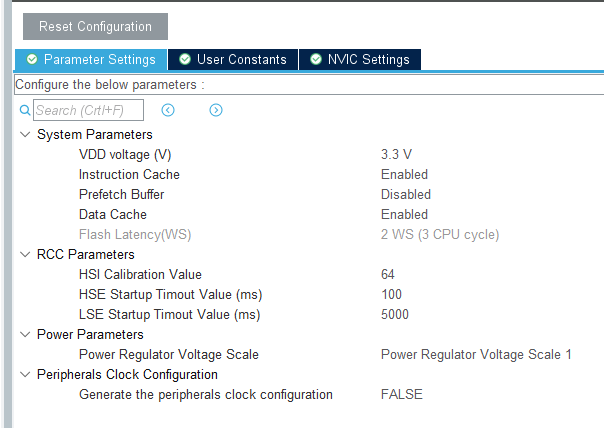
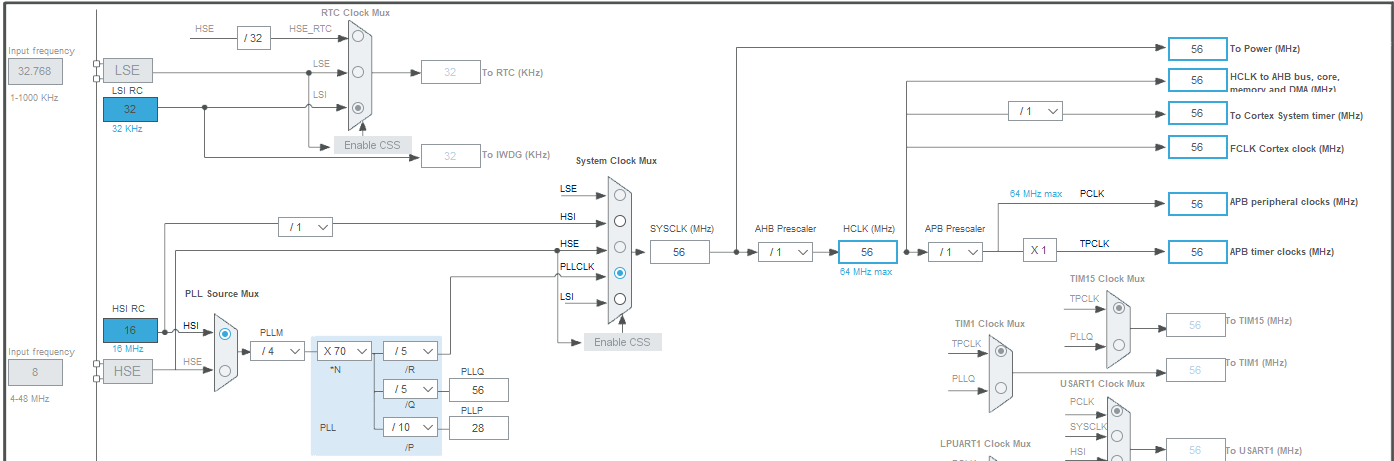
# 内置SPI驱动OLED显示屏

## 创建工程

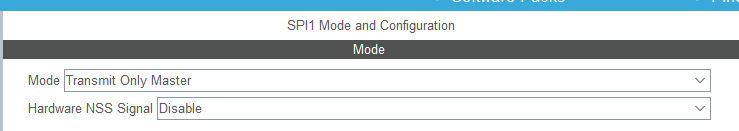
将RCC拉到FALSE



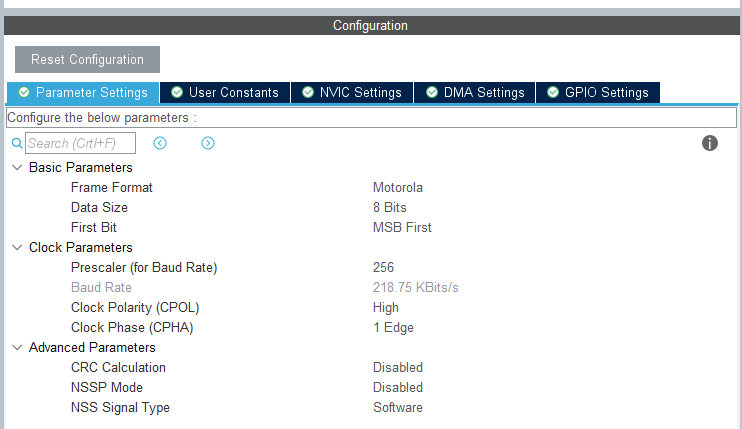
如下图所示配置系统时钟



设定SPI1模式为“主板发送模式”

