1T 8051

8-bit Microcontroller

MS51 / ML51 SPROM
User Manual
NuMicro® 8051 Series

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

Table of Contents

1 OVERVIEW

MS51 & ML51 series with an additional include special 128 bytes security protection memory (SPROM) to enhance the security and protection of customer application. To facilitate programming and verification, the Flash allows to be programmed and read electronically by parallel Writer or In-Circuit-Programming (ICP). Once the code is confirmed, user can lock the code for security.

SPROM start address from FF80H, if the last byte (FFFFH) is the lock bit of SPROM, if this byte value is not FFH, whole SPROM can't be read out, include IAP / ICP or MOVC instruction, also can't setting break point when in OCD mode.

2 SPROM FUNCTION DESCRIPTION

2.1 Security Protection Memory (SPROM)

The security protection memory (SPROM) is used to store instructions for security application. The SPROM includes 128 bytes at location address FF80H ~ FFFFH and doesn't support "whole chip erase command". Figure 2.1-1 SPROM Memory Mapping And SPROM Security Mode shows that the last byte of SPROM (address: FFFFH) is used to identify the SPROM code is non-secured or secured mode.

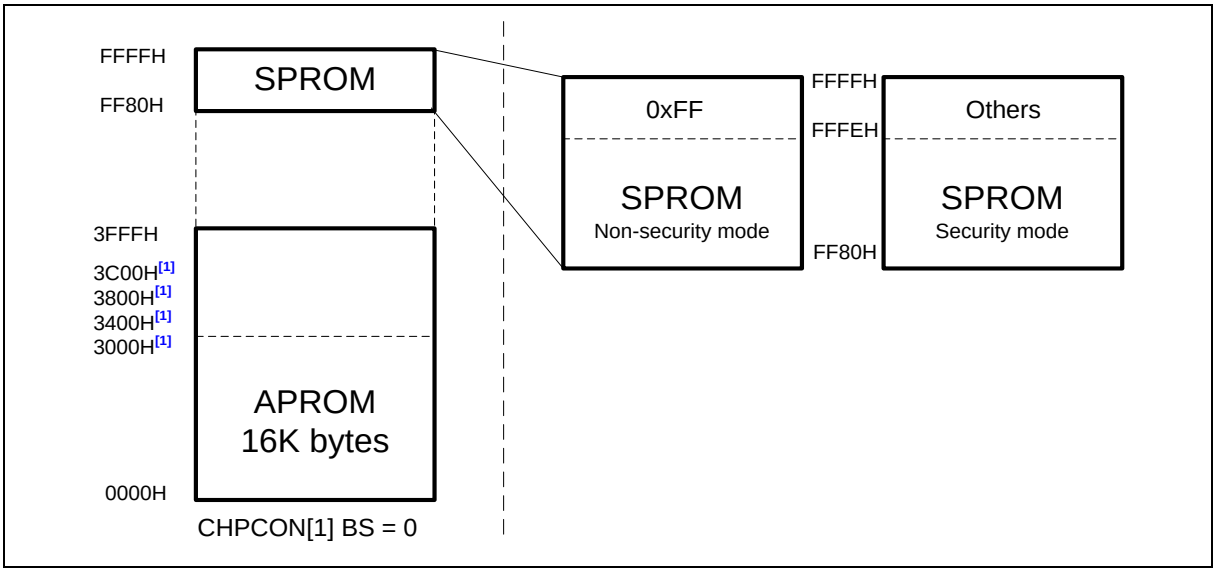


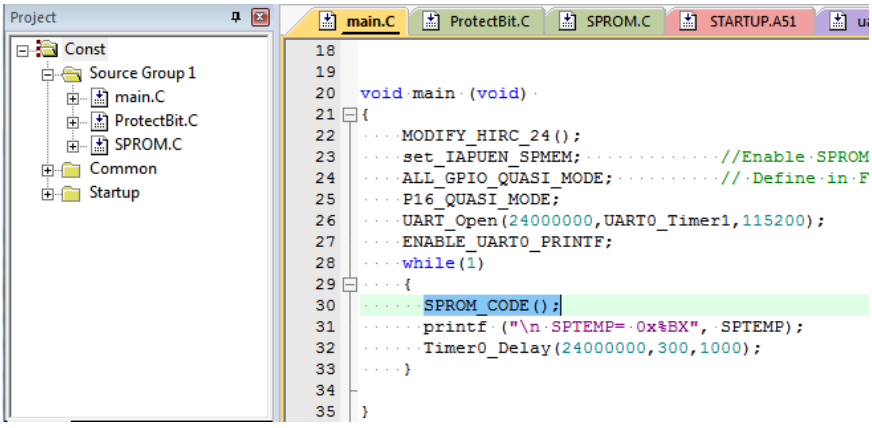
Figure 2.1-1 SPROM Memory Mapping And SPROM Security Mode

(1) SPROM non-secured mode (the last byte is 0xFF). The access behavior of SPROM is the same with APROM and LDROM. All area can be read by CPU or ISP command, and can be erased and programmed by ISP command.

