



# Telit AppZone user manual

Version 0.2.4 – 2011-11-17



**Making machines talk.**

## APPLICABILITY TABLE

PRODUCT
G30



*SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE*

### **Notice**

While reasonable efforts have been made to assure the accuracy of this document, Telit assumes no liability resulting from any inaccuracies or omissions in this document, or from use of the information obtained herein. The information in this document has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies or omissions. Telit reserves the right to make changes to any products described herein and reserves the right to revise this document and to make changes from time to time in content hereof with no obligation to notify any person of revisions or changes. Telit does not assume any liability arising out of the application or use of any product, software, or circuit described herein; neither does it convey license under its patent rights or the rights of others.

It is possible that this publication may contain references to, or information about Telit products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that Telit intends to announce such Telit products, programming, or services in your country.

### **Copyrights**

This instruction manual and the Telit products described in this instruction manual may be, include or describe copyrighted Telit material, such as computer programs stored in semiconductor memories or other media. Laws in the United States and other countries preserve for Telit and its licensors certain exclusive rights for copyrighted material, including the exclusive right to copy, reproduce in any form, distribute and make derivative works of the copyrighted material. Accordingly, any copyrighted material of Telit and its licensors contained herein or in the Telit products described in this instruction manual may not be copied, reproduced, distributed, merged or modified in any manner without the express written permission of Telit. Furthermore, the purchase of Telit products shall not be deemed to grant either directly or by implication, estoppel, or otherwise, any license under the copyrights, patents or patent applications of Telit, as arises by operation of law in the sale of a product.

### **Computer Software Copyrights**

The Telit and 3rd Party supplied Software (SW) products described in this instruction manual may include copyrighted Telit and other 3rd Party supplied computer programs stored in semiconductor memories or other media. Laws in the United States and other countries preserve for Telit and other 3rd Party supplied SW certain exclusive rights for copyrighted computer programs, including the exclusive right to copy or reproduce in any form the copyrighted computer program. Accordingly, any copyrighted Telit or other 3rd Party supplied SW computer programs contained in the Telit products described in this instruction manual may not be copied (reverse engineered) or reproduced in any manner without the express written permission of Telit or the 3rd Party SW supplier. Furthermore, the purchase of Telit products shall not be deemed to grant either directly or by implication, estoppel, or otherwise, any license under the copyrights, patents or patent applications of Telit or other 3rd Party supplied SW, except for the normal non-exclusive, royalty free license to use that arises by operation of law in the sale of a product.

### **VENDOR COPYRIGHT**

Apache Software Foundation Copyright 2004-2005 All Rights Reserved





## 1. Document history

Revision	Date	Changes
0	2010-07-22	First issue
1	2011-11-9	AppZone adjustments for version 0.2.4
2	2011-11-17	Screen shots and content updating



## Contents

<b>1. Document history</b> .....	<b>5</b>
<b>2. Introduction</b> .....	<b>8</b>
2.1. Scope .....	8
2.2. Audience .....	8
2.3. Contact Information, Support .....	8
2.3.1. Required Query Information .....	9
2.4. Text Conventions .....	10
2.5. General Safety .....	10
<b>3. G30 Telit AppZone description</b> .....	<b>13</b>
3.1. System Description .....	13
3.2. Typical System Configuration .....	14
3.3. G30 AppZone SW Platform Configuration .....	15
3.3.1. AppZone Application Execution Flow .....	15
3.4. The AppZone Skeleton .....	17
3.5. G30 Modem to AppZone Application Handshake .....	18
3.6. Interfaces for the AppZone Application .....	19
<b>4. Getting Started</b> .....	<b>20</b>
4.1. Creating and Compiling AppZone Application with Eclipse ...	20
4.2. Telit AppZone Gadget .....	21
4.3. G30 Flash Tool .....	22
4.4. Telit AppZone Flash Tool .....	24
4.5. Eclipse IDE .....	25
4.6. Telit AppZone Compiler .....	27
4.7. Telit AppZone OTA Server .....	28
4.8. Telit AppZone Trace Tool .....	30
4.9. Telit AppZone File System Tool .....	32



<b>5. Using the G30 Telit AppZone application</b>	<b>33</b>
5.1. Flashing an AppZone Application	33
5.2. Telit AppZone Flash tool settings	35
5.3. Enable/Disable M2M Application using AT commands	35
5.4. Recovery Mechanism	35
5.5. OTA (Over The Air) Upgrade	35
<b>6. Appendix A: Capabilities Overview</b>	<b>39</b>
6.1. HW Peripheral	39
6.2. Socket	41
6.2.1. Creating Socket Connections	42
6.3. File System	42
6.4. Clock and Timer	43
6.5. Memory IO	43
6.6. AT Commands	44
6.7. Operating System	44











## Keep away from live circuits

Operating personnel must:

- Not remove equipment covers. Only Factory Authorized Service Personnel or other qualified maintenance personnel may remove equipment covers for internal subassembly, or component replacement, or any internal adjustment
- Not replace components with power cable connected. Under certain conditions, dangerous voltages may exist even with the power cable removed
- always disconnect power and discharge circuits before touching them

## Do not substitute parts or modify equipment

Because of the danger of introducing additional hazards, do not install substitute parts or perform any unauthorized modification of equipment. Contact Telit Warranty and Repair for service and repair to ensure that safety features are maintained.

## Dangerous procedure warnings

Warnings, such as the example below, precede potentially dangerous procedures throughout this manual. Instructions contained in the warnings must be followed. You should also employ all other safety precautions that you deem necessary for the operation of the equipment in your operating environment.

Warning example:



---

### WARNING:

Dangerous voltages, capable of causing death, are present in this equipment. Use extreme caution when handling, testing, and adjusting.

---

## Limitation of Liability

The Products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body; in other applications intended to support or sustain life; for the planning, construction, maintenance, operation or use of any nuclear facility; for the flight, navigation, communication of aircraft or ground support equipment; or in any other application in which the failure of the Product could create a situation where personal injury or death may occur. If CUSTOMER should use any Product or provide any Product to a third party for any such use, CUSTOMER hereby agrees that TELIT is not liable, in whole or in part, for any claims or damages arising from such use, and further agrees to indemnify and hold TELIT harmless from any claim, loss, cost or damage arising from such use.

EXCEPT AS SPECIFICALLY STATED ABOVE, THE PRODUCTS ARE PROVIDED "AS IS" AND TELIT MAKES NO OTHER WARRANTIES EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE REGARDING THE PRODUCTS. TELIT SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, OR ARISING FROM A COURSE OF DEALING OR USAGE OF TRADE.

Under no circumstances shall TELIT be liable to CUSTOMER or any other party for any costs, lost revenue or profits or for any other special, incidental or consequential damages, even if TELIT has been informed of such potential loss or damage. And in no event shall TELIT's liability to CUSTOMER for damages of any nature exceed the total purchase price CUSTOMER paid for the Product at issue in the dispute, except direct damages resulting



from patent and/or copyright infringement, which shall be governed by the "INDEMNITY" Section of this Agreement.

The preceding states TELIT's entire liability for TELIT's breach or failure to perform under any provision of this Agreement.



### 3. G30 Telit AppZone description

AppZone is a SW platform built on top of the G30 modem, providing interfaces to most of the G30 modem functionality.

The G30 AppZone SW Platform is a GSM/GPRS module plus a SW layer that allows integrating customer applications into the module.

**Note:** No external micro is required.

The SW interface gives a safe and controlled access to the GSM/GPRS modem and to various peripherals and HW resources. With AppZone, the developer can easily access most of the G30 functionalities.

The G30 AppZone SW Platform is open and addresses a wide range of different products such as remote monitoring and control, security and surveillance, telemetry, location services, billing, fleet management etc.

By using the AppZone SW platform, it is possible to remove the need for a host processor by allowing the user application to run on the modem processor. Cost and time to market are reduced.

