Telit website Python's scripts

1vv0300808 Rev. 5 2013-05-27

Making machines talk.

Disclaimer

The information contained in this document is the proprietary information of Telit Communications S.p.A. and its affiliates ("TELIT").
The contents are confidential and any disclosure to persons other than the officers, employees, agents or subcontractors of the owner or licensee of this document, without the prior written consent of Telit, is strictly prohibited.
Telit makes every effort to ensure the quality of the information it makes available. Notwithstanding the foregoing, Telit does not make any warranty as to the information contained herein, and does not accept any liability for any injury, loss or damage of any kind incurred by use of or reliance upon the information.
Telit disclaims any and all responsibility for the application of the devices characterized in this document, and notes that the application of the device must comply with the safety standards of the applicable country, and where applicable, with the relevant wiring rules.
Telit reserves the right to make modifications, additions and deletions to this document due to typographical errors, inaccurate information, or improvements to programs and/or equipment at any time and without notice.
Such changes will, nevertheless be incorporated into new editions of this document.

Change Log

Revision	Date	Changes
0	03/03/09	First ISSUE
1	13/02/12	PY scripts are verified to be compatible with HE910. New scripts floattst.py and threadstst.py have been added
2	24/04/12	Split GSM/GPRS script by HE910 scripts.
3	29/05/12	Added PY scripts for GE and HE910
4	21/09/12	Added positioning PY scripts for GE and HE910
5	27/05/13	Added PythonWin 2.7.2 installation

1. Introduction

1.1. Scope

This document gives a first level of support to start working with the Telit Python's scripts and features. The document contains references to the Python Script examples. Examples are organized as follows:

- directory **GSM_GPRS**: scripts can be executed only using Telit GSM/GPRS products supporting Python 1.5.2+ (e.g. GE864, GE865, GL865, GE910-QUAD V3, etc.)
- directory **xE910**: scripts that can be executed only using Telit HE910 & GE910 products (except GE910-QUAD V3)

1.2. Audience

This document is intended for customers who design products that integrate Telit modules and are interested in developing software using the Telit Python's engine and development tools.

1.3. Related Documents

The following documents are related to:

- Telit_Easy_Script_in_Python
- Telit_AT_Commands_Reference_Guide
- Telit_Easy_Script_in_Python_2_7
- Telit_HE910_AT_Commands_Reference_Guide

2. Software Development Tools

The following tools are suggested to start developing with the Telit Python modules:

- PythonWin package 1.5.2+ (for products integrating Python 1.5.2+)
- PythonWin 2.7.2 (for products integrating Python 2.7.2)
- Telit AT Controller or similar AT terminal

2.1. Python installation

2.1.1. PythonWin package 1.5.2+

In order to have software that works correctly the system requirement is PC running Windows 2000 or XP.
To get *PythonWin package 1.5.2+* with the latest version please contacts Technical Support at the email:

TS-EMEA@telit.com
TS-NORTHAMERICA@telit.com
TS-LATINAMERICA@telit.com
TS-APAC@telit.com

or by means the form in Telit's website -> Technical Support -> Contact.

Getting user name and password it is possible to download it from Telit's website -> Download Zone:

http://www.telit.com/en/products/download-zone.php

At the moment, the latest version available is TelitPy1.5.2+_V4.1.exe.

Note: User & Password are available only for Distributors and direct customers.

To install Telit Python package the user needs to execute the exe file TelitPy1.5.2+_V4.1.exe and let the installer use the default settings. The installation contains the Python compiler package.
The Telit Python package is placed in the folder: C:\Program Files\Python\

The correct path in the Windows Environmental variables will be set up automatically.

2.1.2 PythonWin 2.7.2

Download Python 2.7.2 installation from
http://www.python.org/
http://www.python.org/download/releases/2.7.2/
and install it.

Download PythonWin installation related to Python 2.7 from
http://sourceforge.net/projects/pywin32/files/pywin32/
and install it.