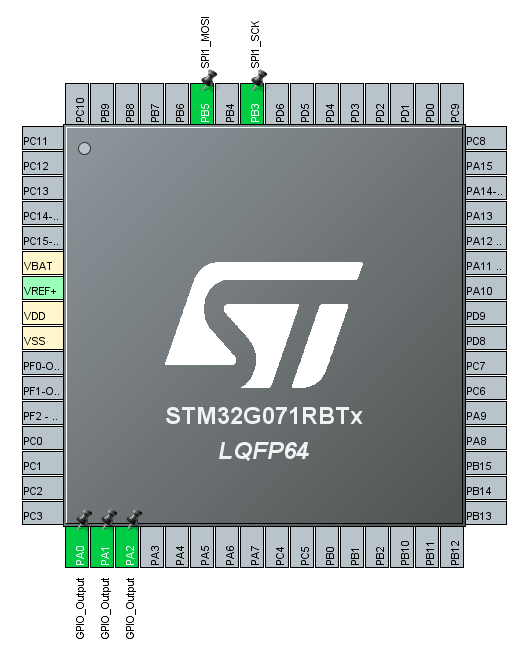


SPI其他设置如下图所示



将PA0，PA1，PA2设置为GPIO\_OUTPUT模式

管脚映射：SPI1\_MOSI映射在PB5引脚，SPI1\_SCK映射在PB3引脚。

创建工程

## 硬件电路连接

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

CN9\_4 PB3→D0 SPI时钟线

CN9\_5 PB5→D1 SPI数据线

CN10\_34 PA2→RES 复位信号

CN8\_2 PA1→DC SPI数据/命令选择

CN8\_1 PA0→CS 片选信号

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

## 添加设计代码

### 变量定义与函数声明如下

uint8\_t aTxBuffer\_T[1] = {0x00}; /\* 发送缓冲区 \*/

#define SET 1

#define RESET 0

#define OLED\_CMD 0

#define OLED\_DATA 1

#define SIZE 16 //字符与数字 字体大小16为大其他为小

#define X\_WIDTH 128

#define Y\_WIDTH 64

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

常用ASCII表

偏移量32

单个字符大小6列1（\*8）行像素

小字体模式 只在下半屏幕输出

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

const unsigned char L6H8[][6] =

{

{0x00, 0x00, 0x00, 0x00, 0x00, 0x00},// space 空格

{0x00, 0x00, 0x00, 0x2f, 0x00, 0x00},// !

{0x00, 0x00, 0x07, 0x00, 0x07, 0x00},// "

{0x00, 0x14, 0x7f, 0x14, 0x7f, 0x14},// #

{0x00, 0x24, 0x2a, 0x7f, 0x2a, 0x12},// $

{0x00, 0x62, 0x64, 0x08, 0x13, 0x23},// %

{0x00, 0x36, 0x49, 0x55, 0x22, 0x50},// &

{0x00, 0x00, 0x05, 0x03, 0x00, 0x00},// '

{0x00, 0x00, 0x1c, 0x22, 0x41, 0x00},// (

{0x00, 0x00, 0x41, 0x22, 0x1c, 0x00},// )

{0x00, 0x14, 0x08, 0x3E, 0x08, 0x14},// \*

{0x00, 0x08, 0x08, 0x3E, 0x08, 0x08},// +

{0x00, 0x00, 0x00, 0xA0, 0x60, 0x00},// ,

{0x00, 0x08, 0x08, 0x08, 0x08, 0x08},// -

{0x00, 0x00, 0x60, 0x60, 0x00, 0x00},// .

{0x00, 0x20, 0x10, 0x08, 0x04, 0x02},// /

{0x00, 0x3E, 0x51, 0x49, 0x45, 0x3E},// 0

{0x00, 0x00, 0x42, 0x7F, 0x40, 0x00},// 1

{0x00, 0x42, 0x61, 0x51, 0x49, 0x46},// 2

{0x00, 0x21, 0x41, 0x45, 0x4B, 0x31},// 3

{0x00, 0x18, 0x14, 0x12, 0x7F, 0x10},// 4

{0x00, 0x27, 0x45, 0x45, 0x45, 0x39},// 5

{0x00, 0x3C, 0x4A, 0x49, 0x49, 0x30},// 6

{0x00, 0x01, 0x71, 0x09, 0x05, 0x03},// 7

{0x00, 0x36, 0x49, 0x49, 0x49, 0x36},// 8

{0x00, 0x06, 0x49, 0x49, 0x29, 0x1E},// 9

{0x00, 0x00, 0x36, 0x36, 0x00, 0x00},// :

{0x00, 0x00, 0x56, 0x36, 0x00, 0x00},// ;

{0x00, 0x08, 0x14, 0x22, 0x41, 0x00},// <

{0x00, 0x14, 0x14, 0x14, 0x14, 0x14},// =

{0x00, 0x00, 0x41, 0x22, 0x14, 0x08},// >

{0x00, 0x02, 0x01, 0x51, 0x09, 0x06},// ?

