Overview

The SIM7600E-H 4G HAT is a 4G/3G/2G communication and GNSS positioning module, which supports LTE CAT4 up to 150Mbps for downlink data transfer. It is pretty low power consumption.

You can connect this 4G module with computer to surf the Internet, or attach it onto Raspberry Pi to enable functions like 4G high speed connection, wireless communication, making telephone call, sending SMS, global positioning, etc.

Note: this is a region-specific module, please check the supported bands before placing order.

Features

- Raspberry Pi connectivity, compatible with Raspberry Pi Zero/Zero W/Zero WH/2B/3B/3B+
- Supports dial-up, telephone call, SMS, MMS, mail, TCP, UDP, DTMF, HTTP, FTP, etc.
- Supports GPS, BeiDou, Glonass, LBS base station positioning
- Onboard USB interface, to test AT Commands, get GPS positioning data, and so on
- Onboard CP2102 USB to UART converter, for serial debugging
- Breakout UART control pins, to connect with host boards like Arduino/STM32
- SIM card slot, supports 1.8V/3V SIM card
- TF card slot for storing data like files, messages, etc.
- Onboard audio jack and audio decoder for making telephone call
- 2x LED indicators, easy to monitor the working status
- Onboard voltage translator, operating voltage can be configured to 3.3V or 5V via jumper
- Baudrate: 300bps ~ 4Mbps (default: 115200bps)
- Autobauding baudrate: 9600bps ~ 115200bps
- Control via AT commands (3GPP TS 27.007, 27.005, and V.25TER command set)
- Supports SIM application toolkit: SAT Class 3, GSM 11.14 Release 99, USAT
- Comes with development resources and manual (examples for Raspberry Pi/Arduino/STM32)
Note: Does not contain Raspberry Pi

Communications Specifications

<table>
<thead>
<tr>
<th></th>
<th>LTE</th>
<th>WCDMA / TD-SCDMA / CDMA 2000</th>
<th>EDGE</th>
<th>GSM/EDGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Band</td>
<td>LTE-FDD B1/B3/B5/B7/B8/B20</td>
<td>UNITS/HSPA+ B1/85/B3</td>
<td>CISM/OPRS/EDGE 900/1800 MHz</td>
<td></td>
</tr>
<tr>
<td>Generation</td>
<td>4G</td>
<td>3G</td>
<td>2G</td>
<td>2G</td>
</tr>
<tr>
<td>Emitting power</td>
<td>0.25W</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Speed</td>
<td>LTE CAT 4 Uplink:50Mbps Downlink:156 Mbps</td>
<td>UNITS Uplink:384Kbps Downlink:384Kbps</td>
<td>HSPA+ Uplink:5.76Mbps Downlink:42Mbps</td>
<td>EDGE Uplink:236.8Kbps Downlink:236.8Kbps</td>
</tr>
<tr>
<td>SIM Card</td>
<td>Normal SIM (Not Included)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Applicable Region</td>
<td>Southeast Asia, West Asia, Europe, Africa</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

GNSS Specifications

- Receiver type
  - 16-channel
  - C/A code
• Sensitivity
  ■ Tracking: -159 dBm (GPS) / -158 dBm (GLONASS) / TBD (BD)
  ■ Cold starts: -148 dBm
• Time-To-First-Fix (open air)
  ■ Cold starts: <35s
  ■ Hot starts: <1s
• Accuracy
  ■ Position: <2.5m CEP

### SMS and Audio Specifications

• SMS
  ■ Supported types: MT, MO, CB, Text, PDU
  ■ Storage: USIM card and ME (default)
• Audio feature
  ■ Supports echo cancellation
  ■ Supports noise reduction

### Other Specifications

• Power supply: 5V
• Operating voltage: 5V/3.3V (configured via jumper)
• Operating temperature: -30°C ~ 80°C
• Storage temperature: -45°C ~ 90°C
• Dimension: 56.21mm x 65.15mm

