');
    }
}

/**
    *函数作用：    OLED 显示数字（十进制，带符号数）
    *参数 Line：    起始行位置，范围：1~4
    *参数 Column：起始列位置，范围：1~16
    *参数 Number：要显示的数字，范围：-2147483648~2147483647
    *参数 Length：要显示数字的长度，范围：1~10
*/
void OLED_ShowSignedNum(uint8_t Line, uint8_t Column, int32_t Number, uint8_t Length)
{
    uint8_t i;
    uint32_t Number1;
    if (Number >= 0)
    {
        OLED_ShowChar(Line, Column, '+');
        Number1 = Number;
    }
    else
    {
        OLED_ShowChar(Line, Column, '-');
```

```
            Number1 = -Number;
        }
        for (i = 0; i < Length; i++)
        {
            OLED_ShowChar(Line, Column + i + 1, Number1 / OLED_Pow(10, Length - i - 1) % 10 + '0');
        }
    }

    /**
        *函数作用：    OLED 显示数字（十六进制，正数）
        *参数 Line：    起始行位置，范围：1~4
        *参数 Column：起始列位置，范围：1~16
        *参数 Number：要显示的数字，范围：0~0xFFFFFFFF
        *参数 Length：要显示数字的长度，范围：1~8
    */
    void OLED_ShowHexNum(uint8_t Line, uint8_t Column, uint32_t Number, uint8_t Length)
    {
        uint8_t i, SingleNumber;
        for (i = 0; i < Length; i++)
        {
            SingleNumber = Number / OLED_Pow(16, Length - i - 1) % 16;
            if (SingleNumber < 10)
            {
                OLED_ShowChar(Line, Column + i, SingleNumber + '0');
            }
            else
            {
                OLED_ShowChar(Line, Column + i, SingleNumber - 10 + 'A');
            }
        }
    }
```

打开 OLED_Font.h 文件，其中 const uint8_t OLED_F8x16[][16]的字模库是固定的，而 const uint8_t OLED_F16x16[][32]中的代码为汉字代码，根据需求从取模软件中获取，新的代码直接替换原有的代码即可使用，该教程代码仅为参考使用，总代码如下：

```
#ifndef __OLED_FONT_H
#define __OLED_FONT_H

/*OLED 字模库，宽 8 像素，高 16 像素*/
```

```
const uint8_t OLED_F8x16[][16]=
{
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//    0

    0x00,0x00,0x00,0xF8,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x33,0x30,0x00,0x00,0x00,//! 1

    0x00,0x10,0x0C,0x06,0x10,0x0C,0x06,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//" 2

    0x40,0xC0,0x78,0x40,0xC0,0x78,0x40,0x00,
    0x04,0x3F,0x04,0x04,0x3F,0x04,0x04,0x00,//# 3

    0x00,0x70,0x88,0xFC,0x08,0x30,0x00,0x00,
    0x00,0x18,0x20,0xFF,0x21,0x1E,0x00,0x00,//$ 4

    0xF0,0x08,0xF0,0x00,0xE0,0x18,0x00,0x00,
    0x00,0x21,0x1C,0x03,0x1E,0x21,0x1E,0x00,//% 5

    0x00,0xF0,0x08,0x88,0x70,0x00,0x00,0x00,
    0x1E,0x21,0x23,0x24,0x19,0x27,0x21,0x10,//& 6

    0x10,0x16,0x0E,0x00,0x00,0x00,0x00,0x00,
    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//' 7

    0x00,0x00,0x00,0xE0,0x18,0x04,0x02,0x00,
    0x00,0x00,0x00,0x07,0x18,0x20,0x40,0x00,//( 8

    0x00,0x02,0x04,0x18,0xE0,0x00,0x00,0x00,
    0x00,0x40,0x20,0x18,0x07,0x00,0x00,0x00,//) 9

