

OMAP-L138_FlashAndBootUtils 使用及编译指导

吴红军

TI 通用数字信号处理系统技术支持

摘要

本文介绍 OMAP-L138_FlashAndBootUtils 工具包基本框架和串口烧写工具 sfh_OMAP-L138.exe 工作原理，使用方法，以及针对客户的硬件如何修改代码，搭建编译环境并重新编译，并总结了使用该工具的常见问题。

关键字：OMAPL138, 串口烧写, UBL

内容

简介 2

1. 串口烧写工具 sfh_OMAP-L138.exe 的工作原理	3
2. 命令格式介绍.....	3
3. sfh_OMAPL138.exe 常用命令的使用	4
3.1 全局擦除 flash 命令	4
3.2 烧写单个 AIS 格式的应用程序	5
3.3 烧写 UBL 和 UBoot	6
4. 根据硬件改动修改 serial flash utility	8
4.1 DDR 参数修改.....	8
4.2 SPI flash	8
4.3 NAND flash.....	8
4.4 NOR flash.....	9
4.5 时钟及 PLL	9
4.6 UART 配置.....	9
5. OMAP-L138_FlashAndBootUtils 编译环境搭建	9
5.1 Cygwin 安装.....	9
5.2 .NET framework.....	11
5.3 C6x Compiler Tools	12
5.4 ARM Compiler tools(CodeSourcery G++ Lite)	12
5.5 CCSV5	12
6. 编译 OMAP-L138_FlashAndBootUtils	12
6.1 查看环境变量	12
6.2 编译工具包	13
7. 使用 serial flash utility 常见问题及解决方法	13
7.1 收不到“BOOTME”	13
7.2 停在 Waiting for SFT on the OMAP-L138.....	13
7.3 烧写中途停止	14
7.4 启动后串口没打印	14

图

图 1	S7 开关设置.....	4
图 2	sfh_OMAPL138.exe -erase 命令运行界面	5
图 3	AIsgen 配置界面.....	5
图 4	sfh_OMAPL138.exe -flash_noubl 命令运行界面	6
图 5	从 UBoot 直接运行界面	6
图 6	sfh_OMAPL138.exe -flash 命令运行界面	7
图 7	S7 开关设置为 SPI Flash 启动	7
图 8	终端调试信息输出.....	8
图 9	cygwin 安装界面 1	10
图 10	cygwin 安装界面 2	10
图 11	cygwin 安装界面 3	10
图 12	cygwin 安装界面 4	10
图 13	cygwin 安装界面 4	11
图 14	windows 环境变量设置	12
图 15	sfh_OMAP-L138.exe 命令运行异常	13
图 16	sfh_OMAP-L138.exe 命令运行异常信息	14

表

表 1	OMAP-L138_FlashAndBootUtils_2_40 工具包目录结构	3
表 2	sfh_OMAPL138.exe 常用命令表.....	4

简介

OMAP-L138 支持多种启动模式，包括 SPI，NAND，NOR 等。TI 为用户提供了两套 flash 烧写工具：

- (1) 使用 TI 在 PSP 包或 OMAP-L138_FlashAndBootUtils 包中提供的 flash writer 的 CCS 工程，通过仿真器连接硬件板，按照提示步骤烧写 flash。
- (2) 使用 OMAP-L138_FlashAndBootUtils 工具包中的串口烧写工具（如 sfh_OMAP-L138.exe），通过串口连接，进行命令行烧写。

本文接下来将针对第二种方式进行详细介绍。

用户可以从开源网站上下载最新的 OMAP-L138_FlashAndBootUtils 版本^[1]。最新的版本为 OMAP-L138_FlashAndBootUtils_2_40，本文以此版本展开讨论。

OMAP-L138_FlashAndBootUtils_2_40 工具包支持 TI 公司的多种芯片平台的 flash 烧写，包括 OMAPL138，AM1808，C6748 等。工具包内的代码是对应 LogicPD 的 OMAL138/AM1808 EVM 板的。如果用于用户自己的硬件板，可能需要修改工具包中相应代码并重新编译。

该工具包中包括多种 flash 工具及代码。主要目录如表 1 所示。

表 1 OMAP-L138_FlashAndBootUtils_2_40 工具包目录结构

根目录	主要子目录及文件	主要目录内容
Common	arch	
	drivers	
	ubl	
OMAP-L138	CCS	flash writer 和 ubl 的 ccs 工程源码包
	Common	
	GNU	sfh_OMAP-L138.exe 等命令行工具

1. 串口烧写工具 sfh_OMAP-L138.exe 的工作原理

sfh_OMAP-L138 是 OMAP-L138_FlashAndBootUtils 包中用于实现串口烧写 flash 的应用程序。与 UartHost.exe^[3]工具不同，UartHost.exe 是 UART 启动工具，不能烧写 flash。

运行 sfh_OMAP-L138.exe 时，通过 UART 启动方式下载并运行 OMAPL138/GNU/sft 目录下的 AIS 格式 bin 文件，完成芯片的初始化操作。然后通过 UART 与 sfh 进行命令交换，实现 flash 的擦除和烧写工作。