### What’s on Board
1. SIM7600E-H
2. CP2102 USB to UART converter
3. NAU8810 audio decoder
4. TXS0108EPWR voltage translator: translates 3.3V/5V into 1.8V
5. MP2128DT power chip
6. MP1482 power chip
7. Raspberry Pi GPIO header: for connecting with Raspberry Pi
8. SIM7600 control interface: for connecting with host boards like Arduino/STM32
9. SIM card slot: supports 1.8V/3V SIM card
10. TF card slot: allows file/SMS/... storage
11. 3.5mm earphone/mic jack
12. USB interface: for testing AT Commands, getting GPS positioning data, etc.
13. USB to UART interface: for serial debugging, or login to Raspberry Pi
14. MAIN antenna connector
15. AUX antenna connector
16. GNSS antenna connector
17. Power switch
18. Network status indicator
19. Power indicator
20. Operating voltage selection jumper:
   VCCIO - 3.3V: set operating voltage as 3.3V
   VCCIO - 5V: set operating voltage as 5V
21. UART selection jumper:

A: access Raspberry Pi via USB to UART
B: control the SIM7600 by Raspberry Pi
C: control the SIM7600 via USB to UART
Contents

Overview......................................................................................................................................................... 1
Features.............................................................................................................................................................. 1
Communications Specifications......................................................................................................................... 2
GNSS Specifications........................................................................................................................................... 2
SMS and Audio Specifications......................................................................................................................... 3
Other Specifications.......................................................................................................................................... 3
What’s on Board.................................................................................................................................................. 3
1. Hardware configuration................................................................................................................................... 8
   1.1. Hardware configuration............................................................................................................................ 8
2. At Test Instructions......................................................................................................................................... 9
   2.1. General AT commands............................................................................................................................. 9
   2.2. Make calls and answer calls................................................................................................................... 11
   2.3. Send and receive messages..................................................................................................................... 12
   2.4. GPS Debugging......................................................................................................................................... 13
   2.5. TF Card Test............................................................................................................................................. 15
   2.6. GPRS Debugging...................................................................................................................................... 16
       Local virtual severs settings..................................................................................................................... 16
       GPRS Test.................................................................................................................................................. 17
3. Using with Raspberry Pi.................................................................................................................................. 18
   3.1. Interface overview.................................................................................................................................... 18
   3.2. UART configuration of Raspberry Pi...................................................................................................... 18
   3.3. Init the Raspberry Pi............................................................................................................................... 19
3.4. Minicom for UART debugging on Raspberry Pi ........................................................................ 20

3.5. Examples .......................................................................................................................... 21
    3.5.1. PhoneCall ..................................................................................................................... 21
    3.5.2. SMS ............................................................................................................................. 22
    3.5.3. GPS ............................................................................................................................... 22
    3.5.4. TCP ............................................................................................................................... 23
    3.5.5. FTP ............................................................................................................................... 23

4. Using with Arduino ........................................................................................................... 24
    4.1. Interface overview ......................................................................................................... 24
    4.2. Install Arduino Library ................................................................................................. 24
        4.2.1. Phonecall .................................................................................................................. 25
        4.2.2. SMS ......................................................................................................................... 25
        4.2.3. GPS ......................................................................................................................... 26
        4.2.4. TCP ......................................................................................................................... 27
        4.2.5. FTP ......................................................................................................................... 27
1. Hardware configuration

1.1. Hardware configuration

This module comes with GSM antenna, LTE antenna and micro USB cable. Besides these you should prepare a 4G sim card and a microphone cable with microphone:

1) Insert the SIM card to the card slot, Insert the headphone cable and connect the LTE antenna.
2) Connect the USB interface of SIM7600E-H 4G HAT to PC with a micro USB cable. Then the PWR indicator will keep bright.
3) Press the PWRKEY button and hold for 1s, the NET indicator will blink as below. Generally, the NET indicator will fast flash firstly (1 time per second), which means that the module has not logged in the Network. After logging in, the indicator become to flash slowly (1 time every three seconds). Up to the local LTE network, this process that logging in will last several seconds to dozens of seconds.

