SIM800 Series_ Embedded AT_Compiling Environment

GPRS Module

Document Title:	SIM800 Series_Embedded AT_Compiling Enviroment
Version:	1.03
Date:	2020.6.15
Status:	Released

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 WIRELESS SOLUTIONS LIMITED COPYING, TO OTHERS AND USING THIS DOCUMENT, ARE FORBIDDEN WITHOUT EXPRESS AUTHORITY BY SIMCOM. OFFENDERS ARE LIABLE TO THE PAYMENT OF INDEMNIFICATIONS. ALL RIGHTS RESERVED BY SIMCOM IN THE PROPRIETARY TECHNICAL INFORMATION ， INCLUDING BUT NOT LIMITED TO REGISTRATION GRANTING OF A PATENT , A UTILITY MODEL OR DESIGN. ALL SPECIFICATION SUPPLIED HEREIN ARE SUBJECT TO CHANGE WITHOUT NOTICE AT ANY TIME.

SIMCom Wireless Solutions Limited

Building B, SIM Technology Building, No.633 Jinzhong Road, Changning District, Shanghai P.R. China

Tel: +86 21 31575100

Email: simcom@simcom.com

For more information, please visit:

https://www.simcom.com/download/list-863-en.html

For technical support, or to report documentation errors, please visit:

https://www.simcom.com/ask/ or email to: support@simcom.com

About Document

Version History

Version	Date	Owner	What is new
V1.01	2013.07.11	Bin Mao	Original
V1.02	2015.08.10	Bin Mao	Update scope
V1.03	2020.06.15	Fan.Fang	All

Scope

This document is applicable to SIM800 series Embedded AT module, include SIM800W, SIM840W, SIM800V, SIM800H, SIM800, SIM800M64, SIM808 and SIM800C.

This document describes the development guide of Embedded AT and relative notes.

Contents
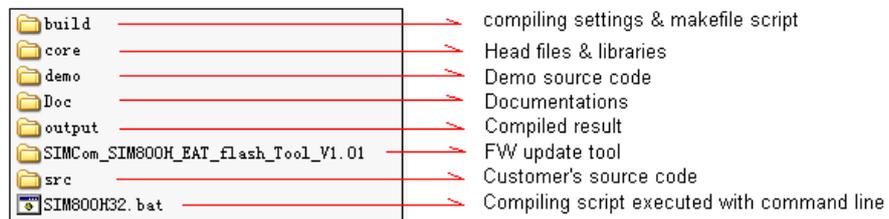
1 Tool and Code Architecture

Compiler: RVCT

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

Compiling tool:winmake (under ..\build\)

Download tool:SIMCom_SIM800H_EAT_flash_Tool_V1.01

Folder list:

2 Compiling

2.1 Compiler Environment

Once RVCT compiler installed successfully, armcc command will show you correct version info as following.

```
f:\Module\SIM800_52D\MT6252MOD\eat\app>armcc
ARM/Thumb C/C++ Compiler, RVCT3.1 [Build 569]

Usage:          armcc [options] file1 file2 ... filen
Main options:
```

SIM800H E-AT project will be compiled with script line.

```
f:\Module\SIM800_52D\MT6252MOD\eat\app>build_SIM800W64.bat
```

2.2 Compiling Process Introduction

Build list description.

File	Function	User-configuration
user.mak	Configuration file for compilingpath,sourcecode folder,and compilingcase,etc.	Yes
Makefile	System entrance	No
option.mak	Configure compiling environment	No
app_build.mak	Compile source code	No

