一. AVR Programmer Software V1.26—2007.8 说明手册

AVR Programmer Software V1.26—2007.8 是 315avr 网站自己开发的一款 AVR 编程软件,支持本站开发的可脱机系列产品:(可脱机 USB AVRISP 下载器, USB AVR Pro/ISP(可脱机高压编程器+ISP 下载器)等)。

1. 软件安装说明

安装之前请下载最新版本软件,将软件解压缩到本地一个文件夹中。

■ 双击软件文件夹中的 setup.exe, 一路单击 "next", 直至 "Finish" 完成软件安装; 软件界面如 下图。

软件正确启动界面:

File Edit Config Operation Help	
ReLoad ID Erase Read ProFlash Pro EE ProFuse Help	
COM: USB Device: ATMEGA16 Check Sum: 0x61ef	-Auto Program Options
	Fuse RC S/N
AVR_Programmer Software V1.26 ATMEGA16 Adapter: ADP_M16_D(DIP),ADP_M16_S(SOIC),ADP_M16_M(MLF)	Frase Device
Buffer Check Sum: 0x61ef	🗹 ID Check
	🔽 Program Flash
	🗖 Program Eeprom
	Program Fuse & Lock
	🗖 Calibrated Internal RC osc.
	₩rite SN
	Config Disable 🔽
	8 8
	Off line Auto
	www.315avr.com

2. 软件界面功能描述

AVR Programmer Software V1.26-2007.8 软件主界面如上图六所示。

Menu Operation (菜单栏)

File Edit Config Operation Help

a. File (文件)

File	Edit	Config	Operatio	n	Help		
Load	Flash Fi	le	1				
Save	Flash Fi	le					
Load Eeprom File			Erase	Kea	Kead Fro Flash		
Save	Eeprom F	ile	Device:	ATI	MEG/	448	_
Load	Project	File					_
Save	Project	File	tware vi.a	2			

Load Flash File: Load File (*.hex*.Bin*.A90) To Flash Data Buffer.

Save Flash File: Save Flash Buffer Data to a File

Load Eeprom File: Load File (*.hex*.eep*.A90) To Eeprom Data Buffer.

Save Eeprom File: Save Eeprom Buffer Data To A File.

Load Project File: Open a Project File (*.prj).

Save Project File: Save Current Config To a Project File.

b. Edit (编辑)



Flash Buffer : Display a hex_edit Dialog for Flash Buffer Data. Eeprom Buffer : Display a hex_edit Dialog for Eeprom Buffer Data. CheckSum: Calculate Flash (Eeprom) Buffer CheckSum.

c. Config (设置)



Picture: Erase tool bar menu

Erase MCU Flash & Eeprom Data and Unlock Lock Bit.

d. Read (读取 Flash, EEPROM, Fuse & Lock bits)



Picture: Read tool bar menu

Read MCU Flash & Eeprom Data and Read Fuse Lock Bit.

e. Pro flash (编程、校验 Flash)



Picture: Pro flash tool bar menu

Write Flash Buffer Data to MCU Flash Memory, and then Verify MCU Flash Memory with Flash Buffer Data.

f. Pro EE (编程、校验 EEPROM)



Picture: Pro Eeprom tool bar menu

Write Eeprom Buffer Data To MCU Eeprom Memory, and then Verify MCU Eeprom Memory With Eeprom Buffer Data.

g. Help (帮助快捷键)



Display User Manual.

h. Status Icon (编程状态指示)

Auto Program: Auto Program Flash&Eeprom Memory and Fuse&Lock Bit with Current Config. Offline: Download Current Config to Config_Memory of Programmer. Then You Can Program the MCU without PC.

e. Help (帮助)

User Help: Display User Manual (用户指南)

About: Display AVR_Prgrammer Software Version Information.