Download pyserial package installation from
http://sourceforge.net/projects/pyserial/files/pyserial/
and install it.

A collection of Python modules written in Python (not built-in) that emulates custom built-in modules is available in xE910\Python272LibPC_Emulation directory.
MDM.py
MDM2.py
SER.py
GPIO.py
GPS.py
Copy these files on PC, in the Python\Lib directory.
These modules make the difference between running the script on PC and on module.

3. Python description

Python scripts are text files stored in NVM inside the Telit module. There's a file system inside the module that allows to write and read files with different names on one single level (no subdirectories are supported).

Attention: it is possible to run only one Python script at the time.

The Python script is executed in a task inside the Telit module at the lowest priority, making sure this does not interfere with GSM/GPRS/UMTS normal operations. This allows serial ports, protocol stack etc. to run independently from the Python script.

The Python script interacts with the Telit module functionality through build-in interfaces.

3.1. Scripts description

3.1.1. ab_test.py

Script description: the script tries to open the "test.bin" file. If the file already exists, the program appends some data; otherwise a new file is created with the "wb" parameter

User inputs: none

Availability: any Telit product supporting Python

3.1.2. ADC_test.py

Script description: the script makes a loop with the following operations (the permitted VMax of the input is 2V):

- the script measures the ADC value at the ADC_IN1 pin, by issuing AT#ADC and AT#ADC=1,2,
- using GPIO.getADC prints the result on the debug port
- sleep for 0,5 sec

User inputs: none

Availability: any Telit product supporting Python

3.1.3. fdi.py

Script description: un-expected power loss with file left open

User inputs: none

Availability: any Telit product supporting Python

3.1.4. FSys_mngtst2.py

Script description: the script modifies the 5th byte inside the file test2.txt (size 4000 KB) stored in flash

User inputs: none

Availability: any Telit product supporting Python

3.1.5. FTP.py

Script description: the script shows capabilities of Telit embedded FTP doing upload and download of a text file

User inputs: edit these values in the script

- SIMPIN = '***' *#SIM PIN*
- GPRS_APN = 'xxx.xxxxxxx.xx' *# APN, ask your network provider the correct value*
- GPRS_USER = "" *# GPRS username*
- GPRS_PASSW ="" *# GPRS password*
- FTPSERVER_ADDR = ftp.byte*** *# FTP server name*
- FTPUSERNAME = '*****' *# FTP username*
- FTPPASSWORD = '*****' *# FTP password*
- FTPFILECONTENT = 'testing 123' *# string to be sent the server*

Availability: any Telit product supporting Python

3.1.6. gpiout_tst2.py

Script description: the script sets the GPIO5 as an output and cycles between 1 and 0 values with a clock period of approximately 2.4 seconds.

User inputs: none.

Availability: any Telit product supporting Python

3.1.7. gpsfence.py

Script description: the script find the geographical position using the gps. When the module move away from this position for a distance bigger than the fence radius, it sends a sms.
User inputs: edit these values in the script

```
FENCE_RADIUS = 50              # outside this radius send an sms

TEL_NUMBER = '1111111111'      #  to be changed,tel number to which send an sms when
                               outside the fence
```

Availability: any Telit product supporting Python and GPS

3.1.8. listenSMS.py

Script description: the script receives an unsolicited result code if a SMS is received and print it on the debug com port. Then the script collects the content of the SMS received and prints it on the debug com port.

User inputs: the user should know the mobile number of the SIM card used by the module.

Availability: any Telit product supporting Python

3.1.9. loc_xE910.py

Script description: the script try to find the geographical position, connecting to opencellid.org and sending the values of the module mmc, mnc, cellid and lac. In response the opencellid site return the latitude and longitude.

