南华大学

计算机学院

实 验 报 告

（ 2023 ~2024 学年度 第 一 学期 ）

|  |  |
| --- | --- |
| 实验名称 | 传感器技术实训报告 |

|  |  |  |  |
| --- | --- | --- | --- |
| 专业 | 医学信息工程 | 班级 | 本2022级 |

|  |  |  |  |
| --- | --- | --- | --- |
| 地点 | 崇业楼202 | 教师 | 李 悛 |

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 贺潇剑 | 学号 | 20220010119 |

传感器技术实训项目报告

1. **实训目的**

1、以实际项目研发为驱动，以掌握基于传感器解决实际项目需求为目标

2、掌握所用的传感器模块

3、掌握单片机编程的方法

4、掌握硬件系统构建方法

5、熟练掌握嵌入式应用程序的编写

6、掌握基于传感器与单片机技术解决实际问题的思路与方法

1. **实验设备**

1、数字温湿度传感器DHT11

2、51单片机开发板STC89C52

3、显示屏LCD1602

1. **项目硬件方案**
2. 开发板

STC89C52是一款经典的51系列单片机，拥有较高的性价比和稳定性。

核心特性：

内核：与传统的8051单片机兼容，采用8051内核。

主频：最高可达24MHz，可以通过外部晶振或内部RC振荡器来设定。

存储器：具有32KB的Flash程序存储器，用于存储用户程序；1280字节的内部RAM和1280字节的扩展RAM。

I/O端口：8位双向I/O端口P0、P1、P2、P3，共32个I/O口。

功能特点：

定时器/计数器：3个16位的定时器/计数器。

中断系统：具有5个中断源，可以处理外部中断、定时器中断等多种中断请求。

串行通信：具备全双工串行通信接口，可用于与其他设备进行串行通信。

看门狗定时器：内置看门狗定时器，用于系统稳定运行。

开发板资源：

LED灯：通常配备若干个LED灯，用于指示或调试程序。

按键：设有若干个按键，用于输入或触发特定功能。

晶振电路：提供外部晶振接口，用户可以选择合适的晶振来设定系统时钟。

串口通信接口：提供串口通信接口，方便与电脑或其他设备进行通信。

电源接口：提供标准的电源接口，一般为USB或DC接口。

编程环境：

STC89C52单片机可以使用Keil等集成开发环境进行编程。

支持C语言和汇编语言编程。

1. 传感器

温湿度传感器DHT11

DHT11是一款性价比高、响应速度快、抗干扰能力强的数字温湿度传感器。

传感器原理：

DHT11温湿度传感器的核心由一个电阻式感湿元件和一个NTC（负温度系数）测温元件组成。电阻式感湿元件的电阻值会随着空气中水分子的吸附而变化，从而测量湿度；而NTC测温元件的电阻值会随着温度的变化而变化，用于测量温度。这两个元件的信号经过一个内置的高性能8位单片机处理，将模拟信号转换为数字信号，并通过单总线接口输出。

具体器件：

电阻式感湿元件：用于测量环境湿度。

NTC测温元件：用于测量环境温度。

8位单片机：处理感湿和测温元件的信号，并控制数据的输出。

OTP内存：存储传感器的校准系数。

通信接口：

电源引脚：DHT11需要3至5.5V的供电电压，通常包括

VDD（正电源）和GND（接地）两个引脚。

数据引脚（DATA）：这是一个单线双向接口，用于与微处理器进行通信。

数据传输过程：

开始信号：主机（微处理器）将DATA线拉低至少18毫秒，以唤醒DHT11。然后主机将DATA线拉高，准备接收DHT11的响应。

响应信号：DHT11在检测到开始信号后，会拉低DATA线80微秒，作为响应信号。随后，DHT11将DATA线拉高80微秒，准备发送数据。

数据传输：DHT11以40位的数据格式发送数据，数据包括 8位湿度整数数据，8位湿度小数数据（目前为0，用于未来扩展），8位温度整数数据，8位温度小数数据，8位校验和（用于检查数据传输的正确性）。每位数据的传输由一个低电平（50微秒）和一个高电平（26至28微秒表示0，70微秒表示1）组成。

