

CE910-SL Software User Guide

1vv0301186 Rev.2 - 2015-11-25



APPLICABILITY TABLE

PRODUCT
CE910-SL

SW Version
22.00.001



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 Italy 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 Italy 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.



Usage and Disclosure Restrictions

License Agreements

The software described in this document is the property of Telit and its licensors. It is furnished by express license agreement only and may be used only in accordance with the terms of such an agreement.

Copyrighted Materials

Software and documentation are copyrighted materials. Making unauthorized copies is prohibited by law. No part of the software or documentation may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, without prior written permission of Telit

High Risk Materials

Components, units, or third-party products used in the product described herein are NOT fault-tolerant and are NOT designed, manufactured, or intended for use as on-line control equipment in the following hazardous environments requiring fail-safe controls: the operation of Nuclear Facilities, Aircraft Navigation or Aircraft Communication Systems, Air Traffic Control, Life Support, or Weapons Systems (High Risk Activities"). Telit and its supplier(s) specifically disclaim any expressed or implied warranty of fitness for such High Risk Activities.

Trademarks

TELIT and the Stylized T Logo are registered in Trademark Office. All other product or service names are the property of their respective owners.

Copyright © Telit Wireless Solutions Co., Ltd.



Contents

1. Introduction	9
1.1. Scope	9
1.2. Audience	9
1.3. Contact Information, Support	9
1.4. Product Overview	9
1.4.1. General Specifications	10
1.5. Document Organization	10
1.6. Text Conventions	11
1.7. Related Documents	11
2. Getting Started	12
2.1. Setting up Service with CDMA Operator	12
2.1.1. CMUX Operational Update	12
3. Basic Operations	13
3.1. Command Syntax	13
3.2. Command Response Timeout	13
3.3. Turning ON the CE910-SL	15
3.4. Turning OFF the CE910-SL	15
3.4.1. Software Shutdown	16
3.4.2. Hardware Shutdown	16
3.4.3. Hardware Reset	16
3.4.3.1. Hardware Unconditional Shutdown	16
3.5. Checking CDMA Device Functionality	18
3.5.1. Baud-rate	18
3.5.2. Network Status	18
3.5.2.1. Query network status	18
3.5.2.2. Common Air Interface parameters (CDMA)	19
3.5.2.3. Check for Received Signal Strength & Quality	20
3.5.2.4. Quick Network Status Checking	21
3.5.2.5. Service status of 1x	22
3.6. Placing a Circuit Switched Data (CSD) call	22
3.7. Windows "System Standby Failed" Error	22
4. Advanced Operations	23



4.1. CDMA Power saving function 23

 4.1.1. Enabling/Disabling the power saving function..... 23

 4.1.2. RI Signal for the specific event on Power Saving Mode..... 25

 4.1.2.1. RI Signal for incoming call 25

 4.1.2.2. RI Signal for incoming SMS 26

 4.1.2.3. RI Signal for socket listen 26

 4.1.2.4. RI Signal for event in power saving mode 27

4.2. SMS 27

 4.2.1. SMS device setup..... 27

 4.2.1.1. Select SMS Message Format 27

 4.2.1.2. Select New Messages indication behavior 27

 4.2.1.3. Select SMS Memory and check for memory space 30

 4.2.2. Writing a New SMS to storage..... 31

 4.2.2.1. PDU Mode Breakdown 32

 4.2.2.1.1.....Coding 7-bit data into octets
34

 4.2.3. Sending a New SMS without Storing It..... 34

 4.2.4. Deleting an SMS..... 36

 4.2.5. Reading an SMS..... 37

 4.2.6. Listing a group of SMSs..... 38

4.3. Using General Purpose Input/Output Pins 41

 4.3.1. GPIO Pin Setup..... 41

 4.3.1.1. Setting a GPIO Pin as OUTPUT 41

 4.3.1.2. Setting a GPIO Pin as INPUT 42

 4.3.2. GPIO Pin Use..... 43

 4.3.2.1. Querying GPIO Pin Status 43

 4.3.2.2. Setting GPIO pin output status 45

 4.3.2.3. Using the GPIO1 Pin as STAT_LED (Alternate Function) 46

 4.3.2.4. Using GPIO6 Pin as ALARM OUTPUT (Alternate Function) 46

4.4. Clock/Alarm Function 47

 4.4.1. Clock Date and Time..... 48

 4.4.1.1. Adjusting the Date and Time 48

 4.4.1.2. Reading the Current Date and Time 49

 4.4.2. Alarm Function..... 50

 4.4.2.1. Setting the Alarm Time and Behavior 50

 4.4.2.2. Stopping the Alarm Activity 51

 4.4.2.2.1.....Exit From the Alarm Status and Shut Down
52



- 4.4.2.2.2.Exit From the Alarm Status and Enter the Normal Operating Mode 52
- 4.4.2.3.Querying the Alarm Status 52
 - 4.4.2.3.1.....Alarm Operation Example 53
- 4.5. Multi-Socket Data Session 55
 - 4.5.1. Connecting to an Echo server using a TCP connection..... 55
 - 4.5.1.1.Configure IP mobility 55
 - 4.5.1.2.Turn on Data Release Notification Message (#DREL) 56
 - 4.5.1.3.Configure the Socket Parameters as Needed 56
 - 4.5.1.4.Verify that no sockets have a current active session 56
 - 4.5.1.5.Activate context 57
 - 4.5.1.6.Connect socket to echo server (XXX.XXX.XXX.XXX) using TCP 57
 - 4.5.1.7.Ensure that Socket Session is Still Open 57
 - 4.5.1.8.Restore socket session 58
 - 4.5.1.9.End socket session 58
 - 4.5.1.10.....Deactivate context 58
 - 4.5.2. Sending and Receiving data to/from an Echo Server while in Command Mode 59
 - 4.5.2.1.Turn on Data Release Notification Message (#DREL) 59
 - 4.5.2.2.Turn off the display of the Escape Character (“+++”) 59
 - 4.5.2.3.Activate context 59
 - 4.5.2.4.Connect socket to echo server (XXX.XXX.XXX.XXX) using UDP 59
 - 4.5.2.5.Send Data in Command Mode 59
 - 4.5.2.6.Check Socket Information, specifically the buffer size ... 60
 - 4.5.2.7.Receive Data in Command Mode 60
 - 4.5.2.8.Close socket connection 61
 - 4.5.2.9.Deactivate context 61
- 4.6. FTP Operations 61
 - 4.6.1. Opening and Closing an FTP Connection..... 62
 - 4.6.2. Setting the FTP Transfer Type..... 62
 - 4.6.3. FTP File transfer to the server..... 62
 - 4.6.4. FTP File download from the server..... 65
- 4.7. Phone as Modem (PAM) 67
 - 4.7.1. External IP Stack..... 67
- 4.8. Email 68
 - 4.8.1. General Information..... 68
 - 4.8.2. Sending an Email..... 68



4.9. Automatic Data/Time updating 69

5. MEID Format and Function 71

5.1. MEID Conversion, HEX to DEC 71

6. Service and Firmware Update 73

6.1. Step-by-Step Upgrade Procedure (TFI) 73

6.2. XFP Tool 76

6.2.1. Step-by-Step Upgrade Procedure..... 76

7. Acronyms and Abbreviations 80

8. Document History 81



1. Introduction

1.1. Scope

This document describes software solutions that can be used to develop a product containing the Telit CE910-SL module, as follows:

- Describing the basic functions of the module
- Suggesting a proper software control solution for each function
- Describing typical wrong solutions and the most common errors to be avoided

This document is not intended to provide an overall description of all software solutions and all products that may be designed.

The solutions suggested serve as a guide or starting point for correctly developing a product with the Telit CE910-SL module.

However, avoiding the wrong solutions and most common errors described here should be regarded as mandatory.

1.2. Audience

This manual is intended for software and applications developers who design products that integrate the CE910-SL module.

1.3. Contact Information, Support

For general contact, technical support, to report documentation errors and to order manuals, contact Telit's Technical Support Center (TTSC) at:

TS-NORTHAMERICA@telit.com, TS-EMEA@telit.com, TS-APAC@telit.com or use <http://www.telit.com/en/products/technical-support-center/contact.php>

For detailed information about where you can buy Telit modules or for recommendations on accessories and components visit <http://www.telit.com>

To register for product news and announcements or for product questions contact Telit's Technical Support Center (TTSC).

Our aim is to make this guide as helpful as possible. Keep us informed of your comments and suggestions for improvements.

Telit appreciates feedback from the users of our information.

1.4. Product Overview

The CE910-SL is a CDMA-1XRTT wireless module designed to have the same form factor of its GSM/GPRS, UMTS/HSPA and EVDO counterpart products of the xE910family: GE910, HE910 and DE910 respectively.



This enables integrators and developers to design their applications once and take advantage of the global coverage and service flexibility allowed by the combination of the most prevalent cellular technologies worldwide.

With its ultra-compact design and extended operating temperature range, the Telit CE910-SL is the perfect platform for low throughput wireless data applications on CDMA 1xRTT network, such as telemetry, telematics, smart metering, health care, tracking applications electronics and security.

1.4.1. General Specifications

Parameter	Description
External access	Code division multiple access
CDMA protocol	CDMA2000 1x
Data Rate	1x: 153.6 Kb/s (full-duplex)
Vocoder	EVRC, 13kQCELP
Operating temperature	-40° - +85°

1.5. Document Organization

This manual contains the following sections:

Section [Introduction](#) provides a scope for this manual, target audience, contact and support information, and text conventions.

Section [Getting Started](#) contains information on how to get started, including provisioning of service.

Section [Basic Operations](#) describes the basic operations of the module.

Section [Advanced Operations](#) provides in-depth information on some of the more advanced operations and commands of the module.

Section [MEID Format and](#) provides information on MEIDs and their representations.

Section [How to update SW](#) provides information on MEIDs and their representations.

Section [Acronyms and Abbreviations](#) provide definitions for all the acronyms and abbreviations used in this guide.



1.6. Text Conventions



Danger – This information **MUST** be followed or catastrophic equipment failure or bodily injury may occur.



Caution or Warning – Alerts the user to important points about integrating the module. If these points are not followed, the module and end user equipment may fail or malfunction.



Tip or Information – Provides advice and suggestions that may be useful when integrating the module.

All dates are in ISO 8601 format, i.e. YYYY-MM-DD.

1.7. Related Documents

The following documents are related to this user guide:

- CE910-SL Product Description
- CE910-SL AT-Command Reference Guide – 80462ST10669A
- CE910-SL Hardware User Guide – 1VV0301187



2. Getting Started

The Telit CE910-SL is a CDMA device operating on the 450 MHz frequency.

Provisioning information is embedded in the module; including the mobile equipment identifier (MEID).

The MEID **cannot** be transferred to another CDMA or GSM device.

The CE910-SL carrier specific modules cannot be transferred to another CDMA carrier.

2.1. Setting up Service with CDMA Operator

All activations begin with contacting the CDMA carrier of choice.

Telit will provide contact information for the carriers upon request.

Before activating on the carrier of choice, the customer will need to provide the MEID(s) of the CE910-SL(s) to be activated. These are listed on the label of the module and can also be displayed using AT commands.

Even though the Telit module is certified on a given carrier, it may be necessary for the integrator to engage carrier certification personnel to obtain active accounts.

2.1.1. CMUX Operational Update



Virus checking software must be turned off before installing and operating the CMUX program. Appropriate exclusions must be added to the virus software.



