TR-900 Series

Multisocket Internet Connection ATC Guide

Date : 24 April 2013

Document Version : 1.2

Our Reference : 02000B34

CONFIDENTIAL

Document History

Revision	Date	Document History
1.0	30 December 2011	▪ Preliminary Release
1.1	1 April 2013	Added: ▪ Open IP Listening Socket $IPLOPEN ▪ Connection at IP Listening Socket $IPL ▪ Close IP Listening Socket $IPLCLOSE ▪ Example of UDP Client
1.2	24 April 2013	Added Error Code

GENERAL NOTE

This document aims to support the application and engineering efforts of iWOW's customers. This document is intended for testing, evaluation, integration, and information purposes only.

iWOW makes every effort to ensure that the quality of the information. However, the content of this documentation is provided on an "as is" basis and may contain deficiencies or inadequacies.

iWOW disclaims any warranty and all responsibility for the application of the device(s) that is made in relation to the accuracy, reliability or contents of this document. iWOW is not liable for any injury, loss or damage of any kind that may incur from the use or reliance of this document.

iWOW reserves the right to make any modifications, additions and deletions to this document due to typographical errors, inaccurate information, or improvement to our products at any time without notice.

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Document Scope

This document describes the AT commands dedicated to Multisocket Internet Connection and iWOW TR-900 Series, namely TR-900 and TR-900C, to manage GPRS-related events and services. The information present in this document is relevant to the TR-900 Series.

1.2 Abbreviations

The following abbreviations are used in this document:

Abbreviation	Description
APN	Access Point Name
ASCII	American Standard Code for Information Interchange. A form of character coding scheme
DLE	Data Link Escape
ETX	End of Text
GPRS	General Packet Radio Service
GSM	Global Standard for Moblie Communications
IP	Internet Protocol
ISP	Internet Service Provider
PPP	Point-to-Point Protocol
TCP	Transmission Control Protocol

1.3 AT Command Presentation Rules

The AT commands are presented in this document as follows:

- **Description** section provides general information on the AT command objective and behavior.
- **Notes** section, if any, will provide remarks on how to use command.
- **Getting/Setting** section provides general syntax for setting and getting the values
- **Command Syntax** section describes command and response syntaxes and examples presenting the real use of that command.
- **Defined Values** section describes all parameters and values supported including default values if available.

1.4 AT Command Syntax

This section describes the AT command format, default value and the possible responses.

Commands always start with the prefix AT$ and end with the <CR> character. Parameters shown in the bracket [ ] are optional, which means when the command is executed without these parameters, their default values are used.

For most command syntaxes except for ATV0 and ATQ1, responses start and end with <CR><LF>

- If command line has been executed successfully, an “OK” is returned.
- If command syntax is incorrect, the “ERROR” string is returned.
- If command syntax is correct but transmitted with wrong parameters:
  - “+CME ERROR: <err>” or “+CMS ERROR” <SmsErr>” string is returned if CMEE is set to 1, or
  - “ERROR” is returned if CMEE is set to 0 (default value)

2 APPLICATION FLOW

This chapter provides a brief flow in setting up and starting an Internet connection. For more detailed examples, please refer to Section 8: Usage Examples.

2.1 Setup Bearer

The application flow for bearer setup is as follows:

a. To establish an Internet connection, TR-900 will dial to the Internet service provider via a bearer.
b. There are two available bearers: GSM and GPRS bearers.
c. Only one bearer is to be connected at one time.
d. For GSM bearer, refer to Section 3: GSM Dialing Services for $ISPDIAL AT Command to setup:
   - ISP dial-up number
   - ISP dial-up Username
   - ISP dial-up Password
e. For GPRS bearer, refer to Section 4: GPRS Dialing Services for $IPAPN AT Command to setup:
   - GPRS APN server name
   - GPRS APN username
   - GPRS APN password
f. Use $IPBEARER AT command to select your preferred bearer.
g. Use $IPStart to connect to the preferred bearer respectively.
h. When bearer is connected, the Serial Port will remain in AT Command Mode.

2.2 Setup Connection Services

The application flow for connection services setup is as follow:

a. After connected to the bearer, setup the preferred Internet Protocol and Socket.
b. The available protocols are:
   - TCP Client
   - TCP Server
   - UDP
c. To setup the protocols, please refer to Section 7: IP Socket Services on $IPCFG AT Command.
d. Configure data exchange settings with $IPDATAMODE and $IPDLE.
e. Connection. (**Note**: use $IPLOPEN for connecting as a listening socket server)

2.3 Transfer Data

a. There are two modes in transferring data (configured by $IPDATAMODE AT Command):
   ii. Data Mode
      - This is a transparent mode.
      - Note: the Data Link Escape mode is configured by $IPDLE.
   iii. Command Mode
      - Use $IPSEND to send data to the remote server/client.
      - $IPRECV unsolicited responses will indicate that data is being received in a specific socket.
      - Use $IPDATA to read the received data.
b. For toggling between online and offline mode, use +++ and ATO.