ATMEGA48 Adapter: ADP_M8_D(DIP),ADP_M8_S(SOIC),ADI

Buffer Check Sum: 0xf995

3. Tool Bar (工具条栏目)



a. Reload (重新载入文件)



Picture: Reload tool bar menu

If Flash & Eeprom File is opened, then reload Flash File to Flash Data Buffer and Reload Eeprom File to Eeprom Data Buffer.

b. ID (读取芯片 I D 号)



Read MCU ID(Signature Bytes)

c. Erase (擦除 Flash, EEPROM, Lock bits)



Picture: Erase tool bar menu

Erase MCU Flash & Eeprom Data and Unlock Lock Bit.

d. Read (读取 Flash, EEPROM, Fuse & Lock bits)



Picture: Read tool bar menu

Read MCU Flash & Eeprom Data and Read Fuse Lock Bit.

e. Pro flash (编程、校验 Flash)



Picture: Pro flash tool bar menu

Write Flash Buffer Data to MCU Flash Memory, and then Verify MCU Flash Memory with Flash Buffer Data.

f. Pro EE (编程、校验 EEPROM)



Picture: Pro Eeprom tool bar menu

Write Eeprom Buffer Data To MCU Eeprom Memory, and then Verify MCU Eeprom Memory With Eeprom Buffer Data.

g. Help (帮助快捷键)



Display User Manual.

h. Status Icon (编程状态指示)



Operation OK



Operation Running



Operation Failed

4. Program options (Include Auto program options)

(编程设置,包括自动编程设置)

a. Select COM Port (选择端口,软件检测可脱机 USB 设备)



You Can Select A Port To Communicate Between PC and Programmer. The Port May USB OR COM1 To COM16.

b. Select Device (选择芯片型号)

TMEGA48	•
	TMEGA48

Select a Device (MCU) That Need To Program.



• **Program Ecprom** Enable OR Disable Program Eeprom Memory When Auto Program Device.

• **Program Fuse & Lock** Enable OR Disable Program Fuse&Lock Bit When Auto Program Device .

• **Calibrated Internal RC osc.** Enable OR Disable Program RC Calibrating Value When Auto Program Device.

• Write SN Enable OR Disable Program Serial Number When Auto Program Device .



• After setup all of the program options, you can press the menu to auto program device.



Press this menu, Download Current Config to Config_Memory of Programmer.

Then You Can Program the MCU without PC.

5. Buffer edit window (缓冲区编辑窗口)

This dialog is used for edit Flash or Eeprom buffer data. You can do two different steps: 1. save buffer data as a file 2. Fill buffer with special value

Flas	h Bui	fer	Edi	t.															
Add	ress	0	1	2	3	4	5	6	7	8	9	A	в	С	D	E	F		
	00000	0000	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0010	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0020	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0030	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0040	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0050	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0060	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0070	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0800	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0090	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	DAOC	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	00в0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	00CO	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	DODO	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	DOEO	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	OFO	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0100	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0110	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	00000	0120	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	
	6																		
									-										
- 28	ive	0.4	n .	_	1	0-1	1	(F	-					10 L				
Fr	om	UXI	J							Fro	m	0	×0				N	1	ОК 🖌
т		0			. 3	Sav	е		- 23	Τn		10	.10	00	-		-		
11		UX	1001	1	1							U	XIU	00			Fill		
_			-	-		_	_			Vali	Ie	0	vn	_	1				Cancel ¥
	GoTo	1	$ \rightarrow $		0×0	-	-			Tal		10	~0				-		
-																			

6. Fuse setting window (熔丝位设置)

This dialog is user for set fuse & lock bits.