#### 3.1. System Description

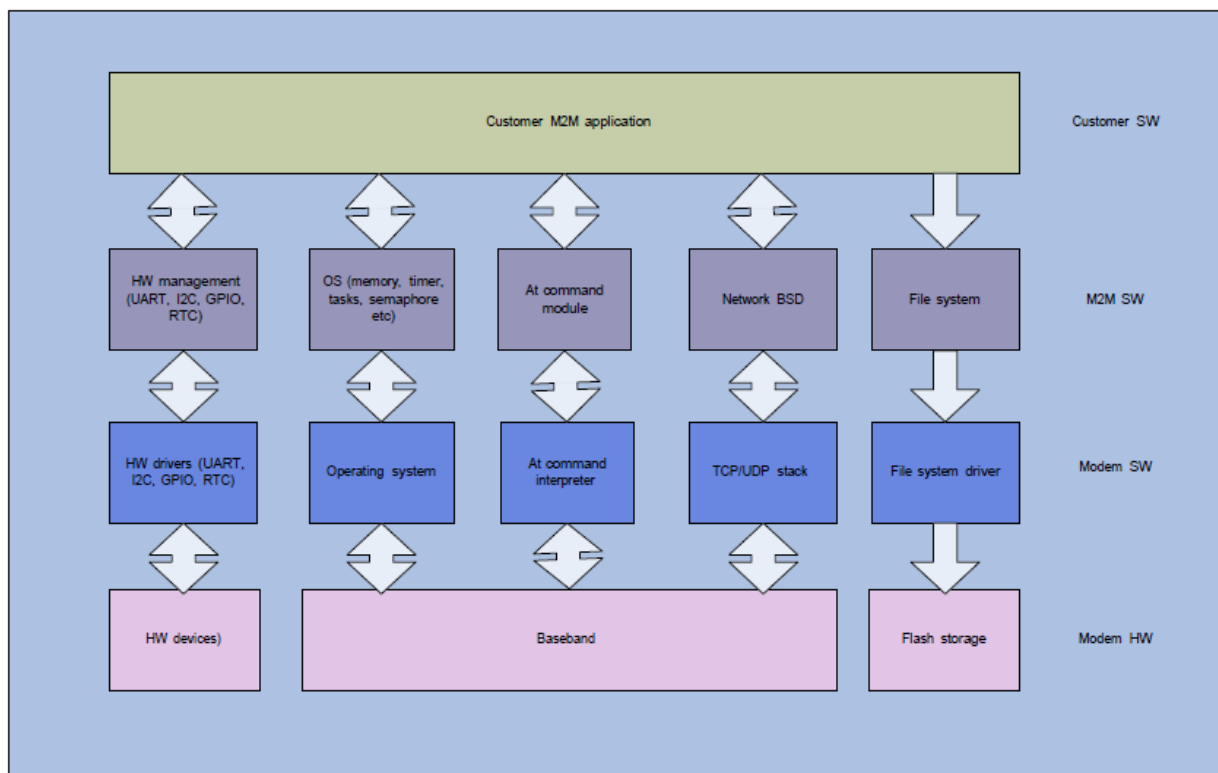


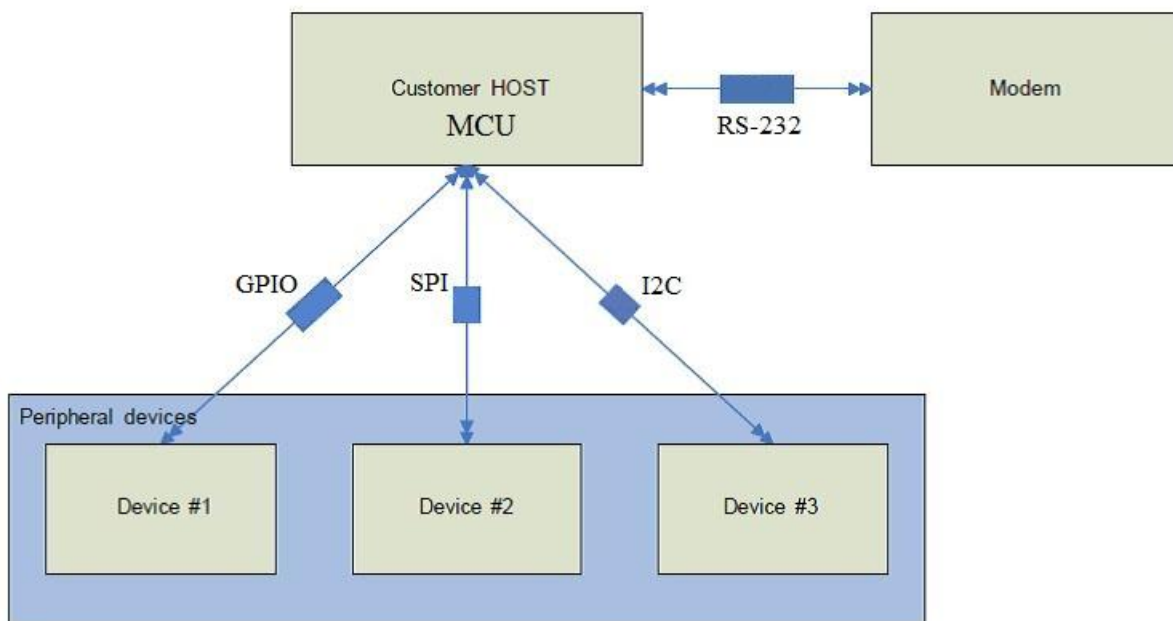
Figure 1: AppZone Platform - High Level Architecture



There are four main SW modules available for the customer application, built on top of an AppZone SW platform:

1. OS: Processes, Signals, Semaphores, Timers, Dynamic Memory Management, etc.
2. HW drivers: GPIO, I2C, SPI, RS232, File-System, Keypad, RTC, etc.
3. GSM/GPRS: Access and control.
4. Networking: BSD socket support, SSL capabilities.

### 3.2. Typical System Configuration

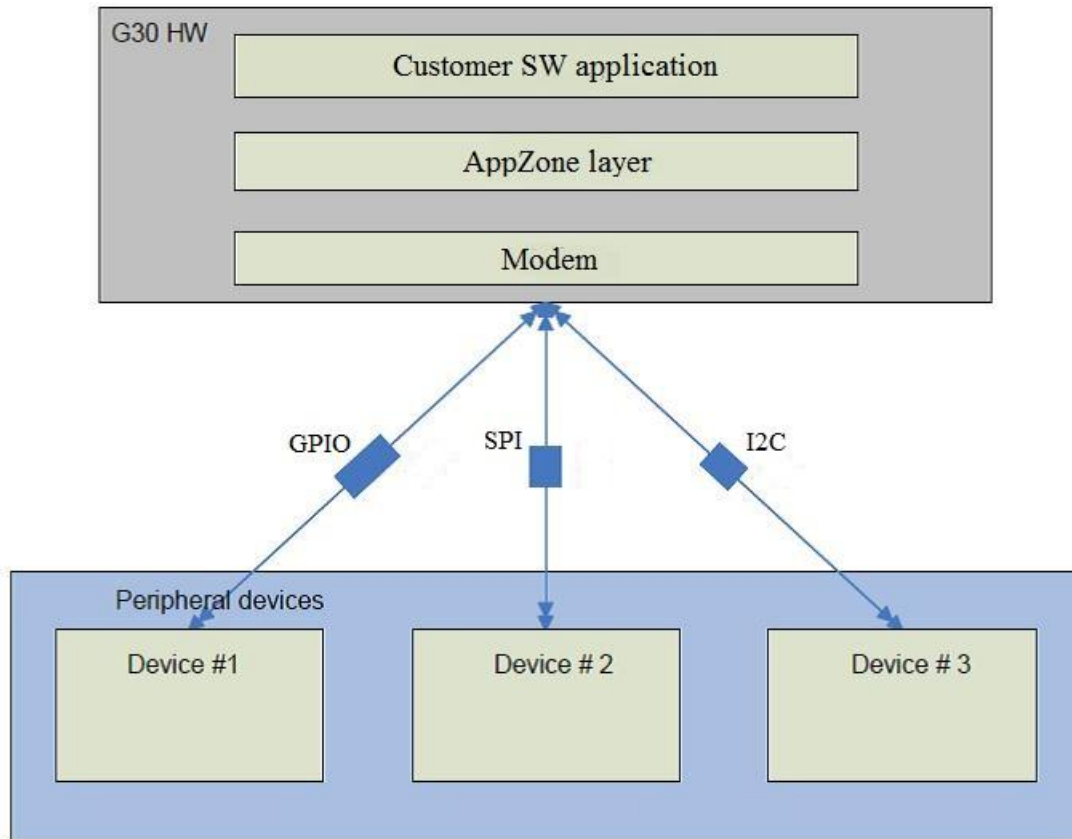


**Figure 2: Typical system configuration**

Figure 2 describes a typical customer configuration without using the AppZone SW platform. Two processors are required: Host and Modem



### 3.3. G30 AppZone SW Platform Configuration



**Figure 3: Telit AppZone system configuration**

Figure 3 describes a typical customer configuration when using the AppZone SW platform. Only one processor is required: Modem.