3. Basic Operations

3.1. Command Syntax

In the next paragraphs the following notations are used:

- <cr> represents the Carriage Return Character (13)
- <lf> represents the Line Feed Character (10)
- <xx> represents a parameter. (<> characters are only for delimiting the parameter and must not be issued to the terminal).
- [<xx>] represents an optional parameter. ([] characters are only for delimiting the optional parameter and must not be issued to the terminal).

3.2. Command Response Timeout

Every command issued to the Telit CE910-SL returns a result response if response codes are enabled (default) (see command ATQn). The time needed to process the given command and return the response varies and may depend on the network also. As a result every command is provided with a proper timeout period. If this time elapses without any result from the operation, the ERROR response is reported.

The timeout period is quite short for internal setup commands, but may be very long for commands that interact with the network (or networks).



NOTE: If no response is received after the timeout has expired, then try repeating the last command and if still no response is received before the timeout, an Unconditional Shutdown **MUST** be issued followed by powering ON the device again.

The table below lists all the commands with timeouts that differ from the default 100 ms:

Command	Time-Out (Seconds)
+CGMI	5
+CGMM	5
+CGMR	5
+CGSN	20
+CSQ	5
+GMI	5
+GMM	5
+GMR	5
+GSN	20
+CSMS	5
+CMGF	5
+CSMP	5
+CNMI	5



Command	Time-Out (Seconds)
+CMGS	180 / 5 for prompt">"
+CMSS	180
+CMGW	5 / 5 for prompt">"
+CMGD	5
+CMGR	5
+CMGL	5
#SHDN	10
#CGMI	5
#CGMM	5
#CGMR	5
#CGSN	20
#CIMI	5

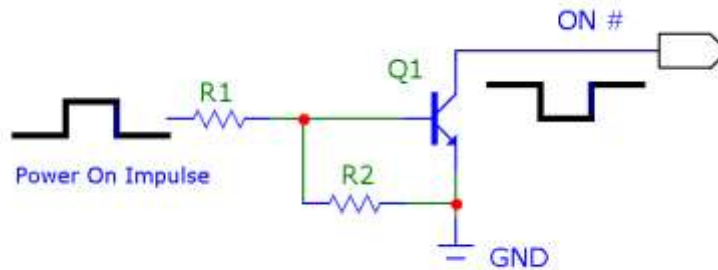


Note: Refer to [3.2.4] Command Response Time-Out of AT commands reference guide document for more information about another AT commands time-out



3.3. Turning ON the CE910-SL

To turn on the CE910-SL the pin ON# must be tied low for at least 1 second and then released. The figure below shows a simple circuit for doing this:



3.4. Turning OFF the CE910-SL

Turning off the device can be done in two ways:

- by software command
- by hardware shutdown

When the device is shut down by software command or by hardware shutdown, it issues to the network a detach request indicating that the device will no longer be reachable.



3.4.1. Software Shutdown

1. Send command AT#SHDN<cr>
2. Wait for response OK

The device shuts down with the following sequence of activities:

1. Detach from the network
2. Module Shutdown

In case the network is unavailable the detach procedure will be attempted for a few seconds (typically 6 seconds). After this timeout the module will be shut down.

3.4.2. Hardware Shutdown

To turn OFF the CE910-SL the pin ON# must be tied low for at least 2 seconds and then released.

The same circuitry for the power ON can be used.

The device shuts down after the release of the ON# pin.



TIP: To check if the device has powered off, the hardware line PWRMON should be monitored. When this line goes low, the device has powered off.

3.4.3. Hardware Reset



WARNING:

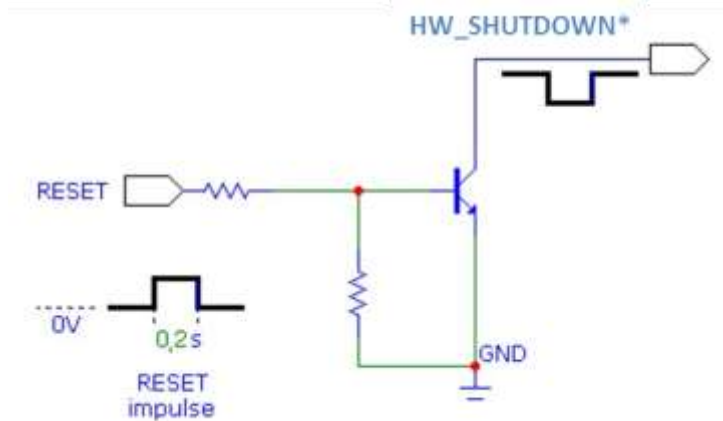
The Hardware Reset must not be used during normal operation of the device since it does not detach the device from the network. It shall be kept as an emergency exit procedure to be performed in the rare case that the device gets stuck waiting for some network message.

3.4.3.1. Hardware Unconditional Shutdown

To unconditionally shutdown the CE910-SL, the pad HW_SHUTDOWN* must be tied low for at least 200 milliseconds and then released.

A simple circuit to unconditionally shutdown the module is illustrated below:





NOTE:

Do not use any pull up resistor on the HW_SHUTDOWN* line or any totem pole digital output. Using a pull up resistor may bring latch up problems on the CE910-SL power regulator and result in improper functioning of the module. The line HW_SHUTDOWN* must be connected only in open collector configuration.

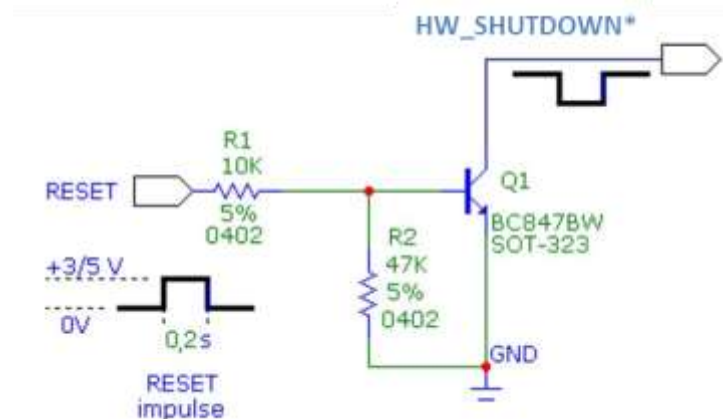


TIP:

The unconditional hardware Shutdown must always be implemented on the boards and the software must use it only as an emergency exit procedure.

For example:

To drive the HW_SHUTDOWN* pad with a totem pole output of a +3/5 V microcontroller (uP_OUT2):



3.5. Checking CDMA Device Functionality

After a proper “power on” the device is ready to receive AT commands on the USB or serial port.

Several things have to be checked in order to be sure that the device is ready for network interaction (data, SMS):

3.5.1. Baud-rate

CE910-SL serial port does not support auto-baud. So the user has to set the correct speed for serial communication before device initialization. If the speed is correct the device responds with OK. The default baud-rate is 115200.

1. Send command AT+IPR=<rate><cr>

where:

<rate> is the port speed and can be 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800 bps.

2. Wait for response OK



TIP: The USB or serial port suggested setting is: port speed 115200, character format 8N1 (8 bit per char, no parity bit, 1 stop bit).

3.5.2. Network Status

3.5.2.1. Query network status

1. Issue command AT+CREG?<cr>
2. Wait for response in format: +CREG: <mode>,<stat>[,<SID>]

where:

<mode> is enable/disable network registration unsolicited result code

0 – Disable network registration unsolicited result code (factory default)

1 – Enable network registration unsolicited result code

2 – Enable network registration unsolicited result code with network system identification data

<stat> is the network registration status

0 – Not registered, not currently searching for a new network operator.

1 – Registered, home network

2 – Reserved

3 – Registration denied

4 – Reserved



5 – Registered, roaming

<SID> is the system identification (only displayed when <mode> is set to “2”)

Example:

AT+CREG?

+CREG: 0,1

OK

3.5.2.2. **Common Air Interface parameters (CDMA)**

Once the module has registered on a network, it is possible to query the module for common network information:

1. Issue command:

AT#CAI?<cr>

2. Wait for response in the format:

<System ID>,<Network ID>,...,<Number of Forward Supplemental Channel>

where parameter value numbers contain information on:

- 0 - Current System ID
- 1 - Current Network ID
- 2 - Current Base Station ID
- 3 - Current Packet Zone ID
- 4 - Current channel number
- 5 - Current Pilot PN Number
- 6 - Current Mobile Station Protocol Revision
- 7 - Current Base Station Protocol Revision
- 8 - Current In Use Protocol Revision
- 9 - Current RSSI Value
- 10 - Current EC/I0 Value
- 11 - Current TX gain
- 12 - Current Rx State
- 13 - Current RX Rate
- 14 - Current TX Rate
- 15 - Current Service Option



- 16 - Current Slot Cycle Index
- 17 - Current Frame Error Rate
- 18 - Current Voice Privacy Mode
- 19 - Current Band

Example

Command:

AT#CAI?<cr>

Response:

#CAI: 4139,41,10323,1,50,216,6,6,6,-90,-9,0,2,0,0,2,0,0,1

OK



NOTE: Since with this command a network scan is performed, some seconds may be required before the output is given.

3.5.2.3. **Check for Received Signal Strength & Quality**

Once the module has registered on a network, it may be useful to know the received signal strength & quality to give the user an indication of the reliability of the network connection.

1. Send command AT+CSQ<cr>
2. Wait for response in the format:

+CSQ: <rss>,<fer>

OK

where:

<rss> is an integer from 0 to 99 that indicates the received signal strength:

<rss> value	Signal strength	Indication (for devices with a User Interface (MMI))
0	-113 dBm or less	Signal is VERY low: at the extreme sensibility limit
1	-111 dBm	MMI may indicate only 1 antenna bar
2	-109 dBm	MMI may indicate only 1 antenna bar
3	-107 dBm	MMI may indicate only 1 antenna bar
4	-105 dBm	MMI may indicate only 1 antenna bar
5	-103 dBm	MMI may indicate only 1 antenna bar
6	-101 dBm	MMI may indicate 2 antenna bars
7	-99 dBm	MMI may indicate 2 antenna bars
8	-97 dBm	MMI may indicate 2 antenna bars
9	-95 dBm	MMI may indicate 2 antenna bars
10	-93 dBm	MMI may indicate 3 antenna bars
11	-91 dBm	MMI may indicate 3 antenna bars



12	-89 dBm	MMI may indicate 3 antenna bars
13	-87 dBm	MMI may indicate 3 antenna bars
14	-85 dBm	MMI may indicate 3 antenna bars
15	-83 dBm	MMI may indicate 4 antenna bars
16	-81 dBm	MMI may indicate 4 antenna bars
17	-79 dBm	MMI may indicate 4 antenna bars
18	-77 dBm	MMI may indicate 4 antenna bars
19	-75 dBm	MMI may indicate 4 antenna bars
20	-73 dBm	MMI may indicate 4 antenna bars
21	-71 dBm	MMI may indicate 4 antenna bars
22	-69 dBm	MMI may indicate 4 antenna bars
23	-67 dBm	MMI may indicate 4 antenna bars
24	-65 dBm	MMI may indicate 4 antenna bars
25	-63 dBm	MMI may indicate 4 antenna bars
26	-61 dBm	MMI may indicate 4 antenna bars
27	-59 dBm	MMI may indicate 4 antenna bars
28	-57 dBm	MMI may indicate 4 antenna bars
29	-55 dBm	MMI may indicate 4 antenna bars
30	-53 dBm	MMI may indicate 4 antenna bars
31	-51 dBm or more	MMI may indicate 4 antenna bars
99	not detected	MMI may indicate flashing antenna bars



NOTE: when <rssI> is less than 6, only 1 MMI antenna bar, the quality of a call will be poor and the call may even drop.