从 OMAP-L138/GNU/sft/build 目录中的 makefile 文件可知，编译 sft 目录内各 bin 文件时需要 sft.c，device.c，uart.c 以及针对不同 flash 对应的 flash 初始化文件。

例如 sft_OMAPL138_SPI_MEM.bin 是由 sft.c，device.c，uart.c 等文件与 spi.c，spi_mem.c，device_spi.c 一起编译，并通过 OMAP-L138/GNU/AISUtils 目录下的 HexAIS_OMAP-L138.exe，根据 sft 目录下的配置文件 sft_hexais.ini 内的配置信息，将.out 文件转换成 AIS 格式的.bin 文件。

2. 命令格式介绍

在命令行下输入 sfh_OMAP-L138.exe -help 输出命令格式说明如下：

sfh_OMAP-L138 <Command> [-targetType <Target>] [-flashType <FlashType>] [<Options>] [<InputFiles>]

各参数说明如下表所示，targetType 默认为 OMAP-L138，FlashType 默认为 SPI_MEM。使用默认选项时相应参数可以不输入。

Option	
command	-erase Global erase of the flash memory device (no input files) -flash_noubl Place single bootable image in the flash memory device (single input file) -flash Place a secondary user boot loader (UBL) and application image in the flash memory device (first input file is UBL binary, second input file is the binary application image) -flash_dsp Place a DSP secondary user boot loader (UBL), an ARM secondary user boot loader and the application image in the flash memory device (first input file is DSP UBL binary, the second file is ARM UBL binary, and the third input file is the binary application image). This is only used for OMAP-L137 devices.
-targetType	Specifies exact target type within OMAP-L138 family (default OMAP-L138) For C6748 devices, use the following option: "-targetType C6748" For AM1808 devices, use the following option: "-targetType AM1808"
-flashType	Specifies exact flash type (default SPI_MEM)

	Currently, the only supported flash types are NAND, NOR, and SPI.
-p <COM PORT NAME>	Allows specifying com port other than default 'COM1' or '/dev/ttyS0'
Option can be the following:	
-h	Show help text
-v	See verbose output from target device
-baud <BAUD RATE>	Allows specifying baud rate other than default (115200)
-APPStartAddr	Changes entry point of application (default 0xC1080000)
-APPLoadAddr	Changes load address of application (default 0xC1080000)
-APPFlashBlock	Changes the block to flash the image into (only for no_ubl mode)

3. sfh_OMAPL138.exe 常用命令的使用

sfh_OMAPL138.exe 常用命令如表 2 所示。

表 2 sfh_OMAPL138.exe 常用命令表

序号	命令格式	含义
1	sfh_OMAPL138.exe -erase	擦除 SPI flash 中的内容
2	sfh_OMAP-L138.exe -flash_noubl <binary application file>	烧写 AIS 格式的 bin 文件，如 uboot-ais.bin
3	sfh_OMAP-L138.exe -flash <UBL binary file> <binary application file>	烧写 ubl 和 uboot

以 SPI flash 启动为例，准备如下文件：

- sfh_OMAPL138.exe，来源于 OMAP-L138_FlashAndBootUtils_2_40 工具包。
- ubl-spi_ais.bin，来源于 OMAP-L138_FlashAndBootUtils_2_40 工具包。
- u-boot.bin，来源于 SDK PSP 目录下的 UBoot。
- u-boot-ais.bin，由 UBoot 编译出的 elf 格式文件 u-boot，通过 AIS 转换工具生成。

首先将 OMAL138/AM1808 EVM 板设置为 UART2 启动模式，S7 开关设置如图 1 所示。用直连串口线连接 PC 与 EVM 板，打开 EVM 板电源。在命令行模式下进入目录：OMAP-L138_FlashAndBootUtils_2_40\OMAP-L138\GNU。

Pin#	1	2	3	4	5	6	7	8
Position	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON

图 1 S7 开关设置

3.1 全局擦除 flash 命令

sfh_OMAPL138.exe -targetType OMAP-L138 -flashType SPI_MEM -erase

```

Attempting to connect to device COM1...
Press any key to end this program at any time.

(AIS Parse): Read magic word 0x41504954.
(AIS Parse): Waiting for BOOTME... (power on or reset target now)
(AIS Parse): BOOTME received!
(AIS Parse): Performing Start-Word Sync...
(AIS Parse): Performing Ping Opcode Sync...
(AIS Parse): Processing command 0: 0x58535901.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Loading section...
(AIS Parse): Loaded 9012-Byte section to address 0x80000000.
(AIS Parse): Processing command 1: 0x58535901.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Loading section...
(AIS Parse): Loaded 752-Byte section to address 0x80002334.
(AIS Parse): Processing command 2: 0x58535906.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Performing jump and close...
(AIS Parse): AIS complete. Jump to address 0x80000000.
(AIS Parse): Waiting for DONE...
(AIS Parse): Boot completed successfully.

Waiting for SFT on the OMAP-L138...

Erasing flash
  7% [██████████]-----]
                                Erasing chip
    
```