Fuse Setting Value: Implet: 0x62 High Byte: 0x41 Extended Byte: 0x61 Lock Byte: 0x61 00- Setif Programming Enable[SELFPRGEN=0] Implet: 0x61	Fuse Set 📃 🗖 🔀
00- Self Programming Enable[SELFPRGEN=0] 01- Reset Disable (Enable PC6 as i/o PINJ[RSTDISBL=0] 02- Debug Wire Enable[DWEM=0]//DISPABLE 03- Serial Programmer Download (SPI) Enable[SPIEN=0] 04- Watch-Dog Timer Always On[WDTON=0] 05- Preserve EEPROM Memory Through The Chip Erase Cycle[EESAVE=0] 06- Brown-out Detection Level At VCC=4.3V[BODLEVEL=100] 07- Brown-out Detection Level At VCC=2.7V[BODLEVEL=101] 08- Brown-out Detection Devel At VCC=1.8V[BODLEVEL=110] 09- Brown-out Detection Devel At VCC=1.8V[BODLEVEL=110] 09- Brown-out Detection Disable[BODLEVEL=111] 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTB0II[CKOUT=0] 12- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=00] 13- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 14- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=00] 15- Int. RC Osc[8MH2]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 15- Int. RC Osc[8MH2]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=01] 16- Int. RC Osc[8MH2]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=01] 17- Int. RC Osc[128KH2; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 18- Int. RC Osc[128KH2; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 18-	Fuse Setting Value: Low Date 0.62 Use Date Date Date Date 0.66 Lost Date 0.6
00- Self Programming Enable[SELFPRGEN=0] 01- Reset Disable (Enable PC6 as i/o PINJ[RSTDISBL=0] 02- Debug Wire Enable[DWEM=0]//DISPABLE 03- Serial Programmer Download (SPI] Enable[SPIEN=0] 04- Watch-Dog Timer Always On[WDT0N=0] 05- Preserve EEPROM Memory Through The Chip Erase Cycle[EESAVE=0] 06- Brown-out Detection Level At VCC=4.3V[BODLEVEL=100] 07- Brown-out Detection Level At VCC=2.7V[BODLEVEL=101] 08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=101] 09- Brown-out Detection Disable[BODLEVEL=111] 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOII[CKOUT=0] 12- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+40ms[CKSEL=0000 SUT=00] 13- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+45ms[CKSEL=0000 SUT=01] 14- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+40ms[CKSEL=0010 SUT=00] 15- Int. RC Osc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+40ms[CKSEL=0010 SUT=01] 14- Ent. RC Osc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 17- Int. RC Osc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=01] 18- Int. RC Osc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=01] 19- Int. RC Osc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC Osc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10]	Tuse betting faide. Low byte: uxbz nigh byte, uxai Extended byte: uxii Luck byte: uxii
01- Reset Disable [Enable PC6 as i/o PIN][RSTDISBL=0] 02- Debug Wire Enable[DWEM=0]//DISPABLE 03- Serial Programmer Download (SPI) Enable[SPIEN=0] 04- Watch-Dog Timer Always On[WDTON=0] 05- Preserve EEPROM Memory Through The Chip Erase Cycle[EESAVE=0] 06- Brown-out Detection Level At VCC=4.3V[BODLEVEL=100] 07- Brown-out Detection Level At VCC=2.7V[BODLEVEL=101] 08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=110] 09- Brown-out Detection Disable[BODLEVEL=111] 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOII[CKOUT=0] 12- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0000 SUT=00] 13- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 14- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=00] 15- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+40ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+40ms[CKSEL=0010 SUT=10] 17- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+45ms[CKSEL=0011 SUT=10] 18- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+45ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+45ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+45ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6	🔲 00- Self Programming Enable[SELFPRGEN=0]
02- Debug Wire Enable[DWEM=0]//DISPABLE 03- Serial Programmer Download (SPI) Enable[SPIEN=0] 04- Watch-Dog Timer Always On[WDTON=0] 05- Preserve EEPROM Memory Through The Chip Erase Cycle[EESAVE=0] 06- Brown-out Detection Level At VCC=4.3V[BODLEVEL=100] 07- Brown-out Detection Level At VCC=1.8V[BODLEVEL=101] 08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=110] 09- Brown-out Detection Disable[BODLEVEL=111] 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOII[CKOUT=0] 12- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=00] 13- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 14- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=00] 15- Int. RC 0sc(8MHz]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc(8MHz]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=10] 17- Int. RC 0sc(128KHz]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 18- Int. RC 0sc(128KHz]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz]; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0	🔲 01– Reset Disable (Enable PC6 as i/o PIN)[RSTDISBL=0]
✓ 03- Serial Programmer Download (SPI) Enable[SPIEN=0] 04- Watch-Dog Timer Always On[WDTON=0] 05- Preserve EEPROM Memory Through The Chip Erase Cycle[EESAVE=0] 06- Brown-out Detection Level At VCC=4.3V[BODLEVEL=100] 07- Brown-out Detection Level At VCC=2.7V[BODLEVEL=101] 08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=110] 09- Brown-out Detection Disable[BODLEVEL=111] ✓ 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOILCKOUT=0] 12- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+40ms[CKSEL=0000 SUT=00] 13- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 15- Int. RC Osc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+465ms[CKSEL=0010 SUT=00] 16- Int. RC Osc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+465ms[CKSEL=0010 SUT=10] 17- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 18- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=20] 19- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=12] 20- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=12] 20- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=12]<	🔲 02- Debug Wire Enable[DWEM=0]//DISPABLE
