**嵌入式技术基础与实践学习札记**

目录

[**第一章、 概述** **3**](#_Toc7808_WPSOffice_Level1)

[1.1.1嵌入式系统的定义 3](#_Toc3605_WPSOffice_Level2)

[1.1.2嵌入式系统的由来及发展历史 3](#_Toc1924_WPSOffice_Level2)

[1.1.3嵌入式系统的分类 3](#_Toc16095_WPSOffice_Level2)

[1.2.1嵌入式系统的学习困惑 3](#_Toc10567_WPSOffice_Level2)

[1.2.2嵌入式系统的知识体系 4](#_Toc1624_WPSOffice_Level2)

[1.3.1微控制器简介 4](#_Toc28987_WPSOffice_Level2)

[1.3.2以MCU为核心的嵌入式测控产品的基本组成 4](#_Toc25986_WPSOffice_Level2)

[1.3.3应用处理器简介 5](#_Toc30795_WPSOffice_Level2)

[1.4.1与硬件相关的术语 5](#_Toc1478_WPSOffice_Level2)

[1.4.1与通信相关的术语 5](#_Toc3519_WPSOffice_Level2)

[1.4.3与功能模块相关的术语 5](#_Toc7881_WPSOffice_Level2)

[**第二章、 ARM Cortex-M0处理器** **5**](#_Toc3605_WPSOffice_Level1)

[2.1.2存储器映像 5](#_Toc570_WPSOffice_Level2)

[2.2.1指令简表与寻址方式 6](#_Toc8170_WPSOffice_Level2)

[**第三章、 存储映像 中断源 硬件最小系统** **6**](#_Toc1924_WPSOffice_Level1)

[3.3.1KL25/26系列存储映像 6](#_Toc3010_WPSOffice_Level2)

[3.3.2KL25/26中断源 6](#_Toc25264_WPSOffice_Level2)

[3.4.1硬件最小系统引脚 6](#_Toc4851_WPSOffice_Level2)

[3.4.2对外提供服务的引脚 6](#_Toc5557_WPSOffice_Level2)

[**第四章、 GPIO及程序框架** **6**](#_Toc16095_WPSOffice_Level1)

[4.1通用I/O接口基本概念及连接方法 6](#_Toc31427_WPSOffice_Level2)

[**第五章、 串行通信模块** **15**](#_Toc10567_WPSOffice_Level1)

[6.1.1串行通信的基本概念 15](#_Toc19402_WPSOffice_Level2)

[6.1.2RS232总线标准 15](#_Toc16269_WPSOffice_Level2)

[6.1.3TTL电平到RS232电平转换电路 16](#_Toc14704_WPSOffice_Level2)

[6.1.4串行通信编程模式 16](#_Toc28467_WPSOffice_Level2)

[**A1、A2组合的四种情况，设置四种keyflag状态** **19**](#_Toc1624_WPSOffice_Level1)

[6.3.1中断的基础知识 19](#_Toc15061_WPSOffice_Level2)

[**第七章、定时器相关模块** **25**](#_Toc28987_WPSOffice_Level1)

[7.2.1脉宽调制PWM通用基础知识 25](#_Toc19355_WPSOffice_Level2)

[7.2.2输入捕捉与输出捕捉 26](#_Toc20213_WPSOffice_Level2)

[**第八章、 GPIO应用——键盘、LED及LCD** **26**](#_Toc25986_WPSOffice_Level1)

[8.1.1键盘模型及接口 26](#_Toc12759_WPSOffice_Level2)

[8.1.2键盘编程基本问题、扫描编程原理及键值计算 26](#_Toc22674_WPSOffice_Level2)

[**第九章、 Flash在线编程** **30**](#_Toc30795_WPSOffice_Level1)

[9.1.1Flash在线编程的基本概念 30](#_Toc29614_WPSOffice_Level2)

[9.1.2KL25/26芯片Flash构件头文件及使用方法 30](#_Toc15391_WPSOffice_Level2)

[**第十章、 ADC，DAC与CMP模块** **32**](#_Toc1478_WPSOffice_Level1)

[10.1.1模/数转换器ADC的通用基础知识 32](#_Toc32365_WPSOffice_Level2)

[10.1.2ADC驱动构件及使用方法 32](#_Toc27092_WPSOffice_Level2)

[**第十一章、 SPI,I2C与TSI模块** **33**](#_Toc3519_WPSOffice_Level1)

[11.1.1串行外设接口SPI的通用基础知识 33](#_Toc16668_WPSOffice_Level2)

[11.1.2 SPI驱动构件头文件及使用方法 34](#_Toc7551_WPSOffice_Level2)

[**第十二章、 USB编程** **34**](#_Toc7881_WPSOffice_Level1)

[（1） 支持即插即用 34](#_Toc5522_WPSOffice_Level2)

[（2） 可以使用总线电源 34](#_Toc10710_WPSOffice_Level2)

[（3） 硬件接口标准化、小巧化 34](#_Toc19200_WPSOffice_Level2)

[（4） 支持多种速度和操作模式 34](#_Toc4944_WPSOffice_Level2)

[**第十三章、 系统时钟与其他功能模块** **35**](#_Toc570_WPSOffice_Level1)

[13.4看门狗 35](#_Toc9699_WPSOffice_Level2)

[**冷复位与热复位的区别** **35**](#_Toc8170_WPSOffice_Level1)

# 概述

## 1.1.1嵌入式系统的定义

嵌入式系统是一种计算机硬件和软件的组合，也许还有机械装置，用于实现一个特定功能。在某些情况下，嵌入式系统是一个大系统或产品的一部分。嵌入式系统置入应用对象内部起信息处理和控制作用的专用计算机系统。

通用计算机是计算机硬件和软件的组合，用作通用计算平台。

嵌入式系统即嵌入式计算机，不以计算机面目出现的计算机，这个计算机系统隐含在各类具体产品中，在这些产品中，计算机程序起到了重要作用。

## 1.1.2嵌入式系统的由来及发展历史

计算机是因为科学家需要一个高速的计算工具而产生的，通常把满足海量高速数值计算的计算机称为通用计算机系统，而把嵌入到实际应用系统中，实现嵌入式应用的计算机称为嵌入式计算机系统。

可以说，是因为通信，测控与数据传输等领域对计算机技术的需求催生了嵌入式系统的产生。

## 1.1.3嵌入式系统的分类

嵌入式系统分类很多，可以按照处理器位数分，也可以按照复杂程度来分，还可以暗号其他分类方法，各有各的特点。可以简单的把嵌入式系统分为电子系统智能化和计算机应用延伸两大类。

## 1.2.1嵌入式系统的学习困惑

基本原则：入门时间较快，硬件成本较少，软硬件资料规范，知识要素较多，学习难度较低。

## 1.2.2嵌入式系统的知识体系

从由浅入深，由简到繁的学习规律，嵌入式学习的入门应该选择微控制器，而不是应用处理器，通过对微控制器基本原则与应用的学习，逐步掌握嵌入式系统的软件与硬件基础。

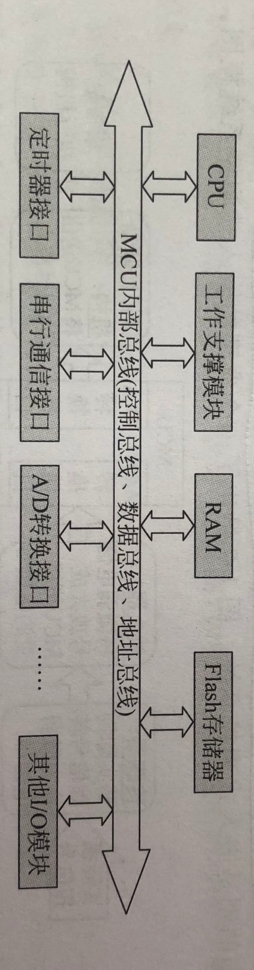
完成一个以MCU为核心的嵌入式系统应用产品设计，需要有硬件、软件相关知识。硬件主要有MCU的硬件最小系统、输入/输出外围电路、人机接口设计。

