

ATxmega64A3 的 bootloader 设计

文档编号	MAN0003A_CH				
文档版本	Rev. A				
文档摘要	ATxmega64A3 的 Bootloader 设计，在没有仿真器的情况下通过串口下载用户应用程序				
关键词	ATxmega、Bootloader、mkII-CN、AVRSTUDIO、调试、串口下载				
创建日期	2009-12-17	创建人员	Robin	审核人员	Hotislandn
文档类型	公开发布/开发板配套文件				
版权信息	Mcuzone 原创文档，转载请注明出处				

更新历史

版本	时间	更新	作者
Rev. A	2009-12-17	初始创建	Robin

www.mcuzone.com

微控电子 乐微电子
杭州市登云路 639 号 2B143
销售 TEL: +86-571-88908193 13957118045
支持 TEL: 13957118045 18913989166 13770507096
FAX: +86-571-88908193
www.mcuzone.com www.atarm.com

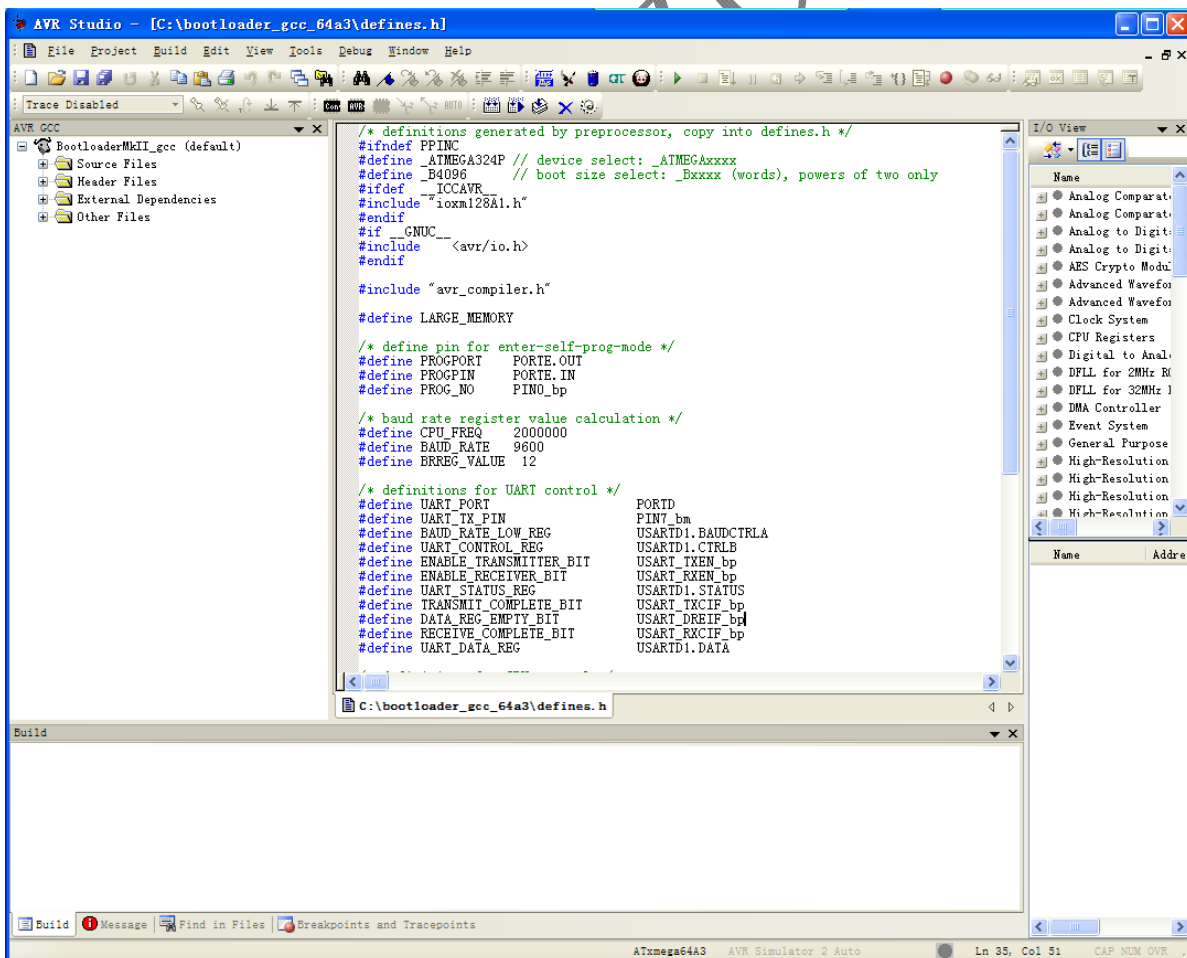
1. 概述

ATxmega 是 ATMEL 推出的一款全新 MCU，与之前的 AVR 相比速度更快，ADC 和 DAC 性能更佳，功耗更低，而且有多达 8 个串口，因此在工控领域有较大用途。目前支持 ATxmega 的仿真器和下载器都屈指可数，虽然 Mcuzone 的 mkII-CN 价格已经很低，但是还有不少人希望有更低成本的方式来学习和评估 xmega。由于 xmega 系列芯片有独立的 bootloader 区段，因此可以用此 bootloader 来实现升级等功能，与早期 AVR 不同的是 xmega 的 bootloader 区是独立于用户 FLASH 的，因此可以更简单的实现程序更新和切换以及代码保护。ATMEL 有提供基于 ATxmega128A1 的 bootloader 参考例子，我们在此基础上将其移植到了 ATxmega64A3 上，并且在本站的 ATxmega64A3 最小系统板上验证通过。下面简要讲述一下移植的过程和 bootloader 的使用。

2. 移植 bootloader

2.1 修改参数

ATxmega128A1 和 ATxmega64A3 的主要差别在于 FLASH 和 SRAM 的大小以及串口的编排，其它基本一致。所以我们只需要修改 FLASH 的 pagesize 的大小和串口编号即可，为了方便 ATxmega64A3 进入 bootloader，我们修改了用来控制进入 bootloader 状态的引脚，修改为 PEO。

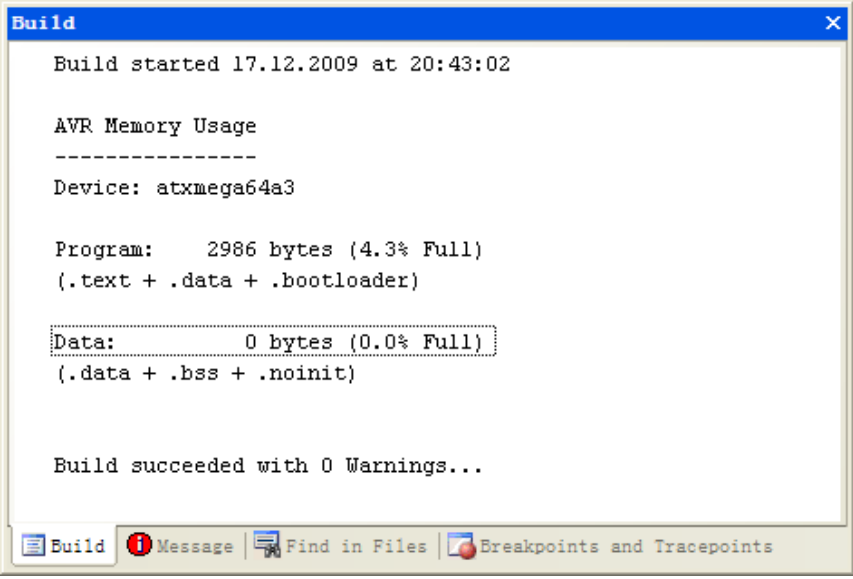


```
AVR Studio - [C:\bootloader_gcc_64a3\defines.h]