If you take too much time to log in and failed, please check that whether the LTE antenna is connected correctly, and whether the SIM card is usable and insert correctly.

4) Install SIM7600 driver (windows driver: www.waveshare.com/wiki/File:SIM7X00-Driver.7z)
Open Device Manager to get the corresponding COM port number of SIM7600. For example, the AT Port is COM19 as below. Users need to choose the correct port according to the Manager.

Figure: Devices Manager
NOTE:
The default hardware of SIM7600CE 4G HAT needs to be turned on by button. If the hardware needs to be turned on automatically, you can use a connection line to connect the PWR and GND pins on the module pin, so that it can automatically turn on, as shown in the figure below.

2. At Test Instructions

2.1. General AT commands

<table>
<thead>
<tr>
<th>Commands</th>
<th>Description</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT</td>
<td>AT test command</td>
<td>OK</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
<td>Result</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------------</td>
<td>--------</td>
</tr>
</tbody>
</table>
| ATE      | ATE1: Enable echo  
           | ATE0: Disable echo  | OK     |
| AT+CGMI  | Module manufacturers                            | OK     |
| AT+CGMM  | Module model                                    | OK     |
| AT+CGSN  | Serial number                                    | OK     |
| AT+CSUB  | Module revision                                 | OK     |
| AT+CGMR  | Firmware revision                               | OK     |
| AT+I PREX| Set baud rate                                    | +IPREX: OK |
| AT+CRESET| Reset module                                    | OK     |
| AT+CSQ   | Check signal quality                             | +CSQ: 17,99 OK |
| AT+CPIN? | SIM Card Status                                 | +CPIN: READY |
| AT+COPS? | Operator selection                              | +COPS: OK |
| AT+CREG? | Network registration                            | +CREG: OK |
| AT+CPSI? | UE system infor                                 |        |
| AT+CNMP  | Mode selection:  
                   | 2: Automatic  
                   | 13: GSM only  
                   | 38: LTE only  
                   | 48: Any modes but LTE  
                   | ... ... | OK     |

For more details, please refer to the documentation:  Series_AT Command Manual_V1.07
2.2. Make calls and answer calls

1) Insert the SIM card, connect the LTE antenna and connect the USB interface of SIM7600E-H 4G HAT to PC. Then press the PWRKEY to power on the board;

2) Check whether the indicators blink correctly (PWR’s and NET’s flashes).

3) Send AT commands as bellow:

<table>
<thead>
<tr>
<th>Commands</th>
<th>Description</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT+CNUM</td>
<td>Phone number (Not all SIM cards Support)</td>
<td>+CNUM OK</td>
</tr>
<tr>
<td>AT+CSDVC</td>
<td>AT+CSDVC=1: Handset output</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>AT+CSDVC=3: Speaker output</td>
<td>OK</td>
</tr>
<tr>
<td>AT+CLVL=?</td>
<td>check volume level</td>
<td>OK</td>
</tr>
<tr>
<td>AT+CLVL=2</td>
<td>volume level set to 2</td>
<td>OK</td>
</tr>
<tr>
<td>ATD&lt;phone_number&gt;;</td>
<td>Make calls</td>
<td>OK</td>
</tr>
</tbody>
</table>
2.3. Send and receive messages

1. Plug the SIM card, connect the LTE antenna and connect the USB interface of SIM7600E-H 4G HAT to PC. Then press the PWRKEY to power on the board;

2. Check whether the indicators blink correctly (PWR’s and NET’s flashes).

3. Send AT commands as bellow:

<table>
<thead>
<tr>
<th>Commands</th>
<th>Description</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT+CMGF=1</td>
<td>select message format</td>
<td>OK</td>
</tr>
<tr>
<td>AT+CSCS=&quot;GSM&quot;</td>
<td>Select TE character set: GSM</td>
<td>OK</td>
</tr>
<tr>
<td>AT+CSMP</td>
<td>set text mode para</td>
<td>OK</td>
</tr>
<tr>
<td>AT+CMGS=&quot;&lt;phonenumber&gt;&quot;</td>
<td>Send message</td>
<td>OK</td>
</tr>
</tbody>
</table>
2.4. GPS Debugging

<table>
<thead>
<tr>
<th>Commands</th>
<th>Description</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT+CGPS</td>
<td>GNSS Power Control:</td>
<td>OK</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>AT+CGPS</td>
<td>Turn on/turn off GPS</td>
<td></td>
</tr>
<tr>
<td>AT+CGPS=1</td>
<td>Turn on GPS</td>
<td></td>
</tr>
<tr>
<td>AT+CGPS=0</td>
<td>Turn off GPS</td>
<td></td>
</tr>
<tr>
<td>AT+CGPSINFO</td>
<td>GNSS navigation information parsed from NMEA sentences</td>
<td></td>
</tr>
<tr>
<td>+CGNSINF</td>
<td>OK</td>
<td></td>
</tr>
</tbody>
</table>

1) Connecting the GPS antenna, and place the receiver on open area outdoor
2) AT+CGPS =1 //Turn on power of GPS
3) Open u-center and set the Port and Baudrate (NMEA Port,COM27)
   - SimTech HS-USB AT Port 9001 (COM25)
   - SimTech HS-USB Audio 9001 (COM24)
   - SimTech HS-USB Diagnostics 9001 (COM28)
   - SimTech HS-USB NMEA 9001 (COM27)
4) AT+CGNSINF //Print the GPS information
5) AT+CGPS =0 //Turn off power of GPS
### 2.5. TF Card Test

1. Plug the SIM card, connect the LTE antenna and connect the USB interface of SIM7600E-H 4G HAT to PC. Then press the PWRKEY to power on the board;
2. Check whether the indicators blink correctly (PWR’s and NET’s flashes);
3. Send AT commands as bellow:
### Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT+FSCD=D:</td>
<td>Select directory</td>
<td>OK</td>
</tr>
<tr>
<td>AT+FSLS</td>
<td>list directories</td>
<td>+FSLs:OK</td>
</tr>
<tr>
<td>AT+CFTRANRX</td>
<td>write data into files</td>
<td>&gt;OK</td>
</tr>
<tr>
<td>AT+CFTRANTX</td>
<td>open file</td>
<td>+CFTRANTX:OK</td>
</tr>
</tbody>
</table>

#### 2.6. GPRS Debugging

**LOCAL VIRTUAL SERVER SETTINGS**

Virtual servers define the mapping between service ports of WAN and web servers of LAN. All requests from Internet to service ports of WAN will be redirected to the computer (web servers of LAN) specified by the server IP. (see your router’s guide manual)
4) Log in Management Console of your router with browser (read your router’s guide manual for specific address)

5) Set Port: 1822 (The Port can’t be conflict to other’s. Here we set 1822)

Set LAN IP address of your computer (you can run CMD on your computer, and execute command `ipconfig` to enquiry the address of IPv4), 192.168.6.168 as examples

6) You can search “IP” on browser to get your WAN IP address.

---

**GPRS TEST**

1. Plug the SIM card, connect the LTE antenna and and connect the USB interface of SIM7600E-H 4G HAT to PC. Then press the PWRKEY to power on the board;

2. Check whether the indicators blink correctly (PWR’s and NET’s flashes).

3. Send AT commands as bellow:

<table>
<thead>
<tr>
<th>命令</th>
<th>说明</th>
<th>返回值</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT+CGDCONT=1,&quot;IP&quot;,&quot;CMNET&quot;</td>
<td>PDP context</td>
<td>OK</td>
</tr>
<tr>
<td>AT+CGREG?</td>
<td>GPRS network status</td>
<td>+CGREG: OK</td>
</tr>
<tr>
<td>AT+CIPMODE=1</td>
<td>TCP/IP mode</td>
<td>OK</td>
</tr>
<tr>
<td>AT+C SokSETPN=1</td>
<td>PDP profile number</td>
<td>OK</td>
</tr>
<tr>
<td>AT+NETOPEN</td>
<td>Open socket</td>
<td>+NETOPEN:</td>
</tr>
<tr>
<td>AT+CIPOPEN=0,&quot;TCP&quot;,&quot;113.81.233.65&quot;,2317</td>
<td>establish connection</td>
<td>+CIPOPEN:</td>
</tr>
<tr>
<td>AT+CIPSEND=0,9</td>
<td>Send data of a specific size</td>
<td>&gt;</td>
</tr>
<tr>
<td>AT+CIPSEND=0,1A</td>
<td>Send data of a fixed size</td>
<td>&gt;</td>
</tr>
<tr>
<td>1A</td>
<td>(HEX format) Tell module to send data</td>
<td>+CIPSEND:</td>
</tr>
<tr>
<td>AT+CIPCLOSE</td>
<td>close connection</td>
<td>+CIPCLOSE:</td>
</tr>
<tr>
<td>AT+NETCLOSE</td>
<td>Close socket</td>
<td>+NETCLOSE:</td>
</tr>
</tbody>
</table>
3. Using with Raspberry Pi

3.1. Interface overview

The default relationship between SIM7600 control pins and Raspberry Pi IOs is shown in Table 1.

Table 1: The relationship between SIM7600 control pins and Raspberry Pi IOs

<table>
<thead>
<tr>
<th>SIM7600</th>
<th>IO of Raspberry Pi B+</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5V</td>
<td>5V</td>
<td>Power supply (5V)</td>
</tr>
<tr>
<td>GND</td>
<td>GND</td>
<td>Ground</td>
</tr>
<tr>
<td>TXD</td>
<td>RXD (BCM P15)</td>
<td>UART pin</td>
</tr>
<tr>
<td>RXD</td>
<td>TXD (BCM P14)</td>
<td>UART pin</td>
</tr>
<tr>
<td>PWR</td>
<td>P22 (BCM P6)</td>
<td>Power up the module</td>
</tr>
<tr>
<td>FLIGHTMODE</td>
<td>P7 (BCM P4), Pull high enable flight mode</td>
<td>Flight mode</td>
</tr>
</tbody>
</table>

3.2. UART configuration of Raspberry Pi

Because UART of Raspberry Pi is used for Linux console output by default, if we want to use the UART, we need to change the settings. Executing this command to enter the configuration page:

```
sudo raspi-config
```
Choose Advanced Options -> Serial -> no, to disable Linux's use of console UART

Open /boot/config.txt file, find the below statement and uncomment it to enable the UART. You can directly append it at the end of file as well.

```bash
enable_uart=1
```

Then reboot.

3.3. Init the Raspberry Pi

1. Download the raspberry pi demo code to /home/pi/ directory.

```bash
wget https://www.waveshare.com/w/upload/2/29/SIM7600X-4G-HAT-Demo.7z
sudo apt-get install p7zip-full
7z x SIM7600X-4G-HAT-Demo.7z -r -o/home/pi
sudo chmod 777 -R /home/pi/SIM7600X-4G-HAT-Demo
```

2. Open the /etc/rc.local file, then add the context below:

```bash
sh /home/pi/SIM7600X-4G-HAT-Demo/Raspberry/c/sim7600_4G_hat_init
```
3.4. Minicom for UART debugging on Raspberry Pi

Inserting the module to Raspberry Pi and plug the jumper B,

Install minicom. minicom is a text-based modem control and terminal emulation program for Linux:

```
sudo apt-get install minicom
```

Execute command: `minicom -D /dev/ttyS0` (ttyS0 is the UART of Raspberry Pi 3B)

Baud rate is 115200 by default. If you need to change the baud rate, for example 9600, you can add the parameter `-b 9600`.