2.4 Close Internet Connection

a. Close all sockets by $IPCLOSE.
b. Close all listening sockets by $IPLCLOSE.
c. Close the bearer by $IPSTOP.

3 GSM DIALING SERVICES

3.1 ISP Dialing $ISPDIAL

Description:

This command sets the Dialup number, username and password required for connecting with the local ISP.

Setting/Getting:

Set value: AT$ISPDIAL = ”<number>”,”<username>”,”<password>”

Get value: AT$ISPDIAL?

Syntax:

AT$ISPDIAL

Command	Possible Response(s)
AT$ISPDIAL=?	OK
AT$ISPDIAL?	$ISPDIAL: ””,””,”” OK
AT$ISPDIAL=”96162531”,” M1”,”1234”	OK
AT$ISPDIAL?	$ISPDIAL: ”96162531”,”M1”,”1234” OK

Defined Values

<number> Decimal phone numbers. (Default: empty).

<username> Alphanumeric ASCII text string up to 64 characters. (Default value: empty).

<password> Alphanumeric ASCII text string up to 64 characters. (Default value: empty).

4 GPRS DIALING SERVICE

4.1 IP Access Point Name $IPAPN

Description:

This command sets the Access Point Name, APN username and APN password required for the GPRS connection.

Setting/Getting:

Set value: AT$IPAPN = <profile>,"<apnsrv>","<apnusr>","<apnpass>" or

AT$IPAPN = "<apnsrv>","<apnusr>","<apnpass>"

Get value: AT$IPAPN?

Note: <profile> can be absent. If <profile> is absent, profile=1 will be used.

Command	Possible Response(s)
AT$IPAPN=?	$IPAPN: (1,2),"APN Server", "APN Username", "APN Password" OK
AT$IPAPN?	$IPAPN: 1,"","",""" $IPAPN: 2,"","",""" OK
AT$IPAPN=1,"sunsurf","user","password"	OK *Note: APN server "sunsurf" with username "user" and password "password" has been set at profile 1*
AT$IPAPN?	$IPAPN: 1,"sunsurf","user","password" $IPAPN: 2,"","",""" OK

Defined Values

<profile> Numeric (1, 2).

<apnsrv> Alphanumeric ASCII text string up to 100 characters. (Default value: empty).

<apnusr> Alphanumeric ASCII text string up to 25 characters. (Default value: empty).

<apnpass> Alphanumeric ASCII text string up to 25 characters. (Default value: empty).

5 CONNECTION SERVICES

5.1 Connection Mode $IPBEARER

Description:

This command defines the bearer for Internet Connection (GSM or GPRS).

Setting/Getting:

Set value: AT$IPBEARER= <value>

Get value: AT$IPBEARER?

Command	Possible Response(s)
AT$IPBEARER=1	OK
AT$IPBEARER?	$IPBEARER: 1 OK

Defined Values

<value>

0 = GSM

1 = GPRS (default)

Note:

For GSM mode, $ISPDIAL parameters will be used in $IPSTART.

For GPRS mode, $IPAPN parameters will be used in $IPSTART.

5.2 Starting a bearer $IPSTART

Description: This command establishes an Internet connection using context defined by $IPBEARER.

Setting: Set value: AT$IPSTART =[<profile>]

Command	Possible Response(s)
AT$IPSTART=1 *Note: Connect*	Ok_Info_GprsActivation OK
AT$IPSTART *Note: Connect*	Ok_Info_GprsActivation OK

Defined Values

<profile> Numeric (1,2). This value corresponds to the <profile> parameter in $IPAPN when $IPBEARER=1. When $IPBEARER=0, <profile> is ignored.

Note:

When $ISPBEARER=0 (GSM mode), $ISPDIAL parameters will be used for the connection.

When $ISPBEARER=1 (GPRS mode), $IPAPN parameters will be used for the connection.

Use AT$IPSTOP command to close the connection.

5.3 Ending a bearer connection $IPSTOP

Description: This command disconnects a GPRS or GSM connection that is established by $IPSTART.

