



到这里呢，学习 51 单片机的准备我们就做完了，也就是说我们可以进入正式学习阶段了，说到学习 51 单片机，考虑到大家多为新手，这里强推大家购买开发板进行学习，推荐购买价位在 100~150 之间的，外设相对较多，大家可以练习的程序也就相对较多，也会学到一些科技作品常用的硬件。下面就开始给大家讲 51 单片机的相关知识了。

```
#include "reg52.h" //此文件中定义了单片机的一些特殊功能寄存器
#include<intrins.h> //因为要用到左右移函数，所以加入这个头文件

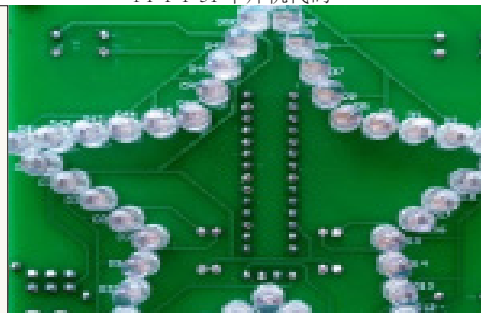
typedef unsigned int u16; //对数据类型进行声明定义
typedef unsigned char u8;

#define led P0 //将P0口定义为led 后面就可以使用led代替P0口

/*
 * 函数名      : delay
 * 函数功能    : 延时函数, i=1时, 大约延时10us
 */
void delay(u16 i)
{
    while(i--);
}

void main()
{
    u8 i;
    led=0x01;
    delay(50000); //大约延时450ms
    while(1)
    {
        for(i=0;i<7;i++) //将led左移一位
        {
            led=_crol_(led,1);
            delay(50000); //大约延时450ms
        }
        for(i=0;i<7;i++) //将led右移一位
        {
            led=_cror_(led,1);
            delay(50000); //大约延时450ms
        }
    }
}
```

P1-1-1 51 单片机代码



P-1-1-3 五角星 LED 灯

可能在学习了后面的知识之后，会产生这样一个疑问，为什么要先学习“led 小灯”呢？我觉得出于两个方面。一是 led 小灯本身相较其他的外设而言，属于简单又泛用的一类，对于初学者而言容易接受。第二就是，在以后学习应用其他外设时，led 小灯可以成为检验外设是否运行成功的一种指标。比如，使用红外感应模块时，读取到温度信号，小灯点亮，反之则不亮之类的。总而言之，led 小灯基础且易学，有兴趣者可以在此基础上编写更多有趣的实验。在此我们就不一一赘述了。

另外，单片机可以实现的功能绝不仅限于此。随着型号的升级和内存的扩大，单片机几乎可以实现任何你想要的功能。比如在智能化家用电器方面，应用单片机替换传统的电子线路，洗衣机，空调等家用电器中都应用了单片机技术。还有我们众所周知的手机，也曾采用过’ /. 3an。

属于专用型单片机；更甚的是，航空航天与国防军事，尖端武器方面，同样也采用了单片机技术，不过至于是如何使用的，我们肯定不知道就是了（摊手.jpg）。





科技入门一本通之 430 系列单片机

写这篇 430 的介绍的前提是大家已经对 51 有了比较深入的了解，于很多单片机通用的知识这里就不在加以介绍了，而且在已经接触过 51 单片机之后，430 就更需要大家的自主学习了，所以对于 430，这里的讲解就不会像 51 单片机一般详细，而是更注重实例的列举，同时讲一下 430 和 51 单片机的一些差距，方便大家比较学习，然后给大家分享几个写 430 常用的库函数。下面我们就进入正题。

一、430 系列单片机简介

mSP430 系列单片机是 TI 公司生产的单片机，如果要参加 TI 杯（也就是小电赛），mSP430 是必须要会用，而且对它的时钟中断这些都需要了解得比较透彻，它包含一个十六位精简指令集（RISC）CPU，单个时钟周期就可以执行一条指令，使用相同晶振，它的速度是 51 单片机的 12 倍，其另一大特点就是它的超低功耗，一个水果电源就可以驱动它工作了。mSP430 系列单片机有很多，这里给大家介绍的是 G2553 系列单片机。一般都会选用 LaunchPad 开发板进行学习，编写 430 系列单片机的软件有两个，IAR 和 DDS，这里给大家介绍的是 IAR 软件，关于 iar 软件的下载，我这里没有安装包，大家去 TI 公司官网上可以找到安装包，百度上有详细的安装教程，或者大家可以关注软件安装管家公众号，在这里可以找到很多软件资源和安装教程，下面给大家分享一个 PPT，主