    0x40,0x40,0x80,0xF0,0x80,0x40,0x40,0x00,
    0x02,0x02,0x01,0x0F,0x01,0x02,0x02,0x00,//* 10

    0x00,0x00,0x00,0xF0,0x00,0x00,0x00,0x00,
    0x01,0x01,0x01,0x1F,0x01,0x01,0x01,0x00,//+ 11

    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x80,0xB0,0x70,0x00,0x00,0x00,0x00,0x00,//, 12

    0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
    0x00,0x01,0x01,0x01,0x01,0x01,0x01,0x01,//- 13
```

```
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x00,0x30,0x30,0x00,0x00,0x00,0x00,0x00,//. 14

0x00,0x00,0x00,0x00,0x80,0x60,0x18,0x04,
0x00,0x60,0x18,0x06,0x01,0x00,0x00,0x00,/// 15

0x00,0xE0,0x10,0x08,0x08,0x10,0xE0,0x00,
0x00,0x0F,0x10,0x20,0x20,0x10,0x0F,0x00,//0 16

0x00,0x10,0x10,0xF8,0x00,0x00,0x00,0x00,
0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//1 17

0x00,0x70,0x08,0x08,0x08,0x88,0x70,0x00,
0x00,0x30,0x28,0x24,0x22,0x21,0x30,0x00,//2 18

0x00,0x30,0x08,0x88,0x88,0x48,0x30,0x00,
0x00,0x18,0x20,0x20,0x20,0x11,0x0E,0x00,//3 19

0x00,0x00,0xC0,0x20,0x10,0xF8,0x00,0x00,
0x00,0x07,0x04,0x24,0x24,0x3F,0x24,0x00,//4 20

0x00,0xF8,0x08,0x88,0x88,0x08,0x08,0x00,
0x00,0x19,0x21,0x20,0x20,0x11,0x0E,0x00,//5 21

0x00,0xE0,0x10,0x88,0x88,0x18,0x00,0x00,
0x00,0x0F,0x11,0x20,0x20,0x11,0x0E,0x00,//6 22

0x00,0x38,0x08,0x08,0xC8,0x38,0x08,0x00,
0x00,0x00,0x00,0x3F,0x00,0x00,0x00,0x00,//7 23

0x00,0x70,0x88,0x08,0x08,0x88,0x70,0x00,
0x00,0x1C,0x22,0x21,0x21,0x22,0x1C,0x00,//8 24

0x00,0xE0,0x10,0x08,0x08,0x10,0xE0,0x00,
0x00,0x00,0x31,0x22,0x22,0x11,0x0F,0x00,//9 25

0x00,0x00,0x00,0xC0,0xC0,0x00,0x00,0x00,
0x00,0x00,0x00,0x30,0x30,0x00,0x00,0x00,//: 26

0x00,0x00,0x00,0x80,0x00,0x00,0x00,0x00,
0x00,0x00,0x80,0x60,0x00,0x00,0x00,0x00,//; 27

0x00,0x00,0x80,0x40,0x20,0x10,0x08,0x00,
0x00,0x01,0x02,0x04,0x08,0x10,0x20,0x00,//< 28
```

```
0x40,0x40,0x40,0x40,0x40,0x40,0x40,0x00,
0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x00,//= 29

0x00,0x08,0x10,0x20,0x40,0x80,0x00,0x00,
0x00,0x20,0x10,0x08,0x04,0x02,0x01,0x00,//> 30

0x00,0x70,0x48,0x08,0x08,0x08,0xF0,0x00,
0x00,0x00,0x00,0x30,0x36,0x01,0x00,0x00,//? 31

0xC0,0x30,0xC8,0x28,0xE8,0x10,0xE0,0x00,
0x07,0x18,0x27,0x24,0x23,0x14,0x0B,0x00,//@ 32

0x00,0x00,0xC0,0x38,0xE0,0x00,0x00,0x00,
0x20,0x3C,0x23,0x02,0x02,0x27,0x38,0x20,//A 33