File Project Build Edit View Tools Debug Window Help
Trace Disabled
AVR GCC
BootloaderMkII_gcc (default)
Source Files
Header Files
External Dependencies
Other Files

/* definitions generated by preprocessor, copy into defines.h */
#ifndef PPINC
#define _ATMEGA324P // device select: _ATMEGAxxxx
#define _B4096 // boot size select: _Bxxxx (words), powers of two only
#endif
#include "ioxm128a1.h"
#endif
#if __GNUC__ < avr/io.h
#endif
#include "avr_compiler.h"
#define LARGE_MEMORY
/* define pin for enter-self-prog-mode */
#define PROGPORT PORTE.OUT
#define PROGPIN PORTE.IN
#define PROG_NO PINO_bp
/* baud rate register value calculation */
#define CPU_FREQ 2000000
#define BAUD_RATE 9600
#define BRREG_VALUE 12
/* definitions for UART control */
#define UART_PORT PORTD
#define UART_TX_PIN PINT_bm
#define BAUD_RATE_LOW_REG USARTD1.BAUDCTRLA
#define UART_CONTROL_REG USARTD1.CTRLB
#define ENABLE_TRANSMITTER_BIT USART1_TXEN_bp
#define ENABLE_RECEIVER_BIT USART1_RXEN_bp
#define UART_STATUS_REG USARTD1.STATUS
#define TRANSMIT_COMPLETE_BIT USART1_TXCIF_bp
#define DATA_REG_EMPTY_BIT USART1_DREIF_bp
#define RECEIVE_COMPLETE_BIT USART1_RXCIF_bp
#define UART_DATA_REG USARTD1.DATA

Build
C:\bootloader_gcc_64a3\defines.h
Ln 35, Col 51 CAP NUM OVR
```