数据格式：数据传输是高位先出（MSB first）。校验和是前四个8位数据的累加和的最低8位。

1. 硬件总体框架

3.1硬件组成

单片机核心

处理器：STC89C52

输出设备

LED灯：8个LED灯，连接到P1端口（P1.0至P1.7）。

蜂鸣器：1个蜂鸣器，连接到P0.7。

输入设备

按键：4个按键，连接到P3端口（P3.3至P3.6），用于用户按键输入。

传感器和显示屏

温湿度传感器DHT11：用于测量环境温度和湿度，数据通过P0.5传输。

LCD1602显示屏：用于显示文本信息，通过L2模块连接。

电源和编程接口

电源：通过数据线连接PC端笔记本供电。

编程接口：使用数据线进行编程内容的下载和串口通信。

3.2硬件资源分配

GPIO口分配

P1端口（P1.0至P1.7）：控制LED的开关。

P0.7：驱动蜂鸣器发声。

P3.3至P3.6：分配给4个按键，用于读取按键状态。

P0.5：分配给DHT11传感器的数据引脚，用于温湿度数据交互。

L2模块：连接LCD1602显示屏，进行数据和控制信号的传输。

通信方式

并行通信：LCD1602显示屏的数据和控制信号传输，需要多个GPIO口来并行发送数据。

串行通信：单片机与PC之间的编程下载和串口发送缓冲区控制。

单总线通信：通过P0.5引脚与DHT11传感器的数据传输，。

3.3功能说明

LED控制：通过编程控制P1端口的各个引脚的高低电平，控制LED灯的亮灭。

蜂鸣器控制：通过P0.7驱动蜂鸣器发声。

按键读取：通过读取P3.3至P3.6的状态，检测按键是否被按下。

温湿度读取：通过P0.5与DHT11进行单总线通信，按照DHT11的时序要求读取温湿度数据。

LCD显示：通过并行通信方式，将文本信息显示在LCD1602屏幕上。

1. **软件方案与设计**

模块设计：

1. LCD1602中显示温湿度数据
2. 温湿度工作 当温度大于等于39度小于等于10度时,蜂鸣器响，小于39度大于10度不响
3. 强制关闭蜂鸣器

对于功能1，我们定义了一个函数StreamLED，其中通过延时函数来控制LED小灯依次亮灭，实现流水灯，然后通过串口发送指令0x01，调用函数StreamLED来实现流水灯效果。

对于功能2，我们定义了一个函数Blink，其中通过五次循环，每次循环控制8盏LED灯同时亮，延时后再控制8盏LED灯同时灭。然后通过串口发送指令0x02，调用函数Blink来实现闪烁灯效果。

对于功能3，其目的是把温湿度传感器的数据显示到LCD屏幕上，通过串口发送指令先读取温湿度传感器数据，然后将其中的温湿度的整数和小数分别解析出来并存储在变量中。然后按照要求的数据格式将每个字节发送到LCD屏幕上展示。

对于功能4，通过串口发送指令0x04，接收到温湿度数据后，判断温度数据是否大于等于39摄氏度或者小于等于10摄氏度，然后进行报警。

对于功能5，通过串口发送指令0x05，控制蜂鸣器引脚为高电平即可。

详细设计

1. **温湿度传感器数据获取：**

// 读取温湿度数据

uchar receive\_byte()

{

  uchar i, temp;

  for (i = 0; i < 8; i++)

  {

    while (!io)

      ;

    delay1();

    delay1();

    delay1();

    temp = 0;

    if (io == 1)

      temp = 1;

    while (io)

      ;

    data\_byte <<= 1;

    data\_byte |= temp;

  }

  return data\_byte;

}

首先从连接到io引脚的传感器中读取一个字节的温度和湿度数据。先等待io引脚变低，此时数据表示可用。然后调用三次延时，读取io引脚的状态并存储在temp中，等待io引脚变高后，将data\_byte左移1位并将temp值添加到最低位。

1. **通过串口发送指令实现：**

void UART\_ISR() interrupt 4

{

  if (RI)

  {

    recv\_data = SBUF;

    UART\_SendByte(SBUF);

    RI = 0;

  }

}

当使用串口发送数据时，会调用中断函数，发送的数据会存放到缓存区，这里缓存区定义为SBUF，由于SBUF会变化，这里用recv\_data来存储发送的数据，然后根据recv\_data来判断是什么指令。

1. **LCD1602显示模块：**

void write\_cmd(uchar cmd)

{

while (lcd\_bz())

;

rs = 0;

rw = 0;

ep = 0;

P2 = cmd;

ep = 1;

ep = 0;

}

LCD1602的控制通过RS、RW、EP三个控制引脚实现。RS为0时写入指令,为1时写入数据;RW为0时写操作,为1时读操作;EP为使能端,高脉冲有效。写入指令时先检查LCD忙状态,然后设置控制引脚,最后通过P2端口写入指令数据

1. **LED流水灯控制：**

void StreamLED()

{

LED0 = 0;

delay(300);

LED0 = 1;

LED1 = 0;

//...

}

通过控制P1口的8个引脚实现LED流水灯效果。每次点亮一个LED并延时300ms后熄灭,然后点亮下一个LED,依次循环实现流水灯效果。LED引脚为低电平点亮,高电平熄灭。

1. **温湿度数据解析**

void receive()

{

uchar T\_H, T\_L, R\_H, R\_L, check;

start();

R\_H = receive\_byte(); //湿度高8位

R\_L = receive\_byte(); //湿度低8位

T\_H = receive\_byte(); //温度高8位

T\_L = receive\_byte(); //温度低8位

check = receive\_byte(); //校验和

num\_check = R\_H + R\_L + T\_H + T\_L;

if (num\_check == check) //校验正确则更新数据

{

RH = R\_H;

TH = T\_H;

//...

