

工程师FAQ

NOVOSENSE
纳芯微电子

NSUC16x0 开发工程师FAQ

作者：MCU AE TEAM

NovoGenius®

上手篇

硬件篇

软件篇



1.1硬件开发板

1.2软件开发环境

1.3调试器

1.4LIN盒子

1.5电机



1.1.1 NSUC16x0的Demo板有哪些特性？



牛天才

回复

NSUC16x0的Demo板是为了帮助客户更快的评估该芯片而设计的。目前硬件已经到V1.3版本。该Demo板主要包括下面的一些功能：

- 1.支持各种不同的电机形式，包括直流有刷电机、三相直流无刷电机以及步进电机控制；
- 2.每个桥臂的电流高达1.4A；
- 3.支持TEST模式切换，可以从FLASH启动或者RAM 启动；
- 4.支持SWD调试接口；
- 5.支持四线的LIN, 包括LIN_IN,LIN_OUT,GND和VSUP, 可以方便的组 网和自动寻址；
- 6.有一个电源指示灯和一个用户可控指示灯；
- 7.支持一个按键输入；
- 8.支持HSBVD高边输出控制；
- 9.各个功能管脚都引出，方便客户评估和使用；

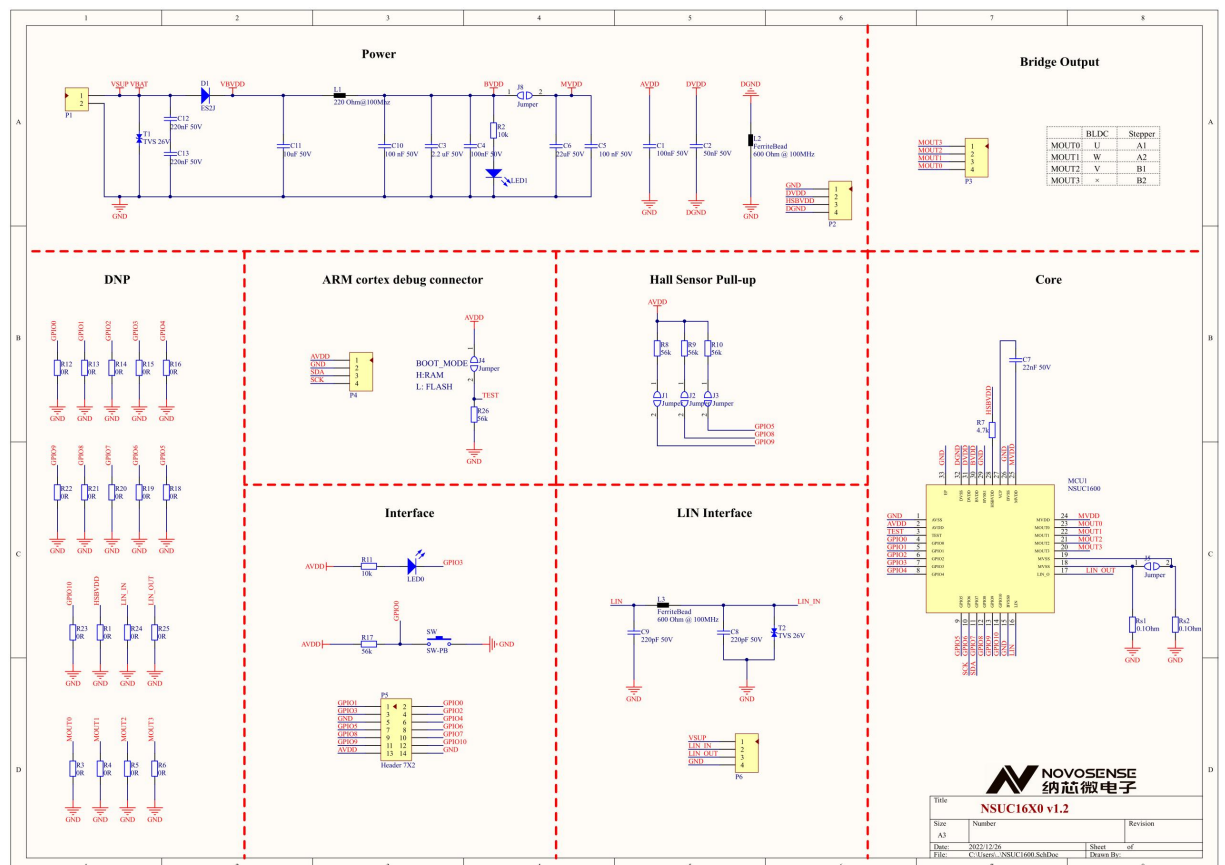


1.1.2 NSUC16x0的Demo板原理图？



牛天才

回复





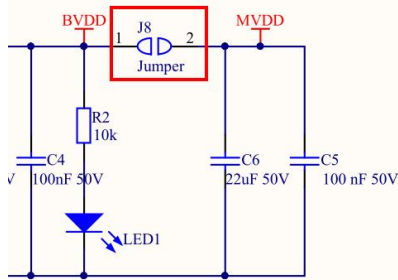
1.1.3 Demo板里的程序明明输出了，但为什么没有电流？

Demo板出厂默认没有跳线，请检查跳线J8是否闭合。如果没有闭合的话，桥臂MVDD就没有供电上。



牛天才

回复



1.1.4 为什么LIN线既有LIN又有LIN_O？在LIN Interface的接口上名字叫LIN_IN和LIN_OUT？



牛天才

回复

芯片的管脚上有LIN和LIN_O。如果应用中不需要自动寻址的功能，那么建议LIN_O悬空即可。如果需要用到自动寻址，比如空调风门或者出风口应用，比较多的相同功能的ECU连接在一起，那么就需要自动寻址的方法来进行ECU的NAD赋值。

LIN_IN和LIN_OUT指的就是连接LIN和LIN_O管脚的外部信号。这两个信号用于将多个ECU串联起来。LIN和LIN_O之间是一个1欧姆的电阻，这个电阻是芯片内部集成的，用万用表通断测量的话，LIN和LIN_O，也就是LIN_IN和LIN_OUT是联通的。