0x08,0xF8,0x88,0x88,0x88,0x70,0x00,0x00,
0x20,0x3F,0x20,0x20,0x20,0x11,0x0E,0x00,//B 34

0xC0,0x30,0x08,0x08,0x08,0x08,0x38,0x00,
0x07,0x18,0x20,0x20,0x20,0x10,0x08,0x00,//C 35

0x08,0xF8,0x08,0x08,0x08,0x10,0xE0,0x00,
0x20,0x3F,0x20,0x20,0x20,0x10,0x0F,0x00,//D 36

0x08,0xF8,0x88,0x88,0xE8,0x08,0x10,0x00,
0x20,0x3F,0x20,0x20,0x23,0x20,0x18,0x00,//E 37

0x08,0xF8,0x88,0x88,0xE8,0x08,0x10,0x00,
0x20,0x3F,0x20,0x00,0x03,0x00,0x00,0x00,//F 38

0xC0,0x30,0x08,0x08,0x08,0x38,0x00,0x00,
0x07,0x18,0x20,0x20,0x22,0x1E,0x02,0x00,//G 39

0x08,0xF8,0x08,0x00,0x00,0x08,0xF8,0x08,
0x20,0x3F,0x21,0x01,0x01,0x21,0x3F,0x20,//H 40

0x00,0x08,0x08,0xF8,0x08,0x08,0x00,0x00,
0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//I 41

0x00,0x00,0x08,0x08,0xF8,0x08,0x08,0x00,
0xC0,0x80,0x80,0x80,0x7F,0x00,0x00,0x00,//J 42

0x08,0xF8,0x88,0xC0,0x28,0x18,0x08,0x00,
```

```
0x20,0x3F,0x20,0x01,0x26,0x38,0x20,0x00,//K 43

0x08,0xF8,0x08,0x00,0x00,0x00,0x00,0x00,
0x20,0x3F,0x20,0x20,0x20,0x20,0x30,0x00,//L 44

0x08,0xF8,0xF8,0x00,0xF8,0xF8,0x08,0x00,
0x20,0x3F,0x00,0x3F,0x00,0x3F,0x20,0x00,//M 45

0x08,0xF8,0x30,0xC0,0x00,0x08,0xF8,0x08,
0x20,0x3F,0x20,0x00,0x07,0x18,0x3F,0x00,//N 46

0xE0,0x10,0x08,0x08,0x08,0x10,0xE0,0x00,
0x0F,0x10,0x20,0x20,0x20,0x10,0x0F,0x00,//O 47

0x08,0xF8,0x08,0x08,0x08,0x08,0xF0,0x00,
0x20,0x3F,0x21,0x01,0x01,0x01,0x00,0x00,//P 48

0xE0,0x10,0x08,0x08,0x08,0x10,0xE0,0x00,
0x0F,0x18,0x24,0x24,0x38,0x50,0x4F,0x00,//Q 49

0x08,0xF8,0x88,0x88,0x88,0x88,0x70,0x00,
0x20,0x3F,0x20,0x00,0x03,0x0C,0x30,0x20,//R 50

0x00,0x70,0x88,0x08,0x08,0x08,0x38,0x00,
0x00,0x38,0x20,0x21,0x21,0x22,0x1C,0x00,//S 51

0x18,0x08,0x08,0xF8,0x08,0x08,0x18,0x00,
0x00,0x00,0x20,0x3F,0x20,0x00,0x00,0x00,//T 52

0x08,0xF8,0x08,0x00,0x00,0x08,0xF8,0x08,
0x00,0x1F,0x20,0x20,0x20,0x20,0x1F,0x00,//U 53

0x08,0x78,0x88,0x00,0x00,0xC8,0x38,0x08,
0x00,0x00,0x07,0x38,0x0E,0x01,0x00,0x00,//V 54

