

# SIM900 ATC Introduction

Albert@2010.1

## ■ Hardware Platform

PNX4851 single solution( BB+Transceiver) + RF7161 + ST/SA(64/32)

## ■ Software Platform

- ◆ Based on ARM926EJ-S, CEVA Teak DSP
- ◆ Run on 156 MHz
- ◆ OS RTK-E: philips real time kernel
- ◆ Build environment
  - RVCT2.2 [ Build 616]
  - Cygwin\_NT-5.1 1.5.18

- 1) Platform ;
- 2) Power off and RF reset ;
- 3) Auto-bauding ;
- 4) Embedded TCPIP protocol ;
- 5) STK ;
- 6) Error code ;
- 7) Hardware flow control is closed in default, should be opened by `AT+IFC=2,2` ;

## ■ Power off & RF reset

- AT+CPOWD add power off in emergency conditions , in this case, "NORMAL POWER DOWN" will not be present.

AT+CPOWD=1 // power off in normal conditions

AT+CPOWD=0 // power off in emergency conditions

- AT+CFUN =<fun>,<rst> add reset parameter.

AT+CFUN=0,1

AT+CFUN=1,1

Note : <fun>supports mode 0,1 and 4. While RF could not be switched between mode 0 and mode 4.

## ■ Auto-Bauding

SIM900 supports baud rate from 1200 bps to 115200 bps, also support auto-bauding mode.

When power on under auto-bauding mode, the first letter of command should be uppercase, such as "AT" or "At", otherwise, module will not give any response. Usually, command with all uppercase letters are strongly recommendatory.

Note, hexadecimal code "49 49 49 49 FF FF FF FF" will be present at the beginning of module power on.

Sample :

III?

AT // after power on, uppercase command should be sent

OK

AT+IPR?