图 2 sfh_OMAPL138.exe -erase 命令运行界面

运行结果如图 2 所示，如果是先上电，后执行该命令，则需要按下板子上的 reset 键输出“BOOTME”给上位机的 sfh。

3.2 烧写单个 AIS 格式的应用程序

sfh_OMAP-L138.exe -targetType OMAP-L138 -flashType SPI_MEM -flash_noubl u-boot-ais.bin

OMAP-L138 的 RBL 支持 AIS 格式，可以在启动过程中通过 AIS 格式文件中的配置信息初始化芯片，如 PLL，DDR 等。这样就可以直接将程序加载到片外内存，而不受片上内存大小限制，省略二级 Boot。

编译 UBoot 时，在 UBoot 根目录下会生成一个 elf 格式的 u-boot 文件，如图 3 所示，在 AISgen.exe 工具里添加 PLL，DDR 等配置参数，将此文件转换成 AIS 格式的 bin 文件，可以直接烧写并启动。

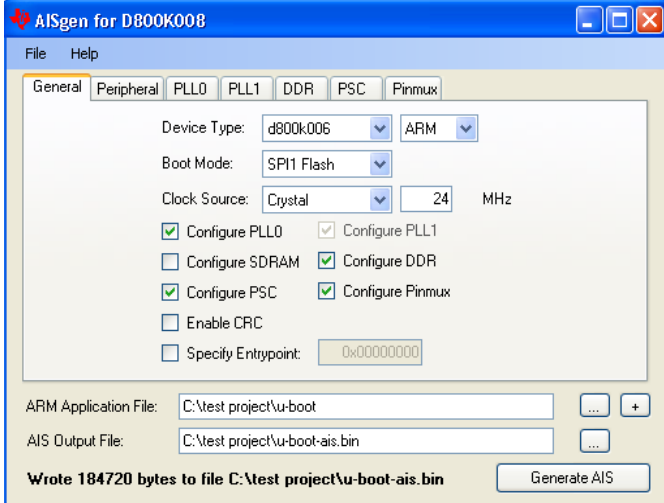


图 3 AISgen 配置界面

该命令将 AIS 格式的 UBoot 文件写到 flash 的起始地址。命令执行显示结果如图 4 所示：重新上电，从图 5 启动信息可见没有 UBL 的运行信息，是直接从 UBoot 运行的。

```
(AIS Parse): Processing command 1: 0x58535901.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Loading section...
(AIS Parse): Loaded 736-Byte section to address 0x800023F0.
(AIS Parse): Processing command 2: 0x58535906.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Performing jump and close...
(AIS Parse): AIS complete. Jump to address 0x80000000.
(AIS Parse): Waiting for DONE...
(AIS Parse): Boot completed successfully.

Waiting for SFT on the OMAP-L138...

Flashing application u-boot-ais.bin (171660 bytes)

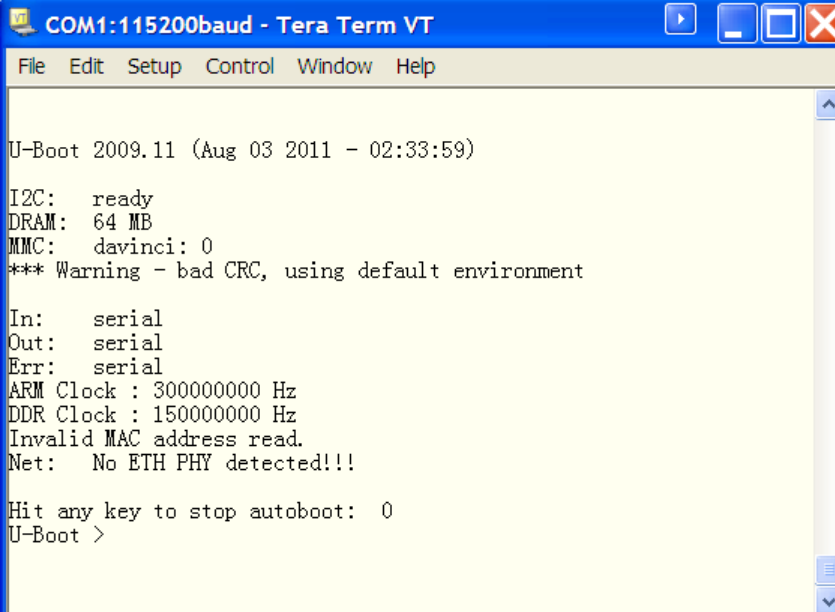
100% [ ████████████████████████████████████████████████████████████████████████████ ]
        Image data transmitted over UART.

100% [ ████████████████████████████████████████████████████████████████████████████ ]
        Application programming complete

Operation completed successfully.
```