Command	Possible Response(s)
AT$IPSTOP *Note: Disconnect*	Ok_Info_GprsDeactivation OK *Note: Connection is disconnected.*

6 DATA EXCHANGE CONFIGURATION

6.1 Configuring data transfer mode $IPDATAMODE

Description:

This command defines the data transfer mode (Data-mode and Command-mode) for which the Serial Port should be after a successful establishment of a socket by $IPOPEN.

Setting/Getting:

Set value: AT$IPDATAMODE = <mode>

Get value: AT$IPDATAMODE?

Command	Possible Response(s)
AT$IPDATAMODE=?	$IPDATAMODE: (0,1) OK
AT$IPDATAMODE?	$IPDATAMODE: 1 OK
AT$IPDATAMODE=0 *Note: Enable command-mode data sending/receiving*	OK
AT$IPDATAMODE?	$IPDATAMODE: 0 OK

Defined Values

<mode>

0 = AT Command-mode.
Serial Port stays in Command-mode after a TCP/UDP socket is established by $IPOPEN.

1 = Data-mode. (default)
Serial Port switches to Data-mode after a TCP/UDP socket is established by $IPOPEN.

Note: This command will not take effect when TCP/UDP socket is already established.

6.2 Data Link Escape mode $IPDLE

Description:

This command enables or disables the coding of ETX (End of Text, 03 HEX). ETX character refers to a request or an indication of end of connection.

Setting/Getting:

Set value: AT$IPDLE = <mode>

Get value: AT$IPDLE?

Command	Possible Response(s)
AT$IPDLE=?	$IPDLE: (0,1) OK
AT$IPDLE?	$IPDLE: 1 OK
AT$IPDLE=0	OK
AT$IPDLE?	$IPDLE: 0 OK

Defined Values

<mode>

0 = Disabled.
In this mode, no specific process is needed on ETX (End of Text, 03 HEX) and DLE (Data Link Escape, 10 Hex) characters between host and TR-900 on the Serial Port.

1 = Enabled (Default)
ETX and DLE characters in the payload data between host and TR-900 on the Serial Port will be preceded by a DLE (Data Link Escape, 10 Hex). An ETX character indicates a request or indication of end of connection.

7 IP SOCKET SERVICES

There are a total of **ten** available sockets. **Five** connections IDs are used (shared) for $IPOPEN and $IPLOPEN with user-defined protocols (TCP-server/TCP-client/UDP), IP addresses and ports. $IPLOPEN will not occupy any of the available sockets.

For example, the socket connections can be set up in the form of two listening servers, seven sockets connected by remote clients, and, three sockets connecting to a remote server.

The supported protocols are:

- TCP Client
- TCP Server
- UDP Client

When $IPLOPEN opens a listening port, an incoming client will be connected to an available local socket. In another words, when receiving a connection request from the remote client, the listening socket does the following:

- Propagate a local client socket that connects to remote client
- Data can be transferred between the remote socket and the propagated socket.
- The listening port remains in listening mode and is ready to listen to requests from other incoming clients.

2000 bytes of incoming data buffer will be allocated to each socket for storing incoming data. The incoming data of a particular socket will be directly forwarded to the Serial Port when the socket is connected in Data-Mode, else, the incoming data will be stored in the buffer. Please refer to $IPRECV, $IPDATA, and $IPO for retrieval of buffered data.

7.1 IP Configuration $IPCFG

Description: This command configures the respective socket connections accordingly.

Setting:

If <protocol> is 1 (TCP Server/Client) or 3 (TCP Server/Client),

Set value: AT$IPCFG=<id>,<protocol>,<remote_IP>,<remote_port>

Command	Possible Response(s)
AT$IPCFG =1,1,"67.123.78.23", 1000 *Note: Set TCP Client at ID 1 to connect to remote server with IP "67.123.78.23" and port 1000*	OK
AT$IPCFG =2,2,5000 *Note: Set TCP Server Index 2 to listen at port 5000.*	OK
AT$IPCFG? *Note: List all settings*	$IPCFG: 1,1,"67.123.78.23", 1000 $IPCFG: 2,2,5000 … $IPCFG: 5,1,"",0

Defined Values

<id> Connection-ID. Numeric 1-5. Corresponding to $IPOPEN and $IPLOPEN.

<protocol> Protocol type. Numeric 1-3.

1 = TCP Client (default)

<remote_IP> Remote Server IP. Supported formats:

- 32-bit number in dotted-decimal notation (i.e. xxx.xxx.xxx.xxx)
- Alphanumeric ASCII text string up to 120 characters. (Default: "0.0.0.0")

<remote_port> Remote Server Port Number. Range from 0 to 65535. (Default value: 0)

<local_port> Local Port Number. Range from 0 to 65535. (Default value: 0)

7.2 Check Local IP Address +CGPADDR

Description: This command queries local IP address (For $BEARER=1 only).