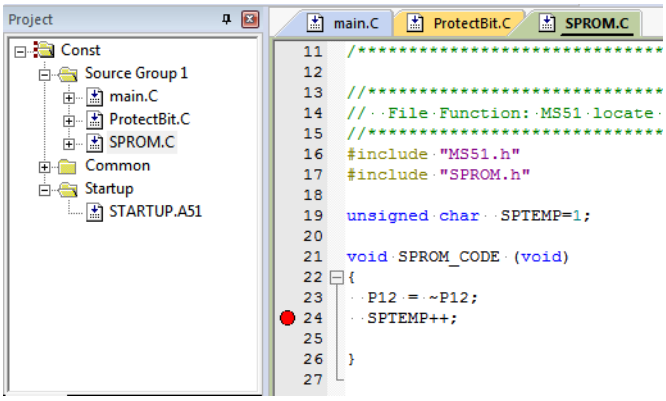
(2) SPROM secured mode (the last byte is not 0xFF). In order to conceal SPROM code in secured mode, CPU only can perform instruction fetch and get data from SPROM when CPU is run at SPROM area. Otherwise, CPU will get all 00H for data access. In order to protect SPROM, the CPU instruction fetch will also get zero value when ICE (OCD) port is connected in secured code. At this mode, SPROM doesn't support ISP program, read or erase.

2.2 How To Coding Sprom Code In Keil Project

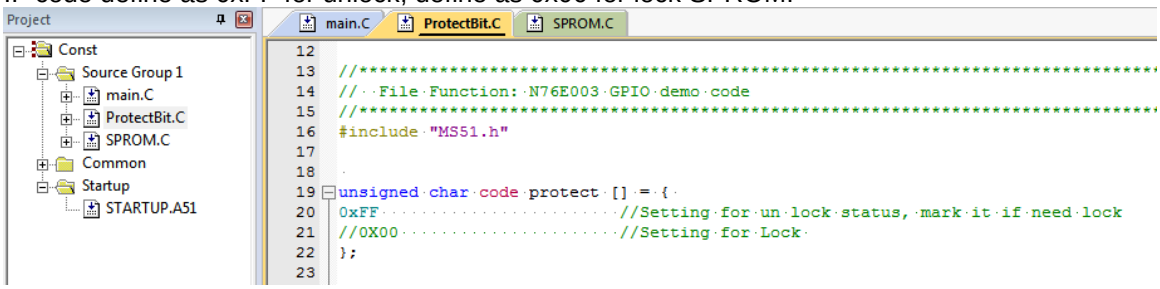
Main loop call SPROM code, for example call SPROM function “SPROM_CODE();”



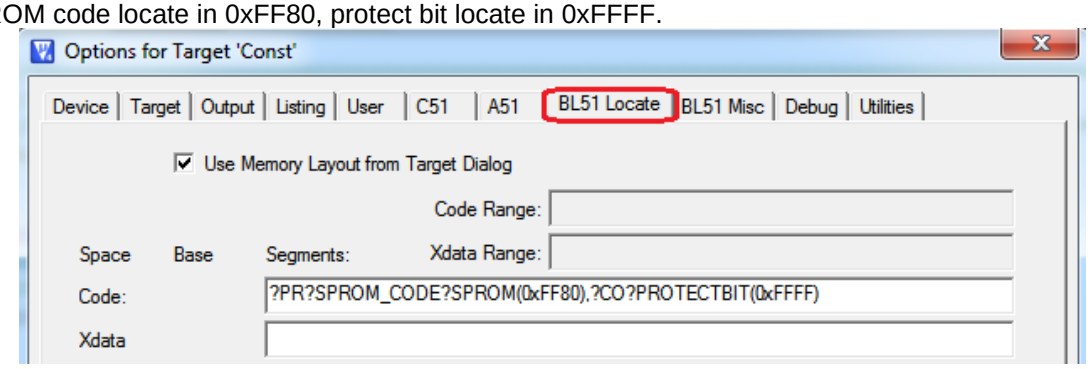
SPROM code: File name SPROM.c.



SPROM Secuirty bit control code: File name ProtectBit.c.
IF code define as 0xFF for unlock, define as 0x00 for lock SPROM.