<fer> is an integer from 0 to 7 and 99 that reports the received signal quality measured on the radio traffic channel.



NOTE: The quality is measured on the traffic channel, hence it is available only during a conversation. The reported value in idle mode must not be considered.

In conversation the quality decreases with the increase of the <fer> number.



NOTE: The <fer> value refers strictly to the CDMA radio channel and is a very technical parameter; it can be used to monitor the voice call quality since the voice quality is inversely proportional to the <fer> number.

3.5.2.4. Quick Network Status Checking

Once the module has registered on a network, it may be useful to know the received signal strength and the network on which the mobile is registered.

1. Send command AT#CAI=?<cr>
2. Wait for response format:
<Current System ID>,<Current Network ID>.....,<Current Band>



3.5.2.5. **Service status of 1x**

It is possible to query the service status of a 1x system:

1. Send command AT#SVCSTAT?<cr>
2. Wait for response in the format:
 <1x_svc_stat>,0

where:

<1x_svc_stat> - Status for 1x service.
 0 – No Service State
 2 – Normal Service State

3.6. **Placing a Circuit Switched Data (CSD) call**

CE910-SL doesn't support CSD call.

3.7. **Windows "System Standby Failed" Error**

When the Windows OS enters standby mode with the USB port on the CE910-SL connected with a terminal, Windows can't close the USB device driver. The solution is to change Telit USB port's property to avoid this pop-up error. Follow the steps below to do this.

Steps to take to avoid error message

1. Click "START" button in Windows
2. Select the "Control Panel"
3. Double-click the "System" icon
4. Select the "Hardware" tab
5. Click on the "Device Manager" button
6. Expand the "Modem" tree
7. Select the "Telit HS Modem"
8. Right click the "Telit HS Modem" to access the "Properties" menu
9. Select the "Power Management" tab
10. Check the box labeled "The computer can power this device off to save power"



4. Advanced Operations

4.1. CDMA Power saving function

The Telit CE910-SL provides a special function that reduces the power consumption during the IDLE time. Thus allowing a longer activity with a given battery capacity. The power saving function can be configured by user needs with following power saving function mode.

Power saving function mode

0 - minimum functionality, In this mode the AT interfaces is not accessible. Consequently, once +CFUN=0 AT commands cannot be sent. Any commands will remain in the input buffer and may delay the output of an unsolicited result code. The first wake-up event stops power saving and takes the ME back to full functionality level +CFUN=1 (Default setting)

1 - mobile full functionality with power saving disabled (factory default)

2 - disable TX

4 - disable both TX and RX

5 - mobile full functionality with power saving enabled

Use the following AT command to select the power saving mode in accordance with your Telit Module:

```
AT+CFUN=[<fun>[,<rst>]]
```

Examples

Check the current mode:

```
AT+CFUN?
```

```
+CFUN: 1 → module with full functionality and power saving disabled (factory setting)
```

```
OK
```



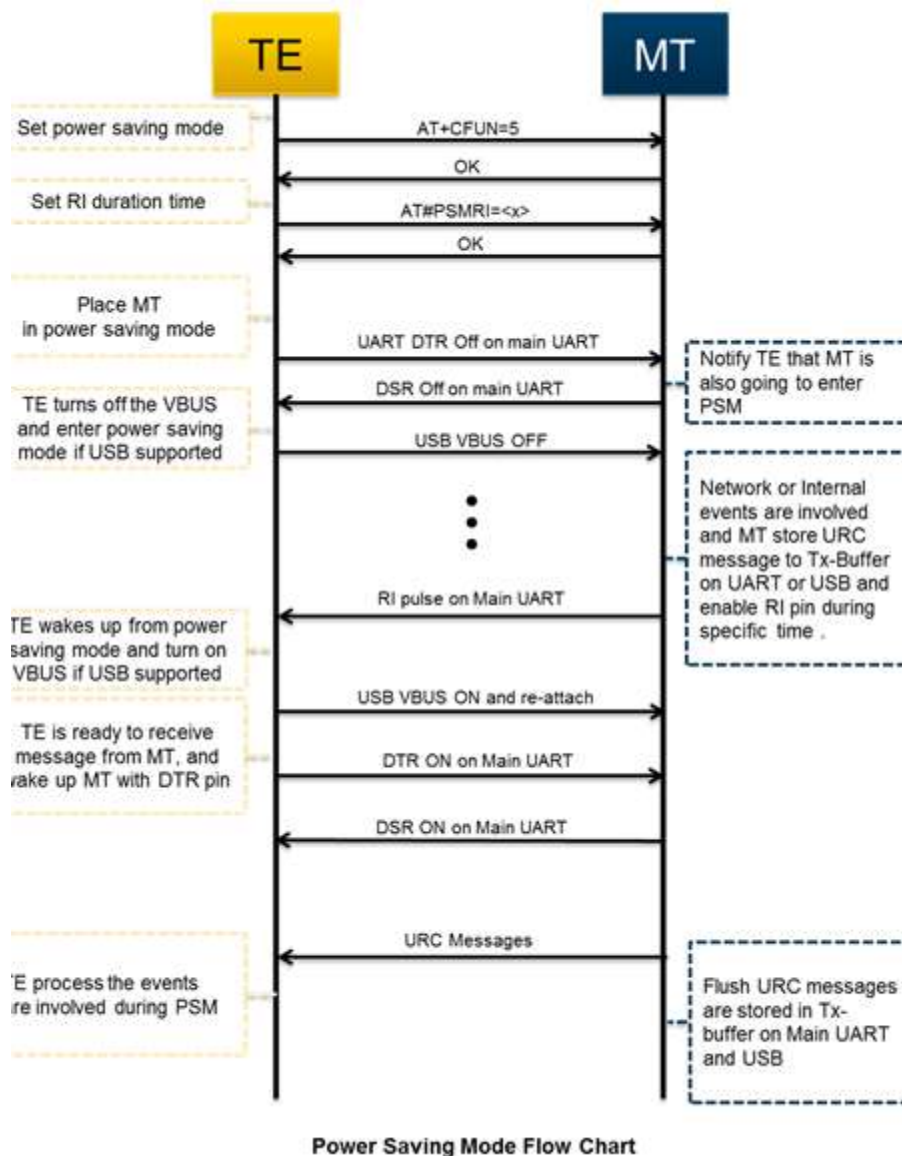
NOTE: URCs and network behavior (incoming calls or SMS) can wake up from CFUN=0 only (RTS not supported)

4.1.1. Enabling/Disabling the power saving function

When DTR (RS232 Pin 4) goes high and USB_VBUS (USB Pin 1) goes low (0v), the OEM application is not going to send any commands and the CE910-SL module can enter the power saving mode.



TE use UART or USB interface shall observe the scenario depicted in below flow chart to place MT in PSM and process the event invoked during PSM



NOTE: In case TE uses both UART and USB as serial interface, TE should do VBUS OFF on USB and DTR Off on UART to place MT in PSM.

NOTE: MT can't keep URC messages more than 10 Kbyte in Tx-buffer. It is recommended that TE shall wake up MT and get URC messages as soon as possible after getting RI pulse from MT.

NOTE: For 2-wire (TX/RX) UART user, use RTS on Main UART to control the flow from modem to DTE is optional (+IFC=2, 2 and #CFLO=1).

NOTE: CE910-SL V_BUS pin should be controlled by application (host or DTE)



NOTE: CE910-SL does not support USB pin emulation in USB AUX port. The customer (application) will have to connect RI/DTR/RTS/CTS) to check pin status.

4.1.2. RI Signal for the specific event on Power Saving Mode

TE can wake up from sleep mode by monitoring RI pin, while it's in sleep mode. MT provides different RI signal type to TE, according to the specific event.

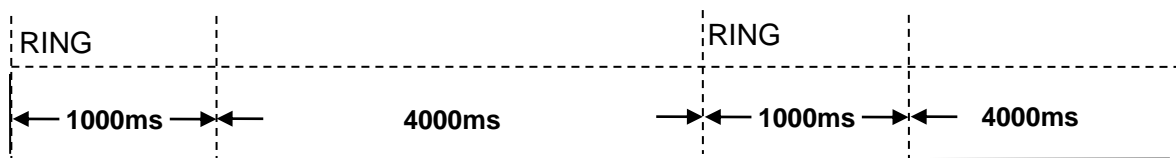
This paragraph deals with following items

- RI Signal for incoming call
- RI Signal for incoming SMS
- RI Signal for socket listen during server mode
- RI Signal for URC message and incoming data during power saving mode

4.1.2.1. RI Signal for incoming call

RI signal for incoming call has different signal in accordance with the value of \R and RING message is sent to TE the instance RI signal is going to is activated, periodically.

In case of the \R value is 0 or 1, RI signal and RING message like as below is generated.



RI signal for \R0 or \R1

In case of the \R value is 2, RI signal and RING message is generated like as below



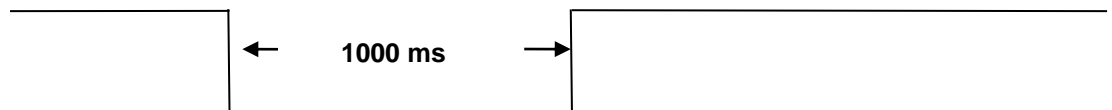
RI signal for \R2



4.1.2.2. RI Signal for incoming SMS

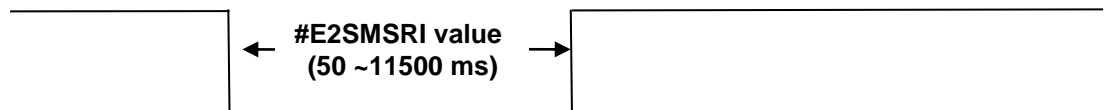
RI signal for incoming SMS has two types in accordance with the value of +CNMI or #E2SMSRI.

In case of +CNMI=3,1, the negative going pulse is generated as below once



RI Signal for +CNMI=3,1

In case of #E2SMSRI = <x>, the negative going pulse is generated as below once



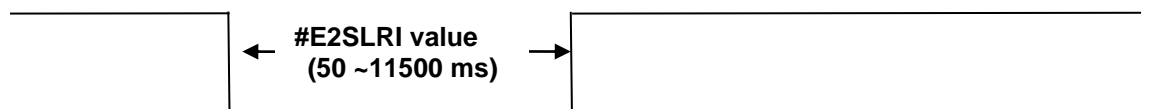
RI Signal for #E2SMSRI



NOTE: In case both +CNMI=3,1 and #E2SMSRI=<x> is issued, RI signal behavior by #E2SMSRI is ignored by +CNMI.

4.1.2.3. RI Signal for socket listen

RI Signal is generated, when modem receive TCP connection request from remote client during socket server mode. This signal is the negative going pulse and is generated, one time.



RI Signal for #E2SLRI





NOTE: The information below written in italics will be present depending on the last setting of +CSDH.

<mode> - unsolicited result codes buffering option

0 - No SMS-DELIVER indications are routed to the TE.