}

}

从DHT11传感器读取5个字节数据,包括湿度高8位、湿度低8位、温度高8位、温度低8位和校验和。通过校验和验证数据的正确性,校验正确则更新全局变量中的温湿度值

1. **蜂鸣器报警控制：**

if (TH >= 39 || TH <= 10) //温度超过阈值时蜂鸣器报警

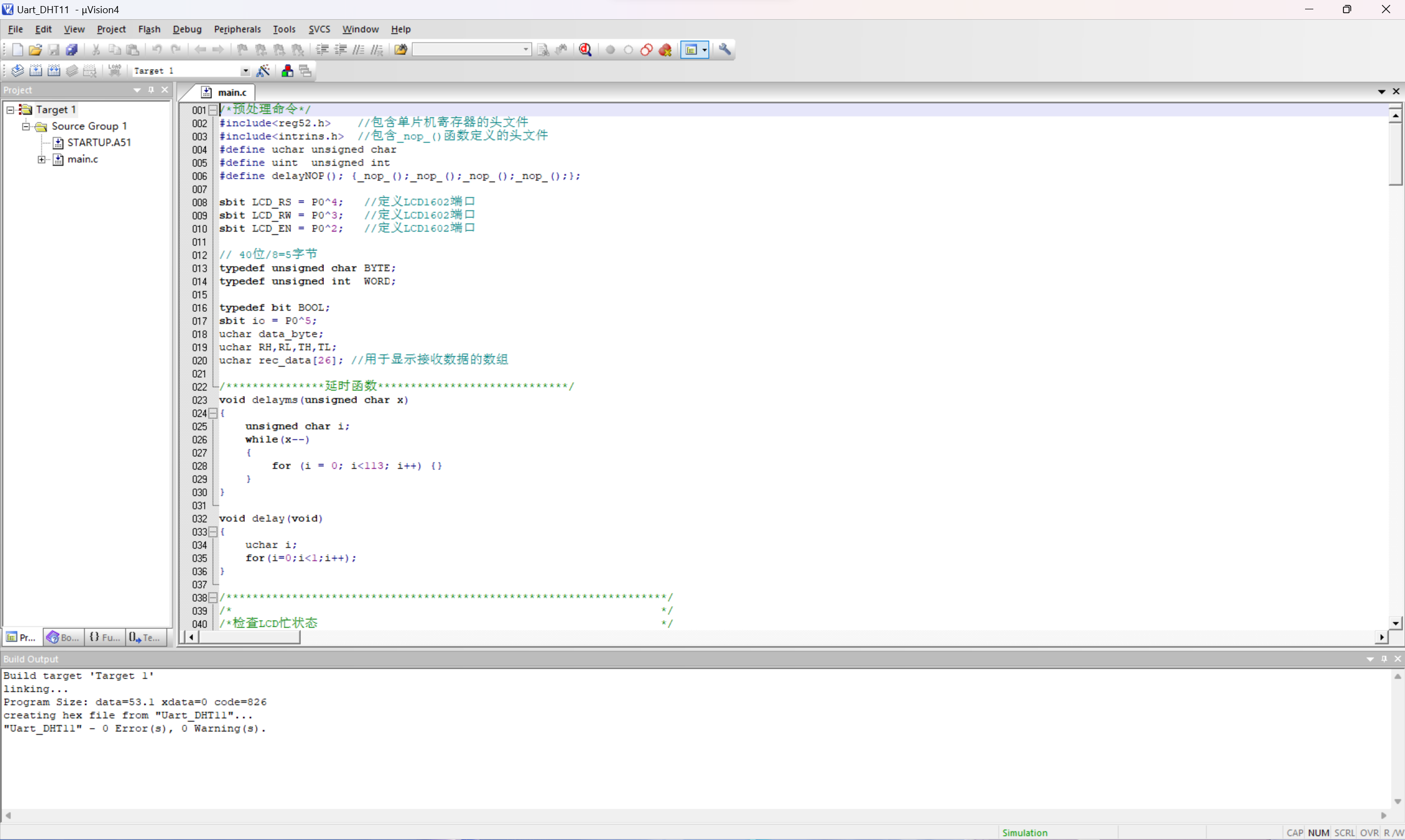
BEEP = 0; //蜂鸣器低电平有效

