Tra le innumerevoli migliorie implementate da Telit nei modem della famiglia xE910 una delle più importante è l'introduzione dell'interprete Python 2.7. In questo articolo vediamo come cominciare a programmare in Python i nuovi terminalei della Gatetel che integrano il modem HE910.

I modem della famiglia xH910 prodotti da Telit (www.telit.com), disponibili nella versione GPRS e UMTS, oltre ad essere più performanti dal punto di vista di prestazione che di consumi energetici, dispongono di un nuovo interprete Python. Sui nuovi modem è infatti presente il più recente l'interprete Pytnon versione 2.7.2 completo di multithreading e floating point. Funzionalità queste ultime due che limitavano notevolmente lo sviluppo di applicazione nella precedente generazione di modem.
Il nuovo modulo, in versione UTMS, è stato inserito dalla GATETEL nei modem GT-HE910 EUD e EUG. Il primo modem integra il Telit HE910-EUD che può operare in GSM: 850/900/1800/1900MHz ed UMTS/HSPA+: 850/900/2100MHz; il secondo modem dispone del Telit HE910-EUG identico alla versione precedente con in più la funzionalità A-GPS.

In questo articolo, dopo una panoramica hardware dei modem Gatetel GT-HE910, vediamo come iniziare a scrivere e caricare nella memoria del modem, 2 MB, un programma in Python.

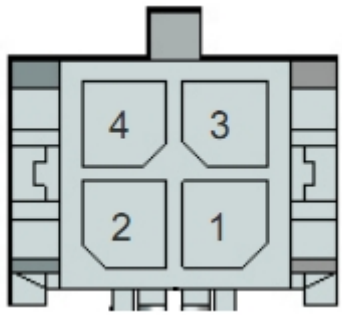
I modem Gatetel GT-HE910-EUx

Il Gatetel GT-HE910-EUx è completo terminale con tecnologia GSM/GPRS e UMTS/HSPA. Ingegnerizzato in un compatto e robusto case di metallo trova posto: un modem Telit HE910-EDU o HE910-EDG (versione con AGPS), un SIM holder di tipo push-push uno, stadio di alimentazione che accetta tensioni da 6V a 50V CC e uno stadio di input output. Quest'ultimo stadio, disponibile su un connettore Molex micro-fit 3mm da 6 poli, fornisce due uscite open-

collector con capacità di 100mA l'una, due ingressi optoisolati e un ingresso ADC da 10Bit che accetta tensioni tra 0 e 55V.

Pin	Descrizione
1	ADC 10bit 0-55V
2	GPIO 2 – OUTPUT Open Collector
3	GPIO 3 – OUTPUT Open Collector
4	GPIO 4 – INPUT
5	GPIO 5 – INPUT
6	GND

I modem della GATETEL si alimentano con una tensione continua compresa tra 6 e 50V da applicare sul connettore Molex micro-fit da 4 poli sito dal alto delle antenne.



Pin	Descrizione
1	POWER
2	IGNITION
3	GND
4	Non connesso

Accanto al Molex da 6 poli sono presenti 2 LED uno verde con etichetta 'POWER' e uno rosso con etichetta 'STATUS CELLULAR'. Il primo LED è attestato alla linea GPIO7 mentre il LED rosso è collegato alla linea STATUS del modem. Entrambe i LED sono gestibili via software Python.

Un semplice script di test in Python

Il programma in Python disponibile nella sezione download di quest’articolo e il cui sorgente è riportato di seguito; mostra come inizializzare e controllare tutti i GPIO del modem GT-HE910-EUx.