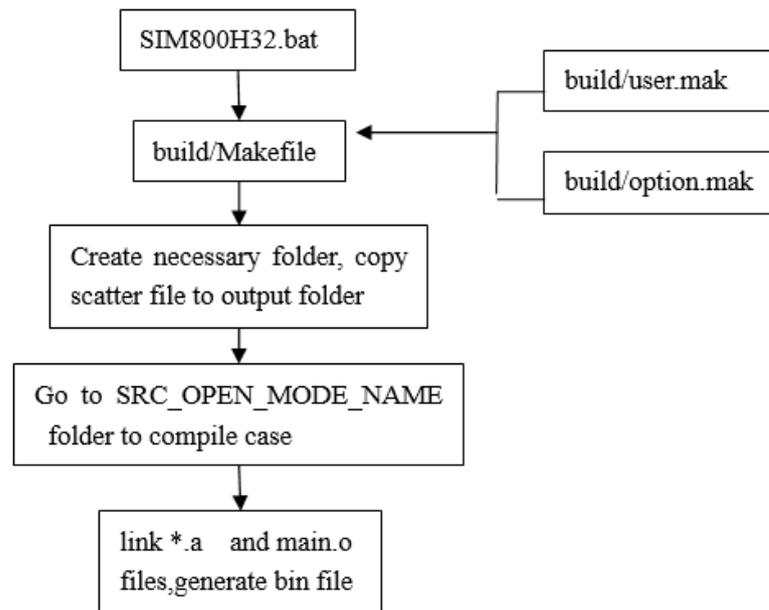Compiling process is as following.

Figure 1 compiling process

Bin file is available in folder defined by OUTPUT_BIN_DIR（user.mak file defined）.
SIM800H and SIM800 app file will be generated to path core/SIM800H32_EMBEDDEDAT/.
Downloading core and app file need SIM800H32_EAT.cfg file. If just update app file, just check app option.

2.3   Client Configuration

Client could modify some settings for user.mak.

#RVCT 3.1 Root Directory
DIR_ARM = C:\Progra~1\ARM\RVCT
#source file folder
SRC_DIR = demo
#current case to be compiled
SRC_OPEN_MODE_NAME = adc

2.4   Add New File to Project

Here, we add a new test case, for example sms case, source cdoe includes main.c, app_sms.c and app_sms.h.

Note: case source should include main.c file for app entrance.

Steps for this case:

➢ Create sms folder;

➢ Generate main.c, app_sms.c and app_sms.h files.

➢ Copy Makefile from demo/adc to root sms folder.

Two options here to orgnizate source files.

➢ compile all *.c files (ensure main.c included)

#First mothod#
S1:= $(wildcard *.c)
SOURCE :=$(S1)

➢ specify source file to compile

#Second mothod#
**SOURCE := main.c **
app_sms.c

*Note, we recommend main.c file for app entrance in EAT project. If there has no mail.c but different source file only, should modify following info to link to main.o file.

MAIN_SRC := app_demo_uart.c
MAIN_OBJ := main.o

➢ modify user.mak file.

 # root directory
SRC_DIR = src
 # case folder to be compiled
SRC_OPEN_MODE_NAME = sms

➢ call "sim800H32.bat all" to compile or "sim800H32.bat clean" to clean generated files.

3 Download

Objective file includes two sections. One is core file for simcom module; another is app file for EATcase.

Normally, core is here, just update app file. But if need new core version to build project, should download these two sections together. Make sure app file is compiled based on this new core file.

Note:

➢ If SIMCom offer new CORE files to customers, CORE must be upgraded. Besides, APP has to be compiled again with new CORE and be downloaded.

➢ Download tools will be different according to different requirements.

4 Debug

Three methods are available to view logs.

➢ eat_uart_write() interface to print log info

➢ Configure DEBUG port as EAT_UART_DEBUG_MODE_UART mode, and eat_trace() interface print log info which could be visible with hyperterminal

➢ Configure DEBUG port as EAT_UART_DEBUG_MODE_TRACE mode, and eat_trace() interface prints log info which only be visible with Catcher tool provided by SIMCom.