04- Watch-Dog Timer Always On[WDTON=0] 05- Preserve EEPROM Memory Through The Chip Erase Cycle[EESAVE=0] 06- Brown-out Detection Level At VCC=4.3V[BODLEVEL=100] 07- Brown-out Detection Level At VCC=2.7V[BODLEVEL=101] 08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=110] 09- Brown-out Detection Disable[BODLEVEL=111] 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOII[CKOUT=0] 12- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+40ms[CKSEL=0000 SUT=00] 13- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 15- Int. RC Osc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 16- Int. RC Osc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC Osc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 18- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=11] 20- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=11] 21- Ext. Low-Freq Cry Osc(32.768KHz); Start-up Time PWRDWN/RESET	🗹 03– Serial Programmer Download (SPI) Enable[SPIEN=0]
05- Preserve EEPROM Memory Through The Chip Erase Cycle[EESAVE=0] 06- Brown-out Detection Level At VCC=4.3V[BODLEVEL=100] 07- Brown-out Detection Level At VCC=2.7V[BODLEVEL=101] 08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=110] 09- Brown-out Detection Disable[BODLEVEL=111] 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOII[CKOUT=0] 12- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0000 SUT=00] 13- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=10] 14- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 15- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18- Int. RC 0sc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=1] 20- Int. RC 0sc(128Hz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=1] 21- Ext. Low-Freq Cry 0sc(32.768KHz); Start-up Time PWRDWN/RESET 1CK/14CK+40ms[CKSEL] Q Writ	🔲 04– Watch-Dog Timer Always On[WDTON=0]
06- Brown-out Detection Level At VCC=4.3V[BODLEVEL=100] 07- Brown-out Detection Level At VCC=2.7V[BODLEVEL=101] 08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=110] 09- Brown-out Detection Disable[BODLEVEL=111] 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOII[CKOUT=0] 12- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0000 SUT=00] 13- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=01] 15- Int. RC 0sc(8MHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc(8MHz): Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc(8MHz): Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 18- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 18- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=20] 19- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=12] 20- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=12] 21- Ext. Low-Freq Cry 0sc[32.768KHz]: Start-up Time PWRDWN/RESET 1CK/14CK+65ms[CKSEL] Image: Read Write Image: Rea	05– Preserve EEPROM Memory Through The Chip Erase Cycle[EESAVE=0]
07- Brown-out Detection Level At VCC=2.7V[BODLEVEL=101] 08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=110] 09- Brown-out Detection Disable[BODLEVEL=111] 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOII[CKOUT=0] 12- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0000 SUT=00] 13- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 15- Int. RC 0sc(8MHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc(8MHz): Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc(8MHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=01] 18- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=01] 19- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=01] 19- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=1] 20- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=1] 21- Ext. Low-Freq Cry 0sc(32.768KHz): Start-up Time PWRDWN/RESET 1CK/14CK+65ms[CKSEL] Image: Read Write Image: Erase Image: Default	06- Brown-out Detection Level At VCC=4.3V[BODLEVEL=100]
08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=110] ✓ 09- Brown-out Detection Disable[BODLEVEL=111] ✓ 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOII[CKOUT=0] 12- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0000 SUT=00] 13- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 15- Int. RC 0sc(8MHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc(8MHz): Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc(8MHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=20] 19- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 20- Int. RC 0sc(128KHz): Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 21- Ext. Low-Freq Cry 0sc(32.768KHz): Start-up Time PWRDWN/RESET 1CK/14CK+0ms[CKSEL] Image: Complexity of the point of	07- Brown-out Detection Level At VCC=2.7V[BODLEVEL=101]
✓ 09- Brown-out Detection Disable[BODLEVEL=111] ✓ 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTBOII[CKOUT=0] 12- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0000 SUT=00] 13- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 15- Int. RC 0sc(8MHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc(8MHz]: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc(8MHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18- Int. RC 0sc(128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=20] 19- Int. RC 0sc(128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=12] 20- Int. RC 0sc(128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=12] 21- Ext. Low-Freq Cry 0sc(32.768KHz]: Start-up Time PWRDWN/RESET 1CK/14CK+0ms[CKSEL] I Image: Provent PWRDWN/RESET 1CK/14CK+0ms[CKSEL] I Image: Provent PWRDWN/RESET 1CK/14CK+0ms[CKSEL]	08- Brown-out Detection Level At VCC=1.8V[BODLEVEL=110]