{0x00, 0x32, 0x49, 0x59, 0x51, 0x3E},// @

{0x00, 0x7C, 0x12, 0x11, 0x12, 0x7C},// A

{0x00, 0x7F, 0x49, 0x49, 0x49, 0x36},// B

{0x00, 0x3E, 0x41, 0x41, 0x41, 0x22},// C

{0x00, 0x7F, 0x41, 0x41, 0x22, 0x1C},// D

{0x00, 0x7F, 0x49, 0x49, 0x49, 0x41},// E

{0x00, 0x7F, 0x09, 0x09, 0x09, 0x01},// F

{0x00, 0x3E, 0x41, 0x49, 0x49, 0x7A},// G

{0x00, 0x7F, 0x08, 0x08, 0x08, 0x7F},// H

{0x00, 0x00, 0x41, 0x7F, 0x41, 0x00},// I

{0x00, 0x20, 0x40, 0x41, 0x3F, 0x01},// J

{0x00, 0x7F, 0x08, 0x14, 0x22, 0x41},// K

{0x00, 0x7F, 0x40, 0x40, 0x40, 0x40},// L

{0x00, 0x7F, 0x02, 0x0C, 0x02, 0x7F},// M

{0x00, 0x7F, 0x04, 0x08, 0x10, 0x7F},// N

{0x00, 0x3E, 0x41, 0x41, 0x41, 0x3E},// O

{0x00, 0x7F, 0x09, 0x09, 0x09, 0x06},// P

{0x00, 0x3E, 0x41, 0x51, 0x21, 0x5E},// Q

{0x00, 0x7F, 0x09, 0x19, 0x29, 0x46},// R

{0x00, 0x46, 0x49, 0x49, 0x49, 0x31},// S

{0x00, 0x01, 0x01, 0x7F, 0x01, 0x01},// T

{0x00, 0x3F, 0x40, 0x40, 0x40, 0x3F},// U

{0x00, 0x1F, 0x20, 0x40, 0x20, 0x1F},// V

{0x00, 0x3F, 0x40, 0x38, 0x40, 0x3F},// W

{0x00, 0x63, 0x14, 0x08, 0x14, 0x63},// X

{0x00, 0x07, 0x08, 0x70, 0x08, 0x07},// Y

{0x00, 0x61, 0x51, 0x49, 0x45, 0x43},// Z

{0x00, 0x00, 0x7F, 0x41, 0x41, 0x00},// [

{0x00, 0x55, 0x2A, 0x55, 0x2A, 0x55},// 55

{0x00, 0x00, 0x41, 0x41, 0x7F, 0x00},// ]

{0x00, 0x04, 0x02, 0x01, 0x02, 0x04},// ^

{0x00, 0x40, 0x40, 0x40, 0x40, 0x40},// \_

{0x00, 0x00, 0x01, 0x02, 0x04, 0x00},// '

{0x00, 0x20, 0x54, 0x54, 0x54, 0x78},// a

{0x00, 0x7F, 0x48, 0x44, 0x44, 0x38},// b

{0x00, 0x38, 0x44, 0x44, 0x44, 0x20},// c

{0x00, 0x38, 0x44, 0x44, 0x48, 0x7F},// d

{0x00, 0x38, 0x54, 0x54, 0x54, 0x18},// e

{0x00, 0x08, 0x7E, 0x09, 0x01, 0x02},// f

{0x00, 0x18, 0xA4, 0xA4, 0xA4, 0x7C},// g

{0x00, 0x7F, 0x08, 0x04, 0x04, 0x78},// h

{0x00, 0x00, 0x44, 0x7D, 0x40, 0x00},// i

{0x00, 0x40, 0x80, 0x84, 0x7D, 0x00},// j

{0x00, 0x7F, 0x10, 0x28, 0x44, 0x00},// k

{0x00, 0x00, 0x41, 0x7F, 0x40, 0x00},// l

{0x00, 0x7C, 0x04, 0x18, 0x04, 0x78},// m

{0x00, 0x7C, 0x08, 0x04, 0x04, 0x78},// n

{0x00, 0x38, 0x44, 0x44, 0x44, 0x38},// o

{0x00, 0xFC, 0x24, 0x24, 0x24, 0x18},// p

{0x00, 0x18, 0x24, 0x24, 0x18, 0xFC},// q

{0x00, 0x7C, 0x08, 0x04, 0x04, 0x08},// r

{0x00, 0x48, 0x54, 0x54, 0x54, 0x20},// s

{0x00, 0x04, 0x3F, 0x44, 0x40, 0x20},// t

{0x00, 0x3C, 0x40, 0x40, 0x20, 0x7C},// u

{0x00, 0x1C, 0x20, 0x40, 0x20, 0x1C},// v

{0x00, 0x3C, 0x40, 0x30, 0x40, 0x3C},// w

{0x00, 0x44, 0x28, 0x10, 0x28, 0x44},// x

{0x00, 0x1C, 0xA0, 0xA0, 0xA0, 0x7C},// y

{0x00, 0x44, 0x64, 0x54, 0x4C, 0x44},// z

{0x14, 0x14, 0x14, 0x14, 0x14, 0x14},// horiz lines

};