通过检测温度值是否超出设定阈值(大于20度或小于10度)来控制蜂鸣器报警。蜂鸣器接P0.7引脚,低电平时蜂鸣器发声。

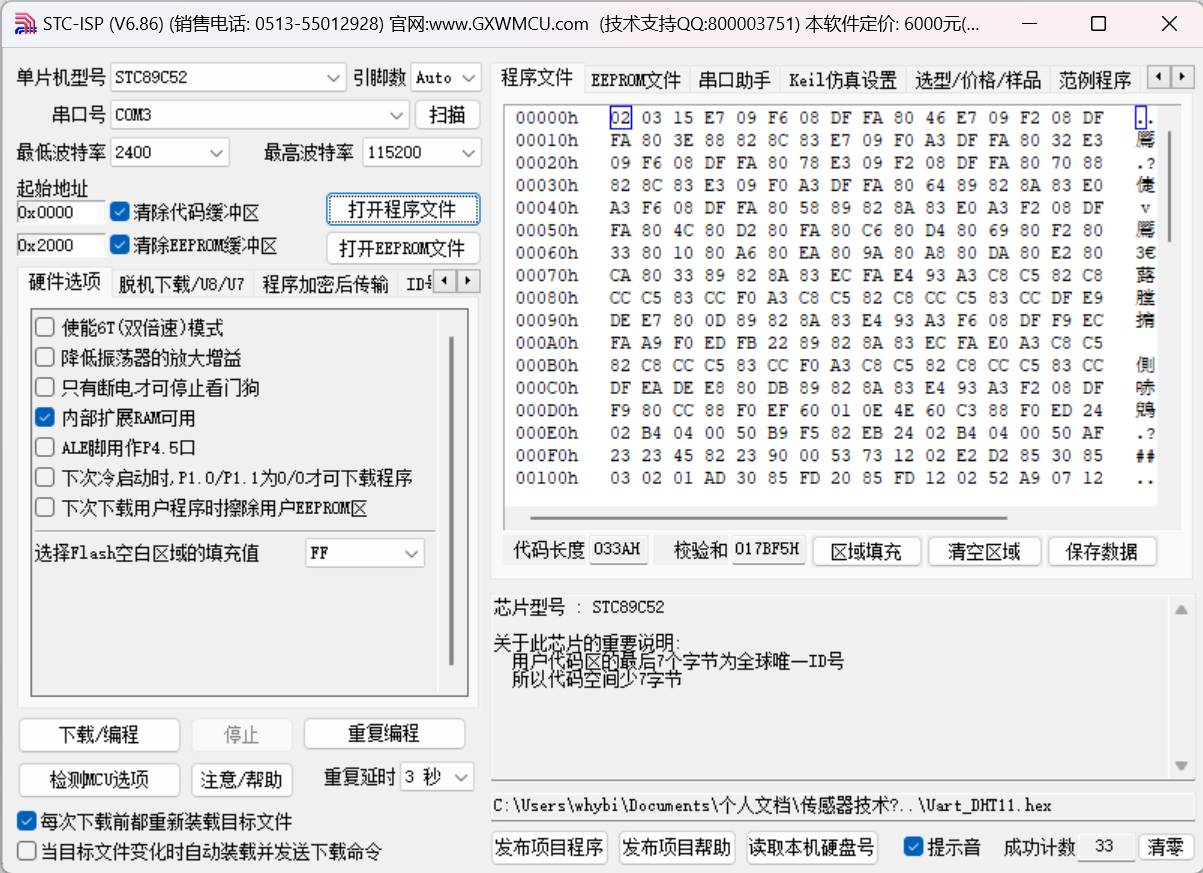
1. **实验步骤与现象**

1、导入程序至开发板：

（1）打开Keil uVision4程序，并导入设计好的代码，并编译成所需的hex文件；