Getting: Get value: AT+CGPADDR

Response: +CGPADDR: 1,<local_IP>

Command	Possible Response(s)
AT+CGPADDR *Note: List local IP address*	+CGPADDR: 1,"172.22.156.68" OK *Note: IP address is displayed during a GPRS connection when an IP is assigned, else it will be omitted.*

Defined Values

<local_IP> Local IP address.

7.3 Check Connection-IDs Status $IPSTAT

Description: This command queries current status and IP addresses of all Connection-IDs.

Getting: Get value: AT$IPSTAT

Response: $IPSTAT: <id>,<protocol>,<status>,<local_port>[,<remote_IP>][,<remote_port>],<socket>

Command	Possible Response(s)
AT$IPSTAT *Note: List all IP Status*	$IPSTAT: 1,1,1,0,"55.186.34.8",5555,1 $IPSTAT: 2,3,0,0,"55.186.34.8",5555 … $IPSTAT: 5,2,4,9000,6,3

Defined Values

<id> Connection-ID. Numeric 1-5.

<protocol> Type of IP protocol to use. Numeric 1-3.

1 = TCP Client

2 = TCP Server

3 = UDP Client

<status> This indicates the current status of the connection

0 = Connection Closed

1 = Active connection in data mode

2 = Active connection in command mode

3 = Active connection in data mode with incoming buffered data

4 = Active connection in command mode with incoming buffered data

5 = Listening connection

<local_port> Local Port.

<remote_IP> Remote Server IP Address.

<remote_port> Remote Server port.

<socket> Assigned Socket.

For <protocol>=1: socket number if connected corresponding to <id> will be displayed.

For <protocol>=2, Client Sockets created on Listening port. Range from 1 to 10.

All sockets are displayed according in first connect sequence separated by commas.

Note:

For client connections, local port is not provisioned and is assigned as 0.

7.4 Check IP Socket Status $IPSOCK

Description: This command checks for current status of the sockets.

Setting/Getting: Get value: AT$IPSOCK

Response: $IPSOCK: <socket>,<status>, <local_port>[,<remote_IP>][,<remote_port>]

Command	Possible Response(s)
AT$IPSOCK	$IPSOCK: 1,1,0,"55.186.34.8",5555 $IPSOCK: 2,2,0,"55.186.34.8",5555 $IPSOCK: 3,0 … $IPSOCK: 10,3,1000

Defined Values

<socket> Socket ID. Numeric 1- 10.

<status> This indicates the current status of the socket.

0 = Socket closed. No socket connection.

1 = Active socket in data mode

2 = Active socket in command mode

3 = Active socket in data mode with incoming buffered data

4 = Active socket in command mode with incoming buffered data

<local_port> Local Port.

<remote_IP> Remote Server IP Address.

<remote_port> Remote Server port

7.5 Set Timeout Timer $IPTXTIMEOUT

Description: This command set and query timeout timer.

Setting/Getting: Set value: AT$IPTXTIMEOUT=<time>

Response: $IPTXTIMEOUT

Command	Possible Response(s)
AT$IPTXTIMEOUT=<time>	
AT$IPTXTIMEOUT?	$IPTXTIMEOUT: 1000
AT$IPTXTIMEOUT=?	$IPTXTIMEOUT: (0-30000)
AT$IPTXTIMEOUT=1000	OK

Defined Values

<time> Time (in milliseconds). Range from 0-30000. Default value is 1000.

7.6 Set Maximum Framesize AT$IPFSIZE

Description: This command sets and query maximum framesize.

Setting/Getting: Set value: AT$IPFSIZE=<value>

Response: $IPFSIZE

Command	Possible Response(s)
AT$IPFSIZE=<value>	
AT$IPFSIZE?	$IPFSIZE: 1000
AT$IPFSIZE=?	$IPFSIZE: (50-1000)
AT$IPFSIZE=1000	OK

Defined Values

<value> Range from 50-1000. Default value is 1000.

7.7 Open IP Connection $IPOPEN

Description: This command opens a socket connection to the user-defined server.