#### 3.3.1. AppZone Application Execution Flow

This paragraph describes the way a typical AppZone application is being executed. Some technical details are involved in order to understand the perception:

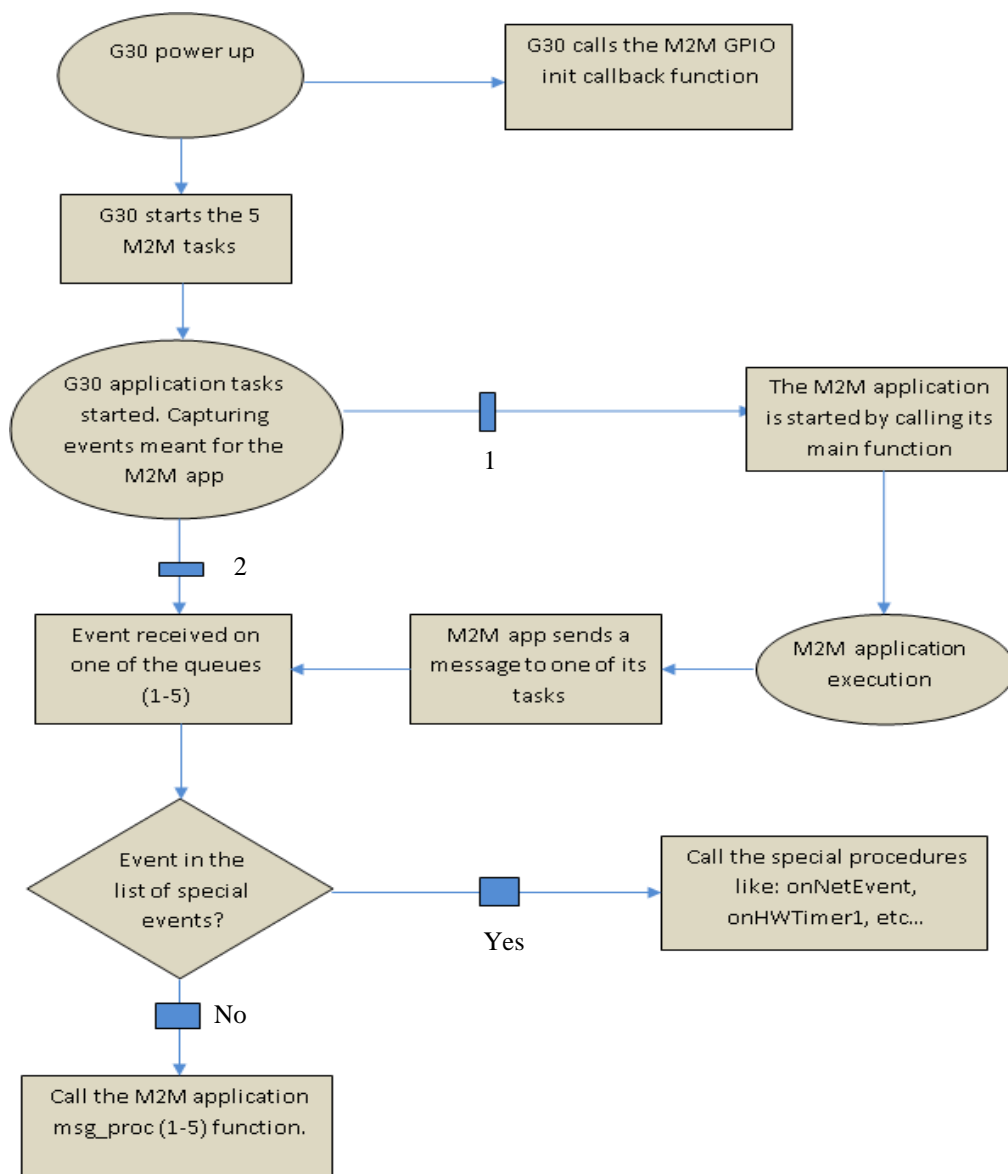
- Each AppZone application has a single entry point (Main).
- There are currently 5 tasks allocated for the AppZone application.
- These tasks are static, meaning, always running.



Whenever an event is received on a queue (every task has its own queue), it is checked against a predefined event list (network, timer, UART, IAT, key, wakeup etc). In case of a predefined event, the event is routed to the AppZone application through a call to its specific callback handler.

In case of a non-predefined event (like errors), the event is routed to the AppZone application through a call to one of its M2M\_msgProc callback function (depending on which task is handling the event).

Non predefined event may be events sent from the modem or events sent by the AppZone application other tasks (the application can send messages from one task to the other).



**Figure 4: This flow chart describes the AppZone application execution flow**





### 3.4. The AppZone Skeleton

The AppZone SW platform comes with a "Skeleton" code (for user development) and header files providing the AppZone interfaces to the G30.

The skeleton files are automatically generated when creating an AppZone project from the Eclipse IDE (see [Creating and Compiling AppZone Application with Eclipse](#) section).

The skeleton files are:

- M2M\_atRsp.c
- M2M\_hwTimer.c
- M2M\_i2c.c
- M2M\_initialize.c
- M2M\_interrupt.c
- M2M\_keyboard.c
- M2M\_main.c
- M2M\_net.c
- M2M\_proc1.c
- M2M\_proc2.c
- M2M\_proc3.c
- M2M\_proc4.c
- M2M\_proc5.c
- M2M\_utilities.c

These files include various handler functions in which the user can implement if wishes to handle these "events".

---

**Important:** Generally, the callback functions (those provided in run time and the static ones) are called on the modem task context. It is extremely important not to implement any complex/heavy operations in these functions. Instead, one can simply send an event to one of the 5 AppZone tasks for handling.

---







## 4. Getting Started

### 4.1. Creating and Compiling AppZone Application with Eclipse

The following steps are required to setup the Telit AppZone development environment:

1. After running the Telit AppZone Installer, the Telit AppZone package will be installed on your computer.

The Telit AppZone package contains the following folders:

- [AppZone Documents](#) – contains informative documents about the package.
- [AppZone Sample Apps](#) – sample applications for demonstration.
- [AppZone SW Files](#) – the latest software files that will be flashed into the module.
- [AppZone Tools](#) – necessary tools for the environment operation including:  
Eclipse, Flash tools, Compiler. Etc.

You can find farther information in each folder.

Go to the Telit AppZone Folders by lunching the [Telit AppZone Gadget](#) and select 'Resources Folder'.

Or alternatively, go to `C:\Program Files\Telit Wireless Solutions\Telit AppZone`.

2. Flash the SW files located in your [AppZone SW Files](#) folder into your G30 module using the G30 Flash Tool (see [G30 Flash Tool](#) section).
3. Create new AppZone project using Eclipse IDE (see [Eclipse IDE](#) section)
4. Flash your AppZone application into your G30 module using the Telit AppZone Flash Tool (see [Flashing an AppZone Application](#) section).
5. Open your terminal to see the application running.



## 4.2. Telit AppZone Gadget

The Telit AppZone Gadget designed to provide the Telit AppZone customers easy access to the main Telit AppZone resources.

**Note:**

The Telit AppZone Gadget will work on Windows Vista and WIN7 x32 and x64 Operating systems.

It is not supported on WIN XP and W2K.



The Gadget will direct you to:

[Eclipse](#) – The Eclipse IDE.

[M2M Flash Tool](#) – The flash tool used to flash your applications into the G30 module.

[Developer Guide](#) – The Developer Guide, where you can find all the API's and the code documentation.

[User Manual](#) – The User Manual, where you will find Comprehensive guidelines on how to work with Telit AppZone

[Resources Folder](#) – The main resources folder where all the relevant Telit AppZone information can be found.

[Telit AppZone](http://appzone.telit.com/) – the Telit AppZone website in the address <http://appzone.telit.com/>, where you can find information and samples.

In order to run the Telit AppZone gadget:

- Enter the WIN7 Gadget screen (right click on your desktop and select 'Gadgets').

Or alternatively, run the command line:

C:\Program Files\Windows Sidebar\sidebar.exe /showgadgets in 32 bit OS

C:\Program Files (x86)\Windows Sidebar\sidebar.exe /showgadgets in 64 bit OS

- Double click the Telit AppZone Gadget.
- The Gadget appearance can be changed to your convenient, by pressing right click on the gadget and set its size and opacity.