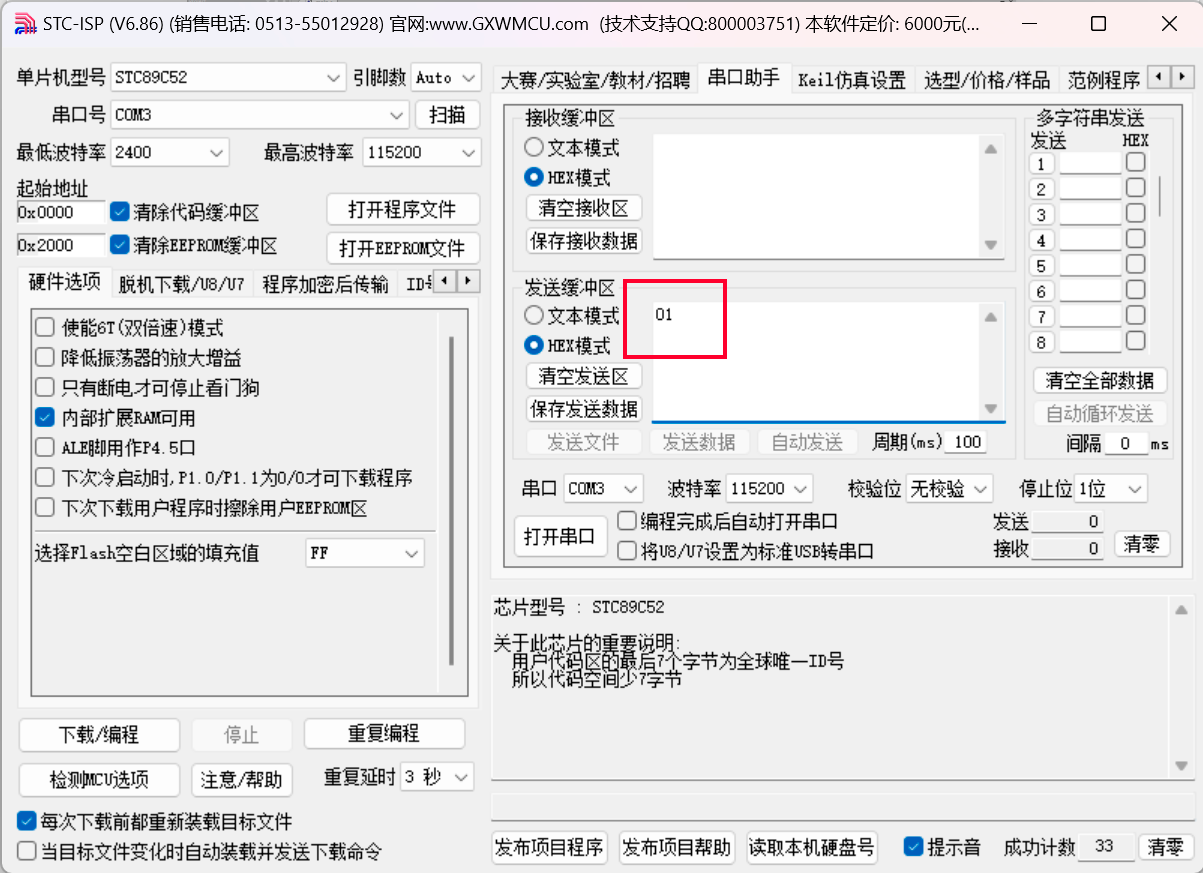
1. 通过isp程序，将编译好的代码通过数据线下载至开发板上：



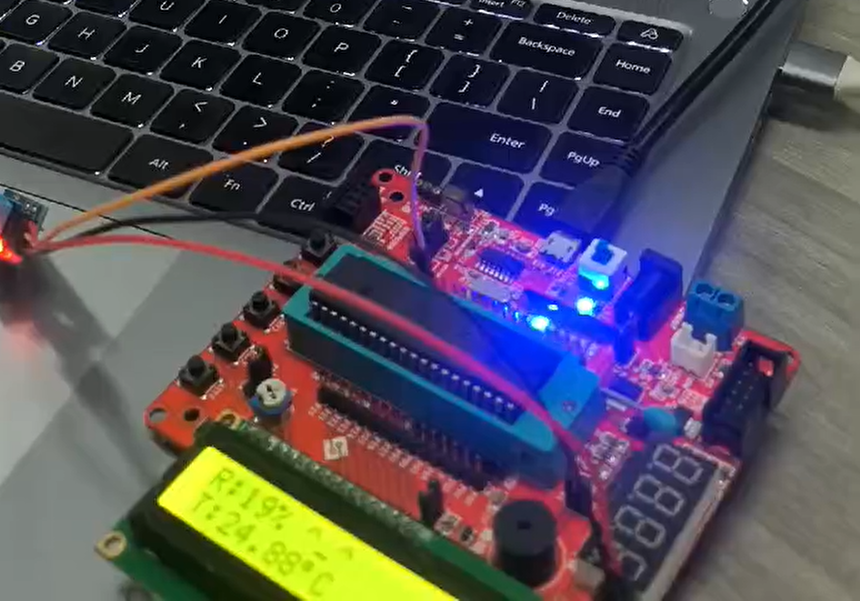
2、对预先设计好的功能进行测试：

（1）LED流水灯：

通过串口助手，导入命令0x01：



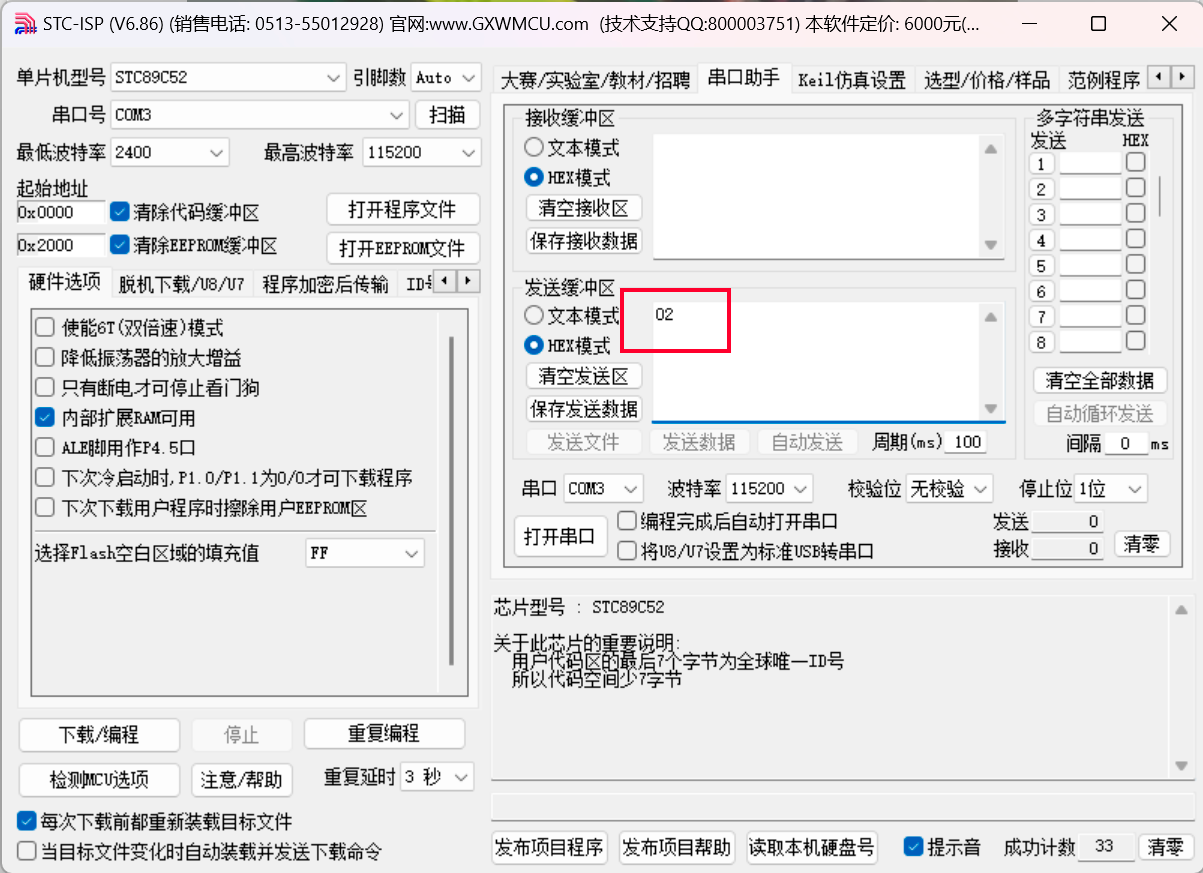