Setting/Getting: Set value: AT$IPOPEN=<id>[,<timeout>]

Command	Possible Response(s)
AT$IPOPEN=2 *Note: Request opening of Connection-ID #2 for data-mode*	$IPCONN: 2,1 Ok_InfoWaitingForData *Note: Connection-ID #2 has been assigned to Socket #1 and ready to transfer data in data-mode*
AT$IPOPEN=1,10 *Note: Request opening of Connection-ID #1 for data-mode with connection timeout of 10s*	$IPCONN: 1,1 Ok_InfoWaitingForData *Note: The socket has opened in data-mode*
AT$IPOPEN=3 *Note: Request opening of Connection-ID #3 for command-mode*	$IPCONN: 3,2 OK *Note: Connection-ID #3 has been assigned to Socket #2 and ready to transfer data in command-mode*

Defined Values

<id> Connection-ID. Numeric 1-5.

<id> must be corresponded to <id> in $IPCFG, otherwise ERROR will be returned.

<timeout> Connection Time-out value (in seconds). Range from 5-120. Optional. If <timeout> is not specified, the connection time-out will be dependent on the network.

Note: Bearer must be started before $IPOPEN can be used.

7.8 Response for IP connected $IPCONN

Description: This unsolicited response will be responded when connection is successfully established with an assigned socket.

Response: $IPCONN=<id>,<socket>

Command	Possible Response(s)
AT$IPOPEN=2 *Note: Request opening of Connection-ID #2 for data-mode*	$IPCONN: 2,1 Ok_InfoWaitingForData *Note: Connection-ID #2 has been assigned to Socket #1 and "Ok_InfoWaitingForData " indicates that the module is ready to transfer data in data-mode*
AT$IPOPEN=3 *Note: Request opening of Connection-ID #3 for command-mode*	$IPCONN: 3,2 OK *Note: Connection-ID #3 has been assigned to Socket #2 and "OK" indicates that the module is ready to transfer data in command-mode*

Defined Values

<id> Connection-ID. Numeric 1-5.

<socket> Assigned IP Socket. Range from 1 to 10.

7.9 Toggling Between Online and Offline Mode in Data-Mode: +++/ATO/AT$IPO

Description:

These commands allow the user to switch the Serial port of the module between Data-Mode (online) and Command-Mode (offline) mode of different data socket connections.

Switching from Data-Mode to Command-Modem:

In Data-mode, send '<1 sec>+++<1 sec>' sequence ("+++" characters between a leading and trailing 1 second idle period) to the module. An "OK" response indicates that the module is switched to Command-Mode successfully.

Switching from Command-Mode to Data-Modem:

To resume online mode for current socket connection, use 'ATO'.

To switch to other sockets connection, use 'AT$IPO'.

A 'CONNECT' response indicates that the module is switched to Data-Mode successfully.

Setting:

Data to Command mode : <1 second>+++ <1 second>

Command to Data mode : ATO or AT$IPO=<socket>

Command	Possible Response(s)
+++ *Note: During a data connection*	OK *Note: switched to Command-Mode. AT-command can be entered.*
AT$IPO=2 *Note: Resume Data connection for socket #2*	CONNECT *Note: switched to Data-Mode*
ATO *Note: Resume Data connection on the current socket*	CONNECT *Note: switched to Data-Mode.*

Defined Values

<socket> Assigned IP Socket. Range from 1 to 10.

7.10 Sending data in Command-Mode $IPSEND

Description: This command sends data to a connected socket.

Setting/Getting: Set value: AT$IPSEND=<socket>,"<data>"

Command	Possible Response(s)
AT$IPSEND=1,"Hello World" *Note: Sending of "Hello World" data through Socket #1 connection*	OK

Defined Values

<socket> Assigned IP Socket. Range from 1 to 10.

<data> Data up to 255 characters. All 7-bit ASCII characters are accepted except ';', '"' and '\'. Use "\XX" to markup binary data (eg.: \03 is to send a binary value 0011).

7.11 Receiving Data on IP Socket $IPRECV

Description: This unsolicited response (in Command-Mode only) indicates that there is an incoming data and has been stored in the incoming data buffer for your retrieval.

Response: $IPRECV: <socket>,<send_mode>

Possible Response(s)
$IPRECV: 2,1 OK *Note: Indication for Pending data in data-mode on Socket #2*
$IPRECV: 3, 0 OK *Note: Indication for Pending data in command-mode on Socket #3*

Defined Values

<socket> Assigned IP Socket. Range from 1 to 10.

<send_mode> Current selected $IPDATAMODE.

0 = Command-Mode: use $IPDATA command to read the data.

1 = Data-Mode: use $IPO command to read the data.

7.12 Retrieve Data in Command-Mode $IPDATA

Description: This command retrieves incoming data from a socket in Command mode.