图 4 sfh_OMAPL138.exe -flash_noubl 命令运行界面

重新上电，从图 5 启动信息可见没有 UBL 的运行信息，是直接从 UBoot 运行的。



```
COM1:115200baud - Tera Term VT
File Edit Setup Control Window Help

U-Boot 2009.11 (Aug 03 2011 - 02:33:59)

I2C: ready
DRAM: 64 MB
MMC: davinci: 0
*** Warning - bad CRC, using default environment

In: serial
Out: serial
Err: serial
ARM Clock : 300000000 Hz
DDR Clock : 150000000 Hz
Invalid MAC address read.
Net: No ETH PHY detected!!!

Hit any key to stop autoboot: 0
U-Boot >
```

图 5 从 UBoot 直接运行界面

3.3 烧写 UBL 和 UBoot

```
sfh_OMAP-L138.exe -targetType OMAP-L138 -flashType SPI_MEM -flash ubl-spi-ais.bin u-boot.bin
```

命令执行结果显示如图 5 所示。该命令将 ubl-spi-ais.bin 写到 flash 的起始地址，u-boot.bin 写到 flash 的 0x10000 的偏移地址。

```

(AIS Parse): BOOTME received!
(AIS Parse): Performing Start-Word Sync...
(AIS Parse): Performing Ping Opcode Sync...
(AIS Parse): Processing command 0: 0x58535901.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Loading section...
(AIS Parse): Loaded 8704-Byte section to address 0x80000000.
(AIS Parse): Processing command 1: 0x58535901.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Loading section...
(AIS Parse): Loaded 824-Byte section to address 0x80002200.
(AIS Parse): Processing command 2: 0x58535906.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Performing jump and close...
(AIS Parse): AIS complete. Jump to address 0x80000000.
(AIS Parse): Waiting for DONE...
(AIS Parse): Boot completed successfully.

Waiting for SFT on the OMAP-L138...

Flashing UBL ubl-spi_ais.bin (8672 bytes) at 0x00000000

100% [ ██████████ ] Image data transmitted over UART.

100% [ ██████████ ] UBL programming complete

Flashing application u-boot.bin (156892 bytes)

100% [ ██████████ ] Image data transmitted over UART.

100% [ ██████████ ] Application programming complete

Operation completed successfully.
    
```

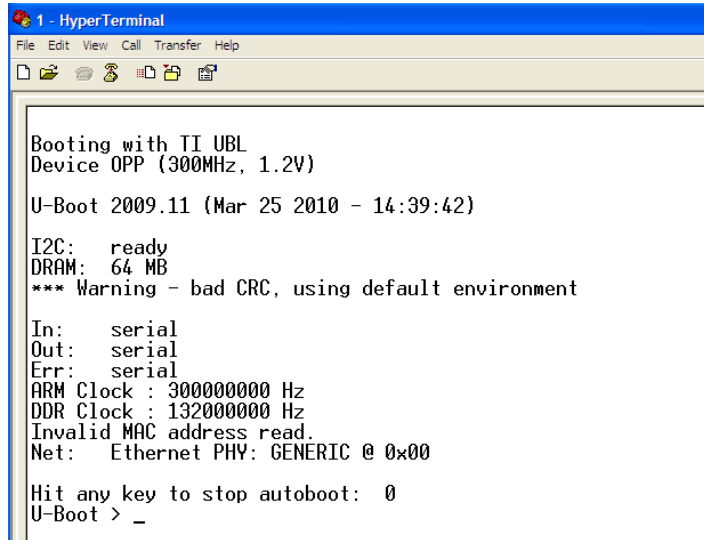
图 6 sfh_OMAPL138.exe -flash 命令运行界面

UBL 和 UBoot 烧写完后，关闭 EVM 板电源，按图 6 所示，将 EVM 板的 S7 开关设为 SPI flash 启动。

Pin#	1	2	3	4	5	6	7	8
Position	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF

图 7 S7 开关设置为 SPI Flash 启动

打开串口终端如 Hyper terminal，打开 EVM 板电源，串口终端输出信息如图 7 所示，表示 UBL 和 UBoot 成功烧写，并成功从 SPI flash 启动



```

1 - HyperTerminal
File Edit View Call Transfer Help

Booting with TI UBL
Device OPP (300MHz, 1.2V)

U-Boot 2009.11 (Mar 25 2010 - 14:39:42)

I2C: ready
DRAM: 64 MB
*** Warning - bad CRC, using default environment

In: serial
Out: serial
Err: serial
ARM Clock : 300000000 Hz
DDR Clock : 132000000 Hz
Invalid MAC address read.
Net: Ethernet PHY: GENERIC @ 0x00

Hit any key to stop autoboot: 0
U-Boot > _

```

图 8 终端调试信息输出

使用 `sfh_OMAP-L138.exe` 烧写 UBL 和 UBoot 时，建议使用工具包自带的 UBL。因为 UBoot 的烧写地址，要与 UBL 的读取地址一致，这样 UBL 运行时，才能正确的读取 UBoot。工具包自带的 UBL 与烧写工具保证了这一点。其它版本的 UBL 需要用户自行确认。