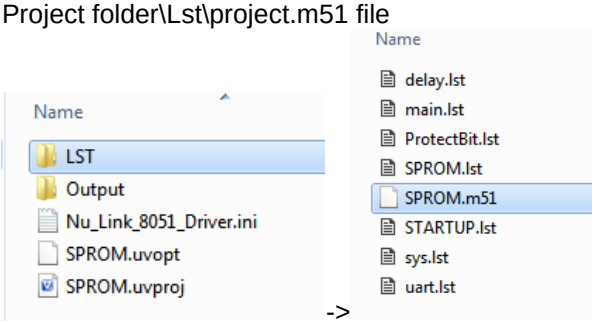


Setting the SPROM address and Protect bit address in options / BL51 Locate.
SPROM code locate in 0xFF80, protect bit locate in 0xFFFF.



To find the option “Code” marco name: check projet.m51 file

Project folder\Lst\project.m51 file



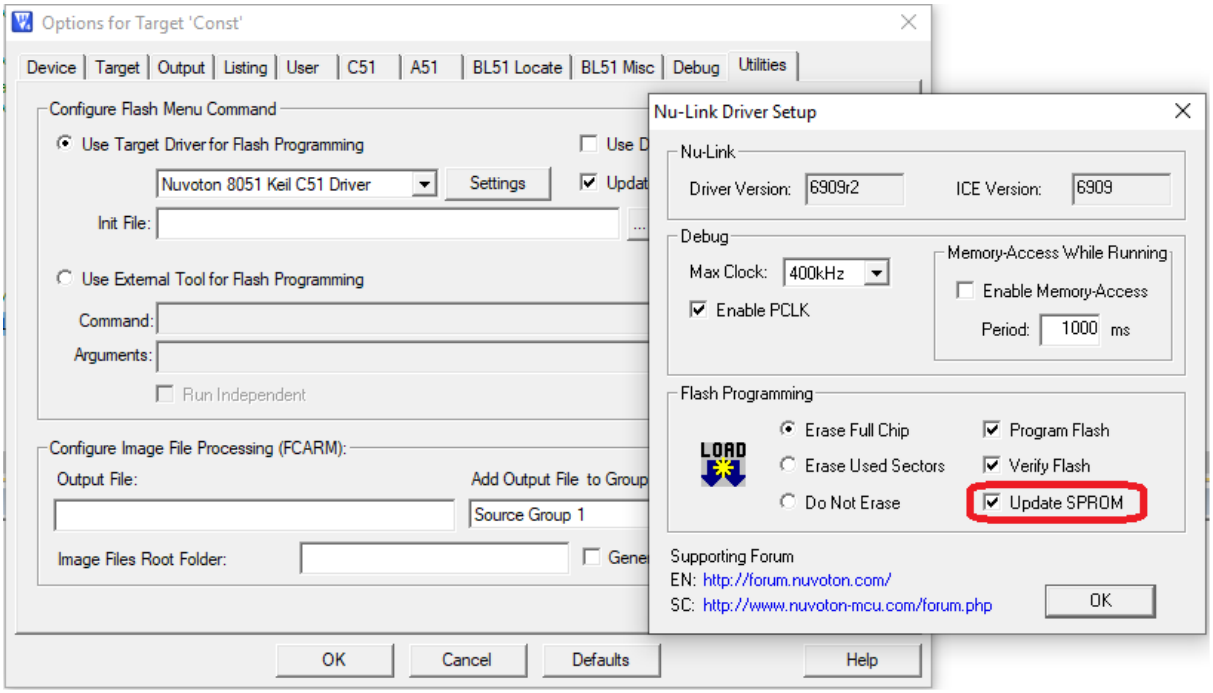
After define the SPROM and SPROM security bit memory mapping
SPROM locate in FF80 and Protectbit locate in FFFF.

```
77          CODE    0D55H   0010H   UNIT        ?CO?MAIN
78          CODE    0D65H   0004H   UNIT        ?C_INITSEG
79                  0D69H   F217H               *** GAP ***
80          CODE    FF80H   0005H   UNIT        ?PR?SPROM_CODE?SPROM
81                  FF85H   007AH               *** GAP ***
82          CODE    FFFFH   0001H   UNIT        ?CO?PROTECTBIT
```

Before difine the location (without setting address in options)