基本知识体系：

1. 掌握硬件最小系统与软件最小系统框架
2. 掌握常用基本输出的概念、只是要素、构件使用方法及构件设计方法
3. 掌握若干嵌入式通信的概念、知识要素、构件使用及设计方法
4. 掌握常用应用模块的构件设计方法及使用方法及数据处理方法
5. 掌握一门实时操作系统RTOS的基本用法与基本原理
6. 掌握嵌入式软硬件的基本调试方法

## 1.3.1微控制器简介

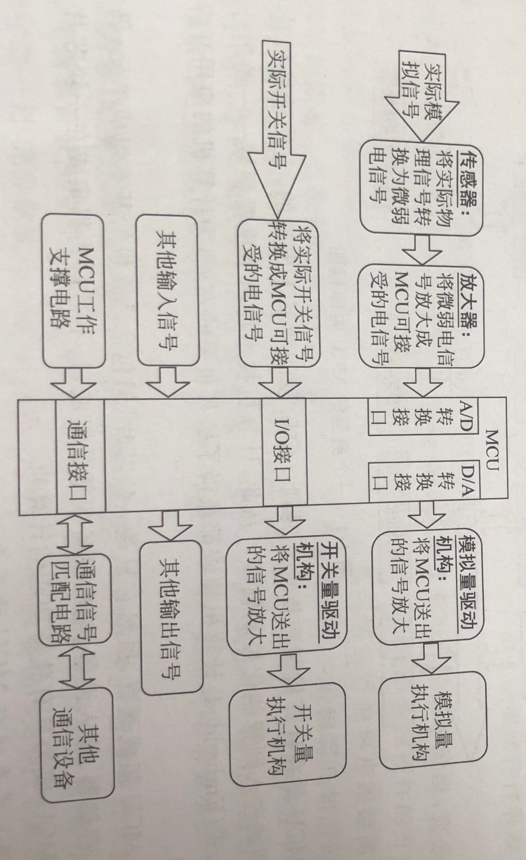
MCU的基本含义：在一块芯片内集成了中央处理单元、存储器、定时器/计数器及多种输入输出I/O接口的比较完整的数字处理系统。



大部分嵌入式系统以MCU为核心进行设计，因此以MCU为核心的系统是应用最广的嵌入式系统。

## 1.3.2以MCU为核心的嵌入式测控产品的基本组成

一个以MCU为核心，比较复杂的嵌入式产品或实际嵌入式应用系统，包含模拟量的输入、输出，开关量的输入、输出及数据通信的部分。



## 1.3.3应用处理器简介

应用处理器的全名是多媒体应用处理器，应用处理器是伴随着便携式移动设备特别是智能手机而产生的。

## 1.4.1与硬件相关的术语

封装、印刷电路板、动态可读写随机存储器、静态可读写随机存储器、只读存储器、闪速存储器、模拟量与开关量。

## 1.4.1与通信相关的术语

并行通信、串行通信、串行外设接口、集成电路互连总线、通用串行总线、控制器局域网、背景调试模式、边界扫描测试协议、串行线调试技术。

## 1.4.3与功能模块相关的术语

通用输入输出、模数转换和数模转换、脉冲宽度调制器、看门狗、液晶显示、发光二极管、键盘。

# ARM Cortex-M0处理器

## 2.1.2存储器映像

CM0+处理器直接寻址空间为4GB，地址范围：0x0000\_0000~0xFFFF\_FFFF。这里所说的存储映像含义：把这4GB空间当做存储器来看待，分成若干区间，都可以安排一些什么实际的物理资源。

## 2.2.1指令简表与寻址方式

指令：与汇编语言类似

寻址方式：立即数寻址、寄存器寻址、偏移寻址、寄存器间接寻址、直接寻址

指令类型：数据传送类指令、数据操作类指令、跳转控制类指令、其他指令

# 存储映像 中断源 硬件最小系统

## 3.3.1KL25/26系列存储映像

存储映像在这里直观的理解为M0+寻址的4GB空间（0x0000\_0000~0xFFFF\_FFFF）被如何使用，都对应了哪些实际的物理介质。

1. 片内Flash区存储映像

KL25/26片内Flash大小为128KB，地址范围是：0x0000\_0000~0x0001\_FFFF，一般被用来存放中断向量、程序代码、常数等，其中192B为中断向量表。

1. 片内RAM区存储映像

KL25/26片内RAM为静态随机存储器SRAM，大小为16KB，地址范围：0x1FFF\_F000~0x2000\_2FFF，一般被用来存储全局变量、静态变量、临时变量等。

1. 其他存储映像

## 3.3.2KL25/26中断源

所谓中断，是指MCU在正常运行程序时，由于MCU内核异常或者MCU各模块发出请求事件，引起MCU停止正在运行的程序，而专区处理异常或执行处理外部事件的程序。

## 3.4.1硬件最小系统引脚

KL25/26硬件最小系统引脚是我们需要为芯片提供服务的引脚，包括电源类引脚、复位引脚、晶振引脚等。

## 3.4.2对外提供服务的引脚

除了需要我们为芯片服务的引脚之外，芯片的其他引脚为我们提供服务，也可以称之为I/O端口资源类引脚。

# GPIO及程序框架

## 4.1通用I/O接口基本概念及连接方法

1. I/O接口的概念
2. 通用I/O
3. 上拉下拉电阻与输入引脚的基本接法：带上拉电阻的连接 带下拉电阻的连接 悬空连接
4. 输出引脚的基本接法

## 4.2.1端口控制模块--------决定引脚复用功能

1. 寄存器映像地址分析
2. 相关名词解释
3. 模拟引脚是指不能够配置成GPIO的引脚
4. 数字引脚是指能够被配置成GPIO的引脚
5. 无源滤波器是由电容、电抗器和电阻器适当组合而成，并兼有无功补偿和调压功能的滤波器。
6. 引脚驱动能力是指引脚放出或吸入电流的承受能力。
7. 转换速率是指电压在高低电平间转换的时间间隔。
8. 数字输出是指芯片只能输出高低电平
9. 数字输入是指芯片只能输入高低电平
10. 引脚复用槽是指信号复用装置与引脚之间的接口
11. wlc俗称写1清0
12. 引脚控制寄存器
13. 全局引脚控制寄存器

## 4.2.2GPIO模块-------对外引脚与内部寄存器

1. KL25的GPIO引脚

可作为GPIO功能的引脚数目及引脚名称：

1. A口有10个引脚
2. B口有12个引脚
3. C口有16个引脚
4. D口又8个引脚
5. E口有15个引脚
6. GPIO寄存器

每个GPIO寄口均有6个寄存器，5个GPIO口共有30个寄存器。各GPIO口的6个寄存器分别是数据输出寄存器、输出置1寄存器、输出清0寄存器、输出反转寄存器、数据输入寄存器、数据方向寄存器。

## 4.2.3GPIO基本编程步骤与基本打通程序

1. 通过端口控制模块的引脚控制寄存器的引脚复用控制字段设定其为GPIO功能
2. 通过GPIO模块相应口的数据方向寄存器来指定相应引脚为输入输出功能
3. 若是输出引脚，则通过设置数据输出寄存器来指定相应引脚输出低电平或者高电平
4. 若是输入引脚，则通过数据输入寄存器获得引脚的状态