要是介绍这款 LaunchPad 以及 iar 软件的使用，还有少许 C 语言重点知识。

链接：<https://wk.baidu.com/view/2256e748ba1aa8114431d9cd?pcf=2>

<https://wk.baidu.com/view/20b4f57a524de518974b7d02?pcf=2>

相信看完再经过一番实战，大家对于这款开发板已经有了一定的了解，关于 iar 软件的使用也没有问题了，虽然这个过程可能花了大家一定的时间，但是请记住，万事开头难嘛，其实学习单片机之前的软件安装和各种配置就是相对很复杂的一环了，所以后续一定要有信心，关于建立工程什么的这里就不在说了，mSP430 单片机相对 51 单片机而言内部资源会多不少，这里不可能给大家一一讲解，或者说我有心一一讲解相信在没有实战的前提下大家也没有耐心看完，所以 430 单片机推荐学法就是用什么学什么，网上有丰富的教程和经验之谈，文末也会给大家分享 mSP430 的中文资料，为了用而学比没有目的的学会高效很多，下面主要就 IO 口配置，时钟系统以及中断来讲一下 430 单片机和 51 单片机的区别所在，讲完就开始给大家分享实用库函数。

二、mSP430 单片机的 IO 口

相较于 51 单片机，430 系列单片机 IO 口的配置相对麻烦，因为不能使用位操作，而要对寄存器进行配置，所以首先我们要了解 IO 口涉及的寄存器。

寄存器名	寄存器功能	读写类型	复位初始值
PXIN	PX 口输入寄存器	只读	无
PXOUT	PX 口输出寄存器	可读可写	保持不变
PXDIR	PX 口方向寄存器	可读可写	0（全部输入）
PXSEL	PX 口第二功能选择	可读可写	0（全部为 IO 口）
PXIE	PX 口中断允许	可读可写	0（不允许中断）
PXIES	PX 口中断沿选择	可读可写	保持不变
PXIFG	PX 口中断标志位	可读可写	0（未发生中断）

例：利用 LaunchPad P1.3 口连接的按键控制 P1.0 口连接的 LED，实现按键按下灯亮，按

键弹起灯灭。

```

#include<msp430.h>
int main(void)
{
    WDTCTL = WDTHOLD + WDTPW; // 关闭看门
    狗（当不用看门狗定时器时，这句话一定需要，
    不然就会一直复位）
    P1SEL &= ~(BIT3+BIT0); // 设置
    P1., P1.3 为 IO 口
    P1DIR |= BIT0; //P1.0 口为输出
    P1DIR &= ~BIT3; //P1.3 口为输入 ‘
    P1REN |= BIT3; // 使能 sw2 为上下拉
    (P1.3)
    P1OUT |= BIT3; // 使能 sw2 上拉
    While(1)
    {
    If((BIT3&P1OUT))

```

```

        P1OUT |= BIT0; // 如果按键按下
        为高电平即灯亮
    else
        P1OUT &= ~BIT0; // 否则灯灭
    }
}

```

三、msp430 基本时钟系统

(1) msp430 基本时钟模块 + 简介：

基本时钟模块 + 支持低系统成本和超低功耗。采用三种内部时钟信号，用户可以选择性能和低功耗的最佳平衡。为了实现无任何外部元件操作，可在全软件控制下，用一个外部电阻、一个或两个外部晶振、或用振荡器来配置基本时钟模块 +。

基本时钟模块 + 有 2 个，3 个或 4 个时钟

- **LFXT1CLK**：低频 / 高频振荡器可以与低频时钟晶振或外接 32768Hz 时钟源，或与标准晶振、振荡器，外部 400KHz ~ 16MHz 的外部时钟源一起使用。

- **XT2CLK**：可以与标准晶振、振荡器，或外部 400KHz ~ 16MHz 的外部时钟源一起使用的可供选择的高频振荡器。

- **DCOCLK**：内部数控振荡器 (DCO)。

- **VLOCLK**：内部超低功耗、12KHz 典型频率的低频振荡器。

(2) msp430 的基本时钟模块 + 可提供的三种时钟信号：

- **ACLK**：辅助时钟。ACLK 是由软件选择来作为 LFXT1CLK 或 VLOCLK。ACLK 经 1, 2, 4, 8 分频后得到。ACLK 可由软件选作各个外围模

块。

- **MCLK**：主机时钟。MCLK 由软件选择作 LFXT1CLK, VLOCLK, XT2CLK (如果片上提供)，或 DCOCLK。MCLK 由 1, 2, 4, 8 分频得到。MCLK 用于 CPU 和系统。

- **SMCLK**：系统子时钟。SMCLK 由软件选择作 LFXT1CLK, VLOCLK, XT2CLK (如果片上提供)，或 DCOCLK。SMCLK 由 1, 2, 4, 8 分频得到。SMCLK 可由软件选作各个外围模块。

(3) 三种时钟信号的区别

- 辅助时钟 ACLK 主要用于低速外设的，可以选作外围模块的时钟信号。

- 主时钟 MCLK 用于 CPU 和系统。

- 系统子时钟 SMCLK 用于高速外围模块。