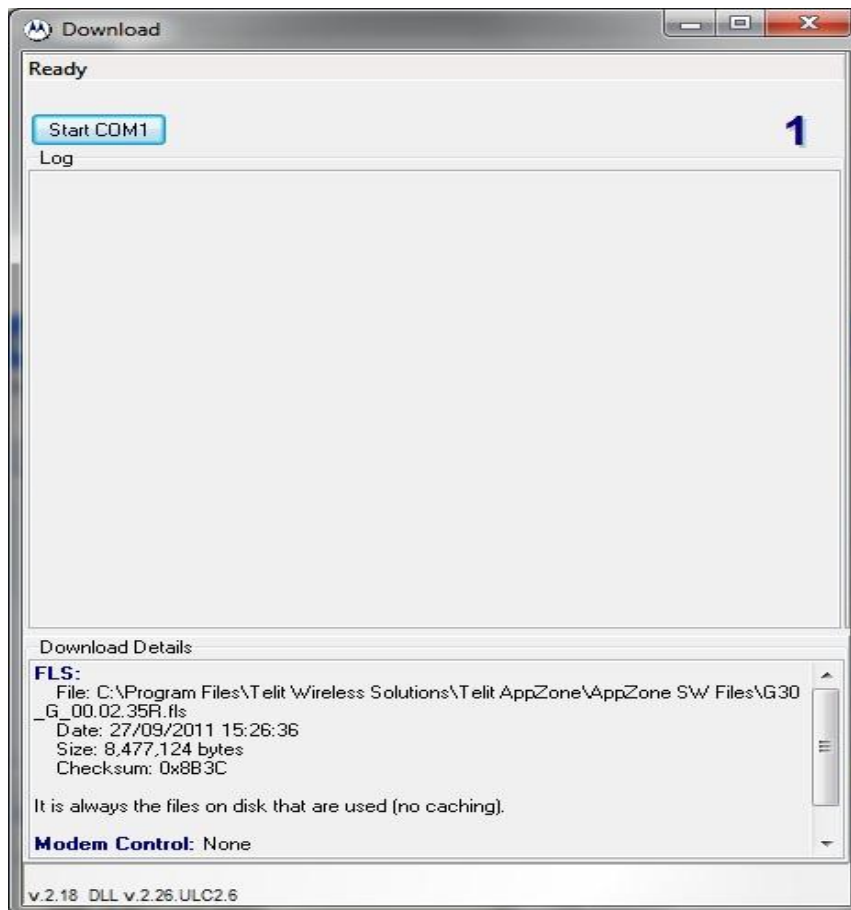


1. Lunch “FlashTool\_E2.exe” located in the **Telit AppZone\ AppZone Tools\G30 Flash Tool** Folder.
2. Select the .fls and .cust Files (located in the **Telit AppZone\ AppZone SW Files** Folder) in the Phone Software and Customization sections respectively.
3. Select the desirable COM Port and Baud Rate and Press Start.

**Note:**

The maximum baud rate when using a serial com port is 115200.

The maximum baud rate when using a serial to USB adapter is 921600.



4. Press the Start COM button.
5. Power cycle the G30 (turn the s200 switch on the evaluation board on and off).
6. When Flashing is completed, the module will be restarted and the new software version will be flashed in the device.







## 4.5. Eclipse IDE

The development is done using the most popular free IDE – “Eclipse”.

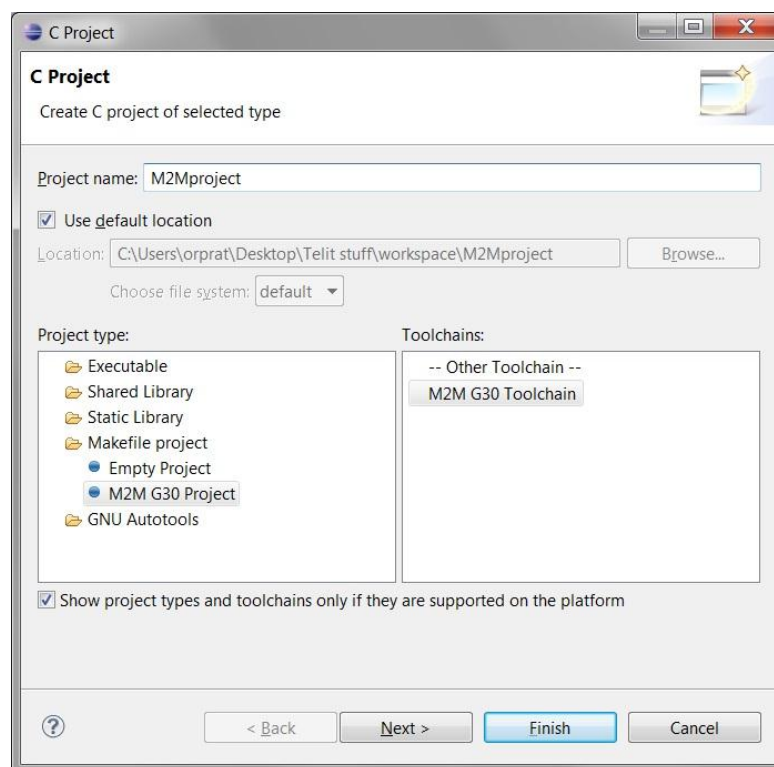
A special Telit AppZone plugin was developed for an easy development of AppZone applications.

The Telit AppZone plugin allows you to use the skeleton files which are automatically generated when creating an AppZone project from the Eclipse IDE.

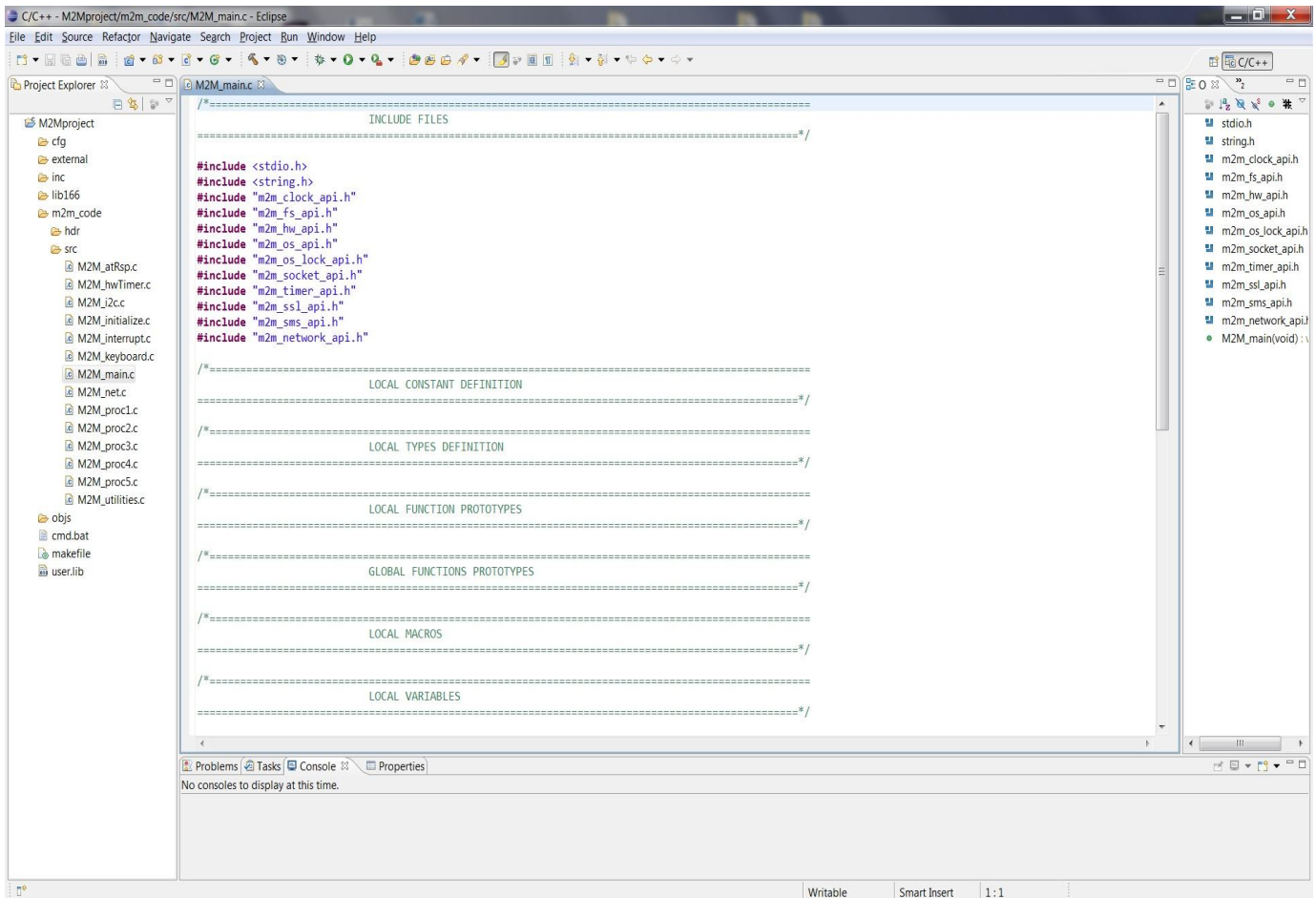
In order to create new AppZone project follow these steps:

1. Run Telit AppZone eclipse.exe (a shortcut will be located on your desktop).  
Or alternatively, lunch it through the Telit AppZone Gadget.  
\* Note that Eclipse depends on the Java runtime environment.
4. Create new C Project --> **Choose Makefile project -> M2M G30 Project -> M2M G30 Toolchain.**

The new AppZone G30 Project wizard will create a new project with all AppZone skeleton files.