## 4.3.1GPIO驱动构件封装要点分析

1. 模块初始化（gpio\_init）

应把引脚设置成GPIO功能，同时定义成输入或输出，若是输出，还要给出初始状态。

1. 设置引脚状态（gpio\_set）

希望通过函数设置引脚是高电平还是低电平。

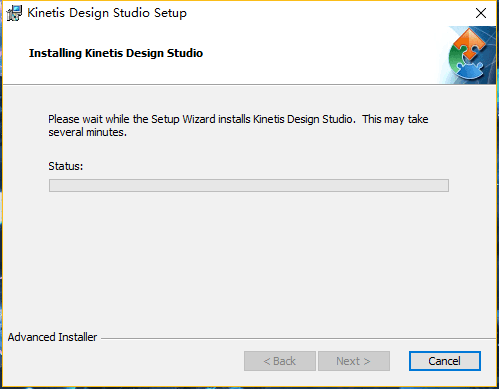
1. 获得引脚状态（gpio\_get）

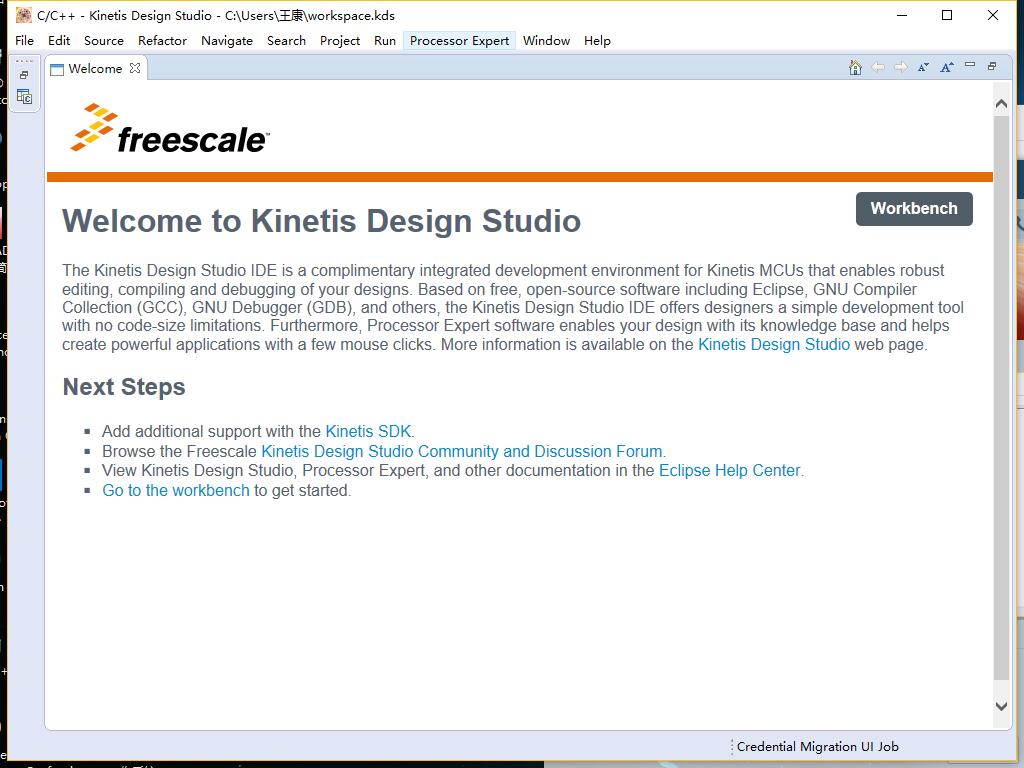
希望通过函数获得引脚的状态是高电平还是低电平。

1. 引脚状态反转（gpio\_reverse）
2. 引脚上下拉使能函数（gpio\_pull）

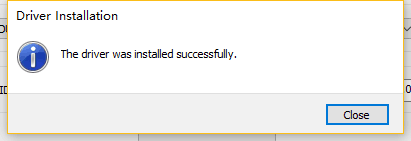
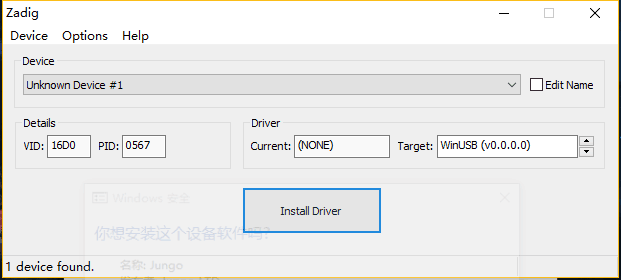
# KL25开发环境搭建