4. 根据硬件改动修改 serial flash utility

开发者自己的硬件板与 TI 的 EVM 比较，会影响工具使用的不同部分可能存在于：

- (1) DDR 型号
- (2) FLASH 型号
- (3) 时钟频率
- (4) UART 配置

4.1 DDR 参数修改

EVM 板用的是 mDDR，大多数用户用的是 DDR2，需要修改 `OMAP-L138/Common/src/device.c` 文件中函数 `Uint32 DEVICE_ExternalMemInit()` 的 5 个 DDR 参数值。这个函数同样被 UBL 调用，如果不用 UBL，则要在 AISgen 界面里配置正确的 DDR 参数。

可使用 TI 提供的 DDR 寄存器配置工具^[5]。根据 DDR 手册进行 DDR 参数的计算，

4.2 SPI flash

如果更改了 SPI flash 的硬件连接，可在 `OMAP-L138/Common/include/device_spi.h` 文件中修改配置。

```

#define DEVICE_SPIBOOT_PERIPHNUM      (1)//选择 SPI1
#define DEVICE_SPI_UBL_HDR_OFFSET     (0*1024)//ubl 在 SPI flash 中的偏移地址
#define DEVICE_SPI_APP_HDR_OFFSET     (64*1024)//application image 在 SPI flash 中的偏移地址

```

如果更换了与 EVM 板不同的 SPI flash，必要时需要修改如下两个文件：

```

OMAP-L138/Common/drivers/include/Spi_mem.h
OMAP-L138/Common/drivers/src/Spi_mem.c

```

4.3 NAND flash

NAND flash 的数据位宽配置在 `OMAP-L138/Common/include/device_async_mem.h` 文件中定义：

```

#define DEVICE_ASYNC_MEM_NANDBOOT_BUSWIDTH (DEVICE_BUSWIDTH_8BIT)

```


如果 NAND flash 更换的与 EVM 上的不同，必要时需要修改如下两个文件
 OMAP-L138\Common\drivers\include\Device_nand.h
 OMAP-L138\Common\drivers\src\Device_nand.c

4.4 NOR flash

NOR flash 是标准设备，除了时序参数之外，一般不需要修改，NOR flash 相关代码文件如下：
 OMAP-L138_FlashAndBootUtils_2_40\Common\drivers\src\nor.c
 OMAP-L138_FlashAndBootUtils_2_40\Common\drivers\include\nor.h
 OMAP-L138\Common\src\device_async_mem.c
 OMAP-L138\Common\include\device_async_mem.h
 OMAP-L138_FlashAndBootUtils_2_40\Common\drivers\src\ async_mem.c

4.5 时钟及 PLL

如果外部输入时钟频率不是 24MHz，则需要 OMAP-L138\Common\src\device.c 文件中函数 DEVICE_PLL0Init()和 DEVICE_PLL1Init()的参数值。
 函数原型 Uint32 DEVICE_PLL0Init(Uint8 clk_src, Uint8 pllM, Uint8 prediv, Uint8 postdiv, Uint8 div1 ,Uint8 div3,

$$\text{Uint8 div7}) \text{ 中 } \text{pllM} \text{ 是 PLL 的倍频系数，主频计算公式为： } F_{CPU} = \frac{F_{OSCIN} * (\text{pllM} + 1)}{\text{prediv} + 1}$$

函数 DEVICE_PLL1Init()参数含义与 PLL0 的类同，为 DDR 提供时钟。

4.6 UART 配置

如果系统主频发生变化，相应的模块时钟也改变，调试串口的分频值需要做相应调整，否则串口输出会因为波特率不正确而出现乱码。根据不同的主频修改 OMAP-L138\Common\src\device_uart.c 中 DEVICE_UART_config 的 UART 时钟分频系数 divider 的值，得到合适的 UART 波特率。

如果不是用 UART2 做为调试串口，修改 OMAP-L138\Common\include\ device_uart.h 文件中的 DEVICE_UART_PERIPHNUM 定义。

5. OMAP-L138_FlashAndBootUtils 编译环境搭建

在根据硬件改动，修改了代码后，需要对工具包重新编译，生成 UBL 和 sfh_OMAP-L138.exe。如果在 Linux 下使用此工具，需要在 Linux 下进行编译（具体编译过程参考参考文献 4）。下面介绍如何在 Windows 环境下搭建 OMAP-L138_FlashAndBootUtils 的编译环境。

编译环境需要的几个工具如下^[6]：

- (1) cygwin
- (2) .NET framework
- (3) ARM Compiler tools(CodeSourcery G++ Lite)
- (4) C6x Compiler Tools
- (5) CCSV5

5.1 Cygwin 安装

从 [http:// www. cygwin. com/](http://www.cygwin.com/) 下载并运行在线安装工具 setup.exe，注意安装路径中不能带有空格。具体步骤如下：