编译结果:



```
Build
-----
Build started 17.12.2009 at 20:43:02

AVR Memory Usage
-----
Device: atxmega64a3

Program: 2986 bytes (4.3% Full)
(.text + .data + .bootloader)

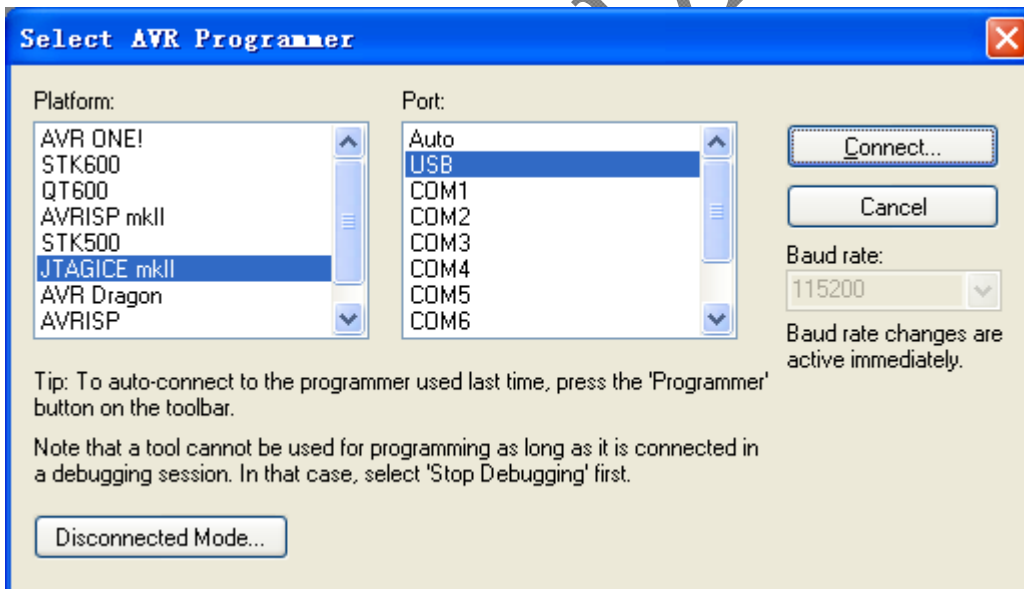
Data: 0 bytes (0.0% Full)
(.data + .bss + .noinit)

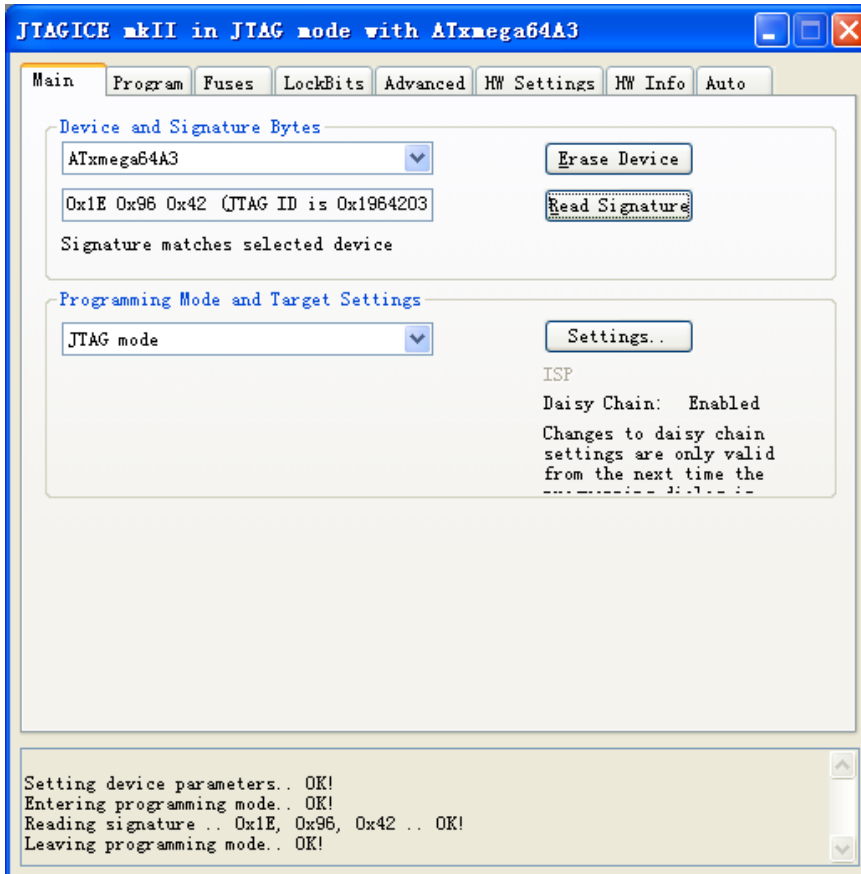
Build succeeded with 0 Warnings...
```