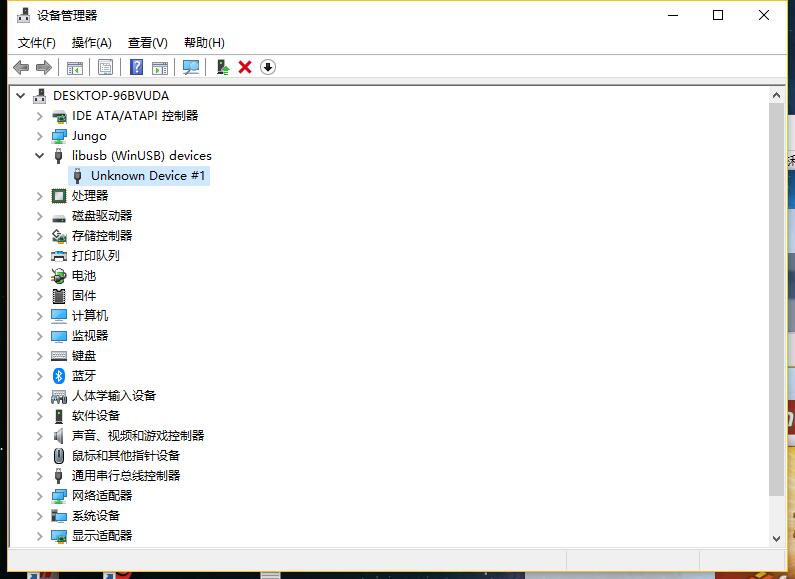
## KDS安装



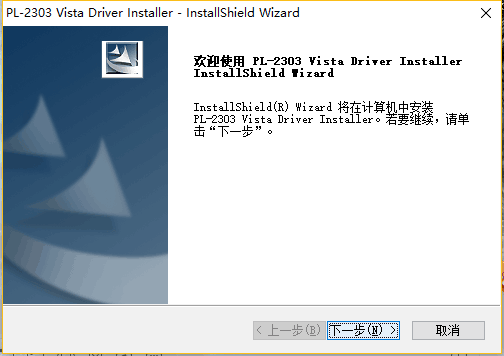


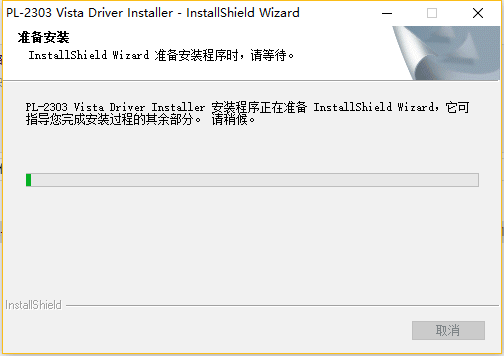
## USBDM辅助软件安装



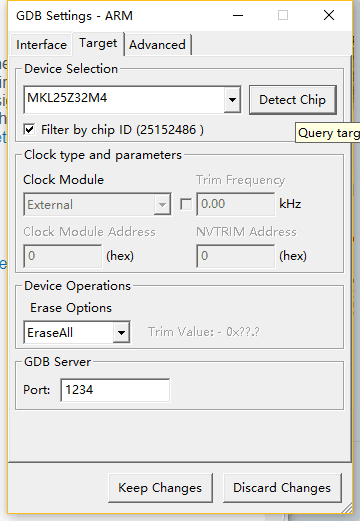


## 调试工具类PL2303alldriver win7-win10

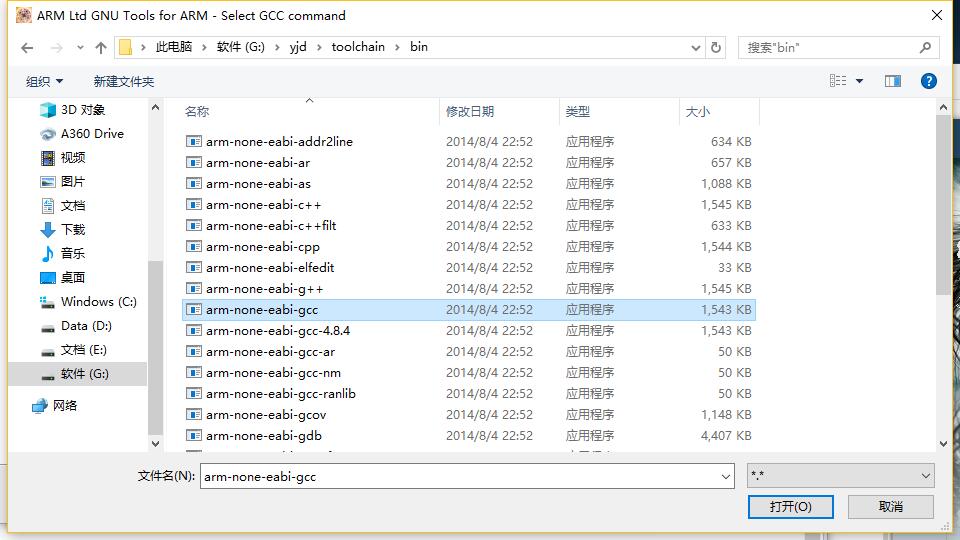


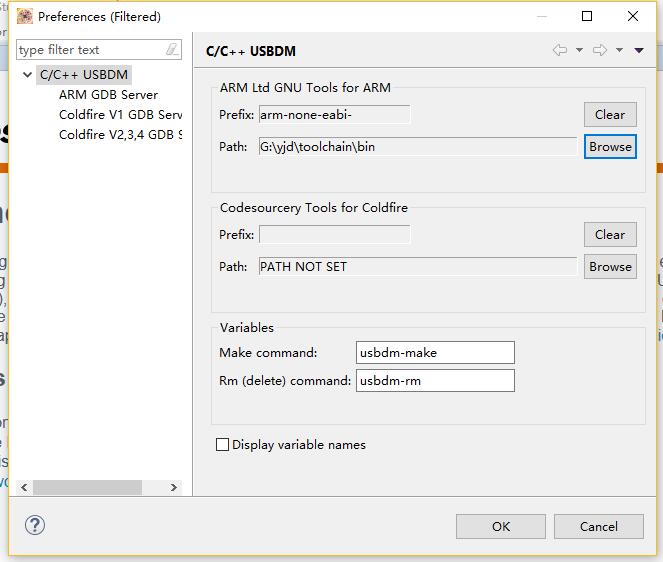


## ARM将程序导入到芯片

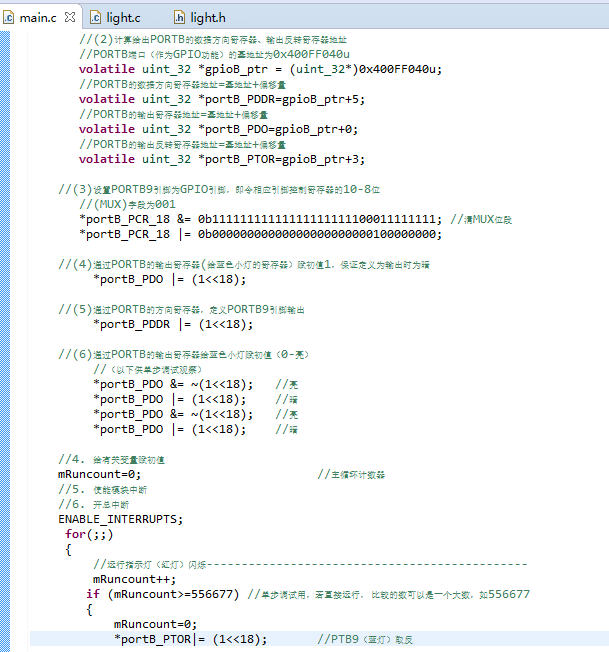
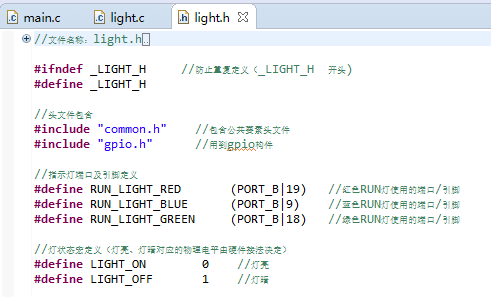