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

常用ASCII表

偏移量32

单个字符大小8列2（\*8）行像素

大字体模式

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

const unsigned char L8H16[][8]=

{

{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00},

{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00},//space 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

};

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

上半个屏幕，下半个屏幕

每个字符占用16列2（\*8）行

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

unsigned char Hzk[][16]=

{

{0x80,0x84,0x44,0x44,0x24,0x14,0x0C,0xFF,0x0C,0x14,0x24,0x44,0x44,0x84,0x80,0x00},

{0x08,0x08,0x08,0x08,0x09,0x49,0x89,0x79,0x0D,0x0B,0x09,0x08,0x08,0x08,0x08,0x00},/\*"李",0\*/

{0x40,0x40,0x42,0x42,0x42,0x42,0x42,0xFE,0x42,0x42,0x42,0x42,0x42,0x40,0x40,0x00},

{0x80,0x80,0x40,0x20,0x10,0x0C,0x03,0x00,0x03,0x0C,0x10,0x20,0x40,0x80,0x80,0x00},/\*"天",1\*/

{0x00,0x02,0x0C,0xC0,0x00,0x20,0xA4,0x64,0xA4,0x3F,0x24,0x64,0xA4,0x20,0x20,0x00},

{0x02,0x02,0x7F,0x00,0x80,0x89,0x44,0x46,0x2B,0x12,0x2A,0x46,0x80,0x81,0x00,0x00},/\*"凌",2\*/

{0x00,0x00,0xFF,0x88,0x88,0x48,0x48,0x00,0x7F,0x88,0x84,0x84,0x82,0xE0,0x00,0x00},

{0x04,0x04,0x05,0x04,0x04,0x04,0x04,0xFF,0x04,0x04,0x04,0x04,0x04,0x04,0x04,0x00},/\*"毕",3\*/

{0x00,0x10,0x60,0x80,0x00,0xFF,0x00,0x00,0x00,0xFF,0x00,0x00,0xC0,0x30,0x00,0x00},

{0x40,0x40,0x40,0x43,0x40,0x7F,0x40,0x40,0x40,0x7F,0x42,0x41,0x40,0x40,0x40,0x00},/\*"业",4\*/

{0x40,0x40,0x42,0xCC,0x00,0x40,0xA0,0x9E,0x82,0x82,0x82,0x9E,0xA0,0x20,0x20,0x00},

{0x00,0x00,0x00,0x3F,0x90,0x88,0x40,0x43,0x2C,0x10,0x28,0x46,0x41,0x80,0x80,0x00},/\*"设",5\*/

{0x40,0x40,0x42,0xCC,0x00,0x40,0x40,0x40,0x40,0xFF,0x40,0x40,0x40,0x40,0x40,0x00},

{0x00,0x00,0x00,0x7F,0x20,0x10,0x00,0x00,0x00,0xFF,0x00,0x00,0x00,0x00,0x00,0x00},/\*"计",6\*/

};

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

图像显示，横排扫描

8（\*8）行128列

无划分

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

unsigned char BMP1[][128] =

{

{0x00,0x03,0x05,0x09,0x11,0xFF,0x11,0x89,0x05,0xC3,0x00,0xE0,0x00,0xF0,0x00,0xF8,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x44,0x28,0xFF,0x11,0xAA,0x44,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x83,0x01,0x38,0x44,0x82,0x92,

0x92,0x74,0x01,0x83,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x7C,0x44,0xFF,0x01,0x7D,

0x7D,0x7D,0x01,0x7D,0x7D,0x7D,0x7D,0x01,0x7D,0x7D,0x7D,0x7D,0x7D,0x01,0xFF,0x00},

{0x00,0x00,0x00,0x00,0x00,0x01,0x00,0x01,0x00,0x01,0x00,0x01,0x00,0x01,0x00,0x01,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x01,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x01,0x00,0x00,0x00,0x00,

0x00,0x00,0x01,0x01,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x01,0x01,0x01,