4.1   EAT_UART_DEBUG_MODE_UART mode

```
app_func_ext1 ():
EatUartConfig_stcfg =
{
    EAT_UART_BAUD_115200,
    EAT_UART_DATA_BITS_8,
    EAT_UART_STOP_BITS_1,
    EAT_UART_PARITY_NONE
};
// configue EAT_UART_2 as debug
eat_uart_set_debug(EAT_UART_2);

// coinfigure DEBUG port as UART mode.
eat_uart_set_debug_config(EAT_UART_DEBUG_MODE_UART, &cfg);
```
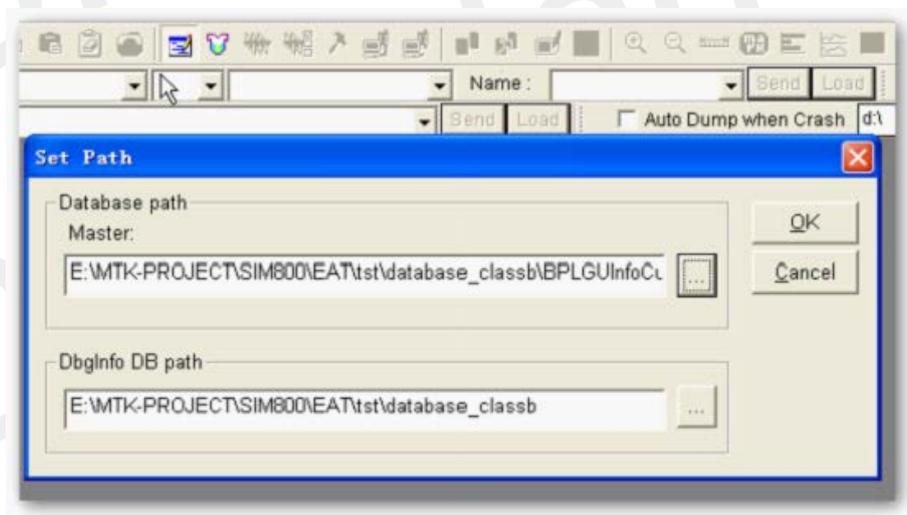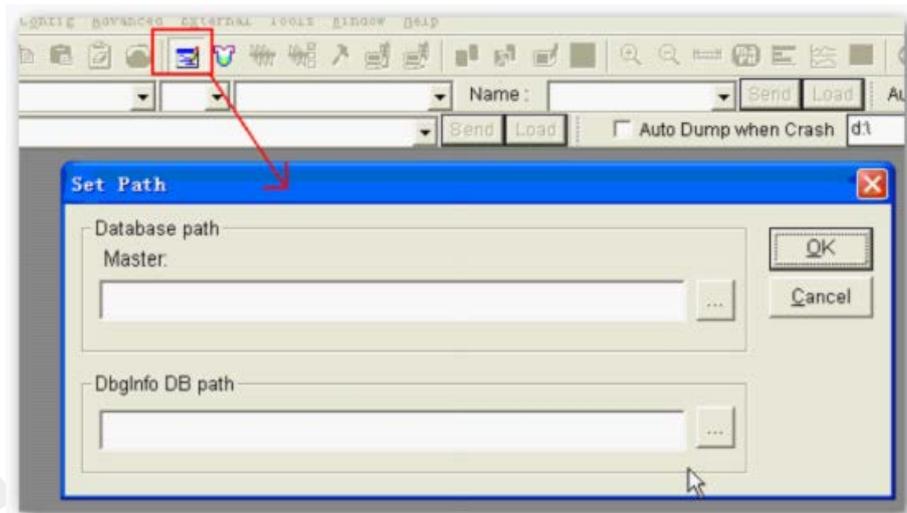
4.2   Catcher Tool Operation