## KDS USBDM调试





## 将开发板的蓝色灯变为其他颜色



控制三色灯颜色的参数在light.h中，19表示红色、9表示蓝色、18表示绿色，只需要将main.c中的对应9改为其他数值即可完成三色灯颜色的转换。

## 4.5工程文件组织框架

1. 工程名与新建工程
2. 工程文件夹内的基本内容

01\_Doc说明文档文件夹，工程改动可以及时记录。

02\_CPU内核文件。

03\_MCU存放芯片头文件及芯片初始化文件。

04\_Linker\_File放置连接文件。

05\_Driver底层驱动文件夹。

06\_App\_Component存放应用构件。

07\_Soft\_Component抽象软件构件文件夹。

08\_Source源程序文件夹。

1. 应用程序源代码文件

Includes.h是main.c使用的头文件，含常量、全局变量声明、外部函数及外部变量的引用。

主程序文件main.c是应用程序的启动后总入口，main函数即在该文件中实现。

中断服务例程文件isr.c是中断处理函数编程的地方。

# 串行通信模块

## 6.1.1串行通信的基本概念

串行通信接口，简称“串口”、UART、SCI。主要掌握异步串行通信的格式与波特率。

1. 异步串行通信的格式



这种格式空闲状态为1，发送器通过发送一个0表示一个字节传输的开始，随后是

数据位。最后发送器发送1位或2位的停止位，表示一个字节传送结束。

1. 串行通信的波特率

每秒内传送的位数叫做波特率。

1. 奇偶校验

知道一个字节的传输是否正确。

1. 串行通信传输方式术语

全双工：数据传送方向是双向的，且可以同时接受与发送数据。

半双工：数据传送也是双向的，只能由一方发送数据，另一方接收数据，不能同时。

单工：数据传送是单向的，一端为发送端，一端为接收端。

## 6.1.2RS232总线标准

MCU引脚输入/输出一般使用TTL电平，即晶体管-晶体管逻辑电平。

RxD：接收数据线

TxD：发送数据线

SG：信号地

## 6.1.3TTL电平到RS232电平转换电路

1. USB转换线、TTL-USB转换线，在PC上安装相应驱动软件，就可以再PC上使用一般的串行通信编程方式，通过USB接口实现与MCU串行通信。

## 6.1.4串行通信编程模式

串行通信接口UART的主要功能是：接收时，把外部的单线输入的数据变成一个字节的并行数据送入MCU内部，发送时，把需要发送的一个字节的并行数据转换为单线输出。

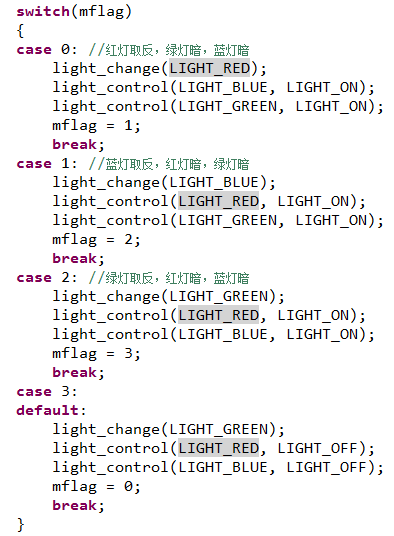
具有波特率寄存器，控制寄存器，状态寄存器。

## 6.2.1UART驱动构件基本要素分析与头文件

UART驱动构件由头文件uart.h及源代码文件uart.c组成，放入uart文件夹，供应用程序开发调用。

UART具有初始化、发送和接收三种基本操作。

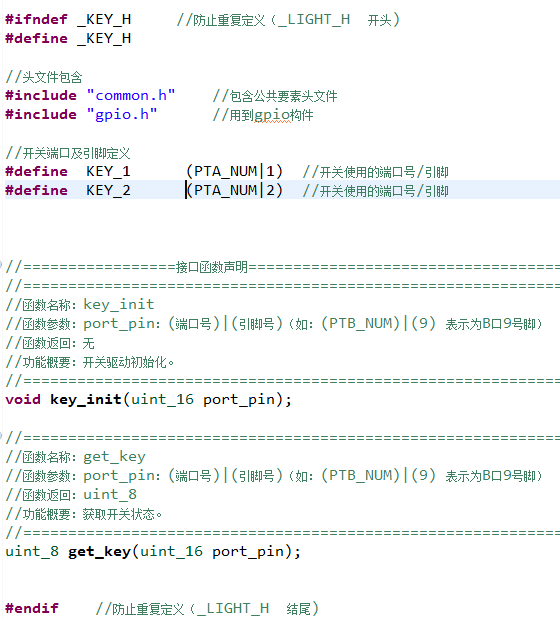
# 实验、流水灯花样



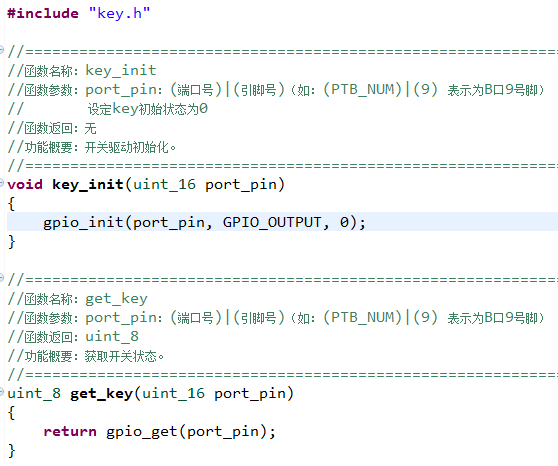
通过mflag的自增，来变换小灯的颜色。

# 实验、KEY构件的增加（两根线控制3种流水灯花样及一种不亮）

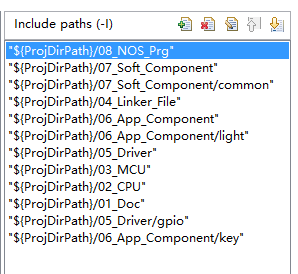
key.h开关使用A1、A2引脚，按键初始化函数声明和获取按键状态函数声明



key.c开关初始化函数和获取按键状态函数定义，默认按键状态是0



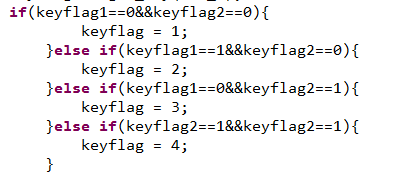
将key构件增加到路径中



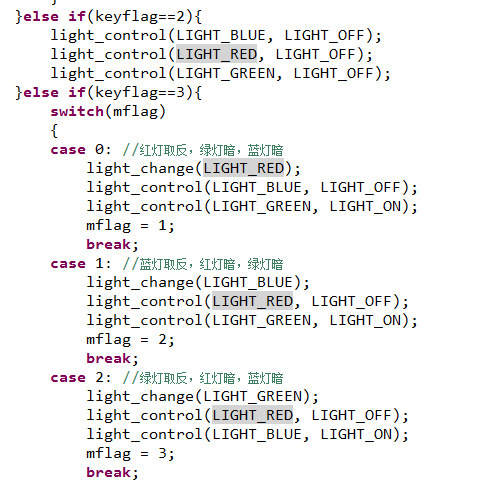
两个按键在main中初始化



A1、A2组合的四种情况，设置四种keyflag状态



keyflag不同值流水灯的不同花样



## 6.3.1中断的基础知识

中断与异常的基本含义：

异常是CPU强行从正常的程序运行切换到由某些内部或外部条件所要求的处理任务上去，这些任务的紧急程度优先于CPU正在运行的人物。而把来自CPU外围设备的强行任务切换请求称为中断。

中断、中断向量表、中断向量号：

可以引起CPU产生中断的外部器件被称为中断源。

一个CPU通常可以识别多个中断源，每个中断源产生中断后，分别要运行相应的中断服务例程，这些中断服务例程的起始地址叫做中断向量地址，放在一段连续的存储区域内，这个存储区被称为中断向量表。

给CPU能够识别的每个中断源编个号，就叫中断向量号。

中断服务例程：

中断时打断当前正在运行的程序，而转去运行的一个中断处理程序，通常被称为中断服务例程。

中断优先级、可屏蔽中断和不可屏蔽中断：

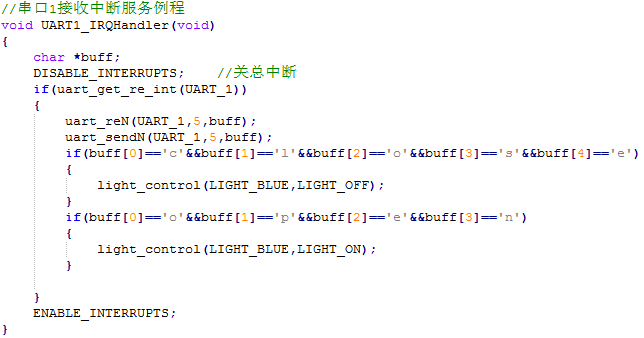
可屏蔽中断是指可通过程序设置的方式决定不响应该中断，即该中断被屏蔽了。不可屏蔽中断反之。

## 6.3.1中断处理的基本过程

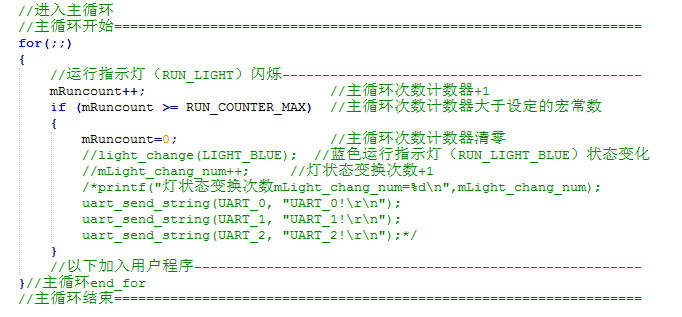
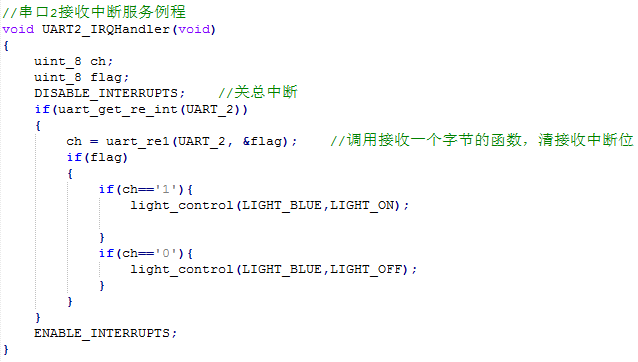
1. 中断请求
2. 中断采样（检测）
3. 中断响应与中断处理