+IPR: 0

OK

ATI

SIM900 R11.0

OK

AT+GSV // this command can show firmware version details

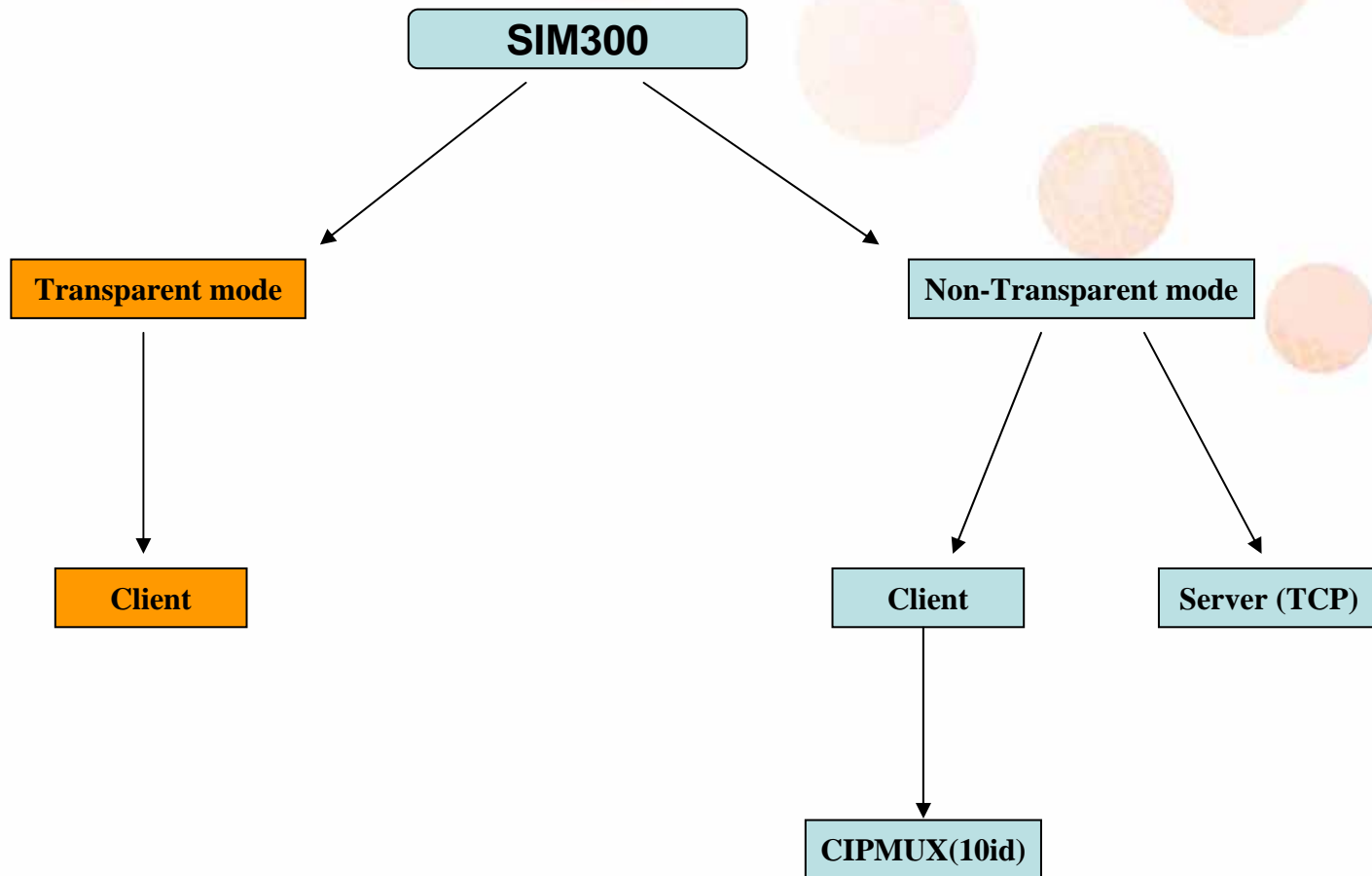
SIMCOM\_Ltd

SIMCOM\_SIM900

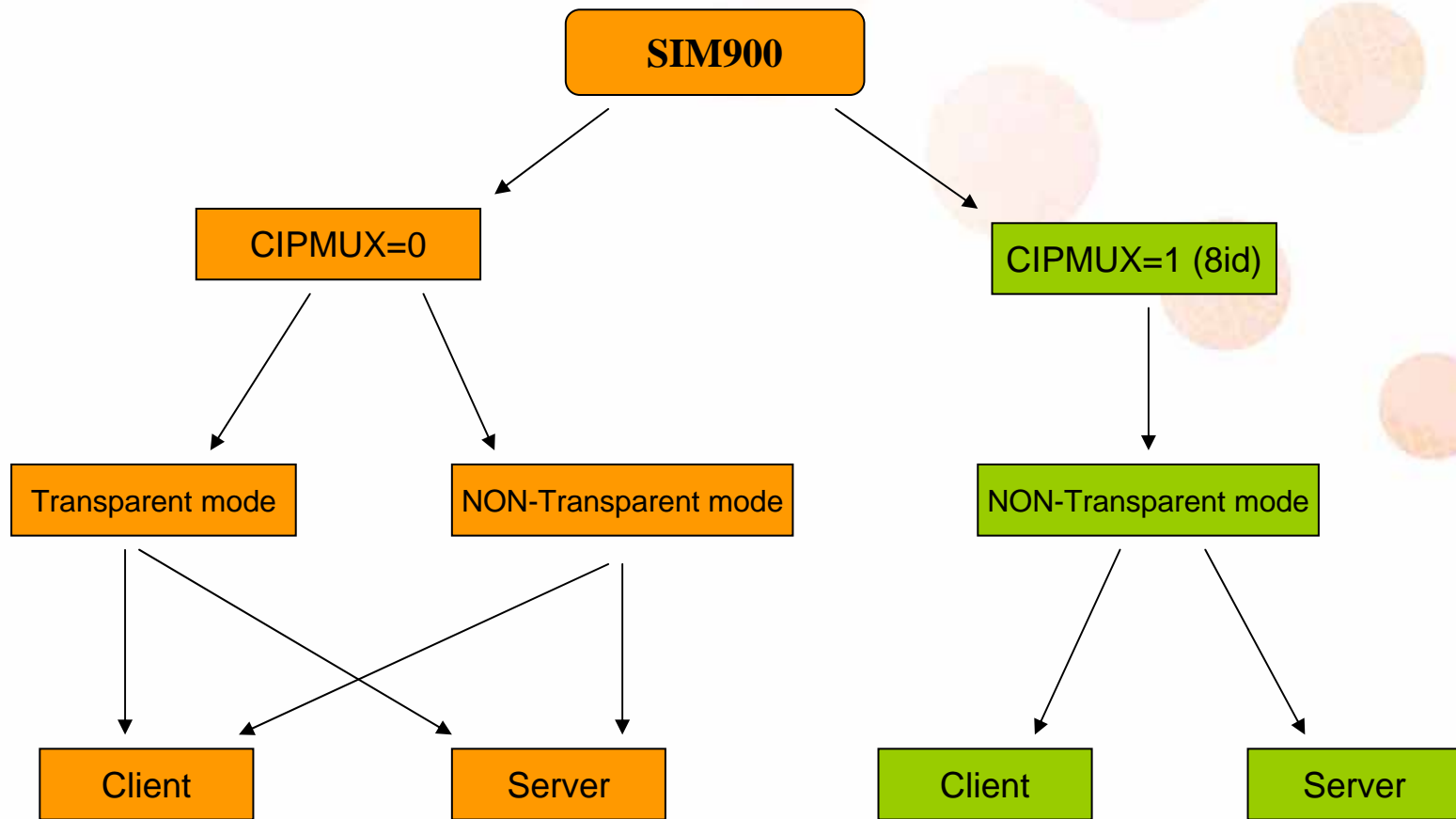
Revision:1137B01SIM900M32\_ST

OK

# SIM900 TCPIP Protocol Introduction







## SIM900 TCPIP protocol special features

- 1) CIPMUX mode;
- 2) DNS parser;
- 3) AT+CIPCLOSE=1;
- 4) AT+CIPSEND?
- 5) AT+CIPQSEND;AT+CIPACK
- 6) TCP SERVER Configuration (nothing to do with AT+CLPORT)

AT+CIPMUX=0

OK

AT+CIPMODE=0 // non-transparent mode

OK

AT+CIPSTART="TCP","116.228.221.51","8011"

OK

CONNECT OK

AT+CIPSEND=5 // send data in fixed length

> HELLO

SEND OK

AT+CIPACK

+CIPACK: 5, 5, 0

OK

AT+CIPSEND

> 0123456789[ctrl+z] // send data by control code

SEND OK

AT+CIPACK

+CIPACK: 15, 15, 0

OK

AT+CIPMUX=0

OK

AT+CIPMODE=1 // transparent mode

OK

AT+CIPSERVER=1,2200 // open

OK

SERVER OK

REMOTE IP: 211.136.130.1

CONNECT

// +++

OK

AT+CIPSERVER=0 // Only close listening status

OK

SERVER CLOSE