1 - Indication of the memory location is routed to the TE using the following unsolicited result code:

+CMTI: <memr>,<index>

where:

<memr> - memory storage where the new message is stored

"ME"

<index> - location on the memory where SM is stored.

2 - New message is routed directly to the TE using the following unsolicited result code:

PDU MODE:

+CMT: ,<length><CR><LF><pdu>

where:

<length> - Length of message

<pdu> - Message data

TEXT MODE:

+CMT: <orig_num>,<callback>,<date>[,<tooa>,<tele_id>
,<priority>,<enc_type>,<udh>,<length>]<CR><LF><data>

where:

<orig_num> - Origination number

Notice: If origination number is "6245" the message is from Email.

This interim solution puts the Internet e-mail address into the User Data field and uses a special address (6245) to label the message. A space will be used as a delimiter between the Internet e-mail address and the text of the message in the User Data field.

<callback> - Callback number

<date> - Received date in form as "YYYYMMDDHHMMSS"

<tooa> - Type of callback number



<tele_id> - Teleservice ID

4097 - page

4098 - SMS message

<priority> - Priority

0 - Normal (factory default)

1 - Interactive

2 - Urgent

3 - Emergency

<enc_type> - Encoding type of message

0 - 8-bit Octet

2 - 7-bit ASCII (factory default)

4 - 16-bit Unicode

<udh> - user data header

<length> - Length of message

<data> - Message data (Indicates the new voice mail count, if <tele_id> is voice mail notification)

Preparation for the examples

AT+CNMI=?

+CNMI: (0-2)

OK

AT+CNMI=1

OK

AT+CNMI?

+CNMI: 1

OK

+CMTI: "ME",98

AT+CNMI=2

OK

AT+CNMI?

+CNMI: 2



OK

+CMT: "01191775982",20071221163655,129,4098,0,2,0,8

TEST SMS

#SMSFULL

Example 1

To eliminate all the unsolicited codes that may be sent when receiving SMS & Status Report, do the following:

1. Send command AT+CNMI= 0<cr>
2. Wait for response OK

Examples for indicating a new message:

Example 2

If a new SMS delivery (AT+CNMI=1) is received and this new message is stored in the "ME" storage at location number 7; the following unsolicited code (if code is enabled) will be displayed:

+CMTI: "ME",7

Example 3

If a new SMS Status Report delivery (AT+CNMI=2) is received and this new message is stored in the "ME" storage at the location number 8; the following unsolicited code will be displayed:

In case of AT+CMGF=0,

+CMT: "",21

06800968616703091028121550100200020002858C

In case of AT+CMGF=1 and AT+CSDH=1,

+CMT: "9086167630","9086167630",20091028124045,129,4098,0,2,0,2

4.2.1.3. Select SMS Memory and check for memory space

There are various types of storage where the SMS can be stored; the Telit CE910-SL provides only "ME" - mobile equipment memory.

The CE910-SL only allows the user to select a "ME" storage for the read-delete, write-send, and reception-saving SMS operations.



1. Send command:

```
AT+CPMS=<memr>,<memw><cr>
```

where:

<memr>: memory storage for Read and Delete commands

"ME"

<memw>: memory to which writing, sending, and receiving operations are made

"ME"

2. Wait for response in the format:

```
+CPMS:<memr>,<usedr>,<totalr>,<memw>,<usedw>,<totalw>
```

```
OK
```

Where:

<memr> - memory from which messages are read and deleted

<usedr> - number of SMS stored into <memr>

<totalr> - max number of SMS that <memr> can contain

<memw> - memory to which writing and sending operations are made

<usedw> - number of SMS stored into <memw>

<totalw> - max number of SMS that <memw> can contain

From this response whether the selected storage has room for new SMSs can be determined. The free positions in the storage X (where X can be r,w,s) are <totalX> -<usedX>.

4.2.2. Writing a New SMS to storage

A new SMS can be written in the selected storage <memw> (in the current SW version only "ME" is supported) and then can be sent to the desired destination.

To write the new SMS, do the following:

1. Send command AT+CMGW="<da>"<cr>

where:

<da> - destination address

2. Wait for prompt ">"
3. Send SMS text (MAX 160 characters)
4. End command with CTRL-Z character (0x1A hexadecimal) or abort command with ESC character (0x1B hexadecimal)





TIP: If command is aborted with ESC character, then the **OK** result code is returned only.

Example

Write a new SMS to storage and the destination address is the number +39338123456789. Assuming the device is set up for text SMS mode as described in the previous paragraphs.

Command:

AT+CMGW="+39338123456789"

Response:

>

Now insert the message text in IRA format (note that the IRA format and ASCII format coincide for the alphabet characters but not for the others).

...the SMS message text will be inserted here....

Conclude text entry with the character CTRL-Z.

In this case, the new SMS was successfully written to the location index 3 of the selected write memory (always "ME" Mobile Equipment memory).

4.2.2.1. **PDU Mode Breakdown**

AT+CMGW=23 //Send message, 23 octets

>07801091346554F307801096224658F110020002000AD19766CDFA32ECD9BC
<ctrl-z>

There are 23 octets in this message except Length of Destination address.

Octet(s)	Description
07	Length of Destination address.
80	Type-of-Address. (80 indicates national format of the phone number).
1091346554F3	The Destination phone number in semi octets (01194356453). The length of the Destination phone number is odd (7), therefore a trailing F has been added as if the phone number were "1091346554F3". Using the unknown format (i.e. the Type-of-Address 81 instead of 91) would yield the phone number octet sequence 7080523185 (0708251358). Note that this has the length 6, which is even.
07	Length of Callback address.
80	Type-of-Address. (80 indicates national format of the



	phone number).
1096224658F1	The phone number in semi octets (01692264851). The length of the phone number is odd 7, therefore a trailing F has been added, as if the phone number were "46708251358F". Using the unknown format (i.e. the Type-of-Address 81 instead of 91) would yield the phone number octet sequence 7080523185 (0708251358). Note that this has the length 6, which is even.
1002	Teleservice Id
00	Priority. 00 - Normal 01 - Interactive 02 - Urgent 03 - Emergency
02	TP-DCS. Data coding scheme. This message is coded according to the 7-bit default alphabet. Having "00" instead of "02" here, would indicate that the TP-User-Data field of this message should be interpreted as 8-bit rather than 7-bit. Data Coding Scheme: 00: Octet (8-bit) 02: 7-bit – ASCII 04: 16-bit Unicode
00	TP-User-Data-Header 00 : Not present TP-User-Data-Header 01 : Present TP-User-Data-Header
0A	TP-User-Data-Length. The length would be the number of octets or septets from TP-User-Data. If TP-User-Data-Header is present, If encoding type is 7bit ASCII <length> value is the sum of the number of septets in TP-User-Data and the number of septets in TP-User-Data-Header (including any padding). Otherwise, <length> value is the sum of the number of octets in TP-User-Data and the number of octets in TP-User-Data-Header If TP-User-Data-Header is not present, If encoding type is 7bit ASCII, <length> value is the number of septets in TP-User-Data. Otherwise, <length> value is the number of octets in TP-User-Data
D19766CDFA32ECD9BC	TP-User-Data. These octets represent the message "hellohello".



4.2.2.1.1. Coding 7-bit data into octets

The message "hellohello" consists of 10 characters called septets when represented by 7 bits each.

These septets need to be transformed into octets for the SMS transfer.

CDMA does not support GSM 7-Bit encoding scheme.

h	e	l	l	o
104	101	108	108	111
1101000	1100101	1101100	1101100	1101111
1101000	1100101	1101100	1101100	1101111

h	e	l	l	o
104	101	108	108	111
1101000	1100101	1101100	1101100	1101111
1101000	1100101	1101100	1101100	1101111

The first septet (h) is turned into an octet by adding the leftmost bit of the second septet. This bit is inserted to the right which yields 1101000+1 = 11010001 ("D1"). The leftmost bit of the second character is then consumed, so the second character (septet) needs two bits (yellow) of the third character to make an 8-bit octet.

This process goes on and on yielding the following octets:

11010001	10010111	01100110	11001101
D1	97	66	CD

11111010	00110010	11101100	11011001	10111100
FA	32	EC	D9	BC

The 9 octets from "hellohello" are D19766CDFA32ECD9BC, where 00 are padding bits.

4.2.3. Sending a New SMS without Storing It

A new SMS can be sent directly to the network without storing it.

1. send command AT+CMGS="<da>"<cr>
where:
<da> - destination address
2. wait for prompt ">"
3. send SMS text (MAX 160 characters)



4. end command with CTRL-Z character (0x1A hexadecimal) or abort command with ESC character (0x1B hexadecimal)
5. wait for response:

Response	Reason	Action
+CMGS: <mr>	Message has been successfully sent. <mr> represents the message reference number.	proceed ahead
OK		
ERROR	some error occurred	Enable the extended error codes report and retry.
OK	command aborted by user	ESC char was issued

Example

Send a new SMS to the destination address number 9194397977.

Assume the device has been set up for text SMS mode as described in the previous paragraphs:

Command:

AT+CMGS="9194397977"

Response:

>

Now insert the message text in IRA format (note that the IRA format and ASCII format coincide for the alphabet characters but not for the others).

....The SMS message text to be sent will be inserted here....

Conclude text with the character CTRL-Z.

Response:

+CMGS: 4

OK

In this case, the new SMS was successfully sent to the SC and its network reference number is 4.



NOTE: Do not confuse message reference with message index position. The first number indicates the network reference for identifying the sent message (the requested status report



will have the same reference number) while the second number indicates the position where the message has been stored in memory.

4.2.4. Deleting an SMS

An already written/received SMS can be deleted from the selected storage.

To delete the SMS its location index is needed:

1. send command **AT+CMGD=<index>[,<delflag>]<cr>**
where:
 <index>: SMS location index
2. Test command shows the valid memory locations and omits the values of <delflag>.
 - 0 Delete the message specified in <index>
3. wait for response:

Response	Reason	Action
OK	Message has been successfully deleted.	Continue
ERROR	Error occurred	Enable the extended error codes report and retry.

Example 1:

Delete a previously written SMS that was written to the storage index position number 3.

Assume the device for has been set up for text SMS mode as described in the previous paragraphs:

Command:

AT+CMGD=3

Response:

OK

In this case, the SMS was successfully deleted.

Example 2:

Delete a received SMS that was stored to the index position number 7:



Command:

AT+CMGD=7

Response:

OK

4.2.5. Reading an SMS

A new SMS can be read using the following command:

1. Send command **AT+CMGR=<index><cr>**
where:
 <index>: SMS location index
2. Wait for response in the format:
+CMGR: <stat>,<orig_num>,<callback>,<date> [,<toa>,<tele_id>,<priority>,<enc_type>,<udh >,<length>] <CR><LF><data>

where:

<stat> - status of the message

"REC UNREAD" - new received message unread

"REC READ" - received message read

"STO UNSENT" - message stored not yet sent

"STO SENT" - message stored already sent

<orig_num> - origination number

<callback> - callback number

<date> - arrival time of the message to the MC

<toa> - type of number

129 - number in national format

145 - number in international format (contains the "+")

<priority> - Priority

0 - Normal (factory default)

1 - Interactive

2 - Urgent

3 - Emergency



<tele_id> - Teleservice ID

4097: page

4098: SMS message

<enc_type> - Data Coding Scheme

<udh> - User data header

<length> - Text length

<data> - Message_data (Indicates the new voice mail count, if <tele_id> is voice mail notification)



NOTE: In both cases if status of the message is “received unread”, status in the storage changes to ”received read”.