1.1硬件开发板 | 1.2软件开发环境 | 1.3调试器 | 1.4LIN盒子 | 1.5电机



1.2.1 NSUC16x0支持哪些软件IDE？



牛天才

回复

NSUC16x0属于ARM CortexM3内核。虽然ARM内核可以有很多不同的开发环境，但Keil一直是比较受广大开发者喜欢的IDE。因此NSUC16x0选择Keil作为其开发环境。用户只需要安装Keil的MDK5xx及以上的IDE，都是可以的。



1.2.2 使用Keil开发应该注意些什么？



牛天才

回复

在使用Keil的MDK5xx开发NSUC16x0产品应用的时候，需要首先安装NSUC16x0的芯片安装包。安装包文件名为Novosns.NSUC16x0.2.1.x.pack(如需最新安装包，请至纳芯微官网下载)

1.3.1 NSUC16x0开发推荐什么调试器呢？



牛天才

回复

NSUC16x0 支持SWD调试口。由于NSUC16x0的IO属于5V系统，因此建议用户使用支持5V电压的J-link。如果使用DP-Link或者ST-Link，大部分都是3.3V的IO，这样的话有较大概率会连接失败。

1.3.2 使用JLINK过程中的注意事项有哪些？



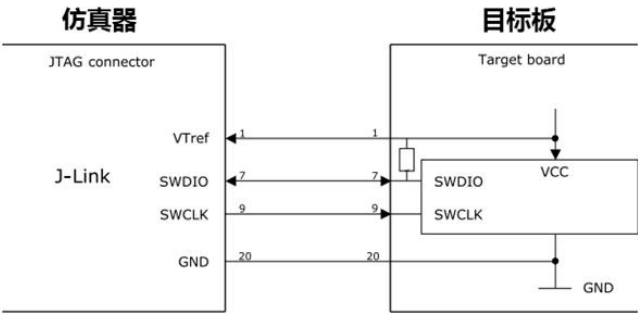
牛天才

回复

1.在购买J-link的时候，需要确保这个J-link是支持5V的，比如会有类似的字样。

V9极速1.8-5V版

2. SWD的连接，Jlink的话，需要连接4根线，VTref需要连接到目标板的AVDD，这个作用是为了检测AVDD的电压，从而确保SWD输出的电压和AVDD匹配。如下图所示：



J-link的20PIN的接头连接如下图所示，VTref - PIN1；SWDIO - PIN7；SWCLK - PIN9；GND - PIN20或者其他。

VTref	1	2	VCC
N/C	3	4	GND
N/C	5	6	GND
SWDIO	7	8	GND
SWCLK	9	10	GND
N/C	11	12	GND
SWO	13	14	GND
RESET	15	16	GND
N/C	17	18	GND
N/C	19	20	GND