Il sorgente Python:

```
import GPIO
import time
import sys
import SER

def INIT_GPIO():
  GPIO.setIOdir(7,0,1)  #Setup output GPIO7 LED green POWER
  GPIO.setIOdir(2,0,1)  #Setup output GPIO2 Pin 2
  GPIO.setIOdir(3,0,1)  #Setup output GPIO2 Pin 3
  GPIO.setIOdir(4,0,0)  #Setup input GPIO4 Pin 4
  GPIO.setIOdir(5,0,0)  #Setup input GPIO5 Pin 5

def PrintDEBUG(txt):
  print txt
  SER.send(txt+'rn')

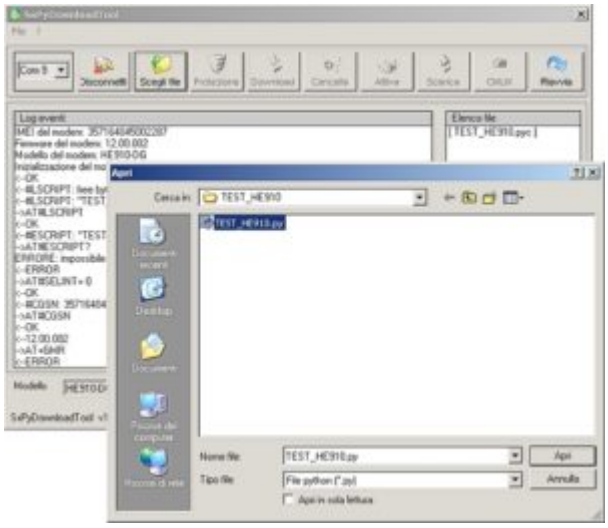
#MAIN
INIT_GPIO()
PrintDEBUG('Test Base Telit HE910')
while(1):
  if(SER.read().find('AT')!=-1): #Blocco  il ciclo se riceve un AT
    GPIO.setIOvalue(7,1) #LED green POWER ON
    sys.exit(1)
  else:
    GPIO.setIOvalue(7,0)  #LED green POWER OFF
  GPIO.setIOvalue(2,1)  #Output Pin 2 attivo
  GPIO.setIOvalue(3,1)  #Output Pin 3 attivo

  PrintDEBUG('Stato Input ----------')
  PrintDEBUG('Input Pin 4: %d' % GPIO.getIOvalue(4) )
  PrintDEBUG('Input Pin 5: %d' % GPIO.getIOvalue(5))
  PrintDEBUG('')
  PrintDEBUG('ADC: ------------------')
  mV = GPIO.getADC(1)
  PrintDEBUG('mV = %d  Vin= %.1f' % (mV, (mV * 0.0488)))
  PrintDEBUG('')
  time.sleep(1.0)
  GPIO.setIOvalue(2,0)  #Output Pin 2 disattivo
  GPIO.setIOvalue(3,0)  #Output Pin 3 disattivo
  GPIO.setIOvalue(7,1) #LED green POWER ON
  time.sleep(1.0)
```

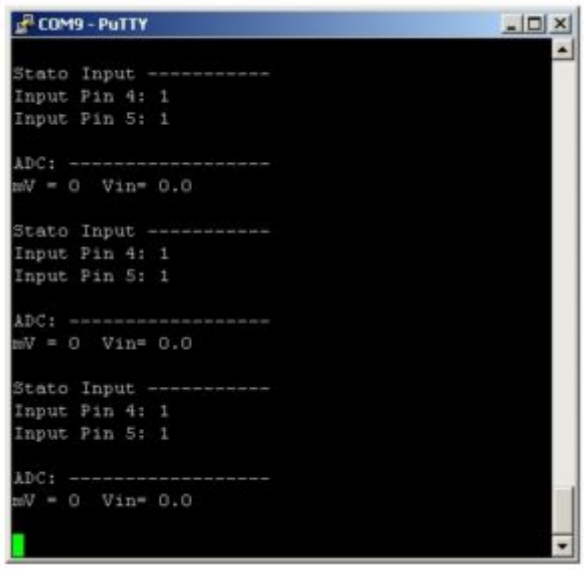
Sulla memoria del modem Telit è possibile caricare il programma direttamente in formato sorgente (.py) o compilato (.pyc). Caricare uno script compilato rende l'avvio di quest'ultimo da parte del modem più veloce.
Per compilare uno script Python procedere come segue:
Scaricare dal sito www.python.org il pacchetto Python 2.7.x compatibile con il proprio sistema operativo ed installarlo su un PC.
Copiare il sorgente in un file di testo e rinominarlo con l'estensione .py es. TEST_HE910.py.
Dal prompt dei comandi compilare lo script con la sintassi che segue:

```
C:\Python27\python.exe -v -S 'C:\Python27\Lib\compileall.py' -l -f
'C:\SCRIPT\TEST_HE910.py'
```

Se non vengono evidenziati errori otteremo come output un file compilato con estensione .pyc es. TEST_HE910.pyc
La procedura per caricare il software python sulla memoria del modulo Telit HE910 è notevolmente semplificata usando il tool
SxPythonDownload scaricabile gratuitamente nella sezione download. Una volta installato il programma SxPythonDownload su
un PC con sistema operativo Windows 2000/XP/Vista/7 è necessario collegare il modem Gatetel alla porta seriale del PC
tramite un cavo seriale a 9 poli dritto. Nel caso il proprio PC non fosse dotato di porta seriale COM è
possibile usare un convertitore USB-Seriale.



Il file TEST_HE910.pyc va selezionato con il tasto ‘Scegli File’, caricato in memoria cliccando sul tasto ‘Download’ e reso eseguibile cliccando sul tasto ‘Attiva’ presente nel software SxPythonDownload. Al riavvio del modem, tasto ‘Reboot’, il lampeggio del led verde indicherà l’esecuzione del programma Python.



L’output del programma è invece disponibile sulla stessa seriale di programmazione aprendo una console.

Link utili

Sito: www.python.org

Sito: www.telit.com

Sito: www.gatetel.com