- Now you can build and use the AppZone platform from the powerful and known Eclipse IDE (that includes the Telit AppZone plugin - com.telit.AppZone.g30).



- As the developer you can automatically save all opened files before build by checking: **Windows --> Preferences --> General --> Workspace --> "Save Automatically Before Build"**.



## 4.6. Telit AppZone Compiler

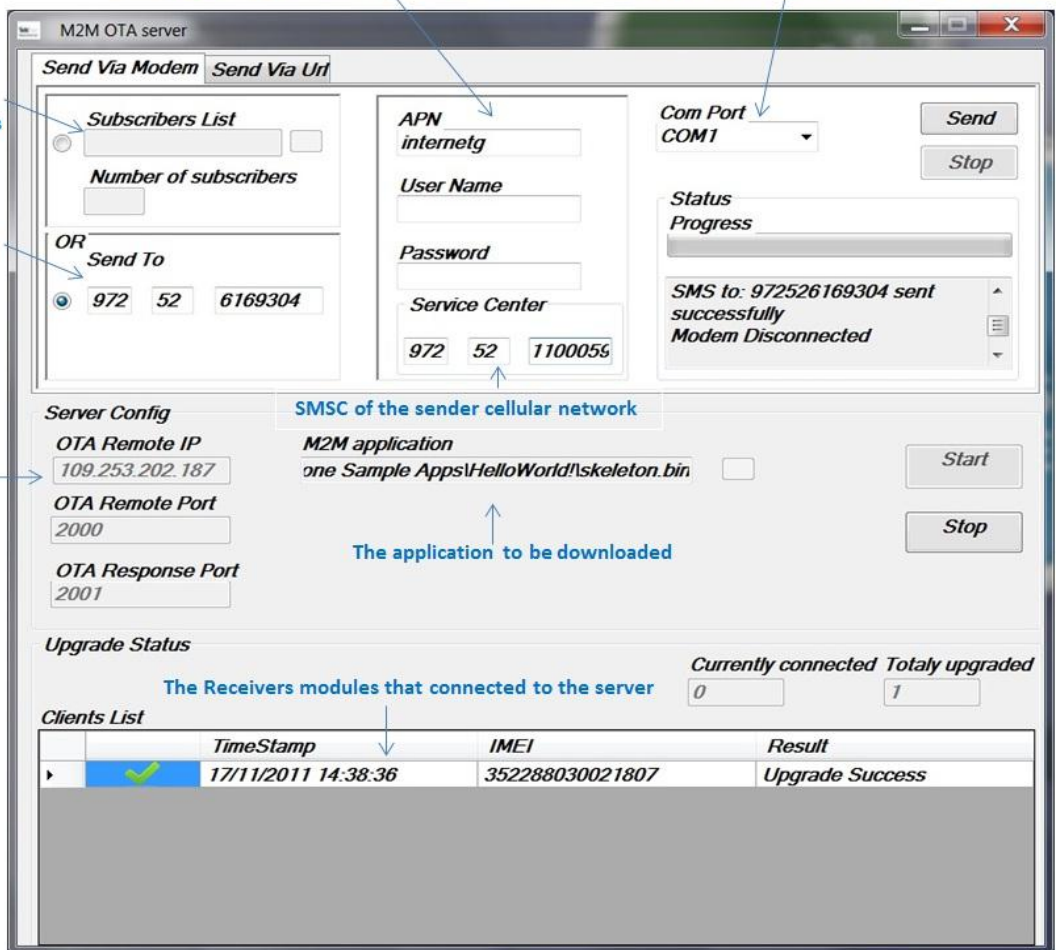
The Eclipse IDE is linked to the C166 compiler in order to compile the AppZone application. The compiler is: Tasking C166 Ver 7.5 R6  
Customer can use Telit's shared compiler server for free (Floating license).  
Customer can purchase a compiler license for its own use.

During the installation, the AppZone server details will be set in your local computer in order to compile your AppZone applications.  
In case you already have a compiler license, you can remove the AppZone server details (7594@84.94.115.140) from the LM\_LICENSE\_FILE which is located in the System Variables (And Restart your computer).





7. Select the APN of the receiving modules.
8. Enter the service center number (SMSC) of the sending module's network.
9. make sure that the receiving modules SIM cards have an external/public IP (verify with the operator).
10. Make sure that the server have an external/public IP.
11. Select the Com Port your sending module is connected to in the 'Com Port' section.
12. Press the Send button in order the send an SMS to the receiving modules you choose.
13. At this point an SMS message will be sent to the receiving modules and they will start downloading the application from the server.  
You can see the download progress in the Clients List table at the bottom.



According to the sender cellular network      The Com port of the sender module's serial connection

For multiple receivers, attach File with the Receivers numbers

The receiver module's number

Server External IP

SMSC of the sender cellular network

The application to be downloaded

The Receivers modules that connected to the server

OTA Remote IP: 109.253.202.187

OTA Remote Port: 2000

OTA Response Port: 2001

APN: internetg

User Name: [empty]

Password: [empty]

Service Center: 972 52 1100056

Com Port: COM1

Send

Stop

Status Progress

SMS to: 972526169304 sent successfully  
Modem Disconnected

Server Config

M2M application: one Sample Apps\HelloWorld!\skeleton.bin

Start

Stop

Upgrade Status

Currently connected: 0

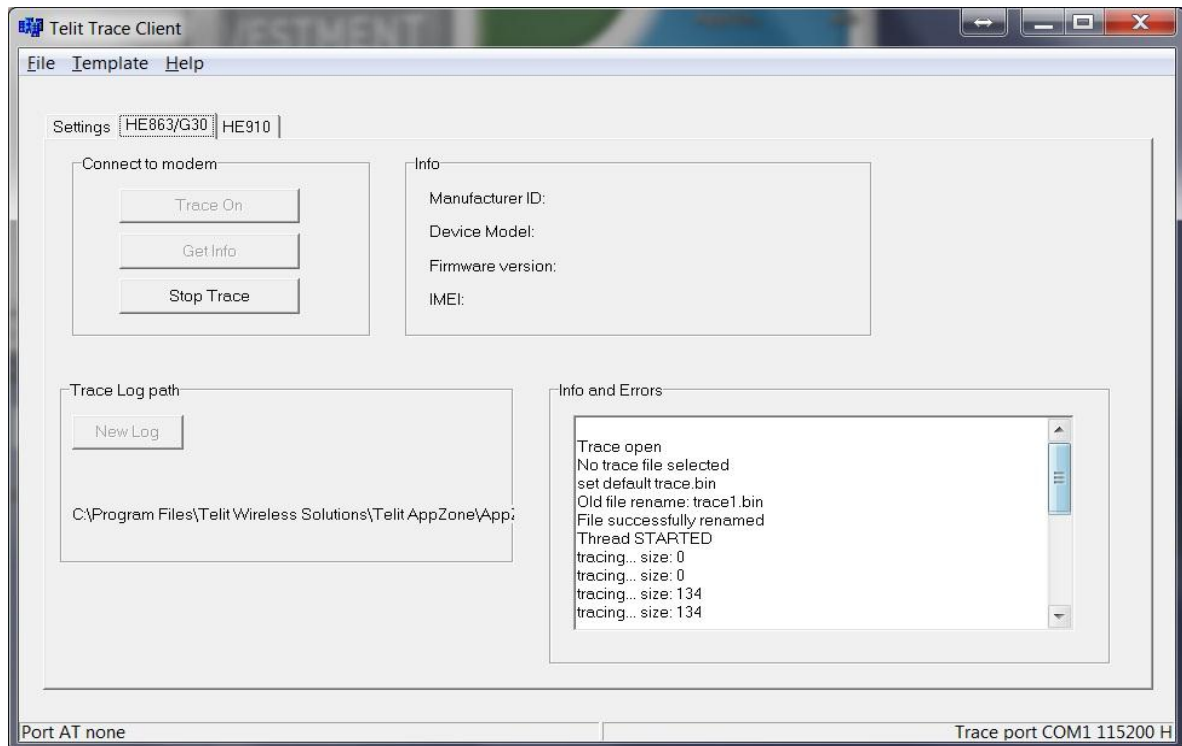
Totally upgraded: 1

Clients List			
	TimeStamp	IMEI	Result
▶	17/11/2011 14:38:36	352288030021807	Upgrade Success

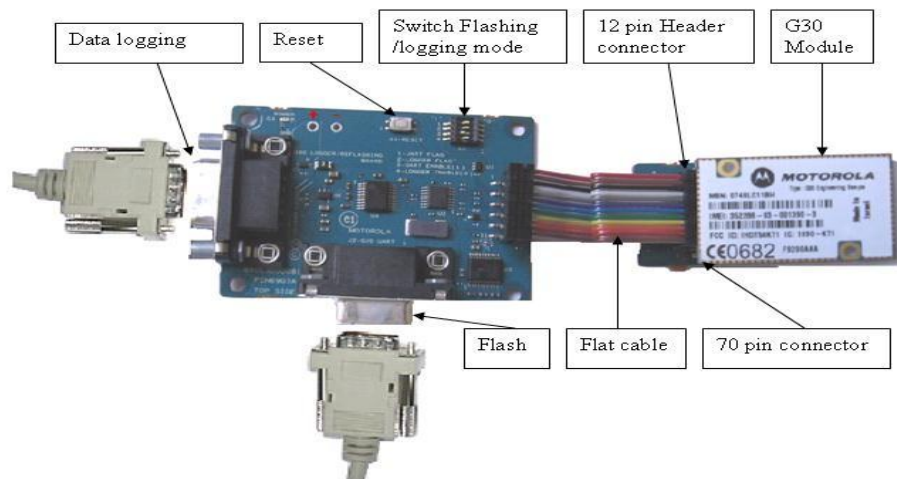
## 4.8. Telit AppZone Trace Tool

The Tracer Tool is used for debugging of the module's behavior.