0xF8,0x08,0x00,0xF8,0x00,0x08,0xF8,0x00,
0x03,0x3C,0x07,0x00,0x07,0x3C,0x03,0x00,//W 55

0x08,0x18,0x68,0x80,0x80,0x68,0x18,0x08,
0x20,0x30,0x2C,0x03,0x03,0x2C,0x30,0x20,//X 56

0x08,0x38,0xC8,0x00,0xC8,0x38,0x08,0x00,
0x00,0x00,0x20,0x3F,0x20,0x00,0x00,0x00,//Y 57
```

```
0x10,0x08,0x08,0x08,0xC8,0x38,0x08,0x00,
0x20,0x38,0x26,0x21,0x20,0x20,0x18,0x00,//Z 58

0x00,0x00,0x00,0xFE,0x02,0x02,0x02,0x00,
0x00,0x00,0x00,0x7F,0x40,0x40,0x40,0x00,//[ 59

0x00,0x0C,0x30,0xC0,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x01,0x06,0x38,0xC0,0x00,//\ 60

0x00,0x02,0x02,0x02,0xFE,0x00,0x00,0x00,
0x00,0x40,0x40,0x40,0x7F,0x00,0x00,0x00,//] 61

0x00,0x00,0x04,0x02,0x02,0x02,0x04,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//^ 62

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x80,//_ 63

0x00,0x02,0x02,0x04,0x00,0x00,0x00,0x00,
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//` 64

0x00,0x00,0x80,0x80,0x80,0x80,0x00,0x00,
0x00,0x19,0x24,0x22,0x22,0x22,0x3F,0x20,//a 65

0x08,0xF8,0x00,0x80,0x80,0x00,0x00,0x00,
0x00,0x3F,0x11,0x20,0x20,0x11,0x0E,0x00,//b 66

0x00,0x00,0x00,0x80,0x80,0x80,0x00,0x00,
0x00,0x0E,0x11,0x20,0x20,0x20,0x11,0x00,//c 67

0x00,0x00,0x00,0x80,0x80,0x88,0xF8,0x00,
0x00,0x0E,0x11,0x20,0x20,0x10,0x3F,0x20,//d 68

0x00,0x00,0x80,0x80,0x80,0x80,0x00,0x00,
0x00,0x1F,0x22,0x22,0x22,0x22,0x13,0x00,//e 69

0x00,0x80,0x80,0xF0,0x88,0x88,0x88,0x18,
0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//f 70

0x00,0x00,0x80,0x80,0x80,0x80,0x80,0x00,
0x00,0x6B,0x94,0x94,0x94,0x93,0x60,0x00,//g 71

0x08,0xF8,0x00,0x80,0x80,0x80,0x00,0x00,
0x20,0x3F,0x21,0x00,0x00,0x20,0x3F,0x20,//h 72
```

```
0x00,0x80,0x98,0x98,0x00,0x00,0x00,0x00,
0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//i 73

0x00,0x00,0x00,0x80,0x98,0x98,0x00,0x00,
0x00,0xC0,0x80,0x80,0x80,0x7F,0x00,0x00,//j 74

0x08,0xF8,0x00,0x00,0x80,0x80,0x80,0x00,
0x20,0x3F,0x24,0x02,0x2D,0x30,0x20,0x00,//k 75

0x00,0x08,0x08,0xF8,0x00,0x00,0x00,0x00,
0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00,//l 76

0x80,0x80,0x80,0x80,0x80,0x80,0x80,0x00,
0x20,0x3F,0x20,0x00,0x3F,0x20,0x00,0x3F,//m 77

0x80,0x80,0x00,0x80,0x80,0x80,0x00,0x00,
0x20,0x3F,0x21,0x00,0x00,0x20,0x3F,0x20,//n 78

0x00,0x00,0x80,0x80,0x80,0x80,0x00,0x00,
0x00,0x1F,0x20,0x20,0x20,0x20,0x1F,0x00,//o 79