➢ Conigure UART2 output log info.

```
void app_func_ext1(void*data)
{
    eat_uart_set_debug(EAT_UART_2);
```
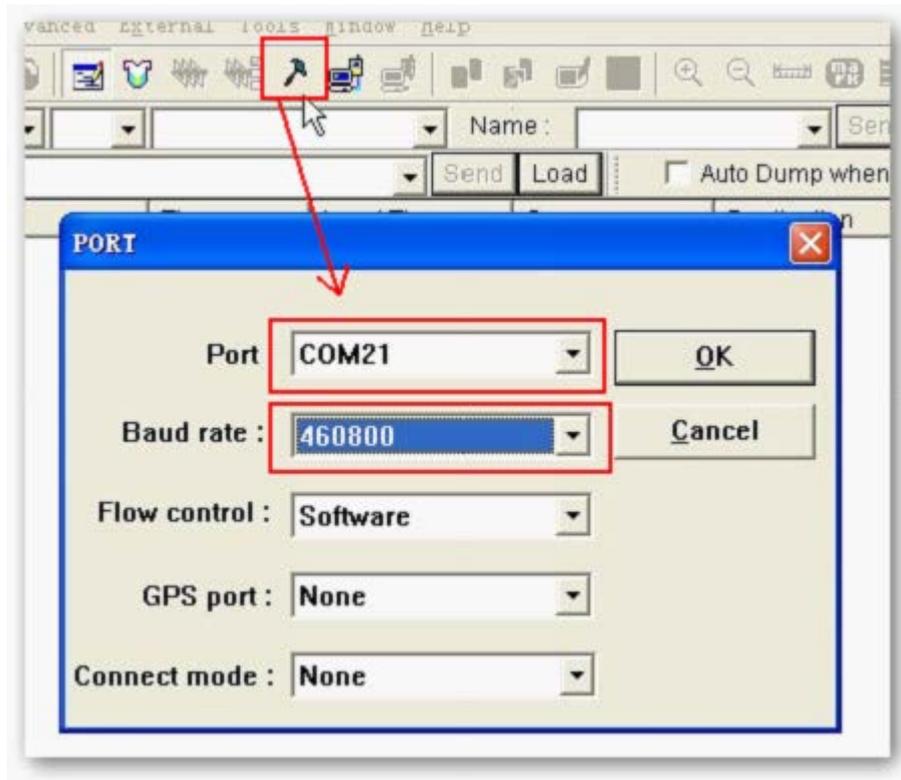
```
......
}
```

➢ Connect UART2 to PC, and launch on Catcher tool.

➢ Pick up data base file.