# 实验、接收1蓝灯亮，接收0蓝灯灭；接收open让蓝灯亮，收到close蓝灯灭。

串口1用指针buff接收串口调试工具发送的字符，判断是close控制蓝灯灭，判断是open控制蓝灯亮。

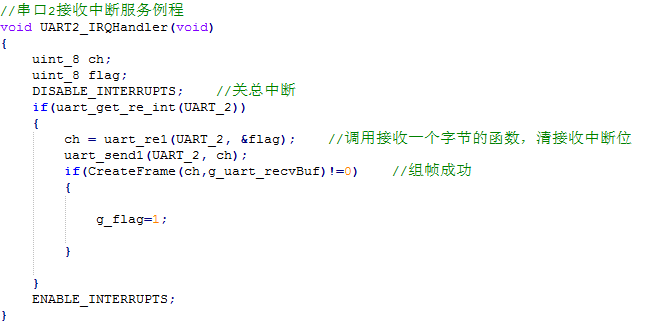


串口2用一个字节变量接收串口调试工具发送的字节数据，判断是1蓝灯亮，判断是0蓝灯灭。



# 实验、组帧实验

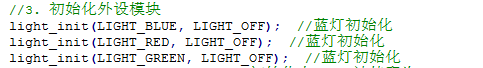
串口2调用CreateFram判断是否组帧成功，成功返回标志为1。



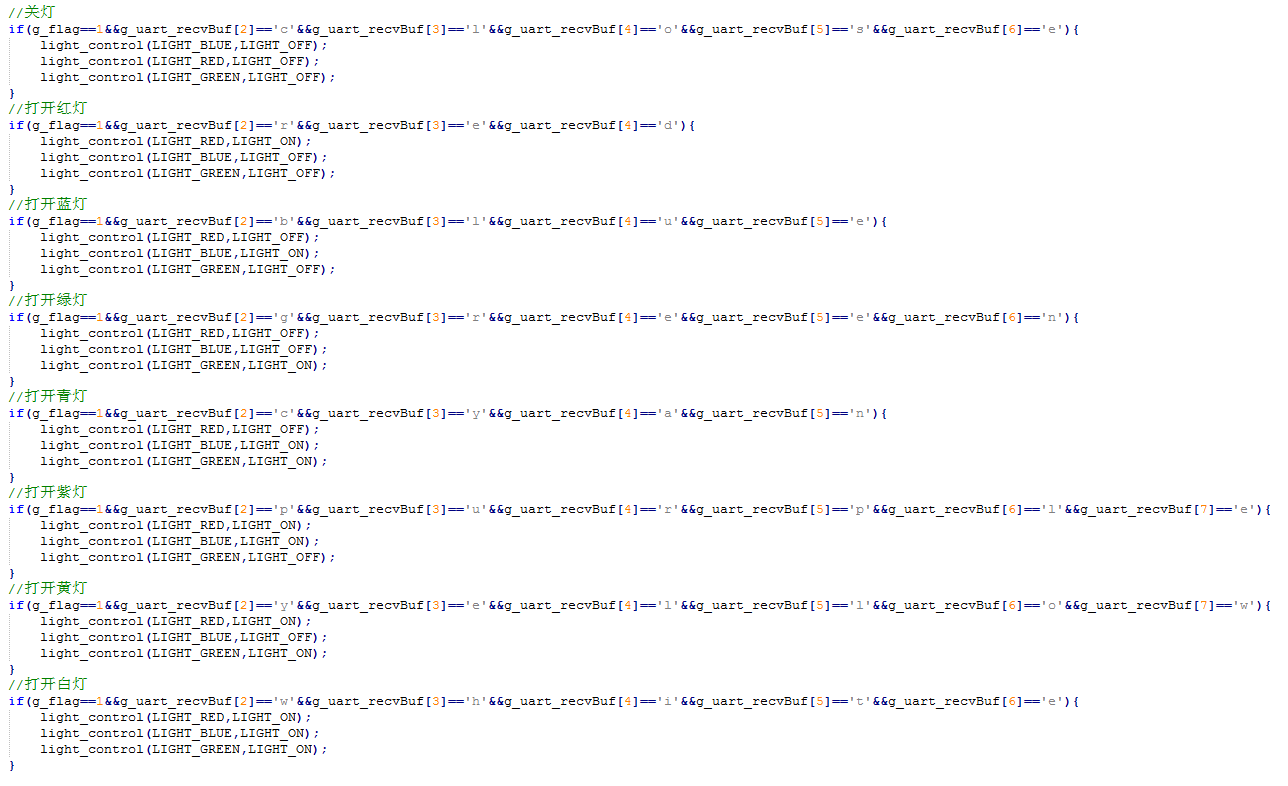
组帧函数方法的具体定义。



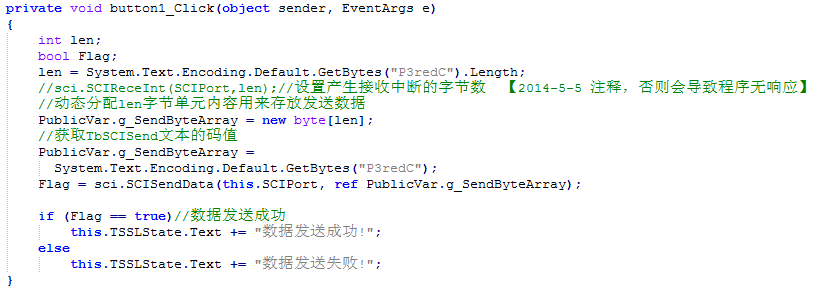
将BLUE、RED、GREEN三种灯初始化。



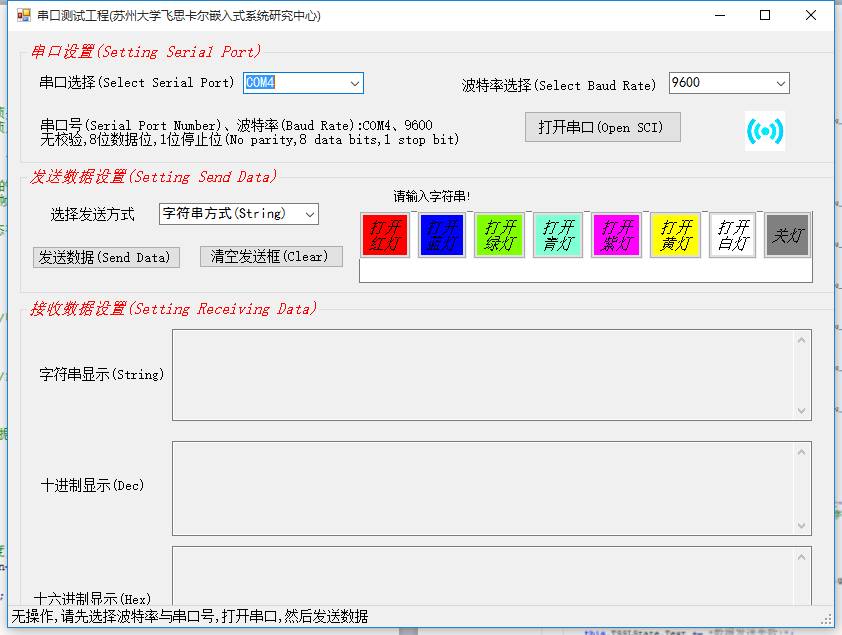
用于处理串口测试程序中书写的8个按钮方法，分别是关灯，打开红灯、蓝灯、绿灯、青灯、紫灯、黄灯、白灯。



以第一个按钮为例，参考原程序中发送数据按钮的代码，将发送的文本数据变为对应等色的组帧字符，例如红灯：P3redC



8个按钮的实现，对应上文的8个条件判断。



# 第七章、定时器相关模块

在嵌入式系统应用中，要求能对外部脉冲信号或开关信号进行计数，这通过计数器完成，实现计数与定时的基本方法有三种：完全硬件方式、完全软件方式、可编程计数器/定时器。

## 7.2.1脉宽调制PWM通用基础知识

1. PWM的基本概念和技术指标

脉宽调制是电机控制的重要方式之一，PWM信号是一个高低电平重复交替输出信号，通常也叫脉宽调制波或PWM波。

1. PWM周期

PWM信号的周期用其持续的时钟周期个数来度量。

1. PWM占空比

PWM占空比被定义为PWM信号处于有效电平的时钟周期数与整个PWM周期内的时钟周期数之比，用百分比表征。

1. PWM极性

PWM极性决定了PWM波的有效电平，正极性表示PWM有效电平为高。

1. 脉冲宽度

脉冲宽度是指一个PWM周期内，PWM波处于高电平的时间。

1. PWM分辨率

PWM分辨率△T是指脉冲宽度的最小时间增量。

1. PWM的对齐方式

PWM有边沿对齐与中心对齐两种对齐方式。

## 7.2.2输入捕捉与输出捕捉

输入捕捉：用来监测外部开关量输入信号变化的时刻。

输出捕捉：用程序的方法在规定的较精确时刻输出需要的电平，实现对外部电路控制。

## 7.3.2TPM构件

TPM模块通常用作输入捕捉、输出比较或PWM输出三种基本功能。

TPM模块计数器位数为16位，计数范围为0~65535。

## 7.4.1TPM模块的结构

每个TPM模块均有状态和控制寄存器，计数器，模数寄存器。另外，还有通道状态和控制寄存器，通道值寄存器，捕捉和比较状态寄存器及配置寄存器。

# GPIO应用——键盘、LED及LCD

## 8.1.1键盘模型及接口

键盘的基本电路为接触开关，通断两种状态分别用0和1表示。

键盘与MCU的连接方式主要有独立方式和矩阵方式。

## 8.1.2键盘编程基本问题、扫描编程原理及键值计算

1. 键盘编程基本问题
2. 键的识别
3. 抖动问题
4. 重键问题
5. 行扫描法识别按键的基本原理

行扫描法识别按键的基本原理就是，把列线n1~n4接到MCU的输入引脚，行线m1~m4接到MCU的输出引脚，则在MCU控制下，使行线m1为低电平0，其余三根行线m2、m3、m4都为高电平1，并读列线n1~n4状态。如果n1~n4都为高电平，则m1这一行上没有键闭合，如果读出列线n1~n4的状态不全为高电平，那么为低电平的列线和m1相交的键处于闭合状态，如果m1这一行上没有键闭合，接着使行m2为低电平，以此类推。