Example

Read the SMS that is stored at the position index 4.

Assuming the device has been set up for text SMS mode as described on the previous paragraphs.

Command:

AT+CMGR=4

Response:

**+CMGR: "REC
READ", "01191775982", "0114933460", 20141023165007, 129, 4098, 0, 2, 0, 8
TEST SMS**

4.2.6. Listing a group of SMSs

An SMS can be grouped into 5 different categories depending on its status:

REC UNREAD	Received messages still not read
REC READ	Received messages already read
STO UNSENT	Stored messages not yet sent
STO SENT	Stored messages already sent



ALL	All types of messages
-----	-----------------------

It is possible to list all the messages in one group:

1. Send command `AT+CMGL=<stat><cr>`

where:

<stat> - status group of the message

"REC UNREAD" - new message

"REC READ" - read message

"STO UNSENT" - stored message not yet sent

"STO SENT" - stored message already sent

"ALL" - all messages

2. Wait for response in the format:

For every message in the group:

```
+CMGL: <index>,<stat>,<oa/da>,<callback>,<date>
[,<tooa/toda>,<tele_id>,<priority>,<enc_type>,<udh>,<length>] <CR><LF><text>
```

where:

<index> - message index position on the storage

<stat> - status of the message

"REC UNREAD" - new message

"REC READ" - read message

"STO UNSENT" - stored message not yet sent

"STO SENT" - stored message already sent

<oa/da> - sender number/destination number

< tooa/toda > - type of number <oa/da>

145 - international number (contains "+" character)

129 - national number

<callback> - callback number

<date> - arrival time of the message to the SC

<tooa> - type of number



129 - number in national format

145 - number in international format (contains the "+")

<priority> - Priority

0 – Normal (factory default)

1 - Interactive

2 - Urgent

3 - Emergency

<tele_id> - Teleservice ID

4097: page

4098: SMS message

<enc_type> - Data Coding Scheme

<length> - text length

<data> - Message_data (Indicates the new voice mail count, if <tele_id> is voice mail notification)



NOTE: If the status of the message is "received unread", status in the storage changes to "received read".

Example

List all the SMSs "received read" that are stored.

Assume the device has been set up for text SMS mode as described in the previous paragraphs.

Command:

AT+CMGL="REC READ"

Response:

+CMGL: 0,"REC READ",,,,,,"01191775982",20071129013659,129,4098,0,2,13
TEST MESSAGE

+CMGL: 1,"REC READ",,,,,,"01191775982",20071217190804,129,4098,0,2,13
TEST MESSAGE1

+CMGL: 2,"REC READ",,,,,,"01191775982",20071221160610,129,4098,0,2,13
TEST MESSAGE2

+CMGL: 3,"REC READ",,,,,,"01191775982",20071224163607,129,4098,0,2,13
TEST MESSAGE3

OK

4.3. Using General Purpose Input/Output Pins

The CE910-SL provides various General Purpose Input/Output pins. These pins can be configured via AT commands as Inputs, Outputs and some of them as "alternate function".

The "alternate function" is supported by the following pins:

- GPIO1, which can be configured to become the STAT_LED output that reflects the modules current network interaction.
- GPIO6, which can be configured to become an alarm output pin that reflects the alarm status

With these pins the application can control external hardware directly using the Telit CE910-SL pins with little or even no hardware added.

4.3.1. GPIO Pin Setup

GPIO pins must be configured to select their direction or alternate function.

4.3.1.1. Setting a GPIO Pin as OUTPUT

When a GPIO is set up as output, the value must be specified that the pin output must take:

1. Send command AT#GPIO=<pin>,<value>,1<cr>

where:

<pin> is the GPIO pin number to which the command applies:

- | | |
|-----------|-------------|
| 1 - GPIO1 | 6 - GPIO6 |
| 2 - GPIO2 | 7 - GPIO7 |
| 3 - GPIO3 | 8 - GPIO8 |
| 4 - GPIO4 | 9 - GPIO9 |
| 5 - GPIO5 | 10 - GPIO10 |

<value> is the GPIO pin value that the pin will assume:

- 0 – LOW
- 1 – HIGH



2. Wait for response OK



WARNING: The #GPIO setting is not saved and will be lost on power off or reset and the GPIOs will be configured as INPUT by default. So at start-up repeat pin initialization commands.

Note that GPIO6 is handled differently from the above. The setting for GPIO6 is maintained even after a shutdown or reset to permit the alarm feature to always work.

Example

Set GPIO3 pin as Output with LOW status:

Command:

AT#GPIO=3,0,1<cr>

Response:

OK

In this case, the GPIO3 pin was successfully put in output direction and its status has been set to LOW.

4.3.1.2. **Setting a GPIO Pin as INPUT**

When a GPIO is set as input, a dummy value for the pin state must be specified:

1. Send command AT#GPIO=<pin>,<dummy_value>,0<cr> where:
 <pin> is the GPIO pin number to which the command applies:

- | | |
|-----------|-------------|
| 1 - GPIO1 | 6 - GPIO6 |
| 2 - GPIO2 | 7 - GPIO7 |
| 3 - GPIO3 | 8 - GPIO8 |
| 4 - GPIO4 | 9 - GPIO9 |
| 5 - GPIO5 | 10 - GPIO10 |

<value> is a dummy value which can be either:

- 0 – dummy value
- 1 – dummy value

2. wait for response OK.





WARNING: The #GPIO setting is not saved and will be lost on power off or reset and the GPIOs will be configured as INPUT by default.

Note that GPIO6 is handled differently from the above. The setting for GPIO6 is maintained even after a shutdown or reset to permit the alarm feature to always work.

Example

Set GPIO4 pin as Input:

Command:

AT#GPIO=4,0,0<cr>

Response:

OK

In this case, the GPIO4 pin was successfully configured for input direction.

4.3.2. GPIO Pin Use

After having set-up the GPIO pin direction the input status of an INPUT pin or set the output status of an OUTPUT pin can be queried.

4.3.2.1. Querying GPIO Pin Status

To query for the pin status:

1. Send command AT#GPIO=<pin>,2<cr>
2. where:

<pin> is the GPIO pin number at which the command applies:

- | | |
|-----------|-------------|
| 1 - GPIO1 | 6 - GPIO6 |
| 2 - GPIO2 | 7 - GPIO7 |
| 3 - GPIO3 | 8 - GPIO8 |
| 4 - GPIO4 | 9 - GPIO9 |
| 5 - GPIO5 | 10 - GPIO10 |

3. Wait for response in the format:
#GPIO: <dir>,<stat>
OK



where:

<dir> - GPIO<pin> direction setting

<stat> - status of the pin

0 - LOW

1 – HIGH



NOTE: In case the GPIO pin direction is set to ALTERNATE FUNCTION (2), then the reported <stat> has no meaning and shall not be regarded as valid and must be treated as a dummy value.



TIP: The query report depends on the pin direction: the actual read pin status if the direction is input; and the previously set pin state if the direction is output. In any case, you can know if the pin at the query moment is high or low and the pin direction.

Example 1

Query the GPIO3 pin for its status:

Command:

AT#GPIO=3,2<cr>

Response:

#GPIO: 0,1

OK

In this case, the GPIO3 pin was set to input direction and its status has been measured to be HIGH.

Example 2

Query the GPIO4 pin for its status:

Command:

AT#GPIO=4,2<cr>

Response:

#GPIO: 1,0

OK

In this case, the GPIO4 pin was set to output direction and its current status is LOW.



Example 3

Query the GPIO6 pin for its status:

Command:

AT#GPIO=6,2<cr>

Response:

#GPIO: 2,0

OK

In this case, the GPIO6 pin was set to "alternate function" direction and therefore works as the alarm output.

The reported status = LOW has no meaning.

4.3.2.2. **Setting GPIO pin output status**

To set the pin status (when pin is set as OUTPUT):

1. Send command AT#GPIO=<pin>,<value>,1<cr>

where:

<pin> is the GPIO pin number at which the command applies:

- | | |
|-----------|-------------|
| 1 - GPIO1 | 6 - GPIO6 |
| 2 - GPIO2 | 7 - GPIO7 |
| 3 - GPIO3 | 8 - GPIO8 |
| 4 - GPIO4 | 9 - GPIO9 |
| 5 - GPIO5 | 10 - GPIO10 |

<value> is the pin value to be set and can be:

- 0 - LOW
- 1 - HIGH

2. wait for response OK

Example

Set the GPIO3 pin HIGH:

Command:

AT#GPIO=3,1,1<cr>



Response:

OK

In this case, the GPIO3 pin was set to output direction and its status has been set to HIGH.

4.3.2.3. Using the GPIO1 Pin as STAT_LED (Alternate Function)

When the GPIO1 is set as STAT_LED function, the pin shows information on the network service availability and call status.

1. Send command AT#GPIO=1,0,2<cr>
2. Wait for response OK

To save the STAT_LED setting for the next power cycle use the &W command.

4.3.2.4. Using GPIO6 Pin as ALARM OUTPUT (Alternate Function)

When GPIO6 pin is set as alarm output function, the pin reports the alarm state according to the +CALA settings.

To set the pin to an alternate function a dummy value must be specified for the pin state:

1. Send command AT#GPIO=6,<dummy_value>,2<cr>

where:

<value> is a dummy value can be either:

- 0 – dummy value
- 1 – dummy value

2. wait for response OK



TIP: Remember that the alternate function places the GPIO6 pin permanently in OUTPUT direction and since the GPIO6 pin value is controlled by the internal software, the corresponding function (+CALA) must be properly setup.



NOTE: The #GPIO6 direction setting is saved and will be kept after a power off or reset.

Example



Set GPIO6 pin as ALARM OUTPUT:

command:

```
AT#GPIO=6,0,2<cr>
```

response:

OK

In this case, the GPIO6 pin was successfully put in alarm output direction.

4.4. Clock/Alarm Function

The Telit CE910-SL provides a Real Time Clock and Alarm embedded in the product. It is therefore possible to set-up the proper time, check the actual time, and set-up an alarm time at which the alarm will be triggered with various behavior depending on the +CALA setting.

The only requirement is that the power input to the Telit CE910-SL has to be guaranteed without interruptions. Since the CE910-SL has no backup battery, it will lose the time setting if its power supply is interrupted.

On Alarm trigger the Telit CE910-SL can:

- Automatically Wake-up fully operative from shutdown as a normal startup.
- Automatically Wake-up from shutdown in a special status, namely "alarm status", where it will not look for or try to register on any network as if it was off, except for the fact that it proceeds with the alarm action and it can receive commands to return completely operative or shutdown immediately.
- If already ON at alarm trigger time, simply proceed with the Alarm action

Once Woken-up the Telit CE910-SL proceeds with the chosen action that can be any of the following options:

- Issue an unsolicited code "+CALA: <user_text>" on the serial port until a 90s timeout expires or a special Wake-up command is received.
- Play an Alarm tone until a 90s timeout expires or a special Wake-up command is received.
- Rise pin GPIO6 until a 90s timeout expires or a special Wake-up command is received.
- Any combination of these actions.