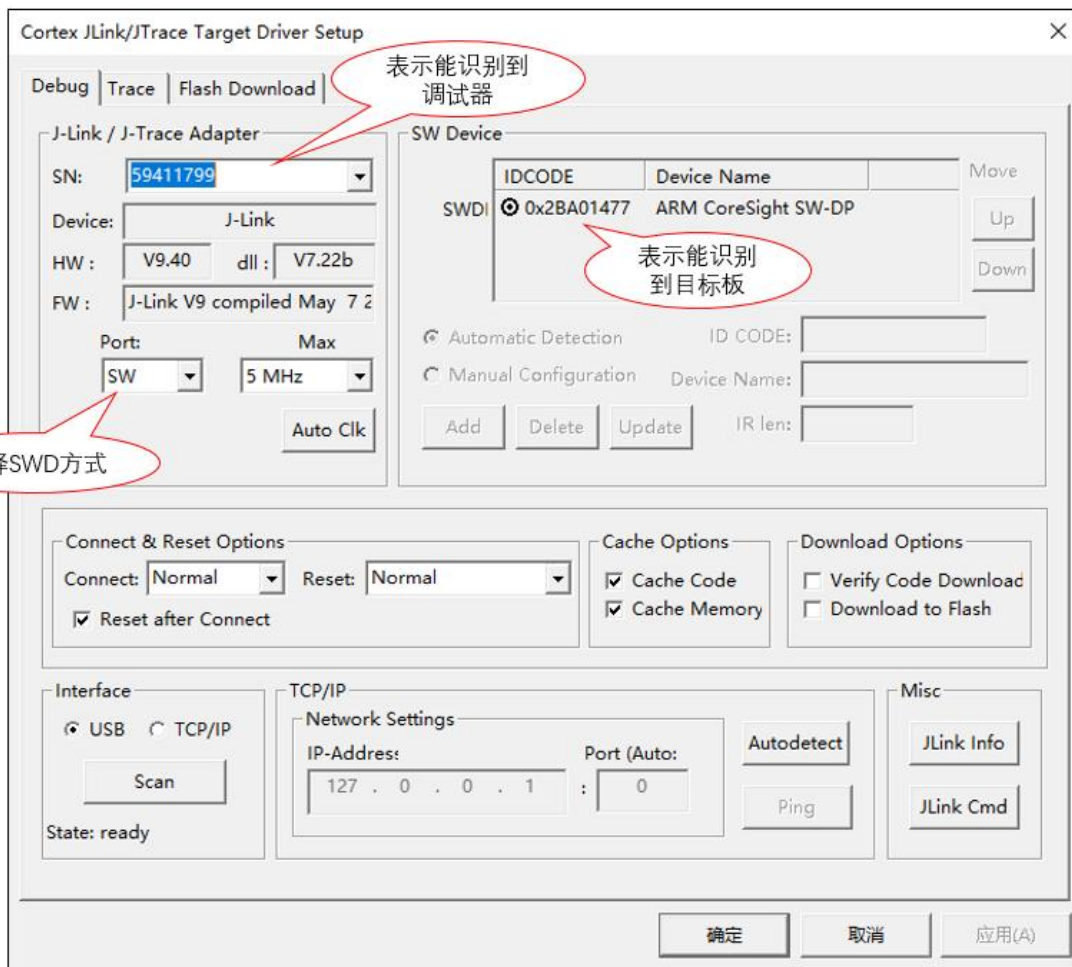
1.3.3 如果J-link连接正确的话，给目标板上电后，在Keil的魔术棒下的Debug选项卡上能读到J-link的信息和芯片的IDCODE。



牛天才

回复

如果J-link连接正确的话，给目标板上电后，在Keil的魔术棒下的Debug选项卡上能读到J-link的信息和芯片的IDCODE。



1.1硬件开发板 | 1.2软件开发环境 | 1.3调试器 | 1.4LIN盒子 | 1.5 电机

1.4.1 LIN盒子能做什么？



牛天才

回复

LIN盒子实际上就是一个LIN-USB工具，主要用于模拟LIN Master，发送LIN的指令和读取LIN Slave的数据。由于NSUC16x0主要是用于LIN通信的执行器，因此LIN盒子是必须要的。



