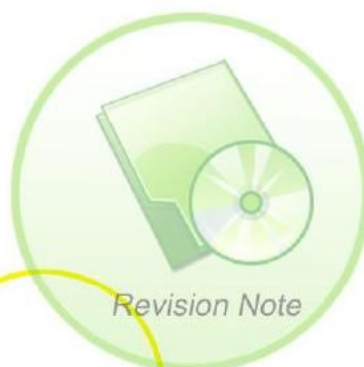




A company of SIM Tech

SIMCom_SIM800H_EAT_User Manual_V1.01



Document Title:	SIMCom_SIM800H_EAT_User Manual
Version:	1.01
Date:	2013-11-27
Status:	Draft
Document Control ID:	AN_SIMCom_SIM800H_EAT_User_Manual_App Note

General Notes

SIMCom offers this information as a service to its customers, to support application and engineering efforts that use the products designed by SIMCom. The information provided is based upon requirements specifically provided to SIMCom by the customers. SIMCom has not undertaken any independent search for additional relevant information, including any information that may be in the customer's possession. Furthermore, system validation of this product designed by SIMCom within a larger electronic system remains the responsibility of the customer or the customer's system integrator. All specifications supplied herein are subject to change.

Copyright

This document contains proprietary technical information which is the property of SIMCom Limited., copying of this document and giving it to others and the using or communication of the contents thereof, are forbidden without express authority. Offenders are liable to the payment of damages. All rights reserved in the event of grant of a patent or the registration of a utility model or design. All specification supplied herein are subject to change without notice at any time.

Copyright © Shanghai SIMCom Wireless Solutions Ltd. 2008

Version History

Data	Version	Description	Author
2013-11-27	V1.01	Created	Jack Sun

SCOPE

This document describes how to build and compile one project with embedded AT feature.

This document could be used for SIM800 series, for example SIM800H.

This document is subject to change without notice at any time.

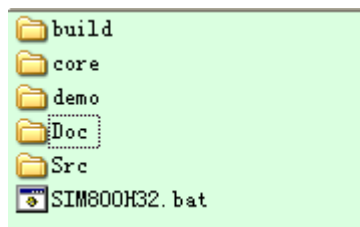
1、 Tool and Code Structure

Compiler: RVCT, Version: ARM/Thumb C/C++ Compiler, RVCT3.1 [Build 569];

Environment: winmake, provided in the build folder;

Download Tool: SIMCom_Multiupdate_tool_for_MTK_platform

The structure is as follows:



Compile configuration and makefile script

Head files and library provided by SIMCom

Demo Source Code

Documentations

Customer Source Code

Compile script

2、 Compile

2.1、 Tool Installation

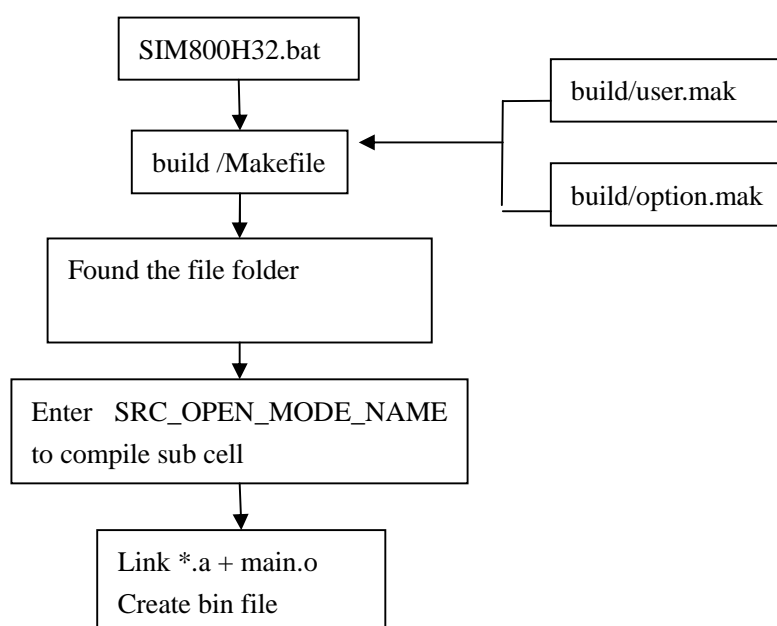
After you install the compiler successfully, there should be some output after you type “armcc” in cmd; the details please refer to the installation user guide;

```
Usage:      armcc [options] file1 file2 ... filen
Main options:
--arm       Generate ARM code
--thumb     Generate Thumb code
--c90       Switch to C mode (default for .c files)
--cpp       Switch to C++ mode (default for .cpp files)
-O0         Minimum optimization
-O1         Restricted optimization for debugging
-O2         High optimization
-O3         Maximum optimization
-Ospace     Optimize for codesize
-Otime      Optimize for maximum performance
--cpu <cpu> Select CPU to generate code for (eg. ARM9E/ARM10E)
--cpu list  Output a list of all the selectable CPUs
-o <file>   Name the final output file of the compilation
-c          Compile only, do not link
--asm       Output assembly code as well as object code
-S          Output assembly code instead of object code
--interleave Interleave source with disassembly (use with --asm or -S)
-E          Preprocess the C source code only
-D<symbol>  Define <symbol> on entry to the compiler
-g          Generate tables for high-level debugging
-I<directory> Include <directory> on the #include search path
Software supplied by: mammoth//ZWTiSO 2005
```