0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x00},

{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x3F,0x3F,0x03,0x03,

0xF3,0x13,0x11,0x11,0x11,0x11,0x11,0x11,0x01,0xF1,0x11,0x61,0x81,0x01,0x01,0x01,

0x81,0x61,0x11,0xF1,0x01,0x01,0x01,0x01,0x41,0x41,0xF1,0x01,0x01,0x01,0x01,0x01,

0xC1,0x21,0x11,0x11,0x11,0x11,0x21,0xC1,0x01,0x01,0x01,0x01,0x41,0x41,0xF1,0x01,

0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x01,0x11,0x11,0x11,0x11,0x11,0xD3,0x33,

0x03,0x03,0x3F,0x3F,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00},

{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xE0,0xE0,0x00,0x00,

0x7F,0x01,0x01,0x01,0x01,0x01,0x01,0x00,0x00,0x7F,0x00,0x00,0x01,0x06,0x18,0x06,

0x01,0x00,0x00,0x7F,0x00,0x00,0x00,0x00,0x40,0x40,0x7F,0x40,0x40,0x00,0x00,0x00,

0x1F,0x20,0x40,0x40,0x40,0x40,0x20,0x1F,0x00,0x00,0x00,0x00,0x40,0x40,0x7F,0x40,

0x40,0x00,0x00,0x00,0x00,0x60,0x00,0x00,0x00,0x00,0x40,0x30,0x0C,0x03,0x00,0x00,

0x00,0x00,0xE0,0xE0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00},

{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x07,0x07,0x06,0x06,

0x06,0x06,0x04,0x04,0x04,0x84,0x44,0x44,0x44,0x84,0x04,0x04,0x84,0x44,0x44,0x44,

0x84,0x04,0x04,0x04,0x84,0xC4,0x04,0x04,0x04,0x04,0x84,0x44,0x44,0x44,0x84,0x04,

0x04,0x04,0x04,0x04,0x84,0x44,0x44,0x44,0x84,0x04,0x04,0x04,0x04,0x04,0x84,0x44,

0x44,0x44,0x84,0x04,0x04,0x84,0x44,0x44,0x44,0x84,0x04,0x04,0x04,0x04,0x06,0x06,

0x06,0x06,0x07,0x07,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00},

{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x10,0x18,0x14,0x12,0x11,0x00,0x00,0x0F,0x10,0x10,0x10,

0x0F,0x00,0x00,0x00,0x10,0x1F,0x10,0x00,0x00,0x00,0x08,0x10,0x12,0x12,0x0D,0x00,

0x00,0x18,0x00,0x00,0x0D,0x12,0x12,0x12,0x0D,0x00,0x00,0x18,0x00,0x00,0x10,0x18,

0x14,0x12,0x11,0x00,0x00,0x10,0x18,0x14,0x12,0x11,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00},

{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

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

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

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},

{0x00,0x7F,0x03,0x0C,0x30,0x0C,0x03,0x7F,0x00,0x00,0x38,0x54,0x54,0x58,0x00,0x00,

0x7C,0x04,0x04,0x78,0x00,0x00,0x3C,0x40,0x40,0x7C,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0xFF,0xAA,0xAA,0xAA,

0x28,0x08,0x00,0xFF,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,

0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x7F,0x03,0x0C,0x30,0x0C,0x03,0x7F,

0x00,0x00,0x26,0x49,0x49,0x49,0x32,0x00,0x00,0x7F,0x02,0x04,0x08,0x10,0x7F,0x00}

};

unsigned char BMP2[][128] =

{

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X80,0X80,0XC0,0XC0,0X60,0X60,0X30,0X30,0X30,

0X30,0X30,0X30,0X30,0X30,0X30,0X30,0X30,0XB0,0X30,0X20,0X60,0X60,0XC0,0XC0,0X80,

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,

0X00,0X80,0XE0,0X78,0X1C,0X0E,0X07,0X03,0X01,0X00,0X80,0X03,0X03,0X01,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X80,0X80,0X01,0X03,0X03,0X00,0X00,0X00,0X00,0X00,0X01,

0X03,0X07,0X0E,0X1C,0X70,0XE0,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0XF0,0XF8,

0X0E,0X07,0X01,0X08,0X1C,0X1C,0X1C,0X1C,0X1C,0X0F,0X07,0X33,0X30,0X60,0X60,0X60,

0X60,0X60,0X60,0X30,0X33,0X03,0X0F,0X0D,0X0C,0X0C,0X0C,0X0C,0X0C,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X87,0XFF,0X30,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X07,0X1F,

0X38,0X70,0X60,0XC0,0XC0,0X80,0X80,0X80,0X80,0X80,0X20,0X60,0X40,0XC0,0XC0,0X80,

0XC0,0XC0,0XC0,0X60,0X30,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X80,0X80,0X80,

0XC0,0X60,0X70,0X38,0X1C,0X0F,0X03,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X80,0XC0,0XE0,0X70,0X1C,0X8F,0XC3,0X61,0X31,0X19,0X08,0X00,0X80,0XC0,0X61,0X39,