1.4.2 LIN盒子推荐型号？



牛天才

回复

目前推荐图莫斯的LIN盒子，原因是纳芯微MCU开发的上位机也都是基于该工具。

该工具性价比比较高，其自带上位机功能也比较丰富。

USB2CAN&LIN总线适配器基础版-重庆图莫斯电子科技有限公司官网



1.1硬件开发板 | 1.2软件开发环境 | 1.3调试器 | 1.4LIN盒子 | 1.5 电机



1.5.1 DEMO配套的步进电机型号是什么？



牛天才

回复

其实没有所谓DEMO配套的步进电机，我们推荐使用12V步进电机，可以使用42BYGH33等常用的步进电机。

1.1 产品板子的TEST引脚该如何处理呢？



牛天才

回复

TEST引脚对于用户来讲，主要的作用是为了芯片boot的选择。芯片复位后会根据TEST脚是高电平还是低电平，选择不同的boot位置。如果TEST是高电平，那么芯片将从RAM开始boot，反之如果是低电平则从FLASH开始boot。建议用户将TEST引脚通过一个10k电阻接地，同时将TEST接一个测试点，这样做的目的是希望如果FLASH启动有问题的话，还可以通过RAM启动来擦除FLASH，可以避免由于程序的操作有问题而导致的板子变砖头。

1.2 HSBVDD是做什么用的，不用的话该如何处理呢？



牛天才

回复

确切的说HSBVDD是一个高边开关，相当于将HSBVDD连到BVDD的一个可控的开关电路。可以通过寄存器控制连接或者关闭，还可以有过流保护功能。比较典型的case是HSBVDD给外部的传感器供电，比如高压的霍尔开关等。如果不用的话，可以将其通过一个电阻10k接地，只要其开关是关闭的，并不会带来电流的增加。也可以将其连接到BVDD上。

1.3 为什么新做的板子连不上Debug呢？



牛天才

回复

新板子连不上Debug，主要可能有下面的一些原因：

- 1.使用的Debug工具，比如ST-link或者DAP-Link的输出是3.3V的，直接连到NSUC16x0上面的话会可能存在逻辑电平识别错误。建议使用5V电平的debug工具；
- 2.Debug的线束太长，如果Debug的杜邦线比较长可能会导致通信问题；
- 3.由于AVDD的纹波较大，也会导致Debug通信不上，建议调整AVDD的电容；
- 4.由于AVDD外部连接较大的电容，比如1uF或者更大的电容。这个大电容可能是AVDD供电的芯片旁边的电容，这样会导致AVDD异常，比如降到4.0V左右。这样会导致Debug连接异常；

1.4 芯片的AVDD可以对外供电么？



牛天才

回复

芯片的AVDD可以对外提供5V，20mA的供电能力。这个输出没有过流保护，只有AVDD的欠压保护。

1.5 芯片的低功耗电流怎样，能满足100uA的要求么？



牛天才

回复

NSUC16x0的低功耗电流sleep模式下典型值是35uA，可以满足100uA的要求。

1.6 芯片的SLEEP模式下要注意些什么？



牛天才

回复

NSUC1610的SLEEP模式下，MCU的内部LDO是关掉的，CPU、NVM、SRAM都处于PowerDown状态。这个时候如果需要保存数据，请先将数据保存在EEPROM里，保存完好之后再进入到SLEEP状态。

1.7 MVSS0和MVSS1是怎么回事，2个脚是联通的么？



牛天才

回复

按照目前的User Guide图示显示，MVSS0与MVSS1是分开的。但实际上MVSS0与MVSS1在芯片内部是联通的，且不能分开。

这点需要注意，User Guide新版本中会更改过来。

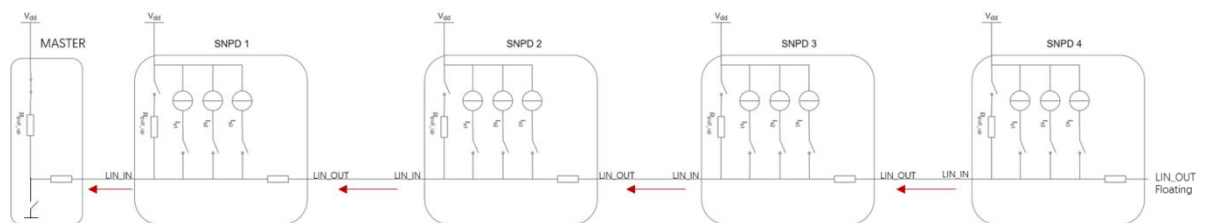
1.8 为什么LIN除了LIN_IN还有LIN_OUT，什么时候需要用到LIN_OUT？