User inputs: edit these values in the script

- GPRS_APN = 'xxx.xxxxxxx.xx' *# APN, ask your network provider the correct value*
- GPRS_USER = "" *# GPRS username*
- GPRS_PASSW ="" *# GPRS password*

Availability: xE910 supporting Python

3.1.10. otaGE.py

Script description: the script is a simple OTA (Over the Air) update of a python script chosen in the SMS sent to the module. Inside the text of the SMS it is written the name of the python file to be downloaded from a FTP server. After the download, the file is saved, enabled and a reboot of the module follows. All the steps are printed on the debug com port. A SMS message is sent back from the module with the result of the update procedure.

User inputs: the user should know the mobile number of the SIM card used by the module, and send an SMS with the name of the python file to update in it.

Availability: any Telit product supporting Python except the xE910.

3.1.11. ota_xE910.py

Script description: the script is a simple OTA (Over the Air) update of a python script chosen in the SMS sent to the module. Inside the text of the SMS it is written the name of the python file to be downloaded from a FTP server. After the download, the file is saved, enabled and a reboot of the module follows. All the steps are printed on the debug com port. A SMS message is sent back from the module with the result of the update procedure. This script is similar to the otaGE.py

User inputs: the user should know the mobile number of the SIM card used by the module and send an SMS with the name of the python file to update in it.

Availability: xE910 supporting Python

3.1.12. PowerSaving.py

Script description: the script does a loop with the following operations:

- put the module in power saving mode for 5 minutes and then wakes it up
- open a TCP/IP socket and send a string to a server
- sleep for 5 seconds.

User inputs: edit these values in the script

- GPRS_APN = 'xxx.xxxxxxx.xx' *# APN, ask your network provider the correct value*
- GPRS_USER = "" *# GPRS username*
- GPRS_PASSW ="" *# GPRS password*
- SERVER_ADDR = 'xxx.xxx.xxx.xxx' *# Target server address*
- SERVER_PORT = xxxx *# Target server port*
- stringtosend = 'testing 123\r' *# String to be sent the server*

Availability: any Telit product supporting Python

3.1.13. send_email.py

Script description: the script sends e-mail using AT#SEMAIL custom command.

User inputs: edit these values in the script

- GPRS_APN = 'xxx.xxxxxxx.xx' *# APN, ask your network provider the correct value*
- GPRS_USER = "" *# GPRS username*
- GPRS_PASSW ="" *# GPRS password*
- SMTP_SERVER = "xxx.xxx.xxx" *# SMTP server*
- SMTP_USERID = "xxxx@xxx.xxx" *# SMTP username*
- SMTP_PASSW = "xxxxx" *# SMTP password*

- FROM_EMAIL_ADDR = " xxxx@xxx.xxx " *# SMTP From*
- TO_EMAIL_ADDR = " xxxx@xxx.xxx " *# SMTP To*
- SUBJECT = "xxxxxxxxxxxx" *# email subject*
- BODY = "testing 123" *# email text body*

Availability: any Telit product supporting Python

Warning: be aware of using an e-mail account that doesn't use the SSL ciphering.

3.1.14. sendSMS.py

Script description: the script sends an SMS to a destination number in text mode.

User inputs: edit these values in the script

- SMSTO = '072*******' *# destination number*
- SMSTEXT = 'testing 123' *# SMS text*

Availability: any Telit product supporting Python

3.1.15. SER_MDM.py

Script description: the script shows the basic idea for a bi-directional SER to MDM bridge

User inputs: none.

Availability: any Telit product supporting Python

3.1.16. SER_send_rcv.py

Script description: the script receives and sends data on SER interface. The user should write something in the Port Terminal after "Write text:" string

User inputs: none.

Availability: any Telit product supporting Python

3.1.17. SERtest.py

Script description: the script sends a text string 'TEST' to the serial port (ASC0) every 500ms.

User inputs: none.