可以看到，开发板上的Led灯珠呈流水灯进行闪烁：



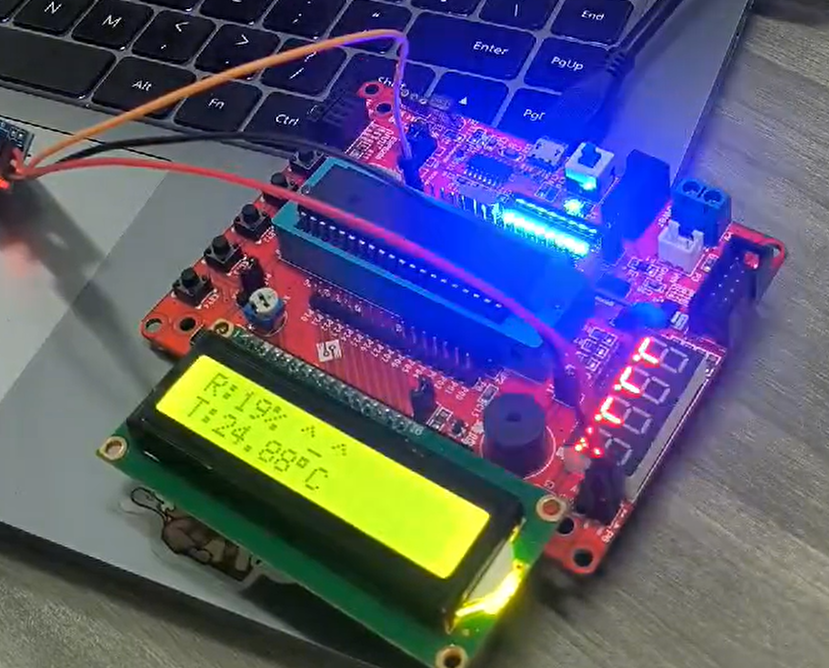
由于设定为流水灯样式的依次闪烁效果，因此图片仅能展示为一个灯珠亮。

1. LED闪烁灯+蜂鸣器

通过串口助手，导入命令0x02：

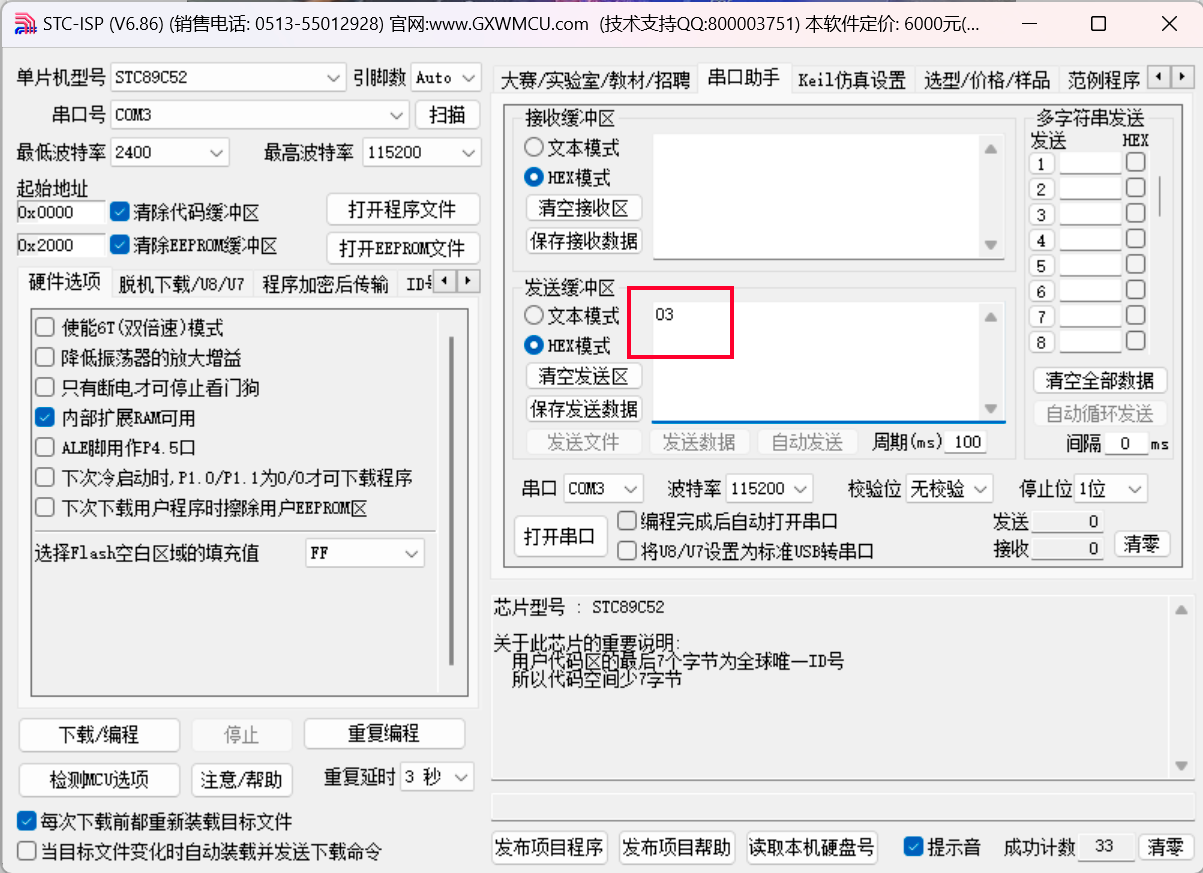