牛天才

回复

当多个相同功能的LIN模块接入系统的时候，需要给不同的LIN Slave赋值相应的NAD。这个时候就需要4线制的LIN。



当使用4线制的LIN的时候，某个节点的LIN_OUT接下一个节点的LIN_IN，依次形成菊花链式连接。

LIN_IN和LIN_OUT之间是一个1Ω的shunt电阻，这个电阻用于采样流过LIN线的电流。具体可以参考文档 Bus Shunt Revision。

1.9 NSUC1610的峰值电流是多少，保护电流呢？



牛天才

回复

NSUC1610峰值电流大小是1.4A，有效值电流约1A。保护电流通常大于峰值电流，NSUC1610的保护电流在1.8A左右，超过该值会触发OC事件。



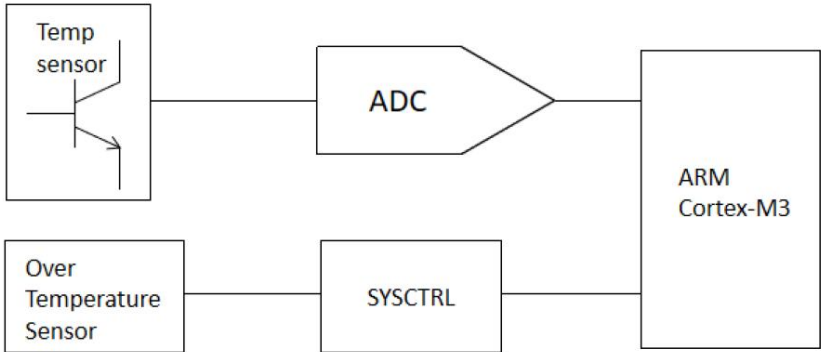
1.1 连接芯片ADC通道的温度传感器怎么计算温度？



牛天才

回复

这部分在用户手册里的第18章节有详细的描述。NSUC1610内部有2颗温度传感器，一颗靠近MOSFET，用于热关断保护；另一颗靠近LDO，且连接到芯片内部的ADC端口，可用于芯片结温实时温度检测。



温度的计算如下所示，通过ADC内部通道采集到DTV的值，通道选择“5”。

[27:24]	ADC sequence fourth conversion source selection
Q4SEL	0: Reserve 1: MVSS0 2: BVDD 3: STDA (differential)
	4: STDB (differential) 5: TEMP (differential)
	6: differential (differential) 7: LGPIO0 8: LGPIO1
	9: LGPIO2 10: LGPIO3 11: LGPIO4 12: LGPIO5
	13: LGPIO4/5 (differential) 14: AVSS 15: VREFADC

室温下，12V供电不跑电机，测试下来一个DTV值是125x左右（芯片之间会有差异）。

根据下面的公式（TST和TSA的值请采用寄存器的值，早期版本的芯片可能寄存器里没有trim的值，就需要用默认的值）

$$T[^{\circ}\text{C}] = (\text{TST} - \text{DTV}) * (\text{TSA} / 4096) [^{\circ}\text{C}]$$

可以计算得到温度值为 26 摄氏度，符合预期。推荐代码：

```
if (ADC->TPSR != 0) {
    i16Temp = (ADC->TPSR_b.TST - i16TempAdc) * ADC->TPSR_b.TSA / 4096;
} else {
    i16Temp = (0x556 - i16TempAdc) * 0x3DA / 4096;
}
```



1.2 过压保护与欠压保护如何设置？



牛天才

回复

过压保护与欠压保护是硬件自动实现的，其中过压保护点并不能设置，而是芯片固定的。欠压保护点有4个档位可选。

BVDD undervoltage threshold	VBVDUn	5.5	6		V	threshold=00
		6.5	7		V	threshold=01
		7.5	8		V	threshold=10
		8.5	9		V	threshold=11
BVDD under voltage hysteresis	VBVDHys	0.3		1	V	1)
BVDD overvoltage threshold	VBVDop	18.2	19	19.7	V	
BVDD overvoltage threshold	VBVDOn	17.2	18	18.5	V	

初始化的代码如下所示：

```
int AMS_BVDDInit(void)
{
    /* Set Interrupt PMU priority */
    LL_NVIC_EnableIRQ(PMU_IRQn);
    LL_NVIC_SetPriority(PMU_IRQn, 1, 1);

    SYSCTRL->PWRCR_b.BOF = 1;      //set the OV filter time = 0.5us
    SYSCTRL->PWRCR_b.BUF = 1;      //set the UV filter time = 0.5us

    SYSCTRL->PWRCR_b.BOIEN = 1;
    SYSCTRL->PWRCR_b.BUIEN = 1;
    /* Enable BVDD OverVoltage switch-off driver */
    SYSCTRL->PWRCR_b.BOBEN = 0;
    /* Enable BVDD UnderVoltage switch-off driver */
    SYSCTRL->PWRCR_b.BUBEN = 0;

    SYSCTRL->PWRCR_b.UVVTH = 3;    // BVDD UV point = 9.5V (3), 8.5V (2); 7.5V (1); 6.5V (0);
    return 0x00;
}
```