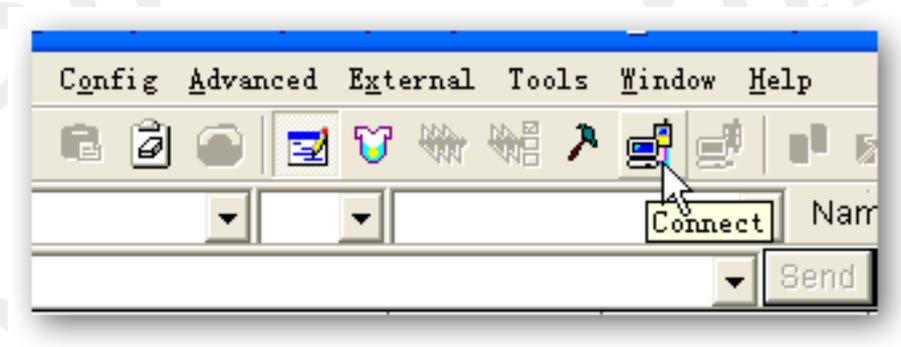Database file is provided by SIMCom, such as BPLGUInfoCustomAppSrcP_xxx _xxx_EMBEDDEDAT.
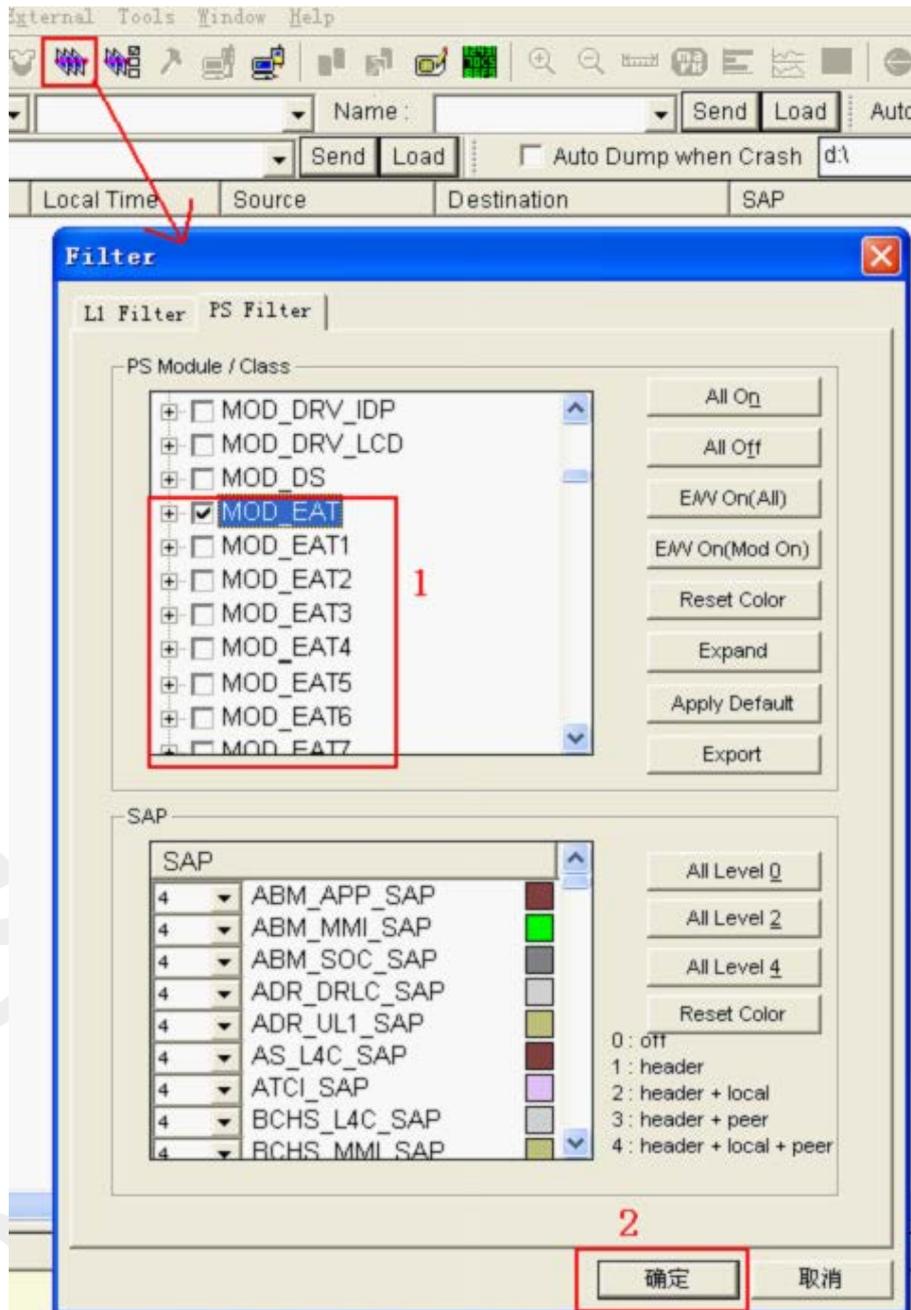