1. 键值计算

键值是MCU获取硬件连接方式下每个按键的具有唯一性的数字表达。

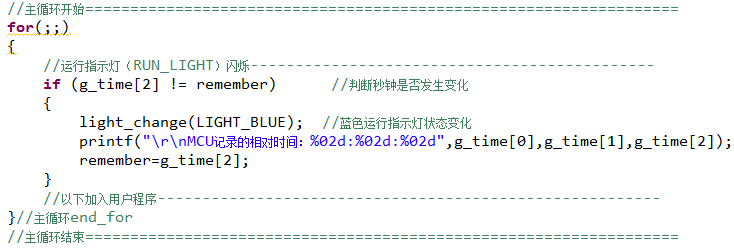
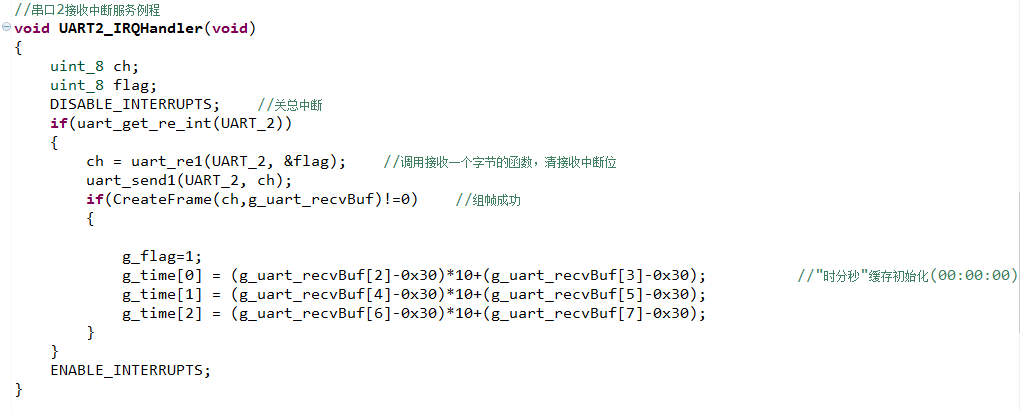
以扫描方式获得键盘的输入值。

## 8.2.1LED数码管基础知识

一个发光二极管形成一段，若要显示数字0~9只需要七段数码管。分为共阴极和共阳极。

# 定时器实验

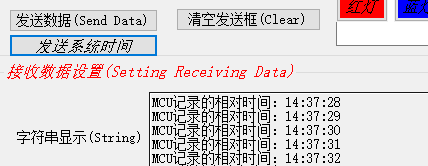
在UART2中设置计时器开始时间从用户发送的时间开始计时。



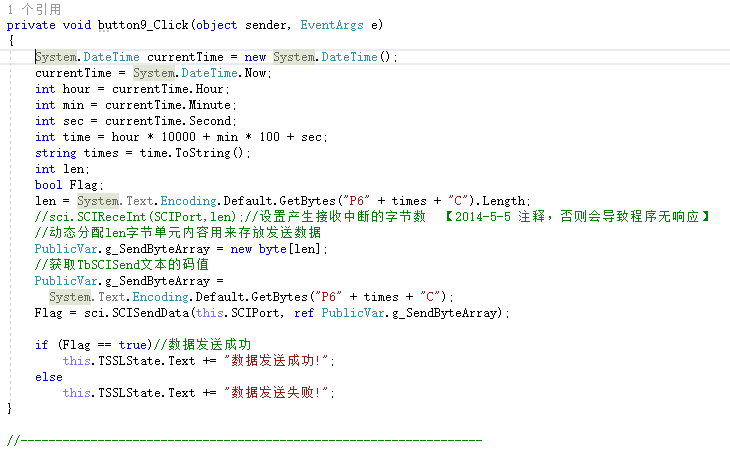
设定时间从14:36:00开始计时。



在C#中通过程序发送当前系统时间给MCU，让MCU从当前时间开始计时。



C#中代码设计。



## 8.3.1LCD的特点和分类

LCD作为电子信息产品的主要显示器件，有其自身的特点：

1. 低电压微功耗
2. 平板型结构
3. 使用寿命长
4. 被动显示
5. 显示信息量大且易于色彩化
6. 无电磁辐射

液晶显示器件分类方法有多种：

1. 按电光效应分类：LCD可分为电场效应类，电流效应类，电热写入效应类和 热效应类。
2. 按显示内容分类：LCD可分为字段型，点阵字符型，点阵图形型。
3. 按LCD的采光方式分类：分为带背光源和不带背光源两大类。

## 8.3.2点阵字符型LCD

点阵字符型LCD专门用于显示数字、字母、图形符号及少量自定义符号。

1. HD44780的引脚信号
2. HD44780的时序信号
3. HD44780的编程结构
4. HD44780的指令集

# Flash在线编程

## 9.1.1Flash在线编程的基本概念

Flash存储器具有固有不易失性，电可擦除，可在线编程，存储密度高，功耗低和成本较低等特点。成为MCU的重要组成部分。

通过编程器将程序写入Flash存储器中的模式被称为写入器编程模式。由于Flash存储器具有电可擦除功能，在程序运行过程中，有可能对Flash存储区的数据或程序进行更新，叫作用户模式或在线编程模式。

擦除操作的含义是将存储单元的内容由二进制的0变成1，而写入操作的含义是将存储单元的某些位由1变成0。

扇区表示在线擦除的最小度量单位。

## 9.1.2KL25/26芯片Flash构件头文件及使用方法

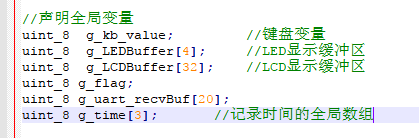
1. 首先要进行初始化Flash模块
2. 在写入之前要根据情况是否先执行擦除操作
3. 通过封装好的入口参数进行传参，进行写入操作
4. 按照逻辑地址读取时，定义足够长度数组
5. 按照物理地址读取时，定义足够长度数组
6. Flash保护函数使用

## 9.2.1Flash保护含义及保护函数

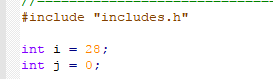
## 9.2.2Flash加密方法与去除密码方法

# 实验、工具箱实现LCD、LED相应实验

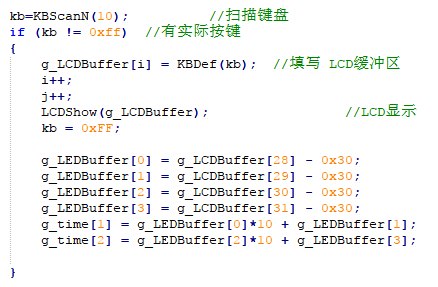
includes.h声明全局变量



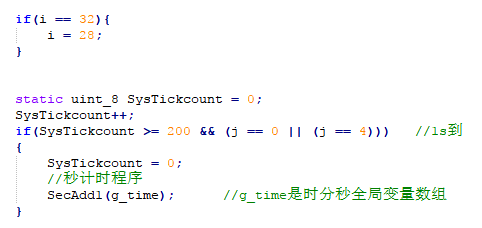
isr.c中设置变量i和j用于对循环进行操作



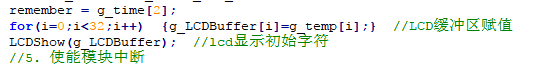
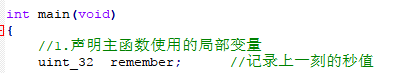
isr.c中有实际按键时，变量i和j进行加一操作，i用于键盘可以循环输入最后四位的值，j用于计时的开始，当键盘在输入时，计时器停止计时。