Setting/Getting: Setting: AT$IPDATA=<socket>[,<size>]

Response: Response: $IPDATA: <socket>,<return_size>

Command	Possible Response(s)
	$IPRECV: 2,0 OK *Note: '2' indicates that there is an incoming data on Socket #2. '0' indicates that Socket #2 is connected in Command-Mode*
AT$IPDATA=2,250 *Note: retrieve 250 bytes of the data*	$IPDATA: 2,250 <data>…….. OK *Note: Response indicating 250 bytes has been successfully retrieved from Socket #2.*
AT$IPDATA=2,250 *Note: retrieve 250 bytes of the data*	$IPDATA: 2,200 <data>…….. OK *Note: Response indicating 200 bytes (instead of 250 bytes) has been successfully retrieved from Socket #2 buffer.*
AT$IPDATA=2,100	$IPDATA: 2,0 OK *Note: Response indicating 0 bytes, i.e. Incoming data buffer is empty and no bytes are retrieved.*

Defined Values

<socket> Assigned IP Socket. Range from 1 to 10.

<size> Size of data to retrieve in bytes. Range 1 to 300. Default is 300.

Note: If no size is specified, default size will be used to retrieve remaining data

<return_size> Number of bytes retrieved from incoming data buffer of the <socket>.

If <return_size>=0, no data is available in buffer.

7.13 Close IP Socket $IPCLOSE

Description:

This command closes one or all socket connection. This command is used only in Command-Mode.

Setting/Getting: Set value: AT$IPCLOSE=<socket>

Command	Possible Response(s)
AT$IPCLOSE=1 *Note: Request closing of socket 1 in command-mode*	Ok_Info_DataClosed OK
AT$IPCLOSE=0 *Note: Request closing of all socket in active or suspended connections*	Ok_Info_DataClosed OK

Defined Values

<socket> Assigned IP Socket. Range from 1 to 10.

When <socket>=0, all sockets will be closed.

7.14 Open IP Listening Socket $IPLOPEN

Description:

This command opens a listening socket. A maximum number of 5 clients are allowed to connect to a server connection at any one time. If the 6th client attempts to connect to server, a server reject message will be sent to the client. Incoming clients are automatically connected. $IPL responses indicate established client connections.

Setting/Getting:

Set value: AT$IPLOPEN=<id>

Command	Possible Response(s)
AT$IPLOPEN=4 *Note: Opens a listening port on Connection-ID #4*	OK *Note: Successfully opened Connection-ID #4 as a Listening Port.*

Defined Values

<id> Connection-ID. Numeric 1-5.

<id> must be corresponded to <id> in $IPCFG, otherwise ERROR will be returned.

7.15 Connection at IP Listening Socket $IPL

Description:

This unsolicited response indicates that a client has successfully connected to the listening port.

Response:

$IPL: <id>,<socket>

Command	Possible Response(s)
Note: Client attempts to connect to the listening port at Connection-ID #4	$IPL: 4, 6 OK *Note: Listening Port at Connection-ID #4 has accepted a client connection; socket was created and assigned to Socket #6. Listening Port at Connection-ID #4 continues to listen for clients.*
Note: Another client attempts to connect to the listening port at Connection-ID #4	$IPL: 4, 7 OK *Note: Listening Port at Connection-ID #4 has accepted another client connection; socket was created and assigned to Socket #7. Listening Port at Connection-ID #4 continues to listen for clients.*

Defined Values

<id> Connection-ID. Numeric 1-5.

<socket> Assigned IP Socket. Range from 1 to 10.

7.16 Close IP Listening Socket $IPLCLOSE

Description:

This command closes/stops a listening port.

Setting/Getting:

Set value: AT$IPLCLOSE=<id>

Command	Possible Response(s)
AT$IPLCLOSE=4 *Note: Close Listening port at Connection-ID #4*	Ok_Info_DataClosed OK

Defined Values

<id> Connection-ID. Numeric 1-5.

7.17 Report IP Error $IPERR

Description:

This command enables/disables response with error and result code relating to the IP connections.

When $IPERR is set to '1' (enabled), IP errors will be responded via $IPERR (format: $IPERR: <socket>,<err>), else only 'ERROR' is responded.