编译生成的 hex 可以在 Mcuzone 网站直接下载到。

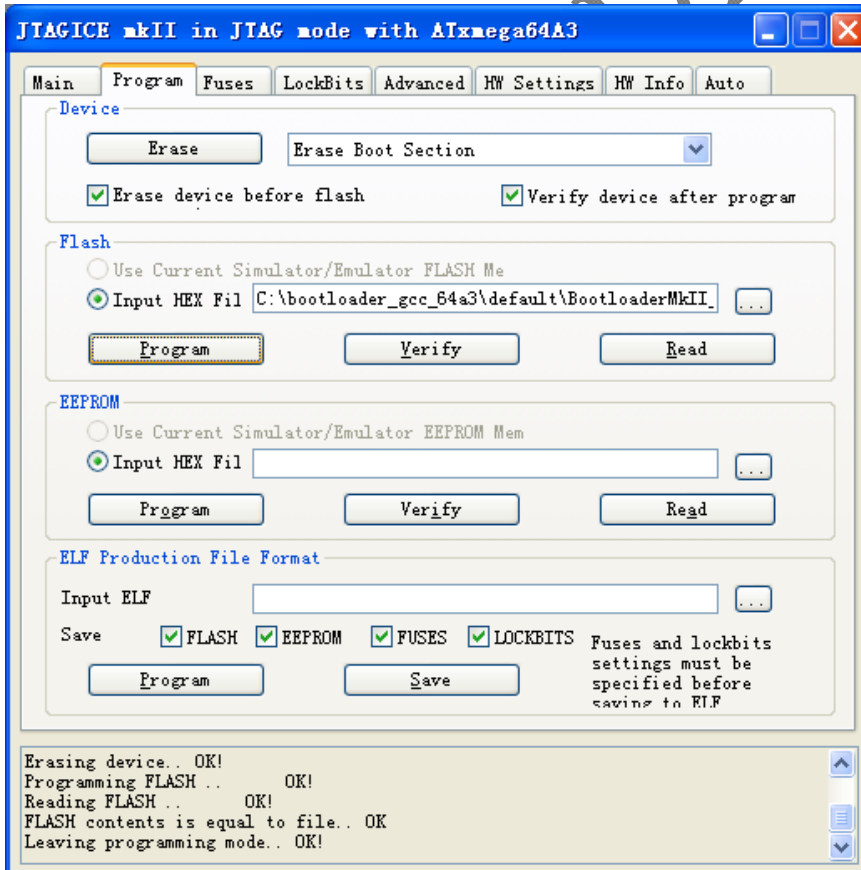
2.2 烧写 bootloader

我们使用本站的 mkII-CN 来烧写 bootloader，首先采用 JTAG 接口来烧写，请将 mkII-CN 的 10 芯线直接插入到 ATxmega64A3 最小系统板的 JTAG 座子，然后打开 AVRSTUDIO 进行连接：