With these features, the Telit CE910-SL can, for example:

- Wake-up itself and its controlling hardware by using the GPIO6 pin at the desired time so timely surveys can be programmed without the need to keep any hardware on, therefore reducing power consumption to a minimum.
- Activate some special hardware on time trigger event with the GPIO6 pin.
- Alert the controlling application that the alarm time has come with the unsolicited code "+CALA:<user_text>"
- Alert the user with the alarm tone played

4.4.1. Clock Date and Time

Before using the Alarm feature, the internal clock must set/adjusted.

4.4.1.1. Adjusting the Date and Time

1. Send command AT+CCLK=<time><cr>

where:

<time> - current time as quoted string in the format: "yy/MM/dd,hh:mm:ss±zz"

where:

yy - year (last two digits are mandatory), range is 00..99

MM - month (two digits are mandatory), range is 01..12

dd - day (two digits are mandatory), range is 01..31 (if the month MM has less than 31 days, the clock will be set for the next month)

hh - hour (two digits are mandatory), range is 00..23

mm - minute (two digits are mandatory), range is 00..59

ss - seconds (two digits are mandatory), range is 00..59

±zz - time zone (indicates the difference, expressed in quarter of an hour, between the local time and GMT; two last digits are mandatory), range is -47..+48

2. Wait for response OK.



NOTE: If the Set command is sent without parameters, the response will be the same as the response for the Read command.



TIP: Remember that the string time has to be encapsulated in (double) quotation marks.



4.4.2. Alarm Function

Once the current time has been set, the alarm function can be set.

4.4.2.1. Setting the Alarm Time and Behavior

1. Send command `AT+CALA="<time>",0,<type>,"<text>"<cr>`

where:

<time> is the Alarm time string of the same format as in the clock set command:

`yy/MM/dd,hh:mm:ss±zz`

where:

yy : two digits year (00-99)

MM : two digits month (01-12)

dd : two digits day (01-31)

hh : two digits hour (00-23)

mm : two digits minute (00-59)

ss : two digits seconds (00-59)

±zz: signed two digits timezone (-47 - +48)

<type> is the Alarm behavior:

0 - Reserved for other equipment use.

1 - The MODULE simply wakes up fully operative as in case of a normal startup. If the device is already ON at the alarm time, then it does nothing.

2 - The MODULE wakes up in "alarm mode" if at the alarm time it was off, otherwise it remains fully operative. In both cases the MODULE issues an unsolicited code every 3s:

`+CALA: <text>`

where:

<text> is the +CALA optional parameter previously set.

The device keeps on sending the unsolicited code every 3s until a #WAKE or #SHDN command is received or a 90s timeout occurs. If the device is in "alarm mode" and it does not receive the #WAKE command within 90s then it shuts down (default).



- Exit from alarm and shutdown the device
- Exit from alarm and enter the normal operational status
- Leave the alarm to go on until the 90s timeout is reached

4.4.2.2.1. Exit From the Alarm Status and Shut Down

1. Send command `AT#SHDN<cr>`
2. wait for response OK

At the OK result code, the device will end any alarm activity and perform a shutdown.

4.4.2.2.2. Exit From the Alarm Status and Enter the Normal Operating Mode

1. Send command `AT#WAKE=0<cr>`
2. wait for response OK

At the OK result code, the device stops any alarm activity and enters normal operating mode. If the device is already in normal operating mode (alarm was triggered when the module was already ON), then this command will terminate the alarm activity only.

4.4.2.3. Querying the Alarm Status

When the device wakes up from an alarm time trigger, the module starts the alarm activity but no network activity, permitting some operations to be done by the controlling application without registering the mobile in the network.

To check it if the module is in the "alarm status" and therefore no network activity is performed or the device is in the normal operating status:

1. Send command `AT#WAKE?<cr>`
2. wait for response in the format:
`+WAKE: <status>`
OK

where:

<status> is the operating mode:

0 - normal operating mode




```
#SS: 3,0
#SS: 4,0
#SS: 5,0
#SS: 6,0
OK
```

4.5.1.5. **Activate context**

1. Send command AT#SGACT=1,1
2. Wait for response OK
3. The IP address should be displayed as #SGACT: XXX.XXX.XXX.XXX

4.5.1.6. **Connect socket to echo server (XXX.XXX.XXX.XXX) using TCP**

1. Send command AT#SD=1,0,XXXXX,"XXX.XXX.XXX.XXX"
2. Wait for response CONNECT
3. Now data could be entered and sent to the echo server.



NOTE: Typed data will not be displayed until it comes back from the server.

4. Use “+++” to suspend the socket session. If data is returned from the echo server while escaped from the session, SRING: 1 will be received.

Example

```
AT#SD=1,0,XXXXX,"XXX.XXX.XXX.XXX" <--- TCP connection to an Echo server
CONNECT <--- Connected to server
Echo test to server <--- Entered text only is displayed when the echo returns
to the module.
Use "+++" to escape socket connection.
OK <--- Successfully suspend socket connection using "+++"
SRING: 1 <--- Notification of data pending on socket 1
```

4.5.1.7. **Ensure that Socket Session is Still Open**

1. Send command AT#SS
2. Wait for response OK



3. A list of the sockets should be displayed and socket one should have information listed.

Example

AT#SS

#SS: 1,3,XXX.XXX.XXX.XXX,40509,XXX.XXX.XXX.XXX,10510

#SS: 2,0

#SS: 3,0

#SS: 4,0

#SS: 5,0

#SS: 6,0

OK

4.5.1.8. Restore socket session

1. Send command AT#SO=1
2. Wait for response CONNECT
3. Once the session is restored and if data is presented , SRING: 1 is visible and now data can continue to be sent to the echo server.
4. Escape the session by using “+++”

4.5.1.9. End socket session

1. Send command AT#SH=1
2. Wait for response OK
3. Verify session has been shut down by sending command AT#SS. The list of sockets should not contain any information.

4.5.1.10. Deactivate context

1. Send command AT#SGACT=1,0
2. Wait for response OK



3. A #DREL notification should be displayed confirming the data session has been released.

4.5.2. Sending and Receiving data to/from an Echo Server while in Command Mode

4.5.2.1. Turn on Data Release Notification Message (#DREL)

1. Send command AT#NOTI=11,1
2. Wait for response OK

4.5.2.2. Turn off the display of the Escape Character ("+++")

1. Send command AT#SKIPESC=1
2. Wait for response OK

4.5.2.3. Activate context

1. Send command AT#SGACT=1,1
2. Wait for response OK
3. IP address should be displayed as #SGACT: XXX.XXX.XXX.XXX

4.5.2.4. Connect socket to echo server (XXX.XXX.XXX.XXX) using UDP

1. Send command AT#SD=1,0,XXXXXX,"XXX.XXX.XXX.XXX"
2. Wait for response CONNECT
3. Now data can be entered and sent to the echo server or go to command mode using "+++"



NOTE: Typed data will not be displayed until it comes back from the server.

Example:

```
AT#SD=1,1,XXXXXX,"XXX.XXX.XXX.XXX" <--- UDP connection to an Echo
server

CONNECT <--- Connected to server

<--- Use "+++" to enter command mode.

OK <--- Successfully suspend socket connection using "+++",
now in command mode
```

4.5.2.5. Send Data in Command Mode

1. Send command AT#SEND=1
2. Wait for response ">"



3. Enter data/text to be sent
4. “Ctrl+Z” to send data
5. Wait for response OK
6. Wait for response SRING:1

Example:

```
AT#SSEND=1                <--- Send data in command mode to socket #1.
> TEST IN COMMAND MODE    <--- "Ctrl+Z" to send
OK
```

```
SRING: 1                  <--- Notification of data arrived
```

4.5.2.6. **Check Socket Information, specifically the buffer size**

1. Send command AT#SI
2. Wait for socket information list and check last parameter for number of new bytes just received in buffer
3. Wait for response OK

Example:

```
AT#SI
#SI: 1,71,49,0,20        <--- 20 new bytes in buffer
#SI: 2,0,0,0,0
#SI: 3,0,0,0,0
#SI: 4,0,0,0,0
#SI: 5,0,0,0,0
#SI: 6,0,0,0, 0
```

OK

4.5.2.7. **Receive Data in Command Mode**

1. Send command AT#SRECV=1,<maxByte>
 - a. <maxByte> is how many Bytes to be read from the buffer
2. Wait for response #SRECV=1,<maxByte>
3. Wait for “Data” from buffer to be displayed
4. Wait for response OK

Example:

```
AT#SRECV=1,20            <--- "#SI" informs us of 20 new bytes in buffer
#SRECV: 1,20
```



TEST IN COMMAND MODE <--- Data is displayed

OK

4.5.2.8. **Close socket connection**

1. Send command AT#SH=1
2. Wait for response OK

4.5.2.9. **Deactivate context**

1. Send command AT#SGACT=1,0
2. Wait for response OK
3. A #DREL notification should be displayed confirming the data session has been released.

4.6. FTP Operations

A set of AT Commands is available to support FTP operations. The first command is called #FTPTO (FTP Time-Out) which defines the time-out for FTP operations. The module already has a factory default time set to 10 seconds.

To modify the setting, the syntax is:

AT#FTPTO=[<tout>]

Where:

<tout> - time-out is 100 ms units

100..5000 – hundreds of ms (factory default is 100)



NOTE: The parameter is not saved in NVM.

Example

Set the timeout to 100 sec.

Command:

AT#FTPTO=1000<cr>

Response:

OK

The timeout has been set to 100 sec.



4.6.1. Opening and Closing an FTP Connection

To open the FTP connection, use the commands below:

```
AT#FTPOPEN=<server:port>,<username>,<password>,<mode>
```

where:

<server:port> - string type, address and port of the FTP server (factory default port21)

<username> - string type, authentication user identification string for FTP

<password> - string type, authentication password for FTP.

<mode> - mode

0 – active mode (default)

1 – passive mode



NOTE: In FTP Open case, the solution dependency limits the maximum time out to 1200 (120 seconds). The FTPTO value that exceeds 1200 is considered as 1200.

To close the FTP connection, use the following command:

```
AT#FTPCLOSE
```

4.6.2. Setting the FTP Transfer Type

To configure the file transfer type, use the command below. The command must be provided during an FTP connection.

```
AT#FTPTYPE=[<type>]
```

Parameter:

<type> - file transfer type:

0 – binary

1 – ASCII



NOTE: The command causes an ERROR result code to be returned if no FTP connection has been opened yet.

4.6.3. FTP File transfer to the server

To send a file to the FTP server during an FTP connection, use the command below:

```
AT#FTPPUT=<filename>
```



where:

<filename> - string type, name under which the file is saved on the server (must have the right extension: ex. If the file you are sending is .txt then the <filename> can be test.txt)

The command opens a data connection and starts sending the file to the FTP server to target file <filename>.

If the data connection succeeds, a CONNECT indication is sent, otherwise a NO CARRIER indication is sent.



NOTE: Use escape sequence +++ to close the data connection.



NOTE: The command causes an ERROR result code to be returned if no FTP connection has been opened yet.

Example

Send a file to the FTP server.

Perform the following steps:

1. Activate the Data Session.

The IP address of the module is provided as a response.

Command:

```
AT#SGACT=1,1<cr>
```

Response:

```
#SGACT: 174.156.82.131
```

```
OK
```

2. Open the FTP connection.

Command:

```
AT#FTPTO=1000<cr> (FTP settings of time-out)
```

Response:

```
OK
```

Command:

```
AT#FTPOPEN="199.188.25.77","user","pass",0<cr>
```



Response:

OK



NOTE: In this case the FTP server port is not specified so the default value 21 is used.