➢ Configure port
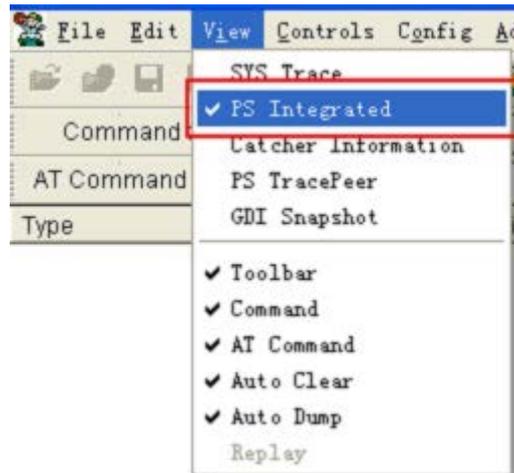
Pick up correct COM port, choose 460800 baud rate.
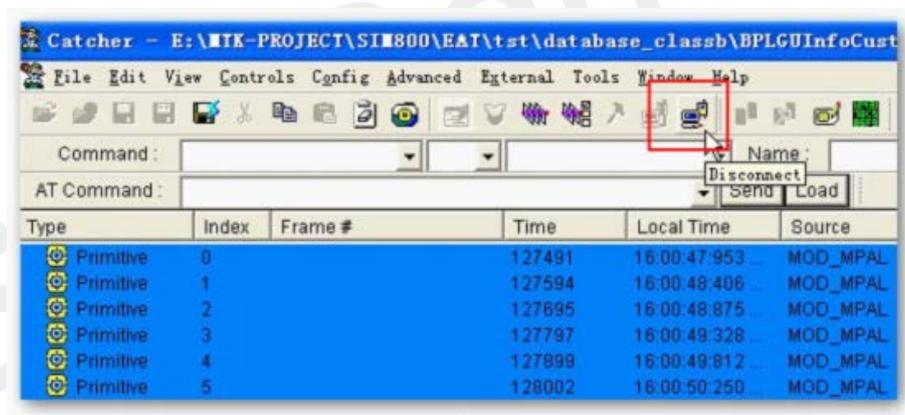
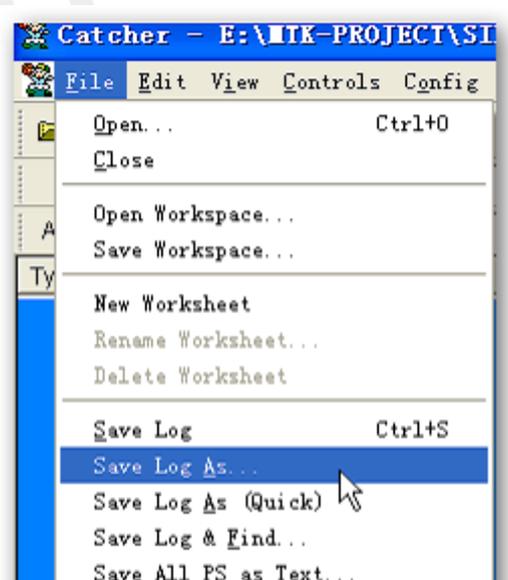➤ Hit "Connect" button to continue.

➤ Configure filter.

➢ Ready to catch log info. Hit "View-> PS Integrated", log trace will be visible.

➢ Disconnect communication to save log.

➢ Hit "File->Save log as…" option to save log.

➢ Log trace is like following.

Primitve	298	1636	15:33:15.360 ...	MOD_DRV_HISR	MOD_EAT	DRIVER_PS_SAP	MSG_ID_UART_READY_TO_READ_IND
Trace	299	1636	15:33:15.360 ...	MOD_EAT			eat_core_uart.c-62:wlen-1 len-0
Trace	300	1636	15:33:15.360 ...	MOD_EAT			can_write_len:1
Trace	301	1636	15:33:15.360 ...	MOD_EAT			tail-11 head-11
Trace	302	1636	15:33:15.360 ...	MOD_EAT			main.c-292:msg 7
Trace	303	1636	15:33:15.360 ...	MOD_EAT			get uart0 buf.
Primitve	304	1636	15:33:15.360 ...	MOD_EAT	MOD_ATCI	INVALID_SAP	MSG_ID_UART_READY_TO_READ_IND
Primitve	305	1637	15:33:15.360 ...	MOD_ATCI	MOD_EAT	INVALID_SAP	MSG_ID_UART_READY_TO_READ_IND
Primitve	306	1637	15:33:15.360 ...	MOD_ATCI	MOD_DS	INVALID_SAP	MSG_ID_READY_UART_WRITE_IND
Trace	307	1637	15:33:15.360 ...	MOD_EAT			main.c-292:msg 4