0x80,0x80,0x00,0x80,0x80,0x00,0x00,0x00,
0x80,0xFF,0xA1,0x20,0x20,0x11,0x0E,0x00,//p 80

0x00,0x00,0x00,0x80,0x80,0x80,0x80,0x00,
0x00,0x0E,0x11,0x20,0x20,0xA0,0xFF,0x80,//q 81

0x80,0x80,0x80,0x00,0x80,0x80,0x80,0x00,
0x20,0x20,0x3F,0x21,0x20,0x00,0x01,0x00,//r 82

0x00,0x00,0x80,0x80,0x80,0x80,0x80,0x00,
0x00,0x33,0x24,0x24,0x24,0x24,0x19,0x00,//s 83

0x00,0x80,0x80,0xE0,0x80,0x80,0x00,0x00,
0x00,0x00,0x00,0x1F,0x20,0x20,0x00,0x00,//t 84

0x80,0x80,0x00,0x00,0x00,0x80,0x80,0x00,
0x00,0x1F,0x20,0x20,0x20,0x10,0x3F,0x20,//u 85

0x80,0x80,0x80,0x00,0x00,0x80,0x80,0x80,
0x00,0x01,0x0E,0x30,0x08,0x06,0x01,0x00,//v 86

0x80,0x80,0x00,0x80,0x00,0x80,0x80,0x80,
```

```
        0x0F,0x30,0x0C,0x03,0x0C,0x30,0x0F,0x00,//w 87

        0x00,0x80,0x80,0x00,0x80,0x80,0x80,0x00,
        0x00,0x20,0x31,0x2E,0x0E,0x31,0x20,0x00,//x 88

        0x80,0x80,0x80,0x00,0x00,0x80,0x80,0x80,
        0x80,0x81,0x8E,0x70,0x18,0x06,0x01,0x00,//y 89

        0x00,0x80,0x80,0x80,0x80,0x80,0x80,0x00,
        0x00,0x21,0x30,0x2C,0x22,0x21,0x30,0x00,//z 90

        0x00,0x00,0x00,0x00,0x80,0x7C,0x02,0x02,
        0x00,0x00,0x00,0x00,0x00,0x3F,0x40,0x40,//{ 91

        0x00,0x00,0x00,0x00,0xFF,0x00,0x00,0x00,
        0x00,0x00,0x00,0x00,0xFF,0x00,0x00,0x00,//| 92

        0x00,0x02,0x02,0x7C,0x80,0x00,0x00,0x00,
        0x00,0x40,0x40,0x3F,0x00,0x00,0x00,0x00,//} 93

        0x00,0x06,0x01,0x01,0x02,0x02,0x04,0x04,
        0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,//~ 94
    };


    /*OLED 字模库，宽 16 像素，高 16 像素*/
    const uint8_t OLED_F16x16[][32]={
    {0x88,0x68,0x1F,0xC8,0x08,0x10,0xC8,0x54,0x52,0xD1,0x12,0x94,0x08,0xD0,
0x10,0x00},
    {0x09,0x19,0x09,0xFF,0x05,0x00,0xFF,0x12,0x92,0xFF,0x00,0x5F,0x80,0x7F,0
x00,0x00},/*"输",0*/
    /* (16 X 16 , 宋体 )*/

    {0x00,0x00,0x7C,0x40,0x40,0x40,0x40,0xFF,0x40,0x40,0x40,0x40,0xFC,0x00,
0x00,0x00},
    {0x00,0x7C,0x40,0x40,0x40,0x40,0x40,0x7F,0x40,0x40,0x40,0x40,0x40,0xFC,0
x00,0x00},/*"出",1*/
    /* (16 X 16 , 宋体 )*/