Setting/Getting: Set value: AT$IPERR=<n>

Command	Possible Response(s)
AT$IPERR=1 *Note: Enable display of IP error result codes*	OK
AT$IPERR=?	$IPERR: (0-2) OK
AT$IPERR?	$IPERR: 1 OK

Defined Values

<n>

0 = disable $IPERR: <socket>,<err> result code and use ERROR instead

1 = enable $IPERR: <socket>,<err> result code and use numeric <err> values (default)
(refer to Section 9.1: IP Error Codes $IPERR for error codes)

2 = enable $IPERR: <socket>,<err> result code and use verbose <err> values
(refer to Section 9.1: IP Error Codes $IPERR for error codes)

8 TCP EXAMPLES

8.1 TCP Client

8.1.1 Command-Mode Sending

Command	Possible Response(s)	Explanation
AT$IPAPN=1,"sunsurf","",""	OK	Set APN server for GPRS Profile 1.
AT$IPAPN?	$IPAPN: 1,"sunsurf","","" $IPAPN: 2,"","","" OK	View GPRS settings.
AT$IPCFG=1,1,"200.34.56.10", 6666	OK	Set Connection-ID #1 to operate as a TCP client. Remote TCP server IP address and port is also specified.
AT$IPDATAMODE=0	OK	Select command-mode for data transfer.
AT$IPSTART=1	Ok_Info_GprsActivation OK	Attach to GPRS specified in GPRS connection profile 1.
AT$IPOPEN=1	$IPCONN: 1,1 OK	Open connection to remote TCP server specified in $IPCFG. This connection has been assigned at Socket #1.
AT$IPSEND=1,"Hello World"	OK $IPRECV: 1,0 OK	Sends "Hello World" to remote TCP echo server. Receives pending data in Socket #1.
AT$IPDATA=1	$IPDATA: 1,11 Hello World OK	Retrieve all pending data in Socket #1.
AT$IPCLOSE=1	Ok_Info_DataClosed OK	Successfully close Socket #1 Connection.
AT$IPSTOP	Ok_Info_GprsDeactivation OK	Close GPRS connection successfully.

8.1.2 Data-Mode Sending

Command	Possible Response(s)	Explanation
AT$IPAPN=1,”sunsurf”,””,””	OK	Set APN server for GPRS Profile 1.
AT$IPAPN?	$IPAPN: 1,“sunsurf”,””,”” $IPAPN: 2,””,””,”” OK	View GPRS settings.
AT$IPCFG=1,1,”200.34.56.10”, 6666	OK	Set Connection-ID #1 to operate as a TCP client. Remote TCP server IP address and port is also specified.
AT$IPDATAMODE=1	OK	Select data-mode for data transfer.
AT$IPSTART=1	Ok_Info_GprsActivation OK	Attach to GPRS specified in GPRS connection profile 1.
AT$IPOPEN=1	$IPCONN: 1,1 Ok_InfoWaitingForData	Open connection to remote TCP server specified in $IPCFG. This connection has been assigned at Socket #1. Data can be sent by keying in.
+++	OK	Exit online mode to offline mode.
AT+COPS?	+COPS: 0,0,"SGP-M1-3GSM"	AT-commands can be entered now.
	$IPRECV: 1,1 OK	Indication of pending data in Socket #1.
AT$IPO=1	CONNECT	Exit offline mode and enter online mode for Socket #1 connection. Data can be sent by keying in.
^C	Ok_Info_DataClosed OK	Closes current socket connection.
AT$IPSTOP	Ok_Info_GprsDeactivation OK	Close GPRS connection successfully.

9 UDP EXAMPLES

9.1 UDP Client

9.1.1 Command-Mode sending

Command	Possible Response(s)	Explanation
AT$IPAPN=1,"sunsurf","",""	OK	Set APN server for GPRS Profile 1.
AT$IPAPN?	$IPAPN: 1,"sunsurf","","" $IPAPN: 2,"","","" OK	View GPRS settings.
AT$IPCFG=1,3,"200.34.56.10", 6666	OK	Set Connection-ID #1 to operate as a UDP client. Remote UDP server IP address and port is also specified.
AT$IPDATAMODE=0	OK	Select command-mode for data transfer.
AT$IPSTART=1	Ok_Info_GprsActivation OK	Attach to GPRS specified in GPRS connection profile 1.
AT$IPOPEN=1	$IPCONN: 1,1 OK	Open UDP connection specified by $IPCFG. This connection has been assigned at Socket #1.
AT$IPSEND=1,"Hello World"	OK	Sends "Hello World" to remote UDP server.
	$IPRECV: 1,0 OK	Receives pending data in Socket #1.
AT$IPDATA=1	$IPDATA: 1,11 Hello World OK	Retrieve all pending data in Socket #1.
AT$IPCLOSE=1	Ok_Info_DataClosed OK	Successfully close Socket #1 Connection.
AT$IPSTOP	Ok_Info_GprsDeactivation OK	Close GPRS connection successfully.