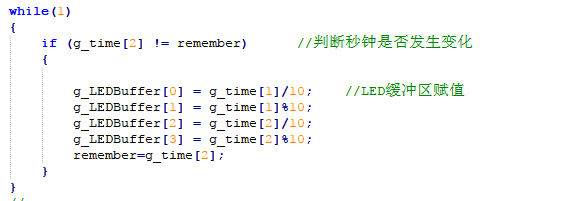
isr.c中j==4即键盘输入完成，计时器开始工作



main函数中声明变量用于对秒值进行记录



main中循环给LED缓冲区赋值



# ADC，DAC与CMP模块

## 10.1.1模/数转换器ADC的通用基础知识

AD转换模块即模/数转换模块，功能是将电压信号转换为相应的数字信号。实际应用中，这个电压信号可能由温度、湿度、压力等实际物理量经过传感器和相应的变换电路转化而来。

与AD转换编程直接相关的基本问题：

1. 转换精度

转换精度就是指数字量变化一个最小量时模拟信号的变化量，也称为分辨率，通常用AD转换器的位数来表征。

1. 转换速度

转换速度通常用完成一次AD转换所要花费的时间来表征。

1. 单端输入与查分输入

单端输入只有一个输入引脚，使用公共地GND作为参考电平。

查分输入比单端输入多了一个引脚，AD采样值是两个引脚的电平差值，有点是降低了干扰，缺点是多用了一个引脚。

1. AD参考电压

AD转换需要一个参考电平。

1. 滤波问题

为了使采样数据更准确，必须对采样的数据进行筛选去掉误差较大的毛刺。

1. 物理量回归

在实际应用中，得到稳定的AD采样值之后，还需要把AD采样值与实际物理量对应起来，这一步称为物理量回归。

## 10.1.2ADC驱动构件及使用方法

## 10.2.1数/模转换器DAC的通用基础知识

当MCU需要把处理后的信息反馈到控制设备上时，就需要把数字量转换成模拟量，完成这种转换的电路称为数/模转换器，DA转换器的工作就是将输入的二进制数字量转换成模拟量，以电压或电流的形式输出。

DA转换器的主要技术指标：

1. 分辨率

分辨率用于表征DA转换器对输入微小量变化的敏感程度。

1. 转换精度

DA转换器的转换精度是指输出模拟电压的实际值与理想值之差。

1. 转换速度

转换速度指从输入的数字量发生突变开始，到输出电压进入与稳定值±0.5最低有效位范围内所需要的时间，称为建立时间。

## 10.2.2DAC驱动构件及使用方法

## 10.3.1比较器CMP的通用基础知识

1. 电压比较器的作用

比较器模块可以比较两路模拟电压。

1. 比较器的分类
2. 模拟比较器
3. 数字比较器

## 10.3.2CMP驱动构件及使用方法

# SPI,I2C与TSI模块

## 11.1.1串行外设接口SPI的通用基础知识

串行外设接口SPI是一种同步串行通信接口，用于微处理器和外围扩展芯片指尖的串行连接，发展成为一种工业标准。

1. 主机与从机概念

一个SPI系统，由一个主机和一个或多个从机构成，主机启动一个与从机的同步通信，从而完成数据交换。

1. 主出从入引脚MOSI与主入从出引脚MISP

主出从入引脚是主机输出，从机出入数据线。

主入从出引脚是主机输入，从机输出数据线。

1. SPI串行时钟引脚

SPI串行时钟引脚用于控制主机与从机之间的数据传输。

1. 时钟极性与时钟相位

时钟极性表示时钟信号在空闲时是高电平还是低电平。

1. 从机选择引脚SS

一些芯片带有从机选择引脚SS，也称为片选引脚。

传输原理：

从主机CPU发出启动传输信号开始，将要传送的数据装入8位移位寄存器，并同时产生8个时钟信号，以此从SCK引脚送出，在SCK信号的控制下，主机中8位移位寄存器中的数据依次从MOSI引脚送出，到从机的MOSI引脚后送入它的8位移位寄存器。

SPI的时序：

1. 空闲电平低电平，上升沿取数
2. 空闲电平低电平，下降沿取数
3. 空闲电平高电平，下降沿取数
4. 空闲电平高电平，上升沿取数

## 11.1.2 SPI驱动构件头文件及使用方法

## 11.2.1集成电路互连总线I2C的通用基础知识

（1）I2C可以翻译为集成电路互连总线，主要用于同意电路板内各集成电路模块之间的连接，I2C采用双向2线制串行数据传输方式。

（2）主机：在I2C总线中，提供时钟信号，对总线时序进行控制的器件。

（3）从机：在I2C系统中，除主机外的其他设备均为从机。

（4）地址：每个I2C器件都有自己的地址，以供自身在从机模式下使用。

（5）发送器与接收器：发送数据到总线的器件被称为发送器，从总线接收数据的器件被称为接收器。

（6）SDA与SCL：串行数据线，串行时钟线。

（7）I2C总线上的信号类型：开始信号 停止信号 重新开始信号 应答信号

（8）一个标准的I2C通信由4部分组成：开始信号 从机地址传输 数据传输 结束信号

（9）在I2C总线系统中，寻址字节由被控器的7位地址位（D7-D1）和一位方向位（D0）组成。方向位为0时，表示主控器将数据写入被控器，为1时表示主控器从被控器读取数据。主控器发送起始信号后，立即发送寻址字节，这是总线上所有器件都将寻址字节中的7位地址与自己的器件地址比较。如果两者相同，则该器件认为被主控器寻址，并发送应答信号，被控器根据数据方向位（R/W）确定自身是作为发送器还是接收器。

（10）I2C系统中，没有两个从机的地址是相同的。

## 11.2.2 I2C驱动构件头文件及使用方法

## 11.3.1触摸感应接口TSI的通用基础知识

# 总结：

最近做期末实验感觉有一个好的分类条理很重要，文件要归类好，不然最后乱在一起会很头大，还有讲义中有的实验需要工具箱，而且还是很多题，所以实验箱希望老师能多发放下来使用，有可能的话能不能一个宿舍带一个回去使用？最后就是加油做实验了！

# USB编程

USB特点如下：

1. 支持即插即用
2. 可以使用总线电源
3. 硬件接口标准化、小巧化
4. 支持多种速度和操作模式

## 12.1.1USB的物理特性

1. USB电缆和接口类型
2. USB差分信号
3. USB总线上的状态与设备速度检测

## 12.1.2USB主机与设备的概念与特性

1. USB主机

USB主机指的是包含USB主控制器，并且能够控制完成与USB设备之间数据传输的设备。

1. 客户软件
2. USB系统软件
3. USB总线接口
4. USB设备（从机）

USB协议中将USB设备定义为具有某种功能的逻辑或者物理实体。

# 系统时钟与其他功能模块

## 13.4看门狗

（1）功能描述

看门狗定时器具有监视系统功能，当运行程序跑飞或者一个系统中的关键系统时钟停止引起严重后果的情形下，无法回到正常的程序上执行，看门狗通过复位系统的方式，将系统带到一个安全操作的状态。正常情况下，看门狗通过与软件的定期通信来监视系统的执行过程，将看门狗定时器清零，即定期喂看门狗。如果应用程序丢失，未能在看门狗计数器超时前清零，则将产生看门狗复位，强制将系统恢复到一个已知的起点。

（2）配置WDOG

（3）其他说明

## 冷复位与热复位的区别

# 总结：

最近一段时间的嵌入式复习，理论方面还需要多理解理解，实验还有些没有做好，正在继续往下做，不会的问同学也能得到解决，自己理解写出来的东西才能印象深刻，新的一年，新的起点，把这个学期的任务做好，完美的度过大三上学期，还有祝老师新年快乐，万事如意。