    {0x00,0x00,0xF8,0x88,0x88,0x88,0x88,0xFF,0x88,0x88,0x88,0x88,0xF8,0x00,0
x00,0x00},
    {0x00,0x00,0x1F,0x08,0x08,0x08,0x08,0x7F,0x88,0x88,0x88,0x88,0x9F,0x80,0
xF0,0x00},/*"电",2*/
    /* (16 X 16 , 宋体 )*/
```

```
    {0x00,0x00,0xFE,0x02,0x82,0x82,0x82,0x82,0xFA,0x82,0x82,0x82,0x82,0x82,0
x02,0x00},
    {0x80,0x60,0x1F,0x40,0x40,0x40,0x40,0x40,0x7F,0x40,0x40,0x44,0x58,0x40,0
x40,0x00},/*"压",3*/
    /* (16 X 16 , 宋体 )*/
    };

    #endif
```

打开 OLED.h 文件，代码如下：

```
    #ifndef __OLED_H
    #define __OLED_H

    void OLED_Init(void);
    void OLED_Clear(void);
    void OLED_ShowChar(uint8_t Line, uint8_t Column, char Char);
    void OLED_ShowString(uint8_t Line, uint8_t Column, char *String);
    void OLED_ShowNum(uint8_t Line, uint8_t Column, uint32_t Number, uint8_t
Length);
    void OLED_ShowSignedNum(uint8_t Line, uint8_t Column, int32_t Number,
uint8_t Length);
    void OLED_ShowHexNum(uint8_t Line, uint8_t Column, uint32_t Number,
uint8_t Length);
    void OLED_ShowBinNum(uint8_t Line, uint8_t Column, uint32_t Number,
uint8_t Length);
    void OLED_ShowChinese(uint8_t Line , uint8_t Column, uint8_t Chi);

    #endif
```

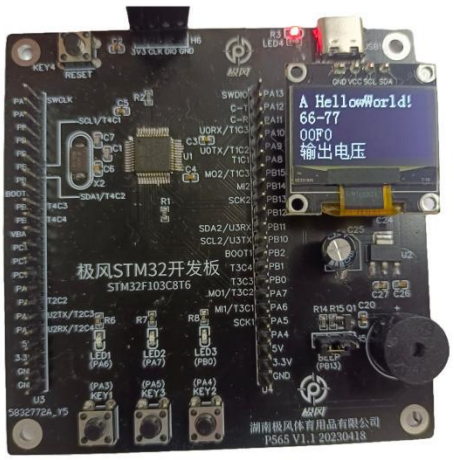
打开 main.c 文件，代码如下：

```
    #include "stm32f10x.h"                  // Device header
    #include "OLED.h"

    int main(void)
    {
        OLED_Init();     //初始化 OLED

    //    显示字符
        OLED_ShowChar(1,1,'A');
```

```
    //  显示字符串
        OLED_ShowString(1,3,"HellowWorld!");

    //  显示十进制正数
    //输入参数（行位置，列位置，想要显示的数字，数字位数）
        OLED_ShowNum(2,1,66,2);

    //  显示十进制有符号数
    //输入参数（行位置，列位置，想要显示的数字，数字位数（不包括符号））
        OLED_ShowSignedNum(2,3,-77,2);

    //  显示十六进制数
    //输入参数（行位置，列位置，想要显示的十六进制数，显示的位数）
        OLED_ShowHexNum(3,1,0XF0,4);

    //  显示汉字
    //输入参数（行位置，列位置，汉字对应的序列号（即 OLED_Font.h 文件里字模库 OLED_F16x16[][32]各汉字对应的数字））
        OLED_ShowChinese(4,1,0);
        OLED_ShowChinese(4,2,1);
        OLED_ShowChinese(4,3,2);
        OLED_ShowChinese(4,4,3);

        while(1)
        {

        }
    }
```

17.5. 显示现象

点击下载后按下开发板的复位键，即可得到如下现象。


A HellowWorld!
66-77
00F0
输出电压
极风STM32开发板
STM32F103C8T6
湖南极风体育用品有限公司
P565 V1.1 20230418