HELLO SIM900 // data from remote client

AT+CIPCLOSE=1 // close socket forcibly without ACK

CLOSE OK

## ■ Step 1

AT+CIPMUX=1     // multiplex IP connection

OK

AT+CIPMODE=0     // this command is invalid under CIPMUX=1 mode

+CME ERROR: operation not allowed

AT+CIPMODE=1

+CME ERROR: operation not allowed

AT+CSTT

OK

AT+CIICR

OK

AT+CIFSR

10.79.161.250     // module has attached GPRS network

## ■ Step 2

AT+CIPSERVER=1,2200

OK

SERVER OK

0, REMOTE IP: 211.136.130.129 // Remote client access this server successfully

AT+CIPSTART=1,"TCP","116.228.221.51","8011"

OK

1, CONNECT OK

AT+CIPSTART=2,"UDP","116.228.221.51","9015"

OK

2, CONNECT OK

AT+CIPSEND=1,5 // send data in fixed length

> HELLO

1, SEND OK

AT+CIPSEND=2 // here, "2" means index 2 connection, not data length

> SIMCOM[ctrl+z] // send data via control code

2, SEND OK

### ■ Step 3

AT+CIPACK=2

+CIPACK: 6, 0, 6 // UDP

OK

AT+CIPACK=1

+CIPACK: 5, 5, 0 // TCP

OK

AT+CIPSEND=2,10

> 0123456789

2, SEND OK

AT+CIPACK=2

+CIPACK: 16, 0, 16

OK

AT+CIPSEND?