PMU的代码如下所示：

```
void PMU_IRQHandler(void)
{
    uint8_t statusov = SYSCTRL->PWRCR_b.BOVST;
    uint8_t statusuv = SYSCTRL->PWRCR_b.BUVST;

    if(statusov)
    {
        faultstate.overr = 1;      //LED slow blink 1Hz
        delay_ms(500);
        GPIO->PDO_b.DO3 = 1;
        delay_ms(500);
        GPIO->PDO_b.DO3 = 0;
    }
    if(statusuv)
    {
        faultstate.uverr = 1;      //LED fast blink 5Hz
        delay_ms(100);
        GPIO->PDO_b.DO3 = 1;
        delay_ms(100);
        GPIO->PDO_b.DO3 = 0;
    }
}
```

这个BOVST和BUVST并不需要软件去清，一旦外部条件变化，会自动置位或者清0。请注意过压和欠压刹车功能不建议使用。



1.3 如何进入SLEEP模式呢？怎么唤醒呢？



牛天才

回复

进入SLEEP模式，其实只需要运行一条语句即可。

```
SYSCTRL->LPCR_b.SLEEP = 1;
```

但SLEEP后，没有办法唤醒，只能通过LIN唤醒。

通过LIN唤醒的话需要配置LIN Wakeup以及LIN break的Debounce设置。

```
SYSCTRL->SYSWKR_b.LINWKE = 1;           //enable LIN wake up;
SYSCTRL->SYSWKR_b.LINDEN = 1;           //enable LIN debounce enable;
SYSCTRL->SYSWKR_b.LINWDBC = 0;         //LIN debounce time setting = 30us;
```



1.4 SLEEP唤醒后如何读取状态标志呢？



牛天才

回复

SLEEP唤醒后，可以通过读取SYSCTRL_APMUSTR寄存器来获取相应的信息。

6.1.2.9 SYSCTRL APMU status register (SYSCTRL_APMUSTR)

- Name: SYSCTRL APMU Status Register
- Reset value: 0x0
- Address offset: 0x2C

bit	31:6	5:0
name		STAR
access		R/W
Reset value		0x0

bits	Instructions
[beat]	Bit[0]:OTD_Flag indicates the overtemperature flag. 1 indicates overtemperature and 0 indicates normal
STAR	Bit[1]:Sleep_Flag, where 1 indicates sleep and 0 indicates no sleep
	Bit[2]:Retention_Mode_Flag indicates the retention mode flag bit. 1 indicates that the retention mode is entered, and 0 indicates that the mode is not entered
	Bit[3]:DVDD_UV_Flag, indicating the undervoltage of the digital power supply. 1 indicates that the digital power supply enters the undervoltage, and 0 indicates that the digital power supply is normal
	Bit[4]:AVDD_UV_Flag indicates the analog power supply undervoltage flag. 1 indicates that the analog power supply enters the undervoltage, and 0 indicates that the analog power supply is normal
	Bit[5]:POR_RESET_Flag, indicating the POR reset flag. 1 indicates that the POR has been reset, and 0 indicates that the POR is normal

SYSCTRL_APMUSTR 的描述如下：

- 0: otd_flag
- 1: sleep_flag
- 2: ret_mode_flag
- 3: dvdd_uv_flag
- 4: avdd_uv_flag
- 5: por_res_flag

上电后读取该寄存器就可以判断是单纯的POR还是SLEEP_FLAG+POR，从而可以实现不同的控制策略。
为了读取这个寄存器，用户需要写3个KEY，之后就可以读取APMUSTR寄存器的值。

```
#define SYSCTRL_UNLOCK_KEY          0x87e4

#define SYSCTRL_TRIM_REG_UNLOCK_KEY1 0x8a3d
#define SYSCTRL_TRIM_REG_UNLOCK_KEY2 0x19ec

/*config sysctrl*/

SYSCTRL->LKKEYR      = SYSCTRL_UNLOCK_KEY;

SYSCTRL->LKKEYR      = SYSCTRL_TRIM_REG_UNLOCK_KEY1;

SYSCTRL->LKKEYR      = SYSCTRL_TRIM_REG_UNLOCK_KEY2;
```