It allows users to collect the binary traces in a single binary file, to be viewed later with by Telit Personnel using additional tools.



The G30 Logger and flashing board is an Adaptor Board that must be put on top of the Developer Board and the module.



G30 Adaptor Board contains Switch for Flashing/logging states:

Switches	Description	Default states
1	UART FLAG	"1"
2	LOGGER FLAG	"1"
3	UART ENABLE	"1"
4	LOGGER ENABLE	"0"

Note: the adaptor board come with all switched at '1' position.  
Make sure to set it according your design.

To start collecting binary trace, perform the following steps:

- Connect your module to the G30 Adaptor Board
- Set your connections settings in the Settings tab
- Make sure that the module TRACE mode is on (send AT+MTRACE = 1 command to the module).
- Press the Start trace button.
- At this point a binary file will be created in the Trace Tool folder, and the binary traces will be stored in the file.
- To stop the trace collecting, press the 'Stop Trace' button.

For more information, see the [Telit AppZone Tracer Tool User Guide](#) file under [Telit AppZone\AppZone Tools\Telit AppZone Trace Tool](#)



## 4.9. Telit AppZone File System Tool

The Telit AppZone File System Tool allows you to access and manage the G30 module's file system in an easy and comfortable way.

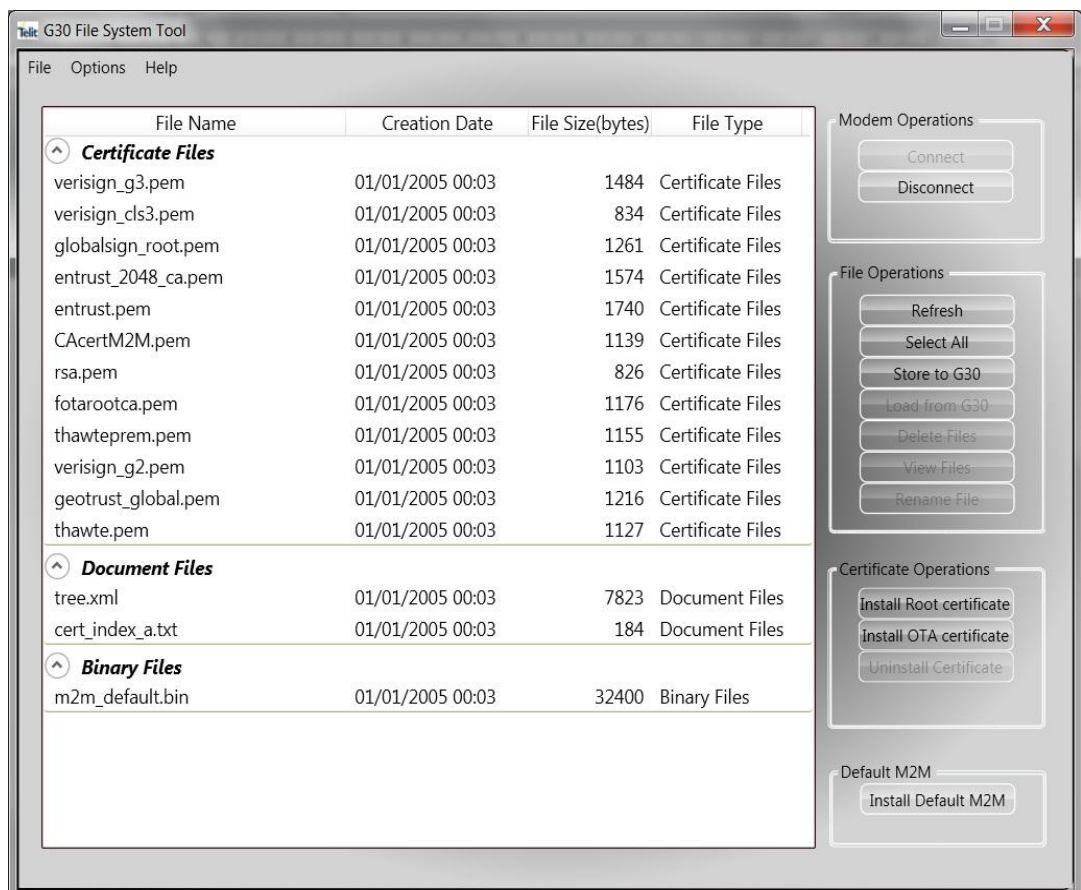
It allows you to load files into the file system, unload files, delete files, see the files content, install client SSL certificates and more.

To store files in your file system, simply press the Store to G30 button and select the file you wish to add to the file system.

To load files from your file system into your computer, select the file you wish to load and press the Load from G30 button.

The m2m\_default.bin file is the module default application, which is restored in case the recovery mechanism discover a memory exception in your application or when you restore M2M app to default using the Telit AppZone Flash Tool.

In case you want to change/edit the module default application, press the install default M2M button and select the binary file of the application you want to become the module's default application.