✓ 10- Divde Clock By 8 Internally[CKDIV8=0] 11- Clock Output On PORTB011[CKOUT=0] 12- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0000 SUT=00] 13- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 15- Int. RC 0sc[8MHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc[8MHz]: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc[8MHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 20- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 21- Ext. Low-Freq Cry 0sc[32.768KHz]: Start-up Time PWRDWN/RESET 1CK/14CK+0ms[CKSEL] Read Write Erase Default OK Cancel X	09- Brown-out Detection Disable[BODLEVEL=111]
11- Clock Output On PORTBUILCKOUTEU] 12- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0000 SUT=00] 13- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0000 SUT=10] 15- Int. RC 0sc[8MHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc[8MHz]: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc[8MHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 18- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 20- Int. RC 0sc[128KHz]: Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=11] 21- Ext. Low-Freq Cry 0sc[32.768KHz]: Start-up Time PWRDWN/RESET 1CK/14CK+0ms[CKSEL] Read Write Erase Default OK Cancel X	10- Divde Clock By 8 Internally[CKDIV8=0]
12- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0000 SUT=00] 13- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0000 SUT=10] 15- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0010 SUT=00] 16- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=10] 19- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=10] 20- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=1] 21- Ext. Low-Freq Cry 0sc(32.768KHz); Start-up Time PWRDWN/RESET 1CK/14CK+05ms[CKSEL] Read Write Erase Default OK Cancel	
13- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0000 SUT=01] 14- Ext. Clock; Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 15- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0010 SUT=01] 16- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=01] 19- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=01] 20- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=1] 21- Ext. Low-Freq Cry 0sc(32.768KHz); Start-up Time PWRDWN/RESET 1CK/14CK+65ms[CKSEL=011 SUT=1] 21- Ext. Low-Freq Cry 0sc(32.768KHz); Start-up Time PWRDWN/RESET 1CK/14CK+0ms[CKSEL] Read Write Erase Default OK Cancel X	12- Ext. Clock; Start-up Time PWRDWN/RESE1 6CK/14CK+Ums[CKSEL=0U00 SU1=00]
14- Ext. Clock, start-up Time PWRDWN/RESET 6CK/14CK+05sits[CKSEL=0000 S01=10] 15- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0010 SUT=01] 16- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=01] 17- Int. RC 0sc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=01] 19- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=01] 20- Int. RC 0sc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=1] 21- Ext. Low-Freq Cry 0sc(32.768KHz); Start-up Time PWRDWN/RESET 1CK/14CK+65ms[CKSEL=0011 SUT=1] 21- Ext. Low-Freq Cry 0sc(32.768KHz); Start-up Time PWRDWN/RESET 1CK/14CK+0ms[CKSEL] Read Write Erase Default OK Cancel	13- EXT. CIOCK, STAT-UP TIME PWRDWN/RESET 0CM/14CK+4.Ims[CKSEL=0000 SUT=10]
15- Int. RC Osc(0MH2), Start-up Time PWRDWN/RESET 6CK/14CK+0Ins[CKSEL=0010 SUT=01] 16- Int. RC Osc(8MH2); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0010 SUT=10] 17- Int. RC Osc(8MH2); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18- Int. RC Osc(128KH2); Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0011 SUT=0] 19- Int. RC Osc(128KH2); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=0] 20- Int. RC Osc(128KH2); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=1] 20- Int. RC Osc(128KH2); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=1] 21- Ext. Low-Freq Cry Osc(32.768KH2); Start-up Time PWRDWN/RESET 1CK/14CK+65ms[CKSEL] Read Write Erase Default OK Cancel	
10° Int. RC Osc(8MHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0010 SUT=10] 18~ Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=0] 19~ Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=0] 20~ Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT=1] 20~ Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=1] 21~ Ext. Low-Freq Cry Osc(32.768KHz); Start-up Time PWRDWN/RESET 1CK/14CK+65ms[CKSEL] Read Write Erase Default OK Cancel	15- Int. RC Osc(0MHz), Start-up Time PWPDWN/RESET 6CK/14CK+0105[CKSEL-0010 S01-00]
Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+0ms[CKSEL=0011 SUT=0(19- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+4.1ms[CKSEL=0011 SUT= 20- Int. RC Osc(128KHz); Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT= 21- Ext. Low-Freq Cry Osc(32.768KHz); Start-up Time PWRDWN/RESET 1CK/14CK+65ms[CKSEL=0011 SUT= Read Write Erase Default OK Cancel	To the the Osciowitz, startup time twitting test social took the test social tool to the test social test social to the test social
Image: Start of the start	18- Int BC OscillarkHzi: Startun Time PWBNWN/BEST 6CK/14CK+0ms(CKSEI =0011 SUIT=0)
20- Int. RC Osc[128KHz]; Start-up Time PWRDWN/RESET 6CK/14CK+65ms[CKSEL=0011 SUT=1 21- Ext. Low-Freq Cry Osc[32.768KHz]; Start-up Time PWRDWN/RESET 1CK/14CK+0ms[CKSEL] Read Write Erase Default OK Cancel X	9 - Int. BC 0scf128KHzi: Start-un Time PWBDWN/BESET 6CK/14CK+4.1ms[CKSE]=0011 SUT=
Read Nrite Erase A Default N OK Cancel	20- Int. BC 0sc(128KHz): Start-up Time PWBDWN/BESET 6CK/14CK+65ms[CKSEL=0011 SUT=1
Read 🚫 Write 🚳 Erase 💉 Default 🕟 OK 🖌 Cancel 🦹	21- Ext. Low-Freg Cry Osci32.768KHz1: Start-up Time PWRDWN/RESET 1CK/14CK+0ms/CKSEL
Read 🌍 Write 🚳 Erase 🚀 Default 🕟 OK 🖋 Cancel 🦹	
	Read 🎒 Write 🚳 Erase 🚀 Default 🕟 OK 🚀 Cancel 🦹