由于设定Led的闪烁模式为：全亮->全暗->全亮循环，因此可以看到Led全亮的情况：

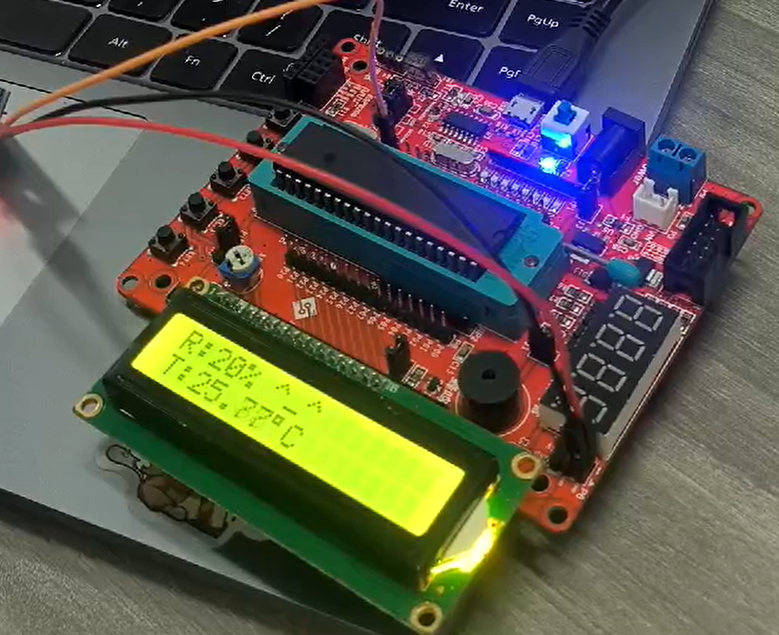


1. 温湿度在LCD1602中显示温湿度数据

通过串口助手，传入命令0x03：



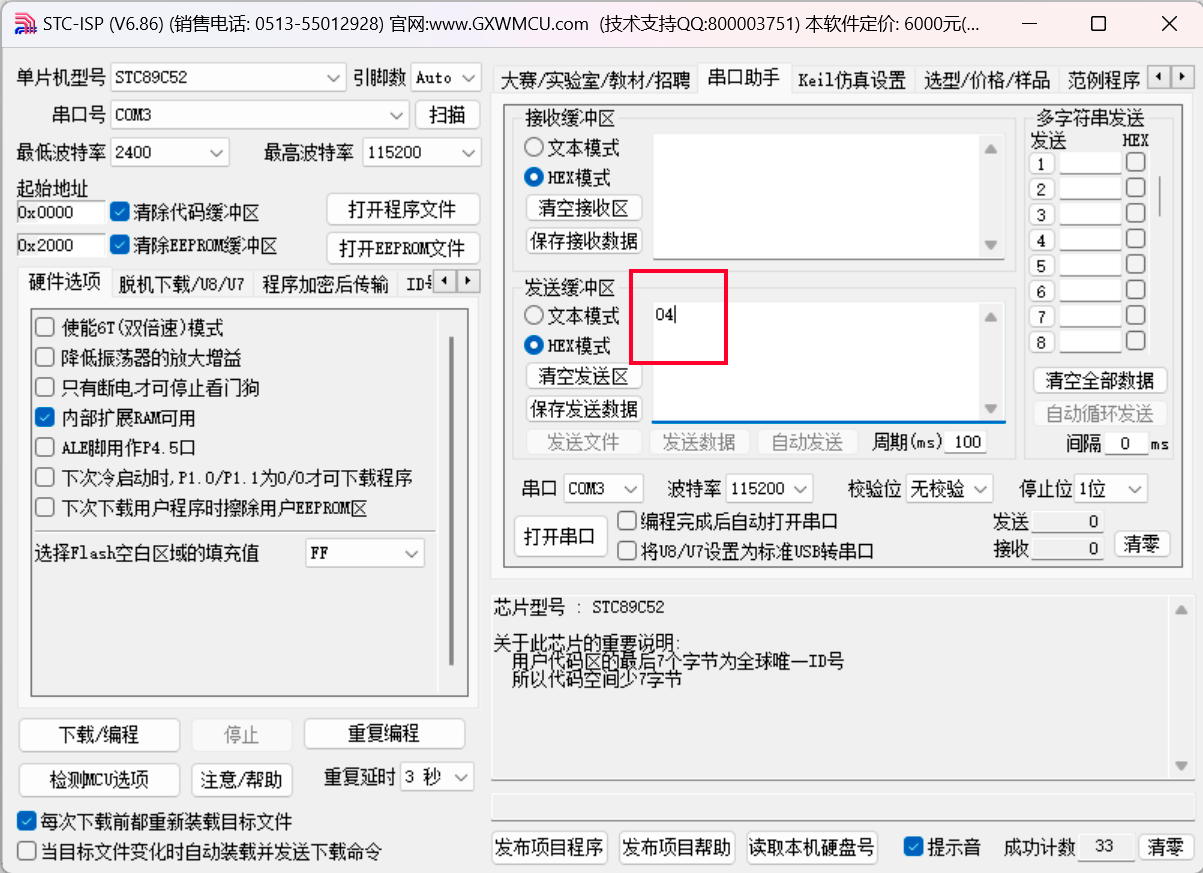
传入该命令后，温湿度传感器所收集到的数据会在LCD1602的液晶显示屏上进行显示：