+CIPSEND: 0,1460

+CIPSEND: 1,1380

+CIPSEND: 2,1460

+CIPSEND: 3,0

+CIPSEND: 4,0

+CIPSEND: 5,0

+CIPSEND: 6,0

+CIPSEND: 7,0

OK

## ■ Step 4

AT+CIPSTATUS

OK

STATE: IP PROCESSING

S: 0,0,"2200""LISTENING"

C: 0,0,"TCP","211.136.130.129","2020","CONNECTED"

C: 1,0,"TCP","116.228.221.51","8011","CONNECTED"

C: 2,0,"UDP","116.228.221.51","9015","CONNECTED"

C: 3,,"","","","INITIAL"

C: 4,,"","","","INITIAL"

C: 5,,"","","","INITIAL"

C: 6,,"","","","INITIAL"

C: 7,,"","","","INITIAL"

AT+CIPCLOSE=0

0, CLOSE OK

Note : under CIPMUX=1mode , after setup server function, 7 connections is maximum and allowed to setup.

AT+CIPMUX=0

OK

AT+CIPMODE=0

OK

AT+CIPSTART="TCP","116.228.221.51","8011"

OK

CONNECT OK

AT+CIPQSEND=1

OK

AT+CIPSEND=5

> HELLO

DATA ACCEPT:5

AT+CIPSEND=10

> 0123456789

DATA ACCEPT:10 // here, data just sent to module buffer successfully

AT+CIPACK

+CIPACK: 15, 15, 0 // you should check the status of data sent from outside

OK



AT+CIPMUX=0

OK

AT+CIPMODE=0

OK

AT+CSTT

OK

AT+CIICR

OK

AT+CIFSR

10.77.87.14

AT+CDNSGIP="WWW.SIM.COM"

OK

+CDNSGIP: 1,"WWW.SIM.COM","58.32.231.146"

AT+CIPSTART="TCP","WWW.YAHOO.COM",80

OK

CONNECT OK

AT+CIPSEND=5

> HELLO

DATA ACCEPT:5

AT+CIPACK

+CIPACK: 5, 5, 0

OK

AT+CIPQSEND=0

OK

AT+CIPSEND=10

> 0123456789

SEND OK

# SIM Application Toolkit (STK)

There just have two AT commands to operate STK functions with SIM900 platform.

- **AT\*PSSTKI=<mode>**

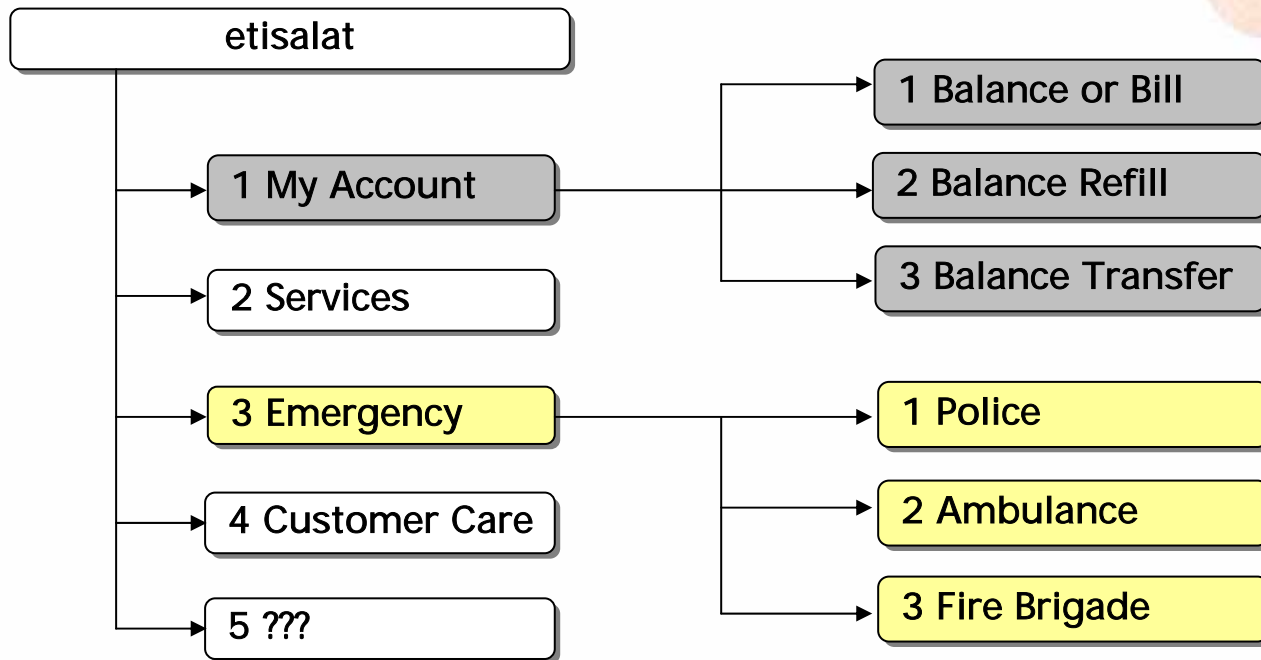
enable or disable STK functions.