0X1D,0X01,0X81,0XE0,0X70,0X38,0X00,0X00,0XF0,0XFE,0X0F,0X03,0X03,0X01,0X01,0X01,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0XE0,0XE0,0XF0,0XF8,0XFE,0XE6,

0XC3,0XC7,0XC6,0X83,0X83,0XF9,0XFC,0X86,0X86,0X82,0X82,0X07,0X0D,0X0C,0X9C,0X9E,

0X9F,0X9B,0X99,0XB0,0XF0,0XF0,0XF8,0XFF,0XF7,0XF0,0XE0,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X01,0X01,0X03,0X03,0X03,

0X07,0X07,0X07,0X07,0X07,0X07,0X07,0X07,0X07,0X07,0X07,0X07,0X07,0X07,0X07,0X07,

0X07,0X07,0X07,0X07,0X03,0X03,0X03,0X03,0X01,0X01,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,0X00,

};

//驱动

void OLED\_WR\_Byte(unsigned char dat,unsigned char cmd);

void OLED\_Set\_Pos(unsigned char x, unsigned char y);

void OLED\_Clear(void);

void OLED\_Init(void);

void OLED\_All(void);

void OLED\_Display\_On(void);

void OLED\_Display\_Off(void);

//字符以及字符串显示

void OLED\_ShowChar(unsigned char x,unsigned char y,unsigned char chr);

void OLED\_ShowString(unsigned char x,unsigned char y, unsigned char \*p);

//中文模式显示

void OLED\_ShowCHinese(unsigned char x,unsigned char y,unsigned char no);

//图片模式显示

void OLED\_DrawBMP(unsigned char x0, unsigned char y0,unsigned char x1, unsigned char y1,unsigned char select\_BMP);

//数字模式显示

int OLED\_pow(unsigned char m,unsigned char n);

void OLED\_ShowNum(unsigned char x,unsigned char y,unsigned int num,unsigned char len,unsigned char size2);

### 主函数代码编写

int main(void)

{

HAL\_Init();

SystemClock\_Config();

MX\_GPIO\_Init();

MX\_SPI1\_Init();

unsigned char t;

OLED\_Init();

OLED\_Clear();

t=' '; //获取需要显示字符的ASC码

while (1)

{

OLED\_Clear();

OLED\_ShowCHinese(0,0,0); //李天凌毕业设计

OLED\_ShowCHinese(18,0,1);

OLED\_ShowCHinese(36,0,2);

OLED\_ShowCHinese(54,0,3);

OLED\_ShowCHinese(72,0,4);

OLED\_ShowCHinese(90,0,5);

OLED\_ShowCHinese(108,0,6);

OLED\_ShowString(0,2,"1.3' OLED TEST");

OLED\_ShowString(20,4,"2021/06/01");

OLED\_ShowString(0,6,"ASCII:");

OLED\_ShowString(63,6,"CODE:");

OLED\_ShowChar(48,6,t);//显示ASCII字符

t++;

if(t>'~')

t=' ';

OLED\_ShowNum(103,6,t,3,16);//显示ASCII字符的码值

HAL\_Delay(3000);

OLED\_All();

HAL\_Delay(1000);

OLED\_Clear();

HAL\_Delay(1000);

OLED\_DrawBMP(0,0,128,8,1); //图片显示(图片显示慎用，生成的字表较大，会占用较多空间，FLASH空间8K以下慎用)

HAL\_Delay(3000);

OLED\_DrawBMP(0,0,128,8,2);

HAL\_Delay(3000);

}

}

### 添加用户定义的功能函数

/\* USER CODE BEGIN 4 \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

向SSD1306写入一个字节。

dat:要写入的数据/命令

cmd:数据/命令选择 0,表示命令;1,表示数据;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void OLED\_WR\_Byte(unsigned char dat,unsigned char select)

{

unsigned char i;

if(select)

HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_1,SET); //数据/命令选择线DC=1输入数据

else

HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_1,RESET); //数据/命令选择线DC=0输入命令

HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_0,RESET); //片选CS=0低电平有效

aTxBuffer\_T[0]=dat;

HAL\_SPI\_Transmit(&hspi1, (uint8\_t \*)aTxBuffer\_T, 1, 5000); //仅发送模式，发送1数据，限时5s

HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_0,SET); //片选CS=1高电平失效

HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_1,SET); //数据/命令选择线DC=1等待输入数据

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

设置光标位置

x为列数0-127共128列（128\*1=128）

y为行数0-7共8行（8\*8=64）

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void OLED\_Set\_Pos(unsigned char x, unsigned char y)

{

OLED\_WR\_Byte(0xb0+y,OLED\_CMD);

OLED\_WR\_Byte(((x&0xf0)>>4)|0x10,OLED\_CMD);

OLED\_WR\_Byte((x&0x0f)|0x01,OLED\_CMD);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

清屏函数。

因为OLED自发光，无背光。

所以清完屏,整个屏幕是黑色的!