2.2、 Compiling Procedure

File	Description	Configured or Not
user.mak	Customer configuration file, configure the path、folder、file name..etc	Yes
Makefile	System Entry Makefile	No
option.mak	Configure compiling environment, system default	No
app_build.mak	Compile source code, system default	No

The compile procedure could show as follows:



2.3、Customer Configuration

Basically, customer needs to modify the file “user.mak” according to his own configuration, open the file by Ultra-Edit; for example:

```

# RVCT 3.1 Root Directory
DIR_ARM = C:\RVCT_EAT      //normally located in the system root
# Source Code Directory
SRC_DIR = demo              //default setting
# Example Sample Name
SRC_OPEN_MODE_NAME = adc    //the example sample which you want to implement
  
```

2.4、Add file

If customer needs to add some source code, for example, add a test code for SMS application, the source code should contain: main.c, app_sms.c, app_sms.h.

Note: App source code must contain one main.c as the entrance;

Steps as follows:

- 1) New a SMS folder in the Src;
- 2) The folder should contain main.c, app_sms.c, app_sms.h;
- 3) Copy the “makefile” from the demo/adc to current SMS folder; there are two ways to compile the source code;
 - a) compile all the “.c” files;


```
#First mothod#
S1 := $(wildcard *.c)
SOURCE:=$(S1)
```
 - b) compile the dedicated source code file


```
#Second mothod#
SOURCE := main.c\ app_sms.c
```
- 4) Modify user.mak file


```
# Directory of source code file
SRC_DIR = src
# the part need to be compiled
SRC_OPEN_MODE_NAME = sms
```
- 5) Run SIM800H32.bat to compile your file, it shows as follows

```
C:\SIM800H_EAT_app_131108\SIM800H_EAT_app_131108>rem CMD ,run "SIM800H32.bat all
" or "SIM800H32.bat clean"
mkdir C:\SIM800H_EAT_app_131108\SIM800H_EAT_app_131108\output
子目录或文件 C:\SIM800H_EAT_app_131108\SIM800H_EAT_app_131108\output 已经存在。
make: [copy] Error 1 (ignored)
mkdir C:\SIM800H_EAT_app_131108\SIM800H_EAT_app_131108\output\obj
子目录或文件 C:\SIM800H_EAT_app_131108\SIM800H_EAT_app_131108\output\obj 已经存
在。
make: [copy] Error 1 (ignored)
mkdir C:\SIM800H_EAT_app_131108\SIM800H_EAT_app_131108\output\log
子目录或文件 C:\SIM800H_EAT_app_131108\SIM800H_EAT_app_131108\output\log 已经存
在。
make: [copy] Error 1 (ignored)
cd demo/gpio && make PARAM=gpio
make[1]: Entering directory `C:/SIM800H_EAT_app_131108/SIM800H_EAT_app_131108/de
mo/gpio'
C:\RUCT_EAT\Programs\3.1\569\win_32-pentium/armcc -c --thumb --cpu ARM7EJ-S --li
ttleend -O3 --remove_unneeded_entities --split_sections -D_SIMCOM_PROJ_SIM800H
__ -D_USB_COM_PORT_ENABLE__ -DAPP_EXAMPLE -DAPP_EXAMPLE2 -IC:\SIM800H_EAT_
app_131108\SIM800H_EAT_app_131108\core/inc -IC:\SIM800H_EAT_app_131108\SIM800H_E
AT_app_131108\demo/gpio -I./ -o C:\SIM800H_EAT_app_131108\SIM800H_EAT_app_13110
8/output/obj/app_demo_gpio.o app_demo_gpio.c 1>&2 2>C:\SIM800H_EAT_app_131108\SI
M800H_EAT_app_131108/output/log/output.log
```

After you finished the compiling, it should create one folder named OUTPUT which contains the compiling result,

log	
obj	
app	4 KB
app.elf	958 KB
app.lis	2,222 KB
app.sym	637 KB

from the log file, you could find the result of your source code, warning or error...etc;

3、Download

SIM800H Application Notes

The target file contains two parts: one is core file provided by SIMCom, another one is the APP file which created by customers themselves;

Normally, the modules shipped to customers should contain the core file already (if order with EAT functionality, others need to upgrade), customer only needs to compile his app file and download it; the details please refer to “SIMCom_SIM800H_EAT_flash_tool_User Manual_V1.02”

Note:

- 1) If SIMCom releases new FW version, customer needs to update his core program correspondingly; meanwhile, the app code also need to be recompiled based on the new library;*
- 2) The download tool should be various according to different demands;*

Contact us:

Shanghai SIMCom Wireless Solutions Ltd.

Add: Building A, SIM Technology Building, No.633, Jinzhong Road, Changning District, Shanghai, P. R. China 200335

Tel: +86 21 3235 3300

Fax: +86 21 3235 3301

URL: www.sim.com/wm