Availability: any Telit product supporting Python

3.1.18. SKT_CL_SR.py

Script description: the script executes following steps:

- script configuration (IP address, remote & local ports, APN, PIN, PUK, NET_Operator)
- check SIM status, insert SIM PIN or PUK when needed, and check for network registration
- socket configuration and GPRS context activation
- client or server selection based on the input char received ('c' or 's' character to select one of the 2 options)
  - start client mode dialling the remote socket (skt1)
  - start server mode configuring firewall and socket listen (skt2)
- when client to server connection is established, every character sent on the serial port of one module is received by the remote module
- suspend the socket with "+++" sequence
- since MDM-SER bridge is active it is possible to send AT commands (AT#SO=1 to reconnect the socket or AT#SH=1 to close the connection)
- the module is rebooted once DCD low status is detected

User inputs: edit these values in the script

- NETWORK= "xxx" *# this is the MNO name*
- PIN = "" *# SIM card PIN*
- PUK = "" *# SIM card PUK*
- NEW_PIN = "" *# SIM card PIN*
- LOCAL_PORT = xxxxx *# Local server port*
- REMOTE_IP = "xxx.xxx.xxx.xxx" *# Remote server address*
- SUBNET_MASK = "255.0.0.0"
- REMOTE_PORT = "xxxx" *# Remote server port*
- APN = "xxx.xxxxxxxx.xx" *# APN, ask your network provider*

the correct value

Availability: any Telit product supporting Python

3.1.19. sock_MDM2.py

Script description: the script sends data to a server through a TCP socket on MDM while using MDM2 for other AT actions

User inputs: edit these values in the script

- GPRS_APN = 'xxx.xxxxx.xx' *# APN, ask your network provider the correct value*
- GPRS_USER = '' '' *# GPRS username*
- GPRS_PASSW ="" *# GPRS password*
- SERVER_ADDR = 'xxx.xxx.xxx.xxx' *# Target server address*
- SERVER_PORT = xxxx *# Target server port*
- stringtosend = 'testing 123\r' *# string to be sent the server*

Availability: any Telit product supporting Python

3.1.20. suHE910.py

Script description: the script works similar to the Network Survey command #CSURV, not present in the HE910 module AT commands. The script takes some time to execute, it does two survey, one for GSM operators and one for the UTRAN operators using the AT#MONI command.

User inputs: none.

Availability: HE910 supporting Python

3.1.21. testStrArg.py

Script description: the script is to test the string allocation. It seems that no space is allocated in the names list for strings delimited by '' and ending with \r

User inputs: none

Availability: any Telit product supporting Python

3.1.22. traceback_1.py

Script description: the script gets exception details and formats them.

User inputs: none

Availability: any Telit product supporting Python

3.1.23. traceonSER.py

Script description: the script redirects print output to SER interface. It's useful to debug scripts without the need to rely on CMUX.

User inputs: none

Availability: any Telit product supporting Python

3.1.24. floattst.py

Script description: the script shows the usage of the float type

User inputs: none

Availability: xE910 supporting Python

3.1.25. sdio.py

Script description: the script package contains 4 files and performs SD card raw data writing and reading using SPI bus. Files contained in the package are:

- main.py

- sd_defs.py
- sd_raw_drv.py
- sdio.py

User inputs: edit these GPIO values in the script

- SCLK = 9 *# SCLK: GPIO number used as SCLK*
- MOSI = 12 *# MOSI: GPIO number used as MOSI*
- MISO = 11 *# MISO: GPIO number used as MISO*
- SS = 8 *# SS: GPIO number used as SS*

Applicability: GSM/GPRS Telit products supporting Python1.5.2+ (e.g. GE864, GE865, GL865, GE910-QUAD V3, etc.)

3.1.26. watchd_test.py

Script description: the script is a test for the watchdog function. When the "i" loop counter reaches 2000, the module goes to sleep for 70 seconds. After 60 sec. of the 70 sec. the module restarts due to watchdog timeout

User inputs: none

Applicability: GSM/GPRS Telit products supporting Python1.5.2+ (e.g. GE864, GE865, GL865, GE910-QUAD V3, etc.)

3.1.27. threadstst.py

Script description: the script shows the usage of threads allocating dynamically two concurrent threads

User inputs: none

Applicability: xE910 supporting Python