Command:

AT#FTPTYPE=0<cr> (FTP settings of file type)

Response:

OK

3. Start the file transfer to the FTP server.

In this example the target file is named “file.txt”.

Command:

AT#FTPPUT=”file.txt”<cr> (sends the file)

Response:

CONNECT

4. Close the data session.

Command:

+++ (escape sequence +++ to close the data connection)

Response:

NO CARRIER

5. Close the FTP connection.

Command:

AT#FTPCLOSE<cr> <closing FTP connection>

Response:

OK

6. Deactivate the Data Session if required.

Command:

AT#SGACT=1,0<cr>

Response:



14. Now that the “New Connection Wizard” has been completed, select to add a shortcut to the desktop if you wish. Click “Finish” to start using this connection
15. Once the wizard is finished a new window will appear titled “Connect Telit CE910”
16. Ensure that “#777” is in the Dial box, if so click “Dial”
17. If everything is successful an internet connection should be established.

4.8. Email

4.8.1. General Information

The CE910-SL only supports SMTP email service and a commercial grade SMTP provider is recommended.

During testing with free SMTP providers not all the accounts were able to send emails even though they might be with the same provider. This could be due to the fact that spammers prefer to use free SMTP email accounts, thus the SMTP servers may block certain messages.

Hotmail and Gmail accounts will not work on the CE910-SL because they use HTTP and SSL respectively. CE910-SL does not support SSL/TLS capability on top of Email operation

4.8.2. Sending an Email

To send an e-mail, do the following.



NOTE: The steps from step 1 to step 8 describe the SMTP parameter settings.

Enter SMTP Server:

1. Send command AT#ESMTP=”smtp.mydomain.com”<cr>
2. Wait for response OK

Enter Sender Email Address:

3. Send command AT#EADDR=me@email.box.com<cr>
4. Wait for response OK

Enter Email Authentication User Name:

5. Send command AT#EUSER=”myE-Name”<cr>
6. Wait for response OK

Enter Email Authentication Password:

7. Send command `AT#EPASSW="myPassword"<cr>`
8. Wait for response OK

Save Email Parameters:

9. Send command `AT#ESAV<cr>`
10. Wait for response OK

Activate Data Connection:

11. Send command `AT#SGACT=1,1<cr>`
12. Wait for response
13. An IP address will be displayed after a data connection is made.

Compose and Send an Email:

14. Send command `AT#EMAILD="user@domain.com","email subject",0<cr>`
15. Wait for ">"
16. Compose body of email
17. Press "Ctrl+Z" to send email
18. Wait for response OK meaning email was sent successfully.
19. If an ERROR is returned send command `AT#EMAILMSG<cr>`. This will return information on the failed email attempt

Close Data Connection:

20. Send command `AT#SGACT=1,0<cr>`
21. Wait for response OK
22. A #DREL will only be displayed if `AT#NOTI=11,1` was entered before ending the data connection

4.9. Automatic Data/Time updating

Use the following AT command to enables or disables the data/time updating. Not all Operators support this feature.

AT#NITZ=<val>,<mode>

Examples



AT#NITZ?

#NITZ: 7,0

OK

AT#NITZ=15,1 ← enable full data/time updating

OK

AT&W0

OK

AT&P0

OK

Power the module OFF/ON.

After CDMA registration, depending on the Network Provider configuration, on the DTE appears the following unsolicited indication:

#NITZ: 10/11/30,14:36:37+04,0 ← date/time and time zone + daylight saving time

AT+CCLK?

+CCLK: "10/11/30, 14:36:42+04,0" ← date/time and time zone + daylight saving time

OK



5. MEID Format and Function

Telit provides Mobile Equipment Identifier (MEID) as part of the Global Hexadecimal Assignment Guidelines and Procedures

The 56-bit identifier structure is compatible between 3GPP IMEI and 3GPP2 MEID. MEIDs Each MS is assigned a unique MEID. The MEID identifies the manufacturer of the MS.

MEID Structure and Format:

The MEID digit range is hexadecimal and syntactically consistent with the IMEI structure. However, the MEID structure does not utilize all of the fields in the exact semantic manner as in IMEI. The MEID numbering space is allocated in a manner that does not impact the decimally encoded IMEI. The MEID structure is also consistent with the MEID allocation scheme which uses 24-bit Serial Numbers.

The MEID structure:



In the case of MEIDs for terminals conforming exclusively to 3GPP2 technology, all of these fields are defined as hexadecimal values with the following valid range:

RR - valid range A0 ... FF – globally administered

XXXXXX - valid range 000000... FFFFFFFF

ZZZZZZ - valid range 000000.... FFFFFFFF

C - valid range 0... F – not transmitted over the air

The MEID does not specify the frequency band, air-interface technology or supported service associated with the MS

5.1. MEID Conversion, HEX to DEC

To convert the MEID from HEX to DEC manually you will need to break the MEID down into two parts. Break the MEID into ‘first 8 digits’ (part 1) and ‘last 6 digits’ (part 2). Convert the two parts to decimal and then bring the two parts together again to make a single number again. See below example using MEID A1000009D6000D.

A1000009 = 2701131785



6. Service and Firmware Update

The **Telit Modules** firmware is updated through the Serial Interface normally used for the AT Commands.

It is suggested to provide an RS232 interface on the User Printed Circuit Board (where the **Telit Module** is soldered) to perform the physical connection between the **Telit module** and a Windows-based PC. That simple circuitry makes the firmware updating easy when a new firmware version is released.

During the User Application development or evaluation phase of the **Telit module**, the RS232 interface or the USB port implemented on the **Telit Evaluation Kit (EVK2)** [6] can be used to connect the **Telit module** to a Windows-based PC on which a dedicated tool for firmware updating is running.

Telit provides the User with two tools to update the firmware of the module. The following paragraphs describe them.



NOTE :

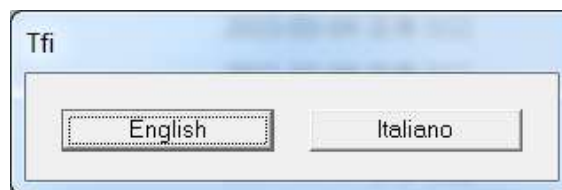
GT terminals are complete encased modems. They do not need the Telit Evaluation Kit (EVK2) to perform testing, evaluation and Firmware Update.

6.1. Step-by-Step Upgrade Procedure (TFI)

The firmware update can be done with a specific software tool provided by Telit that runs on Windows based PCs.

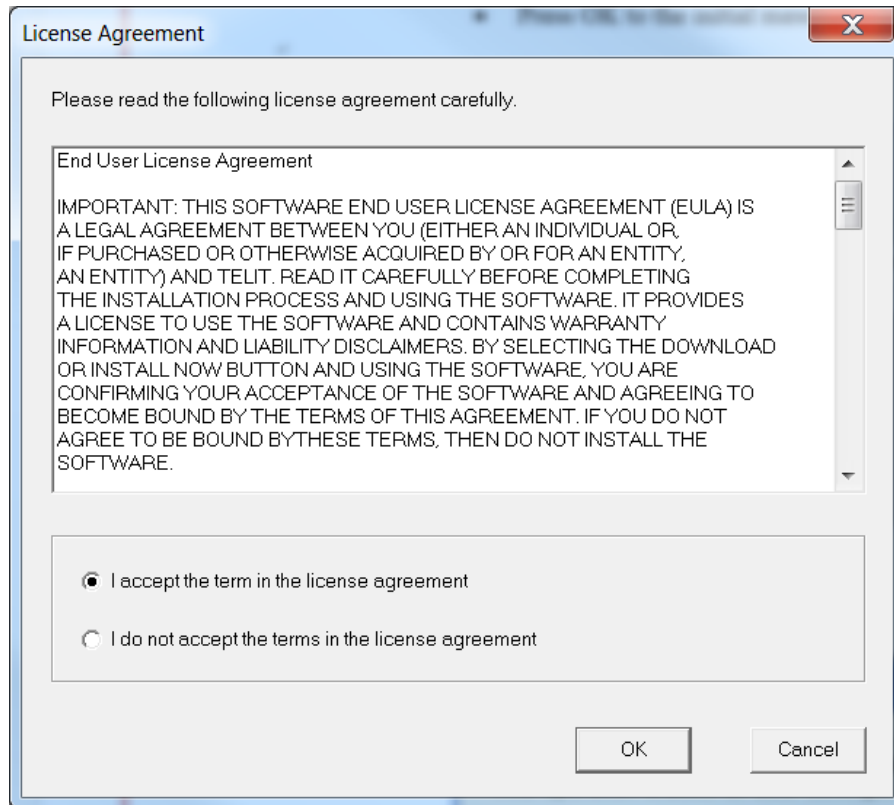
First the program will erase the content of flash memory, and then the program will write on the flash memory. To update the firmware of the module, we suggest the following procedure:

- Run the file **TFI_xxxx.exe**. The following window must be displayed, select the language preferred by pressing the correspondent button.

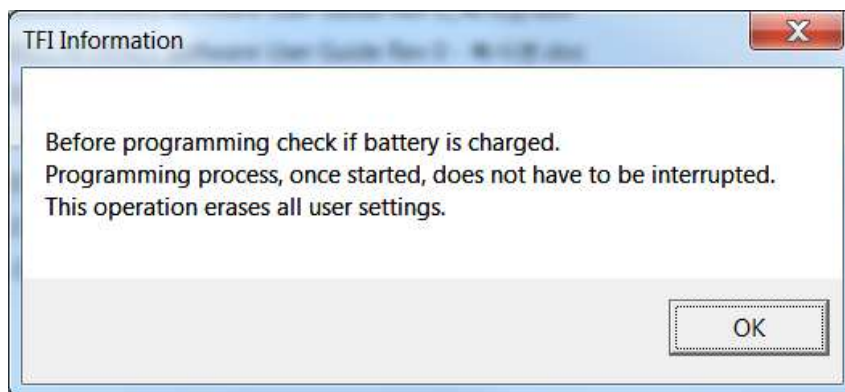


- The End User License Agreement will appear. Please, read it and accept the terms if you are going to proceed.





- Press OK to the initial message.



Note: In connection with the CE910-SL modules, charged battery has to be understood that the power supply must not be disconnected during the firmware update.

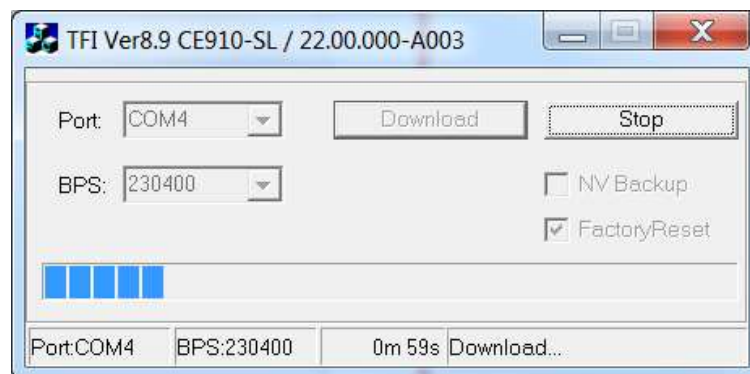
- Download ready screen – If “Diagnostic” COM Port is automatically detected then the baudrate is fixed to 230400. But automatically port detection is depending on Window OS and it is targeting on Window XP. If COM Port is not detecting then must recheck that connecting status of modem USB cable and



USB driver installation. Firmware Version displayed on Title bar is new firmware version and this version will be updated to the module.