You can do four different steps:

- 1. Read fuse & lock bits
- 2. Write fuse & lock bits
- 3. Erase MCU memory and lock bits
- 4. Set fuse & lock bits to factory setting

7. RC Setting window (RC 校准设置)

This dialog is user for set internal RC calibration value.

- 1. Select a system frequency that need to calibration
- 2. Select area (flash or Eeprom area) that store calibration value
- 3. Set store address in flash (or Eeprom) area
- 4. Set compensation value for RC

NOTE: The Calibrated Value That Stored In MCU Memory Is Factory Calibrated Value Add Compensation Value.

Cal RC			
8.0MHz	C Reserve	:]	Read 🊫
C Reserve	C Reserve	:	
Write Area			
Flash	C Eeprom	Address	0×0
Add Value			
Minus	C Plus	Value	0×0

8. Serial number setting window (序列号设置)

This dialog is user for set Serial number.

I Setting -Write Area-			2
Flash	C Eeprom	Start Address: 0	×0
SN Bytes			
I Byte	C 2 Bytes	C 3 Bytes	C 4 Bytes
Byte Endian Big-End Little-En	ian (MSB in Low dian (MSB in Hi	Address) gh Address)	
Init Value:	D×0	ок 🖌	Cancel 🦹

- 1. Select area (flash or Eeprom area) that store serial number
- 2. Set start address and init value
- 3. Set S/N byte number

二. 可脱机 USB AVRISP 下载器 使用说明书

可脱 USB AVRISP 下载器 315avr 技术网自主开发的 AVR 专用 ISP 下载器。是 315avr 技术网开发的可脱 机系列 AVR 编程器产品中的一种。可脱机 ISP 下载,一键操作,快捷方便。USB 接口.支持全系列 AVR 芯片 的 ISP 下载,稳定性好.支持:熔丝位、加密位设置、内部 RC 校准、序列号设置等.极高的下载速度.站长重 点推荐的 AVR 开发生产工具。上位机编程软件使用本站配套开发的 AVR Programmer Software V1.26-2007.8。