- **AT\*PSSTK=<response type>,[<parameter1>,...<parametern>]**

realize all STK functions with this command.

Here, I test a SIM card from Egypt. I just make a simple introduction on how to read and select menu items.

Note, different SIM cards maybe have different STK menu items. Let's view what menu this SIM card has in my cell phone platform.



Note, the fifth menu item is present with Arabic alphabet, my phone does not support this code.

Ok, let's view these menu on SIM900 platform.

power on SIM900, then enable STK function

```
*PSSTK: "SETUP MENU",1,4,"etisalat",0,0,1,0,0,5
```

```
// here have 5 menus
```

Setup the menu with **AT\*PSSTK="SETUP MENU",1,0** command

```
OK
```

```
*PSSTK: "END SESSION"
```

Flesh all menu items with **AT\*PSSTK="GET ITEM LIST",5**command

```
*PSSTK: "GET ITEM LIST",1,128,4,"My account",0,0,0
```

```
*PSSTK: "GET ITEM LIST",2,129,4,"Services",0,0,0
```

```
*PSSTK: "GET ITEM LIST",3,130,4,"Emergency",0,0,0
```

```
*PSSTK: "GET ITEM LIST",4,131,4,"Customer Care",0,0,0
```

```
*PSSTK: "GET ITEM LIST",5,132,2,80040639063106280649,0,0,0
```

```
OK
```

Note, Unicode "0639063106280649" is Arabic alphabet here.

Select the first menu with **AT\*PSSTK="MENU SELECTION",128**command

OK

\*PSSTK: "SELECT ITEM",1,4,"My Account",0,0,1,0,0,**3**

// here, My Account menu have three items.

Download all three items with **AT\*PSSTK="GET ITEM LIST",3**command

\*PSSTK: "GET ITEM LIST",1,12,4,"Balance or Bill",0,0,0

\*PSSTK: "GET ITEM LIST",2,30,4,"Balance Refill",0,0,0

\*PSSTK: "GET ITEM LIST",3,47,4,"Balance Transfer",0,0,0

OK

Select first item with **AT\*PSSTK="SELECT ITEM",1,12,0,0** command

OK

\*PSSTK: "NOTIFICATION",1,18,1,4,"Requesting info",0,0

Response this request with **AT\*PSSTK="NOTIFICATION",1,0** command

OK

\*PSSTK: "DISPLAY TEXT",1,1,1,0,4,"temporary error",0,0



Back to last menu list with **AT\*PSSTK="COMMAND REJECTED",1,17**command

OK

\*PSSTK: "END SESSION"

Select the third menu with **AT\*PSSTK="MENU SELECTION",130**command

OK

\*PSSTK: "SELECT ITEM",1,4,"Emergency",0,0,1,0,0,**3**

// Here, Emergency menu have 3 items

Download all items with **AT\*PSSTK="GET ITEM LIST",3**command

```
*PSSTK: "GET ITEM LIST",1,11,4,"Police",0,0,0
*PSSTK: "GET ITEM LIST",2,23,4,"Ambulance",0,0,0
*PSSTK: "GET ITEM LIST",3,38,4,"Fire Brigade",0,0,0
OK
```

OK, now we can terminate STK function with below command.

```
AT*PSSTK="COMMAND REJECTED",1,16
OK
*PSSTK: "END SESSION"
```

Thank You !