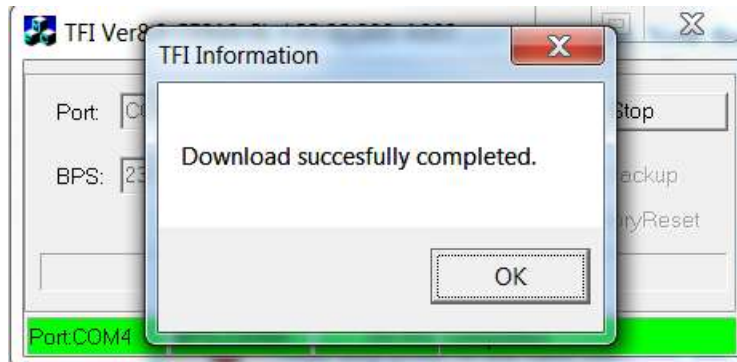


- Select the right COM port and speed. Note that to go faster than 115200 you need a special hardware on the PC. Then Press the Download button and within 5 seconds power-on the CE910-SL.
- Firmware Version displayed on Title bar is current firmware version.



Wait for the end of programming green message OK.





The Telit CE910-SL module is now programmed with the new firmware.

6.2.XFP Tool

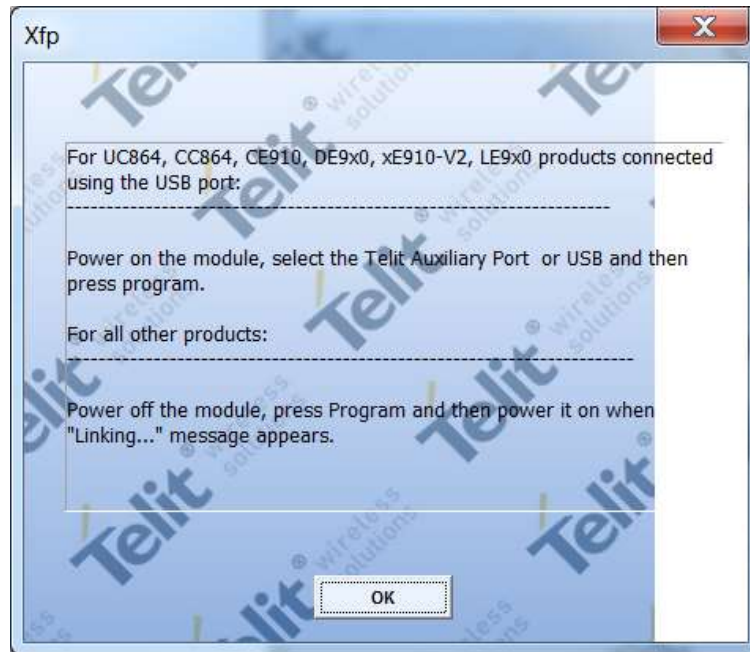
The firmware update of the module can be performed with the Xfp Tool provided by Telit. It runs on Windows based PCs. It erases the flash memory content, and then it downloads the new firmware on the flash memory.

6.2.1. Step-by-Step Upgrade Procedure

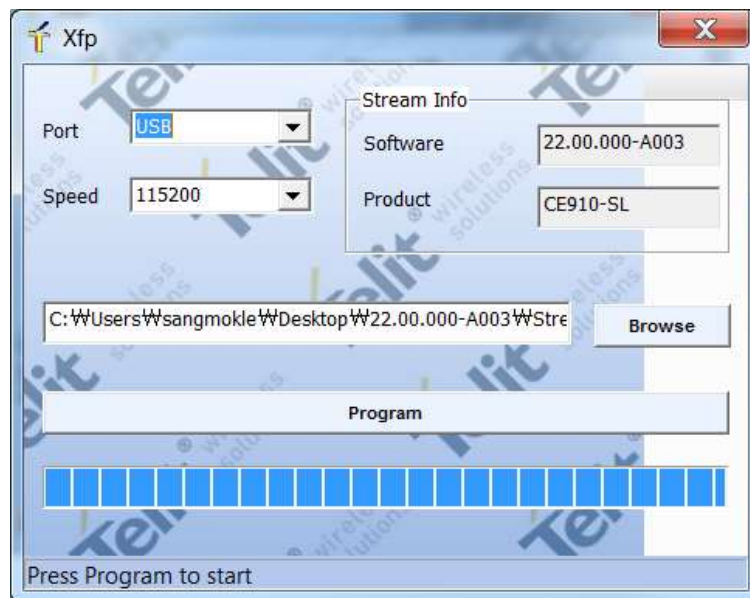
To update the **Telit Module** firmware, follow the procedure:

1. collect information about the **Telit Module** and Software version using the following AT commands:
 - **AT+CGMR<cr>**, returns the Software version information;
 - **AT+CGMM<cr>**, returns the **Telit Module** identification.
2. Switch OFF the Telit Module.
3. Run the *Xfp.exe* tool, the following windows are displayed. Power OFF the module if needed then press OK button.

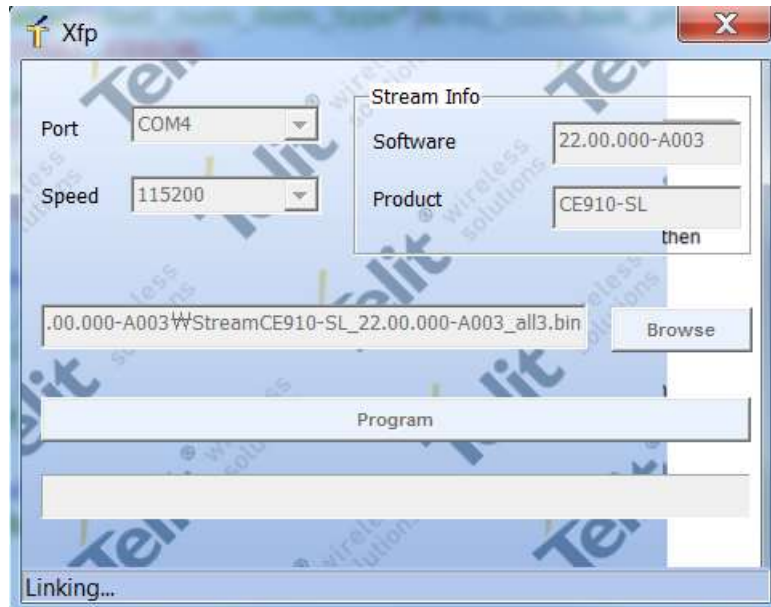




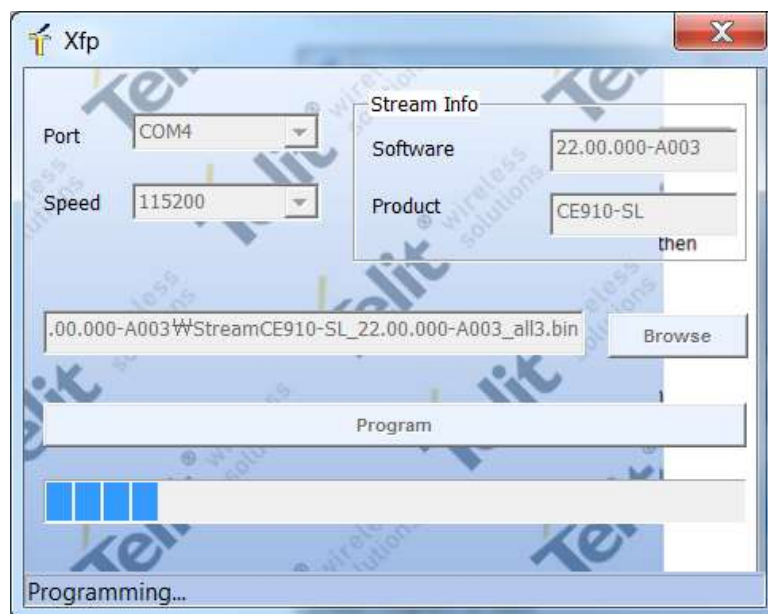
4. After pressing OK button on the screen is displayed only the following windows.



- After selecting COM port, speed¹ and stream file (stream files holds new firmware) press Program button, a flashing blue bar appears on the displayed window. The following window is displayed on the screen.



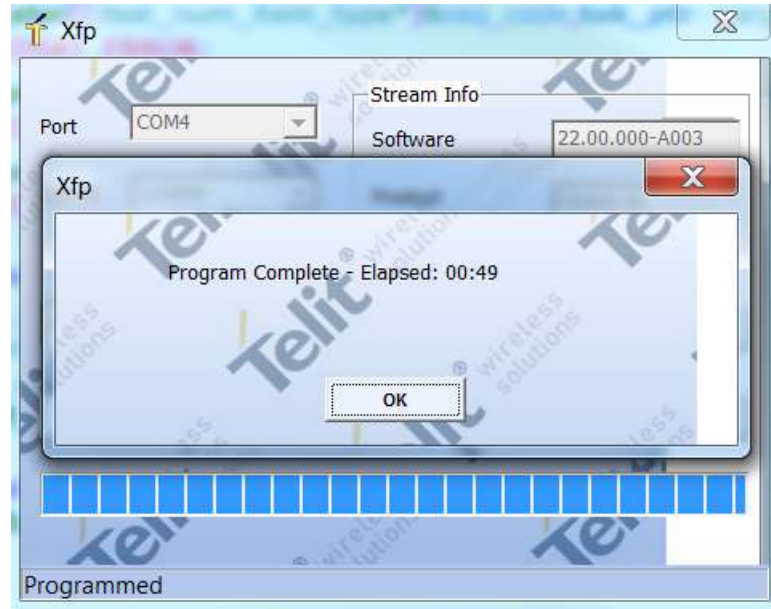
- After powering ON the module the following window is displayed on the screen. The programming is in progress



¹ To use speed greater than 115200 Bps a dedicated hardware on PC is needed.



- The following window is displayed on the screen when the module is successfully programmed.



- Now the [Telit module](#) is programmed with the new firmware. Press OK button and exit the tool.



7. Acronyms and Abbreviations

Term	Definition
ADC	Analog-to-Digital Converter
CDMA	Code Division Multiple Access
DAC	Digital-to-Analog Converter
DTE	Desk Top Equipment
EVRC	Enhanced Variable Rate CODEC
GPIO	General Purpose Input / Output
GPS	Global Positioning System
HF	Hands-free
I2C	Inter-Integrated Circuit
JDR	Jammer Detector
JTAG	Joint Test Action Group(ANSI/ICEEE Std. 1149.1-1990)
ME	Mobile Equipment
MS	Mobile Station
MT	Micro Telephone or HandSet (MT or HS)
PCM	Pulse Coded Modulation
PDM	Pulse Density Modulation (in a DAC)
RTC	Real Time Clock
R-UIM	Removable User Identity Module
S-GPS	Simultaneous-GPS
TA	Terminal Adapter
TE	Terminal Equipment
TGPIO	Telit General Purpose Input / Output
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
VAUX	Voltage Auxiliary
ZIF	Zero Intermediate Frequency



8. Document History

Revision	Date	Changes
0	2014-11-24	Preliminary release for MKT/ENG sample
1	2015-04-07	Updated baud-rate, power saving, GPIO part
2	2015-11-25	Document for Official #1 version