本站可脱机系列产品现包括如下:(可脱机 USB AVRISP 下载器, USB AVR Pro/ISP(可脱机高压编程器+ISP 下载器))。

1. 可脱机 USB AVRISP 下载器功能描述

- · 编程功能: 支持全系列 AVR 的 ISP 编程;
- · 脱机支持: 可脱机 ISP 下载, 一键烧录;
- · 接口类型: USB 接口, USB 供电, 脱机 ISP 下载配 5.2V/500mA USB 电源适配器;
- · ISP 接口: ATMEL 标准 10pins ISP 编程接口, 配 10pin-6pin 转接板;
- · 项目管理: 可保存项目文件.proj,方便项目管理,只需调用保存的项目文件即可烧录,无须再配置;
- · RC 校准: 可实现内部 RC 自动校准功能, 烧录时自动校准内部 RC 精度到+/-2%以内;
- · 熔丝位、加密位设置设置: 支持 ATMEL 标准 STK500 熔丝位、加密位设置模式;
- · 序列号设置设置: 两种可选格式的序列号设置模式;
- ·升级支持:对于新推出的 AVR 产品,315avr 会及时添加到可脱机 USB AVRISP 下载器中;
- · 软件支持: AVR Programmer V1.26-2007.8,315avr 网站自主开发的上位机软件;

2. 连接可脱机 USB AVRISP 下载器到 PC 机

确认 AVR Programmer Software V1.26-2007.8 软件安装正确后,通过套件中匹配的 USB 转接线连接 可脱机 USB AVRISP 下载器到 PC 机,如下图一。

可脱机 USB AVRISP 下载器由 PC 机 USB 供电。

图一: 连接 PC 机后的 LED 指示绿灯亮



3. 安装 USB 驱动

- 若是第一次连接可脱机 USB AVRISP 下载器到 PC 机,系统提示需要安装 USB 驱动,显示如下图二;
- 依次执行图二、三、四、五、六, 手动安装 USB 驱动步骤;
- 以上步骤完成后,系统会提示安装另一个驱动,重复上述步骤,再安装一次;
- 启动软件 AVR Programmer Software V1.26–2007.8,显示如图七,即可进行你需要的编程操作。

图二:手动安装 USB 驱动

找到新的要件向导	
	欢迎使用找到新硬件向导
	Windows 将通过在计算机、硬件安装 CD 或 Windows Update 网站(在您允许的情况下)上查找来搜索当前和更 新的软件。 巍]][[]]]私策略]
	Windows 可以连接到 Windows Update 以搜索软件吗? ② 是,仅这一次(L) ③ 是,这一次和每次连接设备时(L) ④ 否,暂时不(L)
and the second second	单击"下一步"继续。
	<上一步(B) 下一步(B) > 取消

图三:选择从列表和指定位置安装驱动



图四:

注: 在安装软件安装目录下找到该驱动文件夹,如本机安装在 C:/盘,则路径为: C:\Program Files\AVR_Tool\AVR_Programmer\Drvier,如图四选择正确。

找到新的硬件向导		
请选择您的搜索和3	安装选项。	
● 在这些位置上 使用下列的复 到的最佳驱动。	搜索最佳驱动程序(S)。 先框限制或扩展默认搜索,包括本机路径和可移动媒体。会错 程序。	安装找
□搜索可	浏览文件夫 🔹 ? 🔀	
☑在搜索 C:\Pro	选择包含您的硬件的驱动程序的文件夹。	
○不要搜索。剩	🕀 🧰 ArcSoft	
选择这个选项 动程序与您的	🔁 ATAK5278	的服
	E Cavr	
	AVR_Tool	
		IT VIE
	🕀 🛅 BitComet	K (H
置文件(2)	要查看任何子文件夹,请单击上面的 + 号。	~
		time.
	确定即消	Contraction of the local division of the loc

图五:选择好正确的安装驱动路径

找到新的现件向导
请选择您的搜索和安装选项。
 ● 在这些位置上搜索最佳驱动程序(S)。 使用下列的复选框限制或扩展默认搜索,包括本机路径和可移动媒体。会安装找到的最佳驱动程序。 ● 搜索可移动媒体(软盘、CD-ROM)(M) ● 在搜索中包括这个位置(D): C:\Program Files\AVR_Tool\AVR_Programmer\I () 浏览(B) ● 不要搜索。我要自己选择要安装的驱动程序(D)。 选择这个选项以便从列表中选择设备驱动程序。Windows 不能保证您所选择的驱动程序与您的硬件最匹配。
〈上一步 @)下一步 @)〉 取消