1.5 Flash的操作需要注意什么?



牛天才

回复

1.强烈建议Flash的操作通过调用ROM里的Flash函数来实现。

ROM里的Flash操作接口函数只有三个函数（Erase Full、Erase Sector 和 Write Page）

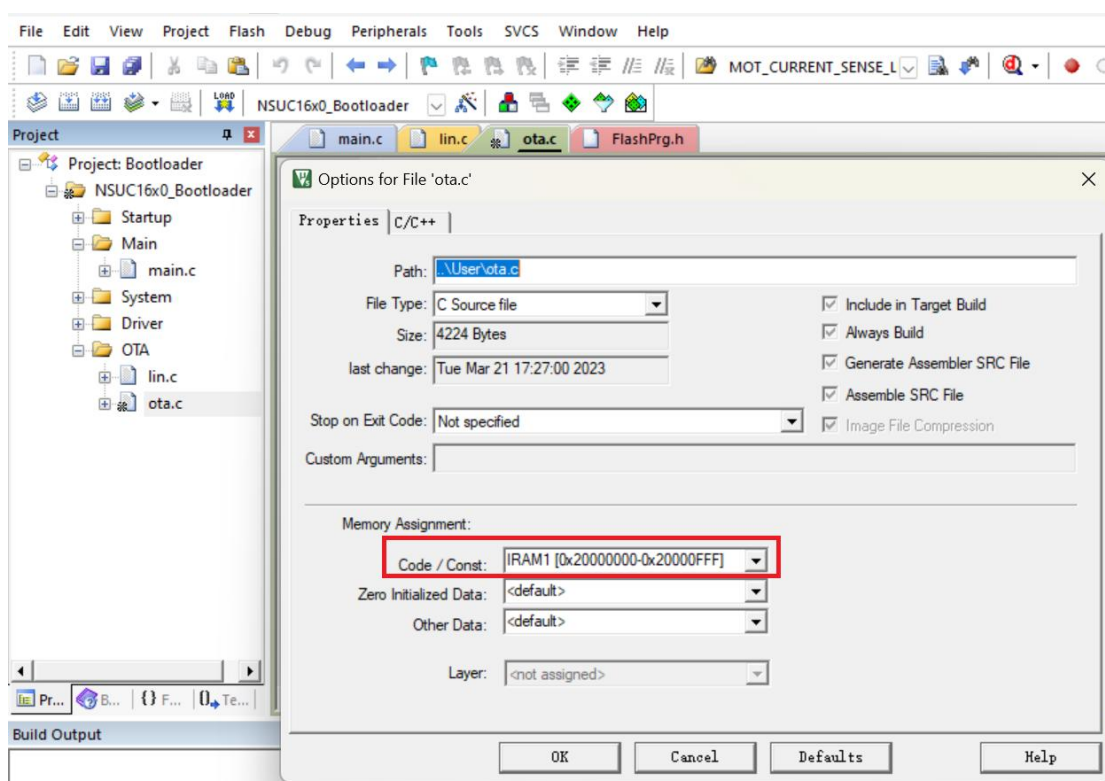
```
#define Write_Page ((unsigned int (*)(unsigned int page_address, unsigned int *data_array))0x01000139)
```

```
#define Erase_Sector ((unsigned int (*)(unsigned int sec_address))0x01000091)
```

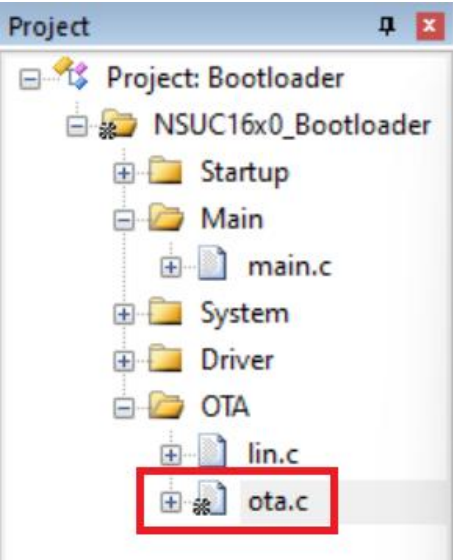
```
#define Erase_Flash ((unsigned int (*)(unsigned int))0x01000001)
```

这3个函数就可以实现Bootloader等常规功能。

调用这些函数的时候，同时需要确保调用这些函数的函数也放到RAM里运行。以纳芯微bootloader程序为例，ota.c 就是整个文件都放在RAM里跑的。



这个设置可以通过魔法棒里的 Code/Const 下拉选项中选择 “IRAM1 (0x20000000-0x20000FFF)” 来实现。一旦成功后，你会发现Project栏目下整个ota.c 的左下角多出来一个小图标。如下图所示。



2.Flash操作的时候确保中断使能是关闭的以及清掉所有pending的中断flag，否则可能导致进入到中断函数，而如果中断函数在Flash中的话就会导致Hardfault。

1.6 为什么我的工程代码会莫名其妙的RESET?