和没点亮一样!!!

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void OLED\_Clear(void)

{

for(unsigned char y=0;y<8;y++) //8行\*8位数据（64）

{

OLED\_Set\_Pos(0,y); //只有行地址不同时需要设置起始位

for(unsigned char x=0;x<128;x++)

OLED\_WR\_Byte(0x00,OLED\_DATA); //列地址是自动加的

}

}

/\*初始化SSD1306\*/

void OLED\_Init(void)

{

HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_2,SET); // 复位信号RES=1

HAL\_Delay(100);

HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_2,RESET); // 复位信号RES=0

HAL\_Delay(100);

HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_2,SET); // 复位信号RES=1

OLED\_WR\_Byte(0xAE,OLED\_CMD);//--turn off oled panel

OLED\_WR\_Byte(0x00,OLED\_CMD);//---set low column address

OLED\_WR\_Byte(0x10,OLED\_CMD);//---set high column address

OLED\_WR\_Byte(0x40,OLED\_CMD);//--set start line address Set Mapping RAM Display Start Line (0x00~0x3F)

OLED\_WR\_Byte(0x81,OLED\_CMD);//--set contrast control register

OLED\_WR\_Byte(0xCF,OLED\_CMD); // Set SEG Output Current Brightness

OLED\_WR\_Byte(0xA1,OLED\_CMD);//--Set SEG/Column Mapping 0xa0左右反置 0xa1正常

OLED\_WR\_Byte(0xC8,OLED\_CMD);//Set COM/Row Scan Direction 0xc0上下反置 0xc8正常

OLED\_WR\_Byte(0xA6,OLED\_CMD);//--set normal display

OLED\_WR\_Byte(0xA8,OLED\_CMD);//--set multiplex ratio(1 to 64)

OLED\_WR\_Byte(0x3f,OLED\_CMD);//--1/64 duty

OLED\_WR\_Byte(0xD3,OLED\_CMD);//-set display offset Shift Mapping RAM Counter (0x00~0x3F)

OLED\_WR\_Byte(0x00,OLED\_CMD);//-not offset

OLED\_WR\_Byte(0xd5,OLED\_CMD);//--set display clock divide ratio/oscillator frequency

OLED\_WR\_Byte(0x80,OLED\_CMD);//--set divide ratio, Set Clock as 100 Frames/Sec

OLED\_WR\_Byte(0xD9,OLED\_CMD);//--set pre-charge period

OLED\_WR\_Byte(0xF1,OLED\_CMD);//Set Pre-Charge as 15 Clocks & Discharge as 1 Clock

OLED\_WR\_Byte(0xDA,OLED\_CMD);//--set com pins hardware configuration

OLED\_WR\_Byte(0x12,OLED\_CMD);

OLED\_WR\_Byte(0xDB,OLED\_CMD);//--set vcomh

OLED\_WR\_Byte(0x40,OLED\_CMD);//Set VCOM Deselect Level

OLED\_WR\_Byte(0x20,OLED\_CMD);//-Set Page Addressing Mode (0x00/0x01/0x02)

OLED\_WR\_Byte(0x02,OLED\_CMD);//

OLED\_WR\_Byte(0x8D,OLED\_CMD);//--set Charge Pump enable/disable

OLED\_WR\_Byte(0x14,OLED\_CMD);//--set(0x10) disable

OLED\_WR\_Byte(0xA4,OLED\_CMD);// Disable Entire Display On (0xa4/0xa5)

OLED\_WR\_Byte(0xA6,OLED\_CMD);// Disable Inverse Display On (0xa6/a7)

OLED\_WR\_Byte(0xAF,OLED\_CMD);//--turn on oled panel

OLED\_WR\_Byte(0xAF,OLED\_CMD); /\*display ON\*/

OLED\_Clear();

OLED\_Set\_Pos(0,0);

}

/\*蓝屏函数,全部点亮\*/

void OLED\_All(void)

{

for(unsigned char y=0;y<8;y++)

{

OLED\_Set\_Pos(0,y);

for(unsigned char x=0;x<128;x++)

OLED\_WR\_Byte(0xff,OLED\_DATA);

}

}

/\*开启OLED显示\*/

void OLED\_Display\_On(void)

{

OLED\_WR\_Byte(0X8D,OLED\_CMD); //SET DCDC命令

OLED\_WR\_Byte(0X14,OLED\_CMD); //DCDC ON

OLED\_WR\_Byte(0XAF,OLED\_CMD); //DISPLAY ON

}

/\*关闭OLED显示\*/

void OLED\_Display\_Off(void)