图六:完成驱动安装

找到新的硬件向导	
	完成找到新硬件向导 该向导已经完成了下列设备的软件安装: AVR USB Composite Device
	要关闭向导, 请单击"完成"。 (上一步 (b) 完成 取消

图七:软件正确启动界面

AVR_Programmer V1.26 2007.8	
File Edit Config Operation Help	
ReLoad ID Erase Read ProFlash Pro EE ProFuse Help	
COM: USB V Device: ATMEGA16 V Check Sum: 0x61ef	Auto Program Options
	Fuse RC S/N
ATMEGA16 Adapter: ADP_M16_D(DIP),ADP_M16_S(SOIC),ADP_M16_M(MLF)	₩ Erase Device
Buffer Check Sum: 0x61ef	🗹 ID Check
	🔽 Program Flash
	🗖 Program Eeprom
	🔽 Program Fuse & Lock
	🗖 Calibrated Internal RC osc.
	₩rite SN
	Config Disable 🔽
	8
	Off line Auto
	www.315avr.com

4. ISP 编程操作说明(在上位机软件 AVR Programmer Software V1.26-2007.8 中进行编程操作)

a) LED (电源指示等说明)

可脱机 USB AVRISP 下载器使用一个双色 LED 等指示各种编程状态。 USB 上电: 红灯闪烁一次,绿灯常亮; 选择软件中的端口 USB,软件检测到可脱机 USB AVRISP 下载器,绿灯常亮; ISP 编程过程中:绿灯闪烁; ISP 编程 OK:绿灯常亮; ISP 编程出错:红灯常亮;

b) Connect (连接)

AVR Programmer Software V1.26—2007.8 软件安装正确后,通过套件中匹配的 USB 转接线连接可脱机 USB AVRISP 下载器到 PC 机,打开软件,如上图七界面。

手动选择 COM 端口 "USB",软件提示框会有相应文字提示,显示软件是否已经自动检测到"可脱机 USB AVRISP 下载器"。

AVR 单片机都带有 Flash 存储器,所有 AVR 器件也都支持在线编程(ISP)。通过 ISP 可以修改 MCU 的程序 存储器,EEPROM,熔丝,加密位等。 AVR 芯片 ISP 编程接口:



可脱机 USB AVRISP 下载器接口为上图中 ATMEL 标准 10pins ISP 编程接口,与双龙并行下载线的输出接口一致。为满足不同需求,出售套件中配有 10pin-6pin 转接板。在与目标板连接时请注意脚位是否对应正确。

正确连接可脱机 USB AVRISP 下载器到 AVR 芯片相应 ISP 接口后,即可在软件中进行编程操作。

c) 在 PC ISP 编程 (可脱机 USB AVRISP 下载器连接到 PC 机编程操作)

具体单项编程操作请参考上位机软件 "AVR Programmer Software V1.26—2007.8"使用说明。以下讲解 软件界面右侧的 "Auto Program Options (自动编程设置)"

如果需要批量 ISP 编程 AVR 芯片,确保您需要的设置完成后(如: Flash, EEPROM, Fuse & lock bits, RC, S/N),即可点击如下图中的 "Auto" 按钮,进行批量 ISP 自动编程。

Auto Progra	am Optio	ns
Fuse	RC	S/N
🔽 Erase 🛛)evice	
团 ID Cheo	:k	
🔽 Program	n Flash	
🗆 Program	n Eepror	n
🗆 Program	n Fuse 8	Lock
🗆 Calibra	ted Inter	nal RC osc.
Write S	N	
	Config D	isable 🗆
	1	8
Off_line		Auto

d) Offline Program (脱机编程)

如果需要批量 ISP 编程 AVR 芯片,建议最好是采用"脱机编程"方式,因为快捷、安全。确保您需要的 设置完成后(如: Flash, EEPROM, Fuse & lock bits, RC, S/N),即可点击如上图中的"Off-line"按钮, 下载您所有的设置信息到"可脱机 USB AVRISP 下载器"中。

由此, 脱离 PC 机, 使用出售套件中的 USB 电源给"可脱机 USB AVRISP 下载器"供电, 只需要操作编程器上的"RUN"按钮, 即可进行批量脱机 ISP 自动编程。