- 1) 选择从互联网安装；也可以选择先下载，再从本地安装。

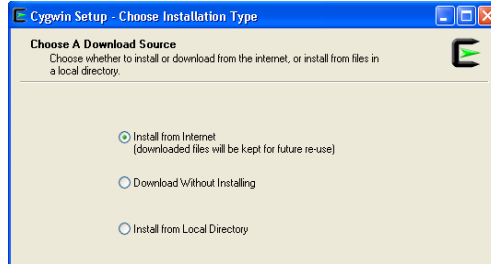


图 9 cygwin 安装界面 1

2) 选择安装路径。

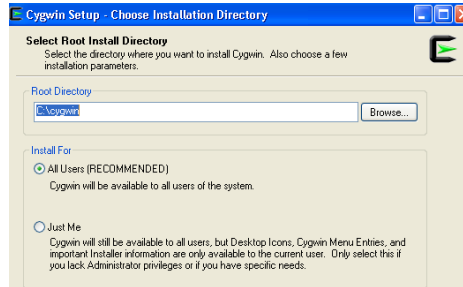


图 10 cygwin 安装界面 2

3) 可选择直接连接或使用代理。

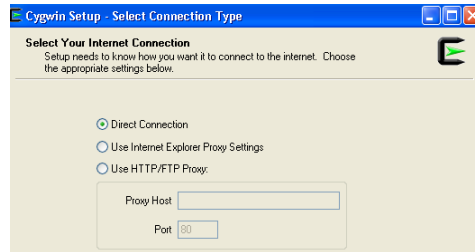


图 11 cygwin 安装界面 3

4) 选择从哪个网址或者服务器上下载。

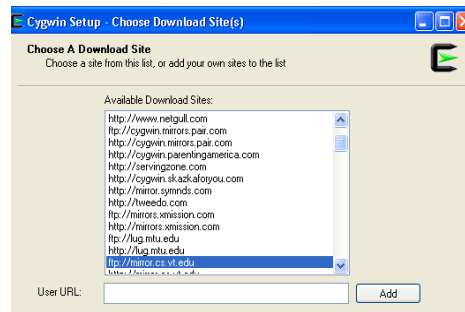


图 12 cygwin 安装界面 4

5) 选择安装包

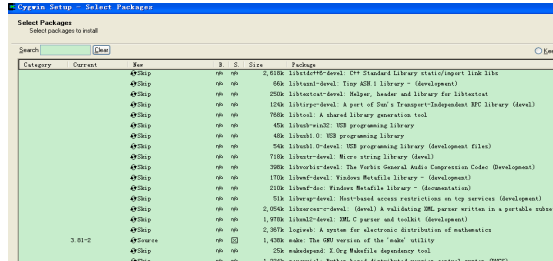


图 13 cygwin 安装界面 4

一般选择以下三项，其他默认即可。

- Devel-->make: The GNU version of the 'make' utility
- Devel-->subversion: A version control subsystem
- Editor-->vim (or similar)

安装后要先运行一次 cygwin 后，才会在 cygwin/home/下出现 Administrator 目录，进入该目录，修改.bashrc。

同时，需要验证变量 TMP 和 TEMP 是否包含/tmp。使用如下命令：

```
echo $TMP
echo $TEMP
```

如果找不到"unset TMP" and "unset TEMP"，可在cygwin\home\[user]\.bashrc 文件中添加如下命令创建：

```
export TMP=/tmp
export TEMP=/tmp
```

设置完后，重新打开cygwin即可。至此，cygwin安装完成。

5.2.NET framework

从下面网址下载安装最新的.NET Framework(4.0 或者更高的版本)

<http://www.microsoft.com/downloads/details.aspx?FamilyID=9cfb2d51-5ff4-4491-b0e5-b386f32c0992&displaylang=en>

安装默认路径为 C 盘。然后将 C#编译器路径添加到 windows 系统环境变量中。进入目录

C:\WINDOWS\Microsoft.NET\Framework 查看软件的版本号并复制，如

C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319。然后右键我的电脑->属性->高级->环境变量，如图 13 所示，选择系统变量中的 path 点击编辑，在原有变量值后粘帖添加刚刚复制的内容，路径之间以';'分隔。至此，Net Framework 安装结束。

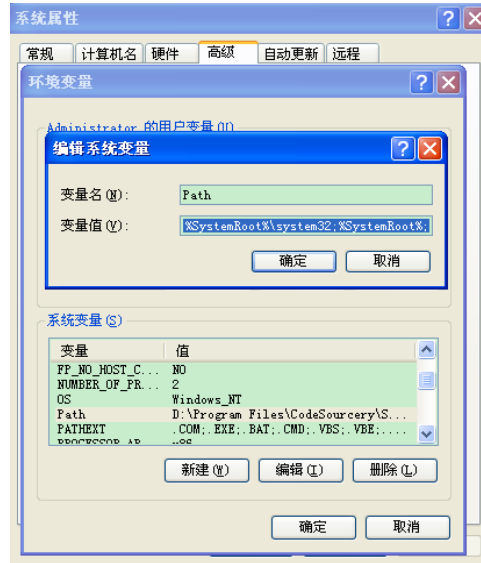


图 14 windows 环境变量设置

5.3 C6x Compiler Tools

在 CCSV5 安装目录下 `ccsv5\tools\compiler\c6000_7.3.4` 即为 C6x 编译器目录，在 CCSV3.3 下的路径为 `CCStudio_v3.3\C6000\cgtools\bin`。如果没有安装 CCS，从下面网址下载 TI 的 C6x 编译工具并安装，如不确认版本则下载最新版本。