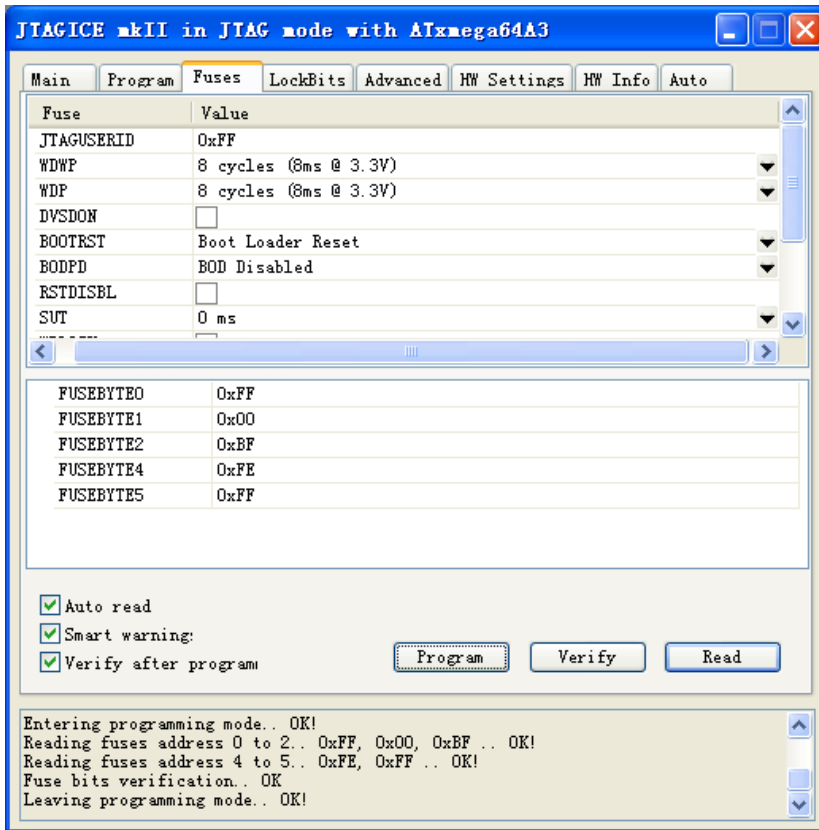




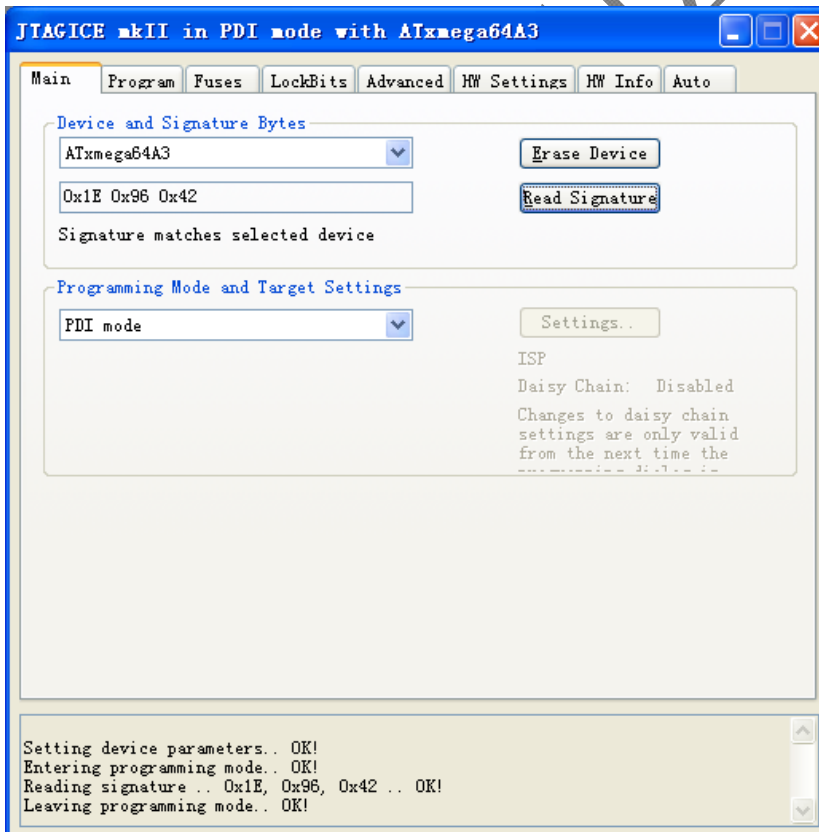
然后进入 Program 选项卡进行 FLASH 编程:



编程结束修改 FUSE，修改为 Boot Loader Reset。



以上是 JTAG 接口烧写，我们也可以用 PDI 接口进行烧写，请将 mkII-CN 的 10 芯线直接插入到 ATxmega64A3 的 PDI 座子。



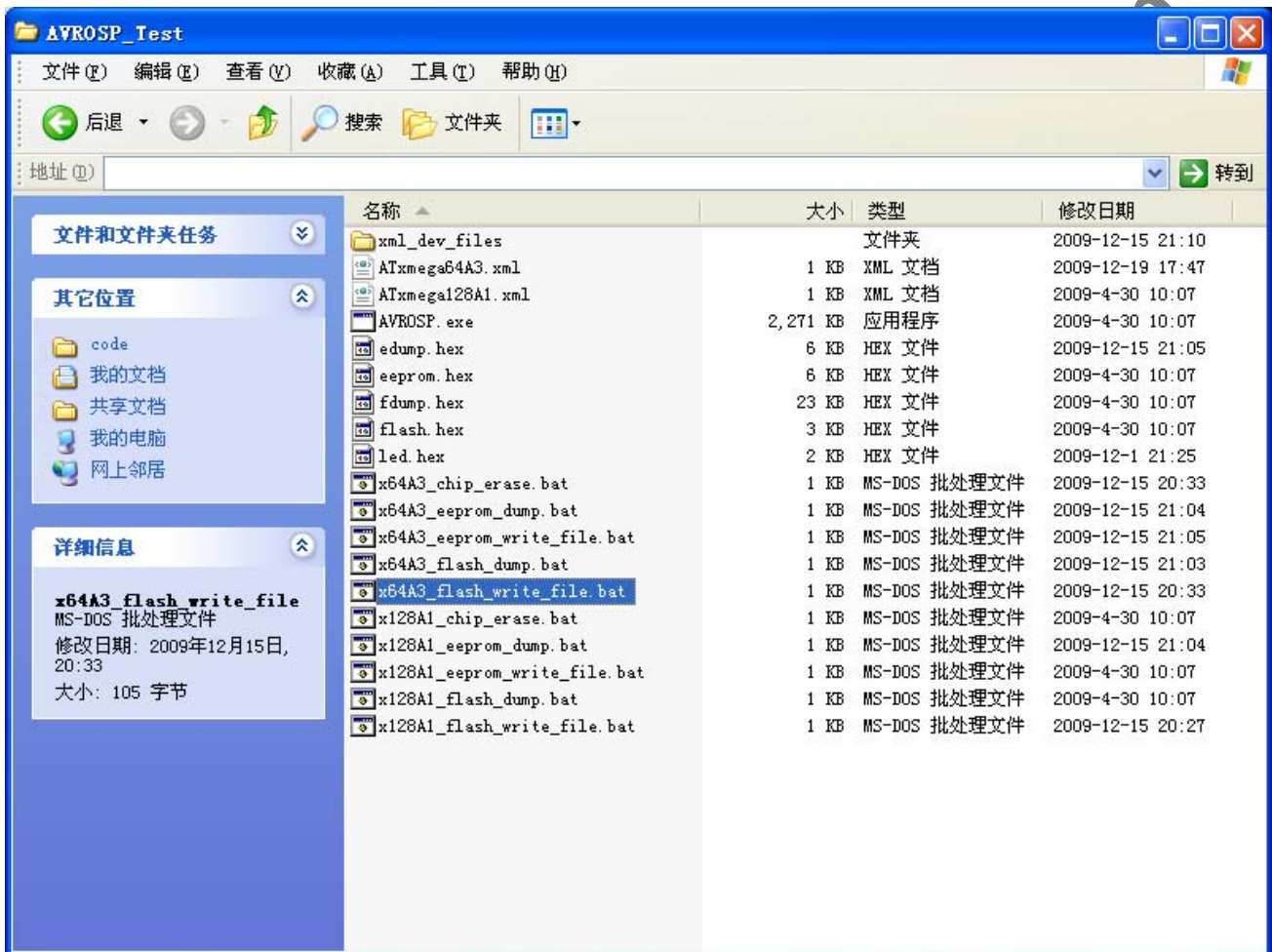
3. 通过 bootloader 更新用户程序

下面以一个简单的 LED 跑马程序为例演示如何通过 bootloader 更新用户程序。

首先了解一下烧写软件 AVROSP，所谓 AVROSP 是指 AVR Open Source Programmer，如果有兴趣了解更多细节，请参阅 ATMEL 的应用笔记 AVR911 和 AVR1605。

下面我们开始演示用户程序更新过程。

进入 AVROSP_Test 文件夹目录：



可以看到有很多 BAT 文件，分成 ATxmega64A3 和 ATxmega128A1 两个 CPU，每个 CPU 平台分别有 5 个 BAT，用户可以逐一尝试，下面我们仅以更新程序 BAT 为例。

在更新应用程序前我们需要了解一下进入 boot 的条件。之前我们已经将 PE0 设置成进入 boot 的控制引脚，在上电后如果 PE0 引脚为低电平，进入更新应用程序状态；如果 PE0 引脚为高电平，则执行应用程序。下面我们是要更新应用程序，所以请将 PE0 引脚跳线到 GND，然后上电。

注意：AVROSP 软件只查找 COM1-COM9，而 ATxmega64A3 最小系统板上使用的 USB 转串口芯片 FT232，如果用户电脑接过很多 USB 转串口设备，则有可能串口编号在 COM9 之后，这个时候请修改串口号到 COM9 之前！

双击运行其中的 x64A3_flash_write_file.bat，运行结果如下：



可以看到 led.hex 文件已经被正确写入到 FLASH。

如何确认程序已经正确写入并能正确执行呢？请将 PE0 的跳线短接至 VCC，然后重新上电，可以看到 PDI

座子边上的两个 LED 指示灯 D1 和 D2 开始闪烁，说明应用程序更新（升级）成功！

如果要再次升级用户程序，只需要将 PE0 跳线接到 GND，然后重新上电并运行 x64A3_flash_write_file.bat。该文件的源码：

```
mode com2 Data=8 Parity=n Baud=9600 DTR=OFF RTS=OFF
```

```
AVROSP -dATxmega64A3 -e -ifled.hex -pf -vf
```

```
Pause
```

用户可以修改串口编号，修改需要更新的文件名。

www.mcuzone.com

AVR JTAGICE mkII-CN 仿真器:



ATxmega64A3 最小系统板:

