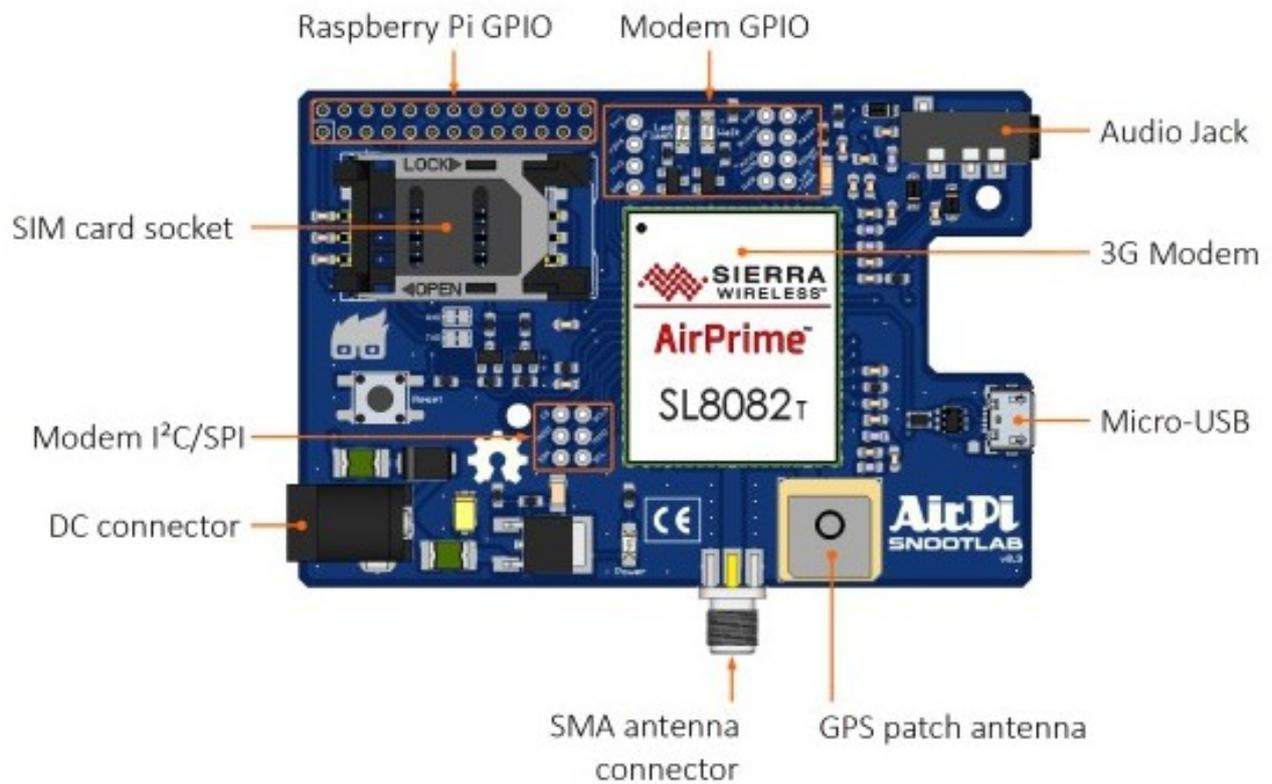




# User manual

## AirPi v0.9



# Contents

1 Foreword.....	3
2 AirPi v0.9 shield connection .....	3
3 Raspberry-Pi configuration.....	3
4 Basic use.....	4
4.1 Echo configuration.....	4
4.2 Serial port speed configuration.....	4
4.3 Visualize the AirPi's temperature or voltage .....	4
4.4 GSM stack activation.....	4
4.5 PIN unlocking.....	4
4.6 SIM unlocking verification.....	5
5 Phone functionalities use.....	5
5.1 Get signal strength.....	5
5.2 Call a number.....	5
5.3 Hang up.....	5
5.4 Show/hide incoming calls origin.....	5
5.5 Pick up the phone.....	6
5.6 Send a SMS.....	6
5.7 Receive a SMS.....	6
5.8 List all available SMS.....	6
5.9 Read an incoming SMS.....	6
5.10 Delete a sent or received SMS.....	7
6 Internet modem use.....	7
6.1 Connexion configuration.....	7
6.2 Connection start.....	7
6.3 Disconnection.....	7
7 GPS use.....	7
7.1 GPS setting.....	7
7.2 Start the GPS.....	8
7.3 Configur the GPS's NMEA flux.....	8
7.4 Get last GPS position known.....	9
7.5 Stop the GPS.....	9
7.6 Mémoriser la configuration du GPS.....	9
7.7 Going further with the GPS (or not ?).....	9
8 Annex.....	9
8.1 AirPi update.....	9
8.1.1 Software version check.....	9
8.1.2 Software update.....	10
8.2 AirPi factory setting.....	11
8.3 Resources.....	11

## 1 Foreword

This tutorial assumes the AirPi v0.9 (more information at <http://airpi.io>) is used with a Raspberry Pi under Raspbian, configured and used with a keyboard and a screen.

Commands sent to the AirPi will be shown like this (in blue).

Answers from the AirPi like this (in red).

**Warning** : The AirPi being totally customizable, this tutorial cannot cover every configuration resultant of undescribed manipulations.

## 2 AirPi v0.9 shield connection

The AirPi and the Raspberry Pi are totally connected together when the shield is plugged onto the Raspberry and the provided micro-USB cable links the two of them.

**Warning** : Powering the whole cluster Raspberry Pi/AirPi v0.9 has to be made solely with the provided power pack, connected with the AirPi. You don't need to plug in the Raspberry Pi, it is powered directly from the shield.

## 3 Raspberry Pi configuration

**Note** : This tutorial assumes only the AirPi is connected to the Raspberry Pi and the access rights to the serial ports are acquired.

You can talk with the AirPi either on the serial link (`/dev/ttyAMA0`) and/or the emulated serial port (`/dev/ttyUSB3`), default speed is **115 200** bauds, with 1 stop bit and no parity. For didactic purposes during this tutorial, open two Putty instances, one on `/dev/ttyAMA0` and another on `/dev/ttyUSB3`, this way you'll understand better how the AirPi works.

Try in both serial terminals the following command to check communication status with the board. Note that the AirPi doesn't send back the commands (echo) by default, don't be surprised.

**AT**

The AirPi should answer « OK » in the terminal where you typed the AT command.

**Warning** : For experienced users using SSH instead of keyboard/screen, remember to alter the network configuration (replace **auto eth0** with **allow-hotplug eth0** in `etc/interfaces`). Otherwise you'll lose the connection each time the AirPi reboots, because of the USB port's weakness (network interface is on USB).

## 4 Basic use

### 4.1 Echo configuration

In order to ease AirPi manipulations, we recommend activating the echo (repetition of the commands sent to the AirPi), with this command :

```
ATE1
```

```
OK
```

**Note** : To permanently save the configuration, type the AT&W command.

### 4.2 Serial port speed configuration

Serial port speed (/dev/ttyAMA0) can be configured with the following command, where x can take the following values : 0 (autobauding), 300, 600, 1 200, 2 400, 4 800, 9 600, 19 200, 38 400, 57 600, 115 200 (default), 230 400, 460 800, 921 600.

```
AT+IPR=x
```

```
OK
```

**Note** : To permanently save the configuration, type the AT&W command.

### 4.3 Visualize the AirPi's temperature or voltage

The following commands inform us about the shield's temperature and power supply voltage :

```
AT!TEMP?
```

```
Temp                               state:                               Normal
Temperature: 40 degC
```

```
AT!PCVOLT?
```

```
Volt                               state:                               Normal
Power supply voltage: 3633mV
```

### 4.4 GSM stack activation

The GSM stack is not active by default, activate it with the command :

```
AT+CFUN=1
```

```
OK
```

### 4.5 PIN unlocking

To use the GSM network we need to unlock the SIM card with the following command (where 1234 is the SIM's PIN) :

```
CPIN=1234
```

```
OK
```

## 4.6 SIM unlocking verification

You may find useful to check your SIM's status with :

`AT+CPIN?`

The AirPi will answer with « **+CPIN: state** », where *state* can be, amongst others :

- **READY** : SIM is unlocked.
- **SIM PIN** : PIN is necessary to unlock SIM.
- **SIM PUK** : PUK 1 is necessary to unlock SIM.

## 5 Phone functionalities use

### 5.1 Get signal strength

It may be useful for a start to check the signal's strength with the command :

`AT+CSQ`

The AirPi will return something like :

**+CSQ: 20,0**

Where 20 is the signal strength level and 0 the error rate.

### 5.2 Call a number

**Note** : Following manipulations assume SIM card is unlocked (cf. Basic use).

To make a vocal call (audio available on 3,5mm mini-jack connector), use the following command where « num » is the number to call.

`ATDnum;`

### 5.3 Hang up

To hang up during an incoming or outgoing call, use the following command :

`ATH`

### 5.4 Show/hide incoming calls origin

A command exists to show or hide the calling number when an incoming call arrives.

`AT+CLIP=1`

**OK**

Replace with `AT+CLIP=0` to disable this function.

**Note** : Use the `AT&W` command to memorize your setting.

## 5.5 Pick up the phone

During a call (the AirPi sends RING for each ring), to accept the communication type the following command :

```
ATA
```

## 5.6 Send a SMS

Note : Following manipulations assume SIM card is unlocked (cf. Basic use)

To send an SMS use the following command :

```
AT+CMGS=0506050606
```

```
>cats are awesome
```

Type your message at the prompt and to end it, type CTRL+Z. The SMS will be sent and the AirPi's reply will contain your message's index :

```
+CMGS: 1
```

```
OK
```

## 5.7 Receive a SMS

Once connected to the GSM network, the AirPi will notify each SMS reception by sending on the terminal :

```
+CMTI: "SM",1
```

Where "1" is the memory index of the received SMS.

## 5.8 List all available SMS

To make a list of every message in the memory, use the following command :

```
AT+CMGL="ALL"
```

Every message starts with a header such as :

```
+CMGL: 2,"REC UNREAD","+330606060606",,"16/02/10,10:15:04+04"
```

In headers you'll find the message's status (read/unread), the sender's phone number and the reception date. The header is followed by the message, OK terminates the message.

## 5.9 Read an incoming SMS

To read a SMS, use the following command where x is the memory index of the SMS :

```
AT+CMGR=x
```

## 5.10 Delete a sent or received SMS

To delete any message stored in the memory, take its index number and type the following command :

```
AT+CMGD=index
```

OK

**Hint** : to delete every SMS in the memory with a single command, type : **AT+CMGD=0,4**

## 6 Internet modem use

**Note** : Following manipulations assume SIM card is unlocked (cf. Basic use).

### 6.1 Connexion configuration

With a Free 3G connection (contact your provider for further information)

```
AT+CGDCONT=1,"IP","free"
```

OK

### 6.2 Connection start

Once the configuration effective, open the connection with the following command :

```
AT!SCACT=1,1
```

OK

If everything went ok, the network interface wwan0 is connected to the Internet.

### 6.3 Disconnection

To close the network connection previously opened, use the following command :

```
AT!SCACT=0,1
```

OK

## 7 GPS use

### 7.1 GPS setting

To activate the AirPi's GPS module, type the following commands :

```
AT !CUSTOM="gpsenable",1
```

OK

**AT&W**

**OK**

**AT!RESET**

Wait a few seconds while the AirPi reboots to take into account your modifications.

Check if the settings have been saved with the command :

**AT!CUSTOM?**

This command returns a list of variables among which you should find this specific line :  
**"GPSENABLE 0x01"**.

Now that the GPS module is active, it's possible to initialize it with the following command :

**AT+GPSINIT=2**

**OK**

**+GPSEVINIT: 1**

## 7.2 Start the GPS

Once the GPS configuration is done, you should start acquisitions with the following command :

**AT+GPSSTART**

**OK**

**+GPSEVSTART: 1**

**+GPSEVPOS: 0**

GPS service is running, you should now see on the terminal (default is **/dev/ttyAMA0**) the NMEA sentences.

## 7.3 Configur the GPS's NMEA flux

When you start the GPS with the command :

**AT+GPSSTART**

the NMEA flux is by default sent to serial port **/dev/ttyAMA0**, but here again you can set things up with the following command :

**AT+GPSNMEA=x [, y]**

Where x can be:

- 0 : No sentence.
- 1 : Sentence on UART 1 (default).
- 3 : Sentence on USB emulated serial port.
- 4 : Sentence on port from which this command was sent.

The optional value allows you to define the period between the generation of two sentences, which can be defined between 0 and 65 535 seconds.

**Note :** When NMEA sentences are generated on a port, you can still converse with the AirPi. Indeed, when you send a command, the NMEA flux is halted and lost until the end of a command or ENTER.

## 7.4 Get last GPS position known

Besides the NMEA output, the GPS also has the following command to get a human-readable geolocalization :

`AT!GPSLOC?`

## 7.5 Stop the GPS

The following command stops the AirPi's GPS service :

`AT+GPSSSTOP`

## 7.6 Save GPS settings

The following command saves the full GPS configuration and applies it automatically at restart :

`AT+GPSSAVE=1`

To cancel the automatic restoration on start, simply use the same command with a 0 instead of the 1 :

`AT+GPSSAVE=0`

## 7.7 Going further with the GPS (or not ?)

For starters, you'll need to install **gpsd** (a daemon that retrieves NMEA data) and **glandkartegt** (a mapping software) on the Raspberry Pi, so type in the terminal :

```
sudo apt-get install gpsd glandkartegt
```

Refer to the relevant packet manuals for further information.

# 8 Annex

## 8.1 AirPi update

### 8.1.1 Software version check

**Warning :** Updating the AirPi can make it unusable.

Before any update, make sure with the following command that your software isn't already up to date with the latest firmware recommended by Snootlab :

## ATI9

```
"DWL", " S4_1_0_21BT R2371 CNSHZ-ED-XP0031 2014/11/27 16:53:47", "", "Sierra Wirel"

"FW", "FW_752_A5_68_F2_5.SL808Fx", "R7.52.2.A5.201411281219.SL8082T", "Sierra Wire"

"MODEM 3G+", "Revision: S4_1_0_21AP R2371 CNSHZ-ED-XP0031 2014/11/27 16:53:47"

"OAT", "v1.3.1.20131223105256", "Extended AT Application", "Sierra Wireless", 71330"

--"Developer Studio", "2.3.2.201310241753"
--"Open AT OS Package", "6.52.0.201311140649"
--"Firmware Package", "7.52.0.201312200625"
--"Internet Library Package", "5.56.0.201305170830"
--"Location Library Package", "2.0.0.201310171016"
--"Security Library Package", "2.2.0.201306261000"
--"eCall Library Package", "1.2.3.201306191228"
--"Jamming Library Package", "1.1.0.201312230837"
--"ExtendedATApplication Library Package", "1.3.1.201312231052"

"ROM", "8400000"

"RAM", "4000000"

"OATRAM", "b00000"

"DWLNAME", "SL808x"

OK
```

The versions of the different software units are located in the first 4 lines.

### 8.1.2 Software update

**Note** : An update is possible with Linux, but with Sierra Wireless's **Developer Studio**.

The AirPi's software is updated and tested in our workshop before being sent, but we invite AirPi owners to update their product.

You'll need first to install the AirPi drivers available on Sierra Wireless's website (Airprime USB driver SL808xT series).

You should, before starting the actual update procedure, stop and uninstall every application with the following commands (wait carefully until they're completed) :

`AT+WOPEN=0`

OK

`AT+WOPEN=4`

OK

Then run the update program available at Sierra Wireless (R7.52.2.A5.201411281219.SL808x exe).

**Warning** : Interrupting the update procedure can make the product useless.

## 8.2 AirPi factory setting

To reinitialize the configuration of the AirPi to its default state, use the following command :

`AT&F0`

OK

## 8.3 Resources

Available at Sierra Wireless :

- AT Commands Interface Guide for Firmware 7.52
- To use the GPS : Location Library for Open AT