The user UART device of Raspberry Pi 2B/Zero is ttyAMA0, and ttyS0 of Raspberry Pi 3B

Testing Bluetooth function as examples:
3.5. Examples

1. Download the demo code from wiki and copy to the Raspberry Pi (/home/pi/SIM7600X)

2. Enter the bcm2835 directory, compile and install the BCM2835 library:

   chmod +x configure && ./configure && sudo make && sudo make install

3. Compile and run the demo (for example:PhoneCall):

   Clean up:   sudo make clean
   Recompile: sudo make
   Run the program: sudo ./PhoneCall
   Combination command: sudo make clean && sudo make && sudo ./PhoneCall

3.5.1. PHONECALL
3.5.2. SMS

3.5.3. GPS
3.5.4. TCP

3.5.5. FTP
4. Using with Arduino

4.1. Interface overview

The default relationship between SIM7600 control pins and Arduino is shown in Table 1.

Table 2: The relationship between SIM7600 control pins and Arduino

<table>
<thead>
<tr>
<th>SIM7600</th>
<th>Arduino UNO / UNO PLUS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5V</td>
<td>5V</td>
<td>Power supply (5V)</td>
</tr>
<tr>
<td>GND</td>
<td>GND</td>
<td>Ground</td>
</tr>
<tr>
<td>TXD</td>
<td>0 (RX)</td>
<td>UART pin</td>
</tr>
<tr>
<td>RXD</td>
<td>1 (TX)</td>
<td>UART pin</td>
</tr>
<tr>
<td>PWR</td>
<td>2</td>
<td>Power up the module</td>
</tr>
</tbody>
</table>

4.2. Install Arduino Library

1. Download the Arduino demo code and copy the Waveshare_SIM7600X_Arduino_Library folder to \{the Arduino software installation path\}/Library/.
2. Run the Arduino IDE, then select the example code as below:
4.2.1. PHONECALL

```
A7
Starting up ...
A7
A7+CREG?
ATD10086;
A7+CNUM
Call disconnected
```

4.2.2. SMS
4.2.3. GPS

```
AT
Starting up...
AT
AT
AT
AT
AT+CEREG?
Setting SMS mode...
AT+CMTS=1
Sending Short Message
AT+CMGS="1E-168"
www.waveshare.com
Sent successfully
Setting SMS mode...
AT+CMTS=1
AT+CPMS="SM", "SM", "SM"
AT+CNGR=1

Autoscroll  Newline  9600 baud  Clear output
```
4.2.4. TCP

<table>
<thead>
<tr>
<th>Command</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CREG?</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CSq</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CREOP</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CPSI?</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;COREB9?</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CSOCKCONNECT=1, &quot;IP&quot;, &quot;CMNET&quot;</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CSOCKOPEN=1</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CIFMODE=0</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;NETOPN</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;IPADDR</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CIFOPEN=0, &quot;TCP&quot;, &quot;118.190.83.84&quot;, 2317</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CIFSEND=0</td>
<td>-</td>
</tr>
<tr>
<td>WaveShare</td>
<td>-</td>
</tr>
<tr>
<td>AI-&lt;CIFCLOSE=0</td>
<td>-</td>
</tr>
</tbody>
</table>

4.2.5. FTP
AT
AT+CIPORD=21
AT+CIPNODE=1
AT+CIPFFTPE=A
AT+CIPSERVER=113.81.238.52
AT+CIPUSER="user"
AT+CIPFPF="waveshare"
Download file from FTP...
AT+CIPGETFILE="index.htm",0
Upload file to FTP...
AT+CIPFFHTFILE="index.htm",0