可以看到当前湿度为20%，温度为25.77℃。

1. 温湿度工作 当温度大于等于39度小于等于10度时,蜂鸣器响，小于39度大于10度不响

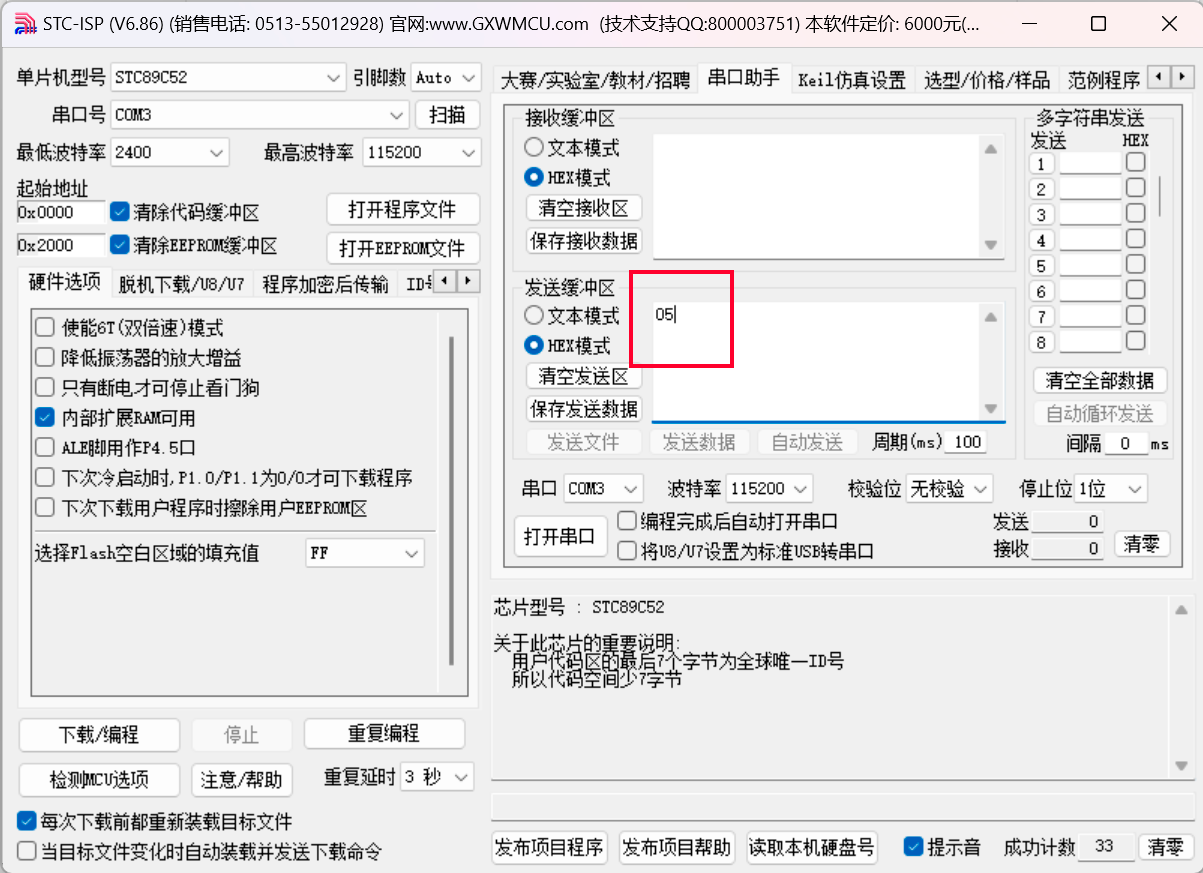
通过串口助手，传入命令0x04：



当满足设定的条件之后，蜂鸣器便会响起。

1. 强制关闭蜂鸣器

通过串口助手，传入命令0x05：



此时原本响起的蜂鸣器将会关闭，用于在实验过程中手动将蜂鸣器关闭。

1. **总结与分析**

通过本项目实践，我成功实现了以下目标：

项目中，合作利用DHT11温湿度传感器采集环境数据，并通过单片机进行处理。

熟悉了DHT11温湿度传感器、LCD1602液晶显示屏模块，了解了它们的工作原理及接口方法。

学会了单片机编程方法，掌握了STC89C52单片机的编程技巧。

初步掌握了嵌入式应用程序的编写，实现了传感器数据读取、处理及显示等功能。

在项目实施过程中，我负责了以下工作：

编写了LCD和蜂鸣器的控制程序，实现了规律执行的功能，用于提示环境异常。

运用了中断技术，提高了系统响应速度和运行效率。

了解了显示器的内容显示方法。

完成了温湿度传感器的数据读取。

通过本次项目，我对STC89C52单片机的布局、组件、程序下载、硬件连接等方面有了了解，并在实践中提高了编程能力。总之，本次项目使我受益匪浅，为我在嵌入式领域的发展打下了坚实基础。