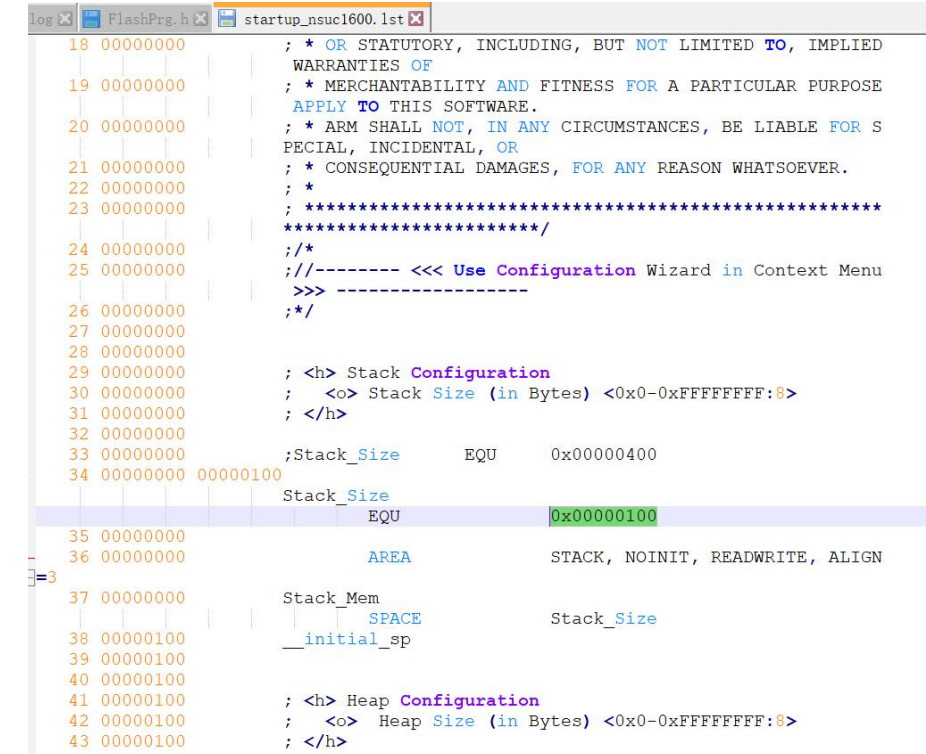
牛天才

回复

使用纳芯微的例程，如果用户出现了莫名其妙的RESET现象，请考虑stack不够引起的问题。例如当用户额外设置了较大的数组变量，而stack的默认值是 0x100的话就会出现stack不够的场景。这个时候会出现reset的问题。

解决的办法是增大Stack或者减少局部变量的长度，比如将0x100增大到0x400。

这个修改是在startup_nsuc1610.lst文件里进行，如下图所示：



EQU 0x00000100 改成 EQU 0x00000400 即可。



1.7 在Keil里如何将某个.c文件放到RAM里面运行?



牛天才

回复

请参考Q5的答案，可以通过魔法棒工具直接制定特定的.c文件到RAM中即可。



1.8 NSUC1610过温硬件保护如何开启?



牛天才

回复

NSUC1610过温硬件保护需要软件配置后才会开启，参考代码如下：

```
SYSCTRL->PWRCR_b.TSF = 0;
```

```
SYSCTRL->PWRCR_b.TSREN = 1;
```

销售联系方式: sales@novosns.com; 获取更多信息: www.novosns.com

重要声明

本文件中提供的信息不作为任何明示或暗示的担保或授权,包括但不限于对信息准确性、完整性,产品适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的陈述或保证。

客户应对其使用纳芯微的产品和应用自行负责,并确保应用的安全性。客户认可并同意:尽管任何应用的相关信息或支持仍可能由纳芯微提供,但将在产品及其产品应用中遵守纳芯微产品相关的所有法律、法规和相关要求。

本文件中提供的资源仅供经过技术培训的开发人员使用。纳芯微保留对所提供的产品和服务进行更正、修改、增强、改进或其他更改的权利。纳芯微仅授权客户将此资源用于开发所设计的整合了纳芯微产品的相关应用,不视为纳芯微以明示或暗示的方式授予任何知识产权许可。严禁为任何其他用途使用此资源,或对此资源进行未经授权的复制或展示。如因使用此资源而产生任何索赔、损害、成本、损失和债务等,纳芯微对此不承担任何责任。

有关应用、产品、技术的进一步信息,请与纳芯微电子联系 (www.novosns.com)。

苏州纳芯微电子股份有限公司版权所有