{

OLED\_WR\_Byte(0X8D,OLED\_CMD); //SET DCDC命令

OLED\_WR\_Byte(0X10,OLED\_CMD); //DCDC OFF

OLED\_WR\_Byte(0XAE,OLED\_CMD); //DISPLAY OFF

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

在指定位置显示一个字符,包括部分字符

x:0~127列 y:0~7行（8行\*8位）

mode:0,反白显示;1,正常显示

size:选择字体 16/12

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void OLED\_ShowChar(unsigned char x,unsigned char y,unsigned char chr)

{

unsigned char nomber=0,i=0;

nomber=chr-' ';//得到偏移后的值即ASC码偏移量 设置空格为0号字符

if(x>X\_WIDTH-1) //如果超出这一行自动跳转到下一行（+2）

{

x=0;

y=y+2;

}

if(SIZE ==16) //按照16号字体显示 显示方式与原理同汉字

{

OLED\_Set\_Pos(x,y);

for(i=0;i<8;i++)

OLED\_WR\_Byte(L8H16[nomber\*2][i],OLED\_DATA);

OLED\_Set\_Pos(x,y+1);

for(i=0;i<8;i++)

OLED\_WR\_Byte(L8H16[nomber\*2+1][i],OLED\_DATA);

}

else //字体除了16以外都是小字体，设置其他值无用

{

OLED\_Set\_Pos(x,y+1); //设置起始地址为下半屏幕

for(i=0;i<6;i++)

OLED\_WR\_Byte(L6H8[nomber][i],OLED\_DATA);

}

}

/\* 显示一个字符串\*/

void OLED\_ShowString(unsigned char x,unsigned char y,unsigned char \*chr)

{

unsigned char i=0;

while (chr[i]!='\0') //不是字符串的结束则一直循环

{

OLED\_ShowChar(x,y,chr[i]); //在x，y处显示字符

x+=8; //x=x+8 列地址加8准备显示下一字符

if(x>120) //位置不够显示当前字符，去下一行显示

{

x=0;

y+=2;

}

i++; //扫描下一字符

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

显示汉字

x列y行 从0开始

no寻址需要显示的汉字

两个for循环，上下半屏分开显示

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void OLED\_ShowCHinese(unsigned char x,unsigned char y,unsigned char no)

{

unsigned char t; //循环用的变量

OLED\_Set\_Pos(x,y); //设置上半屏起始地址

for(t=0;t<16;t++)

OLED\_WR\_Byte(Hzk[2\*no][t],OLED\_DATA);

OLED\_Set\_Pos(x,y+1); //下半屏起始地址，比上半屏多一列

for(t=0;t<16;t++)

OLED\_WR\_Byte(Hzk[2\*no+1][t],OLED\_DATA);

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

显示显示BMP图片128×64

起始点坐标(x,y)

列x的范围0-127，行y的范围0-7

x0y0起始，x1y1结束

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void OLED\_DrawBMP(unsigned char x0,unsigned char y0,unsigned char x1,unsigned char y1,unsigned char select\_BMP)

{

for(unsigned char y=y0;y<y1;y++)

{

OLED\_Set\_Pos(x0,y);

for(unsigned char x=x0;x<x1;x++)

{

if(select\_BMP==1)

OLED\_WR\_Byte(BMP1[y][x],OLED\_DATA);

else if(select\_BMP==2)

OLED\_WR\_Byte(BMP2[y][x],OLED\_DATA);

else ;

}

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

幂函数m^n

用来辅助获取要显示的某一位

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

int OLED\_pow(unsigned char m,unsigned char n)

{

unsigned int result=1;

while(n--) result\*=m;

return result;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

显示2个数字

x,y :起点坐标

len :数字的位数

size:字体大小

num:数值(0~4294967295);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void OLED\_ShowNum(unsigned char x,unsigned char y,unsigned int num,unsigned char len,unsigned char size2)

{

unsigned char t,number\_byte; //循环与输出位

unsigned char enshow=0; //显示控制变量，显示完置一

for(t=0;t<len;t++)

{

number\_byte=(num/OLED\_pow(10,len-t-1))%10; //输出位，从高到低检测

if(number\_byte!=0) //显示内容非0，开启显示（之后不再显示空格）

{

enshow=1;

OLED\_ShowChar(x+(size2/2)\*t,y,number\_byte+'0');

}

else if(enshow==0) //显示内容为0，且没有开启显示，显示空格

OLED\_ShowChar(x+(size2/2)\*t,y,' ');

else //开启显示之后，正常显示0

OLED\_ShowChar(x+(size2/2)\*t,y,'0');

}

}

/\* USER CODE END 4 \*/