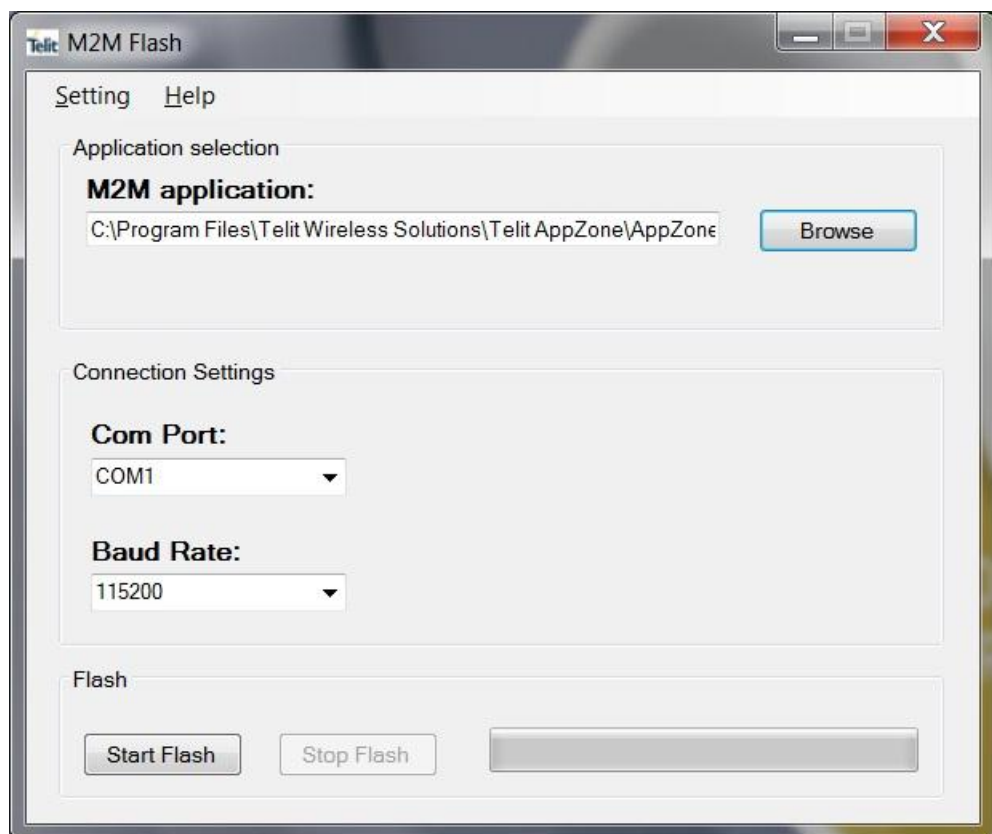


## 5. Using the G30 Telit AppZone application

### 5.1. Flashing an AppZone Application

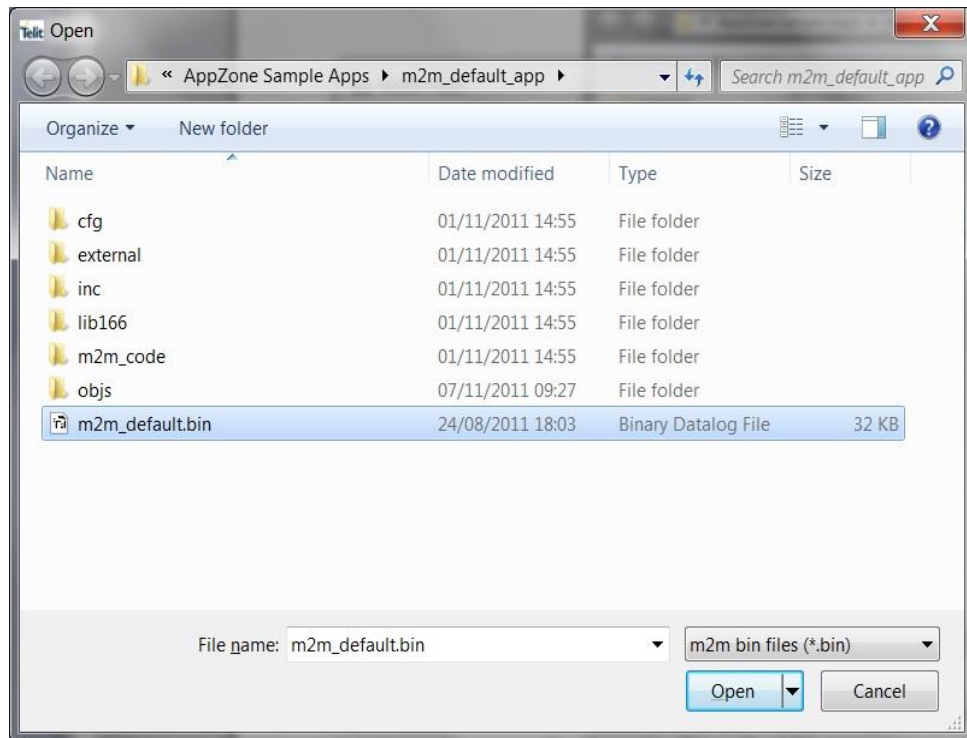
To flash an M2M Application, perform the following steps:

1. Enter the Telit AppZone Flash Tool, by selecting M2M Flash Tool in the Telit AppZone Gadget screen.  
Or alternatively, Launch "M2MGuiFlash.exe" located at <C:\Program Files\Telit Wireless Solutions\Telit AppZone\AppZone Tools\Telit AppZone Flash Tool>.
2. Connect the G30 module to the PC using RS232.



3. Press Browse button and choose the application SW binary file you would like to flash into G30 module





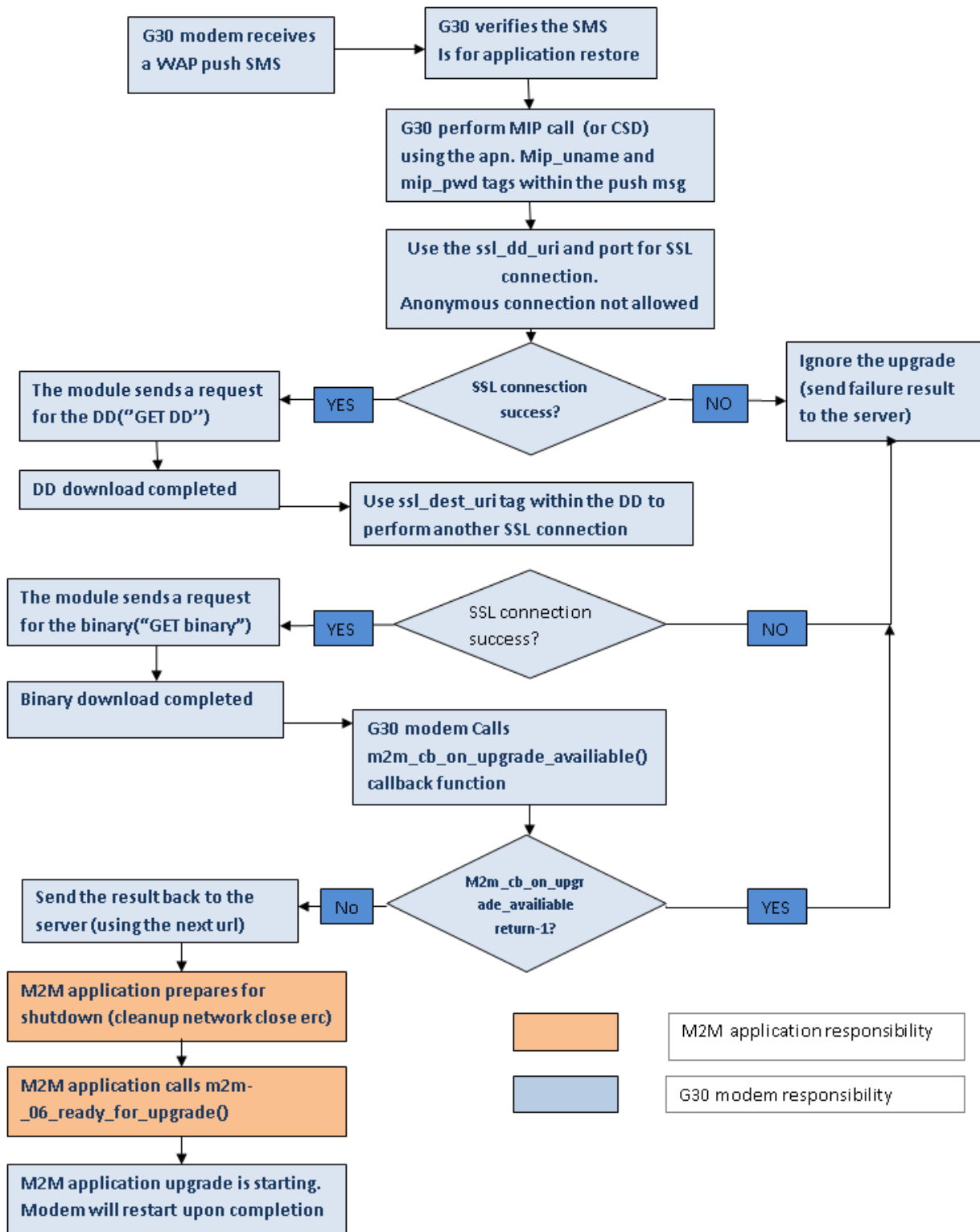
4. Select the COM port G30 is connected to.
5. Select the desired Baud Rate (between 300 and 115200).
6. Press the "Start Flash" button.
7. After a few seconds, Telit AppZone Flash tool shall start flashing the G30 module with the new application.
8. When Flashing is completed, the module will be restarted and the new M2M application will be run on the device.

After M2M application Flash successfully completed:

1. Connect HyperTerminal to G30 (set port settings to 115200,8,None,1,Hardware).
2. Power cycle the G30 (turn it off and on).
3. At this point, the AppZone controls the G30.
4. All data sent to/from the hyper terminal is managed by the M2M application.







**Figure 5: AppZone OTA Application Upgrade Process**



## OTA request for application restore

The module supports restoring its AppZone application to default upon a request made over the air. The server can send a request over the air, asking the module to use a predefined application instead of the current one.

This predefined application will be stored in the file system prior to the request, using the following file name: AppZone\_default.bin

The mechanism is similar to the OTA upgrade, however only the first step is relevant here (WAP push message only, no DD download):

- The server sends a WAP push message to the module. The message format is same as the OTA upgrade, except for the following addition:

<internal-ota>true</internal-ota>

Value needs to be "true" in order to perform internal OTA upgrade.

- The module will attempt to create an SSL connection using the data in the WAP push message.
- If SSL handshake is completed successfully (no anonymous connection is allowed, and fota certificate is used just like OTA upgrade), the connection is closed and the default APPZONE application will be installed.

- The response result will be sent to the next\_url+next\_url\_port fields of the WAP push message (900 for success and 500 for error). Message format is same as OTA upgrade.