https://www-a.ti.com/downloads/sds_support/TICodegenerationTools/download.htm

将 C6x 编译器路径添加到 windows 环境变量。

5.4 ARM Compiler tools(CodeSourcery G++ Lite)

从下面网址下载安装 GCC 交叉编译器。注意交叉编译工具的安装目录名不能含有空格，否则编译时会报错。

<https://sourcery.mentor.com/sgpp/lite/arm/portal/release858>

修改 `OMAP-L138_FlashAndBootUtils_2_40\Common\build.mak` 文件中的 `ARM_TOOLS_PATH` 变量为交叉编译器的安装路径。例如 `ARM_TOOLS_PATH?=C:\CodeSourcery\arm-2009q1-203\`

5.5 CCSV5

如果需要用 CCS 编译工程，则安装 CCS。

6. 编译 OMAP-L138_FlashAndBootUtils

6.1 查看环境变量

打开 cygwin，输入 `export`，查看环境变量是否设置成功。显示如下，黑色加粗的目录为之前添加的环境变量。

```
declare -x PATH="/usr/local/bin:/usr/bin:/cygdrive/c/ti/xdctools_3_22_01_21:/cygdrive/c/Program
Files/CodeSourcery/Sourcery
G++
Lite/bin:/cygdrive/c/WINDOWS/system32:/cygdrive/c/WINDOWS:/cygdrive/c/WINDOWS/System32/Wbem:/
cygdrive/c/Program Files/Common Files/Roxio Shared/DLLShared:/cygdrive/c/Program Files/Common
Files/Roxio
Shared/10.0/DLLShared:/cygdrive/c/Program
Files/Windows
Imaging:/cygdrive/c/ti/xdctools_3_22_01_21:/cygdrive/c/Program Files/Texas Instruments/C6000 Code
```

Generation Tools 7.3.4/bin:/cygdrive/c/WINDOWS/Microsoft.NET/Framework/v4.0.30319"

如未添加成功，可以使用如下命令添加：

```
export PATH=<arm-compiler-root>/bin:<MS-.NET-Root>:<C6000-Compiler-Root>/bin:$PATH
```

6.2 编译工具包

```
cd OMAP-L138
```

如果只是编译命令行工具（如 sfh_OMAP-L138），而不需要重新编译 CCS 工程，则进入 GNU 目录进行编译；如果只需要重新编译 CCS 工程，则进入 CCS 目录进行编译；如果两者都需要，则在 OMAP-L138 这个目录进行编译。编译命令如下：

```
make clean
make
```

7. 使用 serial flash utility 常见问题及解决方法

7.1 收不到“BOOTME”

在带电情况下，复位芯片，如果还是收不到正确的“BOOTME”，检查 BOOTMODE 设置，串口终端波特率等参数设置，串口线连接方式。

7.2 停在 Waiting for SFT on the OMAP-L138...

如图 14 所示，这种情况的原因通常有两种：一是由于 DDR 参数配置不正确从而导致程序运行异常，不能返回正确的指令给主机；二是硬件原因，可能是焊接，或硬件连接上的错误造成的。可以通过仿真器连接目标板用 BSL 库提供的测试程序验证一下 DDR，和相应的 flash。如果没有仿真器，可以通过间接的方法，比如用 AISgen 工具包里提供的 UART 启动工具 UartHost.exe 来加载运行测试程序，检测 DDR，flash 等，将测试结果通过串口回传显示，或者设置 GPIO 的状态来反映。

```
C:\WINDOWS\system32\cmd.exe - sfh_OMAP-L138.exe -erase
(AIS Parse): Read invalid BOOTME string.
(AIS Parse): Boot aborted.
Booting SFT failed. Trying again (you may need to reset the target)...
(AIS Parse): Read magic word 0x41504954.
(AIS Parse): Waiting for BOOTME... (power on or reset target now)
(AIS Parse): BOOTME received!
(AIS Parse): Performing Start-Word Sync...
(AIS Parse): Performing Ping Opcode Sync...
(AIS Parse): Processing command 0: 0x58535901.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Loading section...
(AIS Parse): Loaded 8704-Byte section to address 0x80000000.
(AIS Parse): Processing command 1: 0x58535901.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Loading section...
(AIS Parse): Loaded 824-Byte section to address 0x80002200.
(AIS Parse): Processing command 2: 0x58535906.
(AIS Parse): Performing Opcode Sync...
(AIS Parse): Performing jump and close...
(AIS Parse): AIS complete. Jump to address 0x80000000.
(AIS Parse): Waiting for DONE...
(AIS Parse): Boot completed successfully.

Waiting for SFT on the OMAP-L138...
```