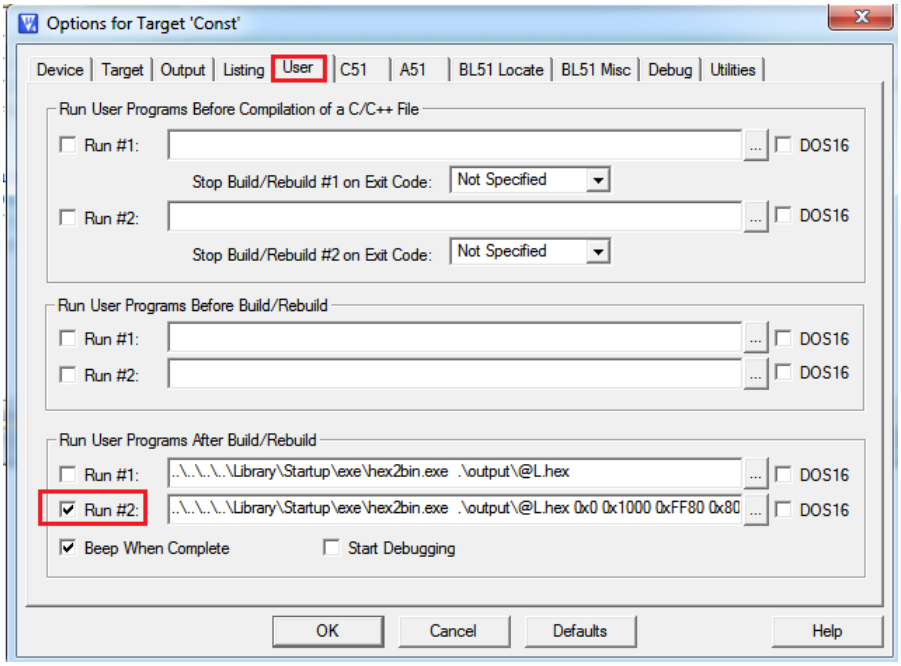
```
77          CODE    0D55H   0010H   UNIT        ?CO?MAIN
78          CODE    0D65H   0005H   UNIT        ?PR?SPROM_CODE?SPROM
79          CODE    0D6AH   0004H   UNIT        ?C_INITSEG
80          CODE    0D6EH   0001H   UNIT        ?CO?PROTECTBIT
```

Before download or enter debug mode enable “Update SPROM” is necessary.

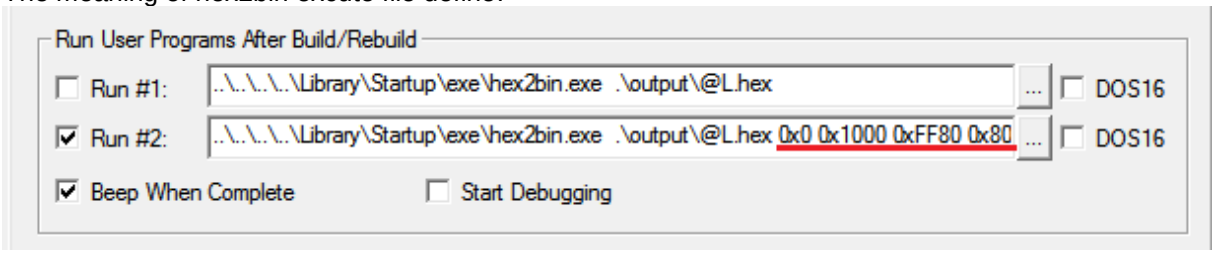


2.3 How To Create The Bin File of APROM and SPROM

Choose run “Options -> User -> Run User Programs After Build/Rebuild as following”



The meaning of hex2bin excute file define:



APROM start address	APROM bin file size define	SPROM start address	SPROM bin file size define
0x0	0x1000	0xFF80	0x80

To fine APROM size check with .m51 file.
For example in this project the APROM size must be larger than 0x0D69.

```
77          CODE    0D55H     0010H     UNIT         ?CO?MAIN
78          CODE    0D65H     0004H     UNIT         ?C_INITSEG
79                  0D69H     F217H                  *** GAP ***
80          CODE    FF80H     0005H     UNIT         ?PR?SPROM_CODE?SPROM
81                  FF85H     007AH                  *** GAP ***
82          CODE    FFFFH     0001H     UNIT         ?CO?PROTECTBIT
```

Note: this hex2bin.exe is only released by nuvoton. Please download nuvoton MS51 BSP package.

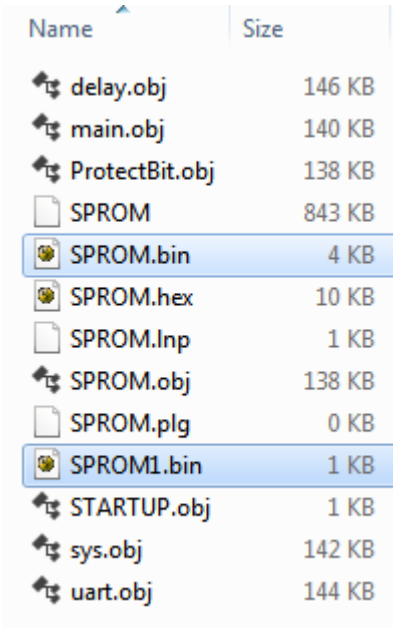
Github: https://github.com/OpenNuvoton/MS51_BSP_KEIL

https://github.com/OpenNuvoton/ML51_BSP_KEIL

Find in “Output” folder to find the bin file

For APROM code: SPROM.bin

For SPROM code: SPROM1.bin



3 REVISION HISTORY

Date	Revision	Description
2019.05.206	1.00	Initial release

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.