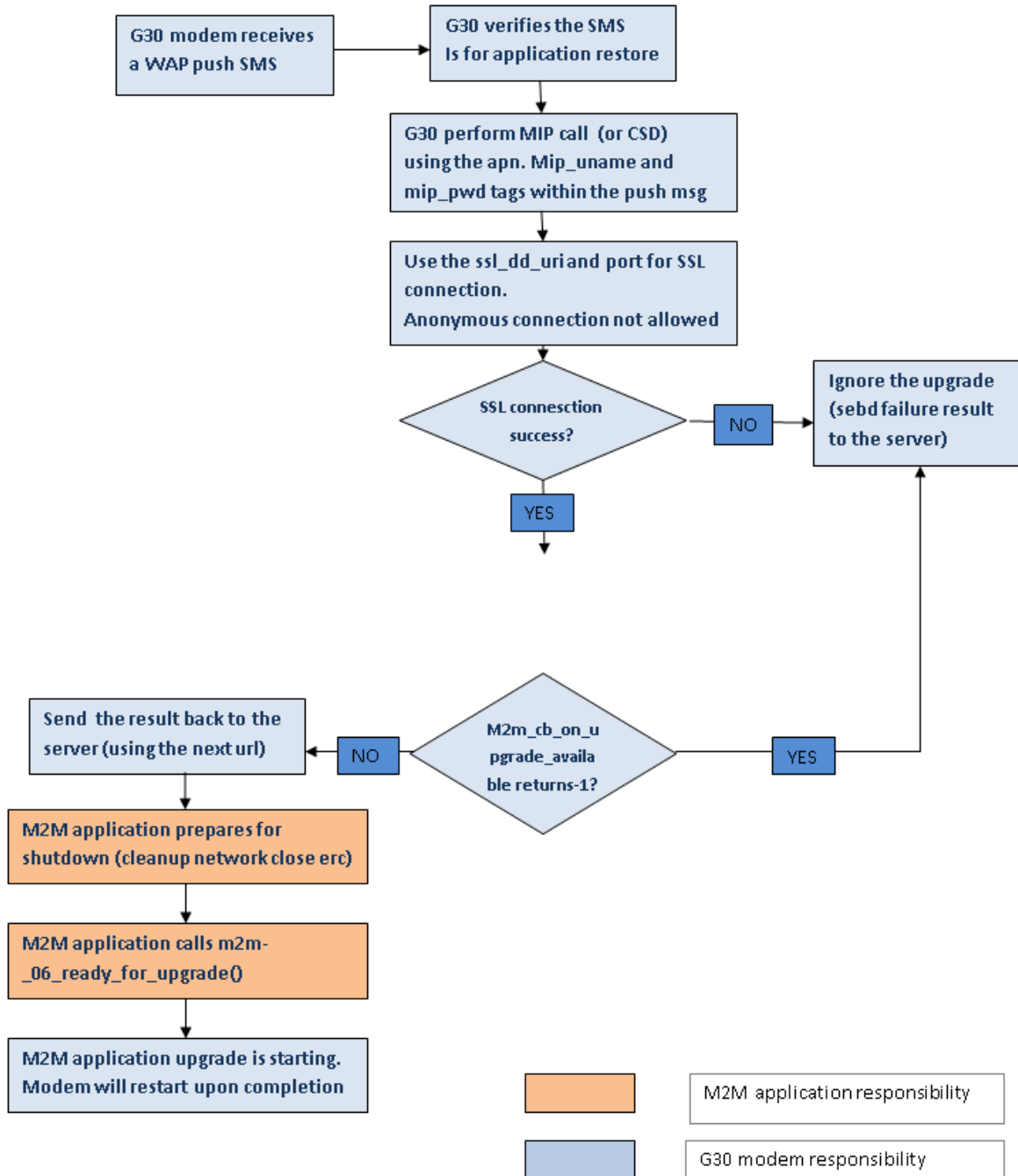


Figure 6: AppZone OTA Application Restore Process



## 6. Appendix A: Capabilities Overview

You can find additional information about the following subjects in the Telit App Zone Developer Guide.

To see the Developer Guide, launch your [Telit AppZone Gadget](#) and select 'Developer Guide'. Or alternatively, go to Telit AppZone\AppZone Documents\Developer Guide Folder and open the Telit AppZone developer guide.chm file.

### 6.1. HW Peripheral

The AppZone provides capabilities to access and control the following HW interfaces:

#### 1. UART

The Telit AppZone G30 module supports one serial UART RS232 interface.

The UART Allows output of both string and binary data through the UART interface. The UART interfaces includes: open, read, write, ioctl and close.

Only one UART client can control the UART at a given time. The UART provides the following ioctl options:

- **Blocking mode**

The UART port can be configured as blocking or non-blocking.

In case of blocking, the read/write function will not return until all data has been read/written. Default is blocking.

- **AT mode**

The UART port can be configured as an AT parser capable. In case of AT mode, any data received by the UART is checked against an AT pattern.

The receive callback, if supplied, will be called only after the whole at command has been received, or in case the data is not in at command format.

Default is off.

- **Receive callback**

The user may specify a callback function to be called when data is received by the UART.

A minimum data size must be specified by the user.

The maximum data to be returned is 1024 bytes.

In case no receive callback is specified, the template receive callback function will be called.

- **HW options get/set**

The user can get or set the UART HW options.

The following combinations are supported for setting the UART HW options:

databits(7), parity(none), stopbits(1,2)

databits(8), parity(none, 1, 2), stopbits(1,2)

databits(9), parity(none), stopbits(1,2)

The following baud rates are supported:

300,1200,2400,4800,9600,19200,38400,57600,115200,230400,460800,921600



## 2. GPIO

Allows configuration of GPIO pins, as well as read/write.

The Telit AppZone G30 module supports up to 14 GPIO's.

Some pins are considered interruptible (62 and 16) and AppZone provides a way to enable/disable interrupts on these GPIO's.

The GPIO ports are configured using the M2M\_initGPIO callback function.

The following GPIO ports can be configured for AppZone application use:

@70 Pin Conn.	I/O	
62	I/O	Interruptible GPIO SPI Interrupt Input. Once configured for APPZONE application use, logger will not be working.
16	I/O	Interruptible GPIO Wake-Up In (I) This GPIO signal is the recommended method to temporarily wake-up G30 from low power mode.
28	I/O	GPIO
30	I/O	GPIO
38	I/O	GPIO
40	I/O	GPIO
42	I/O	GPIO
26	I/O	GPIO Wake-Up Out (O) The wakeup-out (WKUPO) signal is an active low output, which is designed to support a low power mode feature in the host application.
71 (LGA, pad12)	I/O	GPIO
49	I/O	GPIO GPRS (O) The GPRS output signal indicates the network GPRS connection status.
41	I/O	GPIO Antenna Detect (O) Antenna detection which senses the physical connection and removal of an antenna or antenna circuit on the G30 antenna connector.
23	I/O	GPIO Ring Indicator (O) The RI signal (Ring Indication) indicates of an incoming call.
32	I/O	GPIO I2C bus data line
34	I/O	GPIO I2C bus clock line









The following open modes are available:

- **Read**

Allows reading only from the opened file.

- **Append**

Allows writing to the opened file at the end. Writing at the end of file.

- **Modify**

Allows modifying file's data.

- **Write**

Create a file if not exist or truncate to 0 bytes, if exist. Writing at the end of the file.

- **Truncate**

The file will be opened in append mode after being truncated to 0 bytes.

- **Create**

Open a non existing file in append mode.

## 6.4. Clock and Timer

The AppZone provides support for RTC (Real Time Clock) as well as timers.

The current consumption while in RTC is 75  $\mu$ A (while in DRX 9 the current consumption is 1.7 mili A).

- The RTC interfaces include date and time.

In both cases, set and get are supported.

RTC provides an alarm interface. Setting an alarm will enable the device to wakeup (if in sleep mode) or even power-up upon expiration.

Only one alarm can be configured at a given time.

- The timer interfaces includes both SW and HW timers.

Two HW timers are supported.

In both cases (HW and SW), a callback function will be called upon expiration.

## 6.5. Memory IO

AppZone provides memory I/O interfaces.

The heap memory is provided exclusively for the AppZone application use. These interfaces include:

- Memory allocation.
- Memory free.
- Memory re-allocate.

The total pool size is 300KB.

Currently, maximum buffer size (i.e. block size) is 8KB.







name - Sender name. Currently has no affect. version - SW version.  
ssl\_dest\_url - ip based url for downloading the binary file for the upgrade. Mandatory field.  
ssl\_dest\_port - port for use with the ssl\_dest\_url. Mandatory field.  
file\_size - The binary file size. Mandatory field.  
next\_url - The URL (ip based) to be used upon upgrade complete (TCP connection). The module will send the upgrade result to this URL in the following format:  
imei:XXXX result:YYYY  
Where XXXX is the module IMEI and YYYY is the result (900 for success, 500 for error).  
next\_url\_port - The port to be used with next\_url.

### **OS independent**

Creating and compiling of AppZone applications are not dependent in the computer's Operating System and can be done in different OS besides Windows.  
Your AppZone project needs to be linked to the AppZone skeleton and then it can be compiled using command line operations.  
Therefore, it can be done also without using the Eclipse IDE or the Windows based Tools.