图 15 sfh_OMAP-L138.exe 命令运行异常

重要声明

德州仪器(TI) 及其下属子公司有权根据 JESD46 最新标准, 对所提供的产品和服务进行更正、修改、增强、改进或其它更改, 并有权根据 JESD48 最新标准中止提供任何产品和服务。客户在下订单前应获取最新的相关信息, 并验证这些信息是否完整且是最新的。所有产品的销售都遵循在订单确认时所提供的TI 销售条款与条件。

TI 保证其所销售的组件的性能符合产品销售时 TI 半导体产品销售条件与条款的适用规范。仅在 TI 保证的范围内, 且 TI 认为有必要时才会使用测试或其它质量控制技术。除非适用法律做出了硬性规定, 否则没有必要对每种组件的所有参数进行测试。

TI 对应用帮助或客户产品设计不承担任何义务。客户应对其使用 TI 组件的产品和应用自行负责。为尽量减小与客户产品和应用相关的风险, 客户应提供充分的设计与操作安全措施。

TI 不对任何 TI 专利权、版权、屏蔽作品权或其它与使用了 TI 组件或服务的组合设备、机器或流程相关的 TI 知识产权中授予的直接或隐含权限制作出任何保证或解释。TI 所发布的与第三方产品或服务有关的信息, 不能构成从 TI 获得使用这些产品或服务的许可、授权、或认可。使用此类信息可能需要获得第三方的专利权或其它知识产权方面的许可, 或是 TI 的专利权或其它知识产权方面的许可。

对于 TI 的产品手册或数据表中 TI 信息的重要部分, 仅在没有对内容进行任何篡改且带有相关授权、条件、限制和声明的情况下才允许进行复制。TI 对此类篡改过的文件不承担任何责任或义务。复制第三方的信息可能需要服从额外的限制条件。

在转售 TI 组件或服务时, 如果对该组件或服务参数的陈述与 TI 标明的参数相比存在差异或虚假成分, 则会失去相关 TI 组件或服务的所有明示或暗示授权, 且这是不正当的、欺诈性商业行为。TI 对任何此类虚假陈述均不承担任何责任或义务。

客户认可并同意, 尽管任何应用相关信息或支持仍可能由 TI 提供, 但他们将独力负责满足与其产品及其应用中使用的 TI 产品相关的所有法律、法规和安全相关要求。客户声明并同意, 他们具备制定与实施安全措施所需的全部专业技术和知识, 可预见故障的危险后果、监测故障及其后果、降低有可能造成人身伤害的故障的发生机率并采取适当的补救措施。客户将全额赔偿因在此类安全关键应用中使用任何 TI 组件而对 TI 及其代理造成的任何损失。

在某些场合中, 为了推进安全相关应用有可能对 TI 组件进行特别的促销。TI 的目标是利用此类组件帮助客户设计和创立其特有的可满足适用的功能安全性标准和要求的终端产品解决方案。尽管如此, 此类组件仍然服从这些条款。

TI 组件未获得用于 FDA Class III (或类似的生命攸关医疗设备) 的授权许可, 除非各方授权官员已经达成了专门管控此类使用的特别协议。

只有那些 TI 特别注明属于军用等级或“增强型塑料”的 TI 组件才是设计或专门用于军事/航空应用或环境的。购买者认可并同意, 对并非指定面向军事或航空航天用途的 TI 组件进行军事或航空航天方面的应用, 其风险由客户单独承担, 并且由客户独力负责满足与此类使用相关的所有法律和法规要求。

TI 特别标示了符合 ISO/TS16949 要求的特定组件, 此类组件主要针对汽车用途。凡未做如此标示的组件则并非设计或专门用于汽车用途; 如果客户在汽车应用中使用任何未被指定的产品, 则 TI 对未能满足应用要求不承担任何责任。

产品	应用
数字音频	www.ti.com.cn/audio 通信与电信 www.ti.com.cn/telecom
放大器和线性器件	www.ti.com.cn/amplifiers 计算机及周边 www.ti.com.cn/computer
数据转换器	www.ti.com.cn/dataconverters 消费电子 www.ti.com.cn/consumer-apps
DLP® 产品	www.dlp.com 能源 www.ti.com.cn/energy
DSP - 数字信号处理器	www.ti.com.cn/dsp 工业应用 www.ti.com.cn/industrial
时钟和计时器	www.ti.com.cn/clockandtimers 医疗电子 www.ti.com.cn/medical
接口	www.ti.com.cn/interface 安防应用 www.ti.com.cn/security
逻辑	www.ti.com.cn/logic 汽车电子 www.ti.com.cn/automotive
电源管理	www.ti.com.cn/power 视频和影像 www.ti.com.cn/video
微控制器 (MCU)	www.ti.com.cn/microcontrollers
RFID 系统	www.ti.com.cn/rfidsys
OMAP应用处理器	www.ti.com.cn/omap
无线连通性	www.ti.com.cn/wirelessconnectivity 德州仪器在线技术支持社区 www.deyisupport.com

邮寄地址: 上海市浦东新区世纪大道 1568 号, 中建大厦 32 楼 邮政编码: 200122
Copyright © 2012 德州仪器 半导体技术 (上海) 有限公司