9.1.2 Data-Mode Sending

Command	Possible Response(s)	Explanation
AT$IPAPN=1,”sunsurf”,””,””	OK	Set APN server for GPRS Profile 1.
AT$IPAPN?	$APN: 1,“sunsurf”,””,”” $APN: 2,””,””,”” OK	View GPRS settings.
AT$IPCFG=1,3,”200.34.56.10”, 6666	OK	Set Connection-ID #1 to operate as a UDP client. Remote UDP server IP address and port is also specified.
AT$IPDATAMODE=1	OK	Select data-mode for data transfer.
AT$IPSTART=1	Ok_Info_GprsActivation OK	Attach to GPRS specified in GPRS connection profile 1.
AT$IPOPEN=1	$IPCONN: 1,1 Ok_InfoWaitingForData	Open UDP connection specified by $IPCFG. This connection has been assigned at Socket #1. Data can be sent by keying in.
+++	OK	Exit online mode to offline mode.
AT+COPS?	+COPS: 0,0,"SGP-M1-3GSM"	AT-commands can be entered now.
	$IPRECV: 1,1 OK	Indication of pending data in Socket #1.
AT$IPO=1	CONNECT	Exit offline mode and enter online mode for Socket #1 connection. Data received is displayed. Data sending is resumed by keying in.
^C	Ok_Info_DataClosed OK	Closes current socket connection.
AT$IPSTOP	Ok_Info_GprsDeactivation OK	Close GPRS connection successfully.

10 ERROR CODES

10.1 IP Error Codes $IPERR

The following table lists the Error Codes values and text from $IPERR response.

Syntax:

$IPERR: <socket>,<err>

<err> Numeric	<err> Verbose	Description
3000	Error_Info_AsyncError	An asynchronous error network event has occurred
3001	Error_Info_InvalidParameter	A parameter given to the function is invalid
3002	Error_Info_InternalError	An internal error has happened
3003	Error_Info_AddressInUse	The address or port is already in use
3004	Error_Info_OutOfMemory	There is not enough memory to fulfill the request
3005	Error_Info_NotSupported	The socket is not of a type that can support this operation
3006	Error_Info_Unreachable	The specified host cannot be reached
3007	Error_Info_ConnRefused	The connection to the specified address was refused by the remote host
3008	Error_Info_AlreadyConnected	The request could not be fulfilled because the socket is already connected
3009	Error_Info_ConnTimeout	The connection attempt timed out without establishing a connection
3010	Error_Info_HostNotFound	The specified host could not be found in the DNS
3011	Error_Info_TempDNSError	A temporary DNS error has occurred. Retrying the query may be successful
3012	Error_Info_PermDNSError	A permanent DNS error has occurred
3013	Error_Info_NoIPAddress	The specified name has been found in the DNS, but no IP address is available
3014	Error_Info_MsgTooBig	The size of the data buffer is too large for a UDP socket
3015	Error_Info_ConnReset	The connection has been reset by the remote peer
3016	Error_Info_ConnAborted	The connection was aborted due to timeout or some other error condition
3017	Error_Info_NoBufSpace	Sending failed temporarily because the space to buffer

<err> Numeric	<err> Verbose	Description
		the message was exhausted.
3018	Error_Info_NetworkLost	The operation failed because TCP/IP's bearer connection has been disconnected
3019	Error_Info_NotRdy	The operation failed because the bearer connection has not been opened.
3020	Error_Info_BearerNotRdy	The bearer connection could not be opened because the mobile is not yet completely attached to the network. A retry at a later time may be successful.
3021	Error_Info_InProgress	The operation failed because a similar operation is already in progress.
3022	Error_Info_BearerAlreadyOpen	The operation failed because a bearer connection is already open.
3023	Error_Info_MEBusy	Mobile equipment is not ready for TCP/IP connectivity
3024	Error_Info_BearerNotOpen	Bearer is not open
3025	Error_Info_ConnNotOpen	Connection is not open yet
3026	Error_Info_BearerOpenFail	Bearer open fail
3027	Error_Info_SockCreateFail	Socket create fail
3028	Error_Info_OperNotAllowed	Operation not allowed

10.2 IP Display Messages

The following table listed possible responses on IP multi-socket commands and events.

Display Message	Description
Ok_Info_GprsActivation	GPRS bearer context is activated - ready to be used by socket connection
Ok_Info_GprsDeactivation	GPRS bearer context is de-activated
Ok_Info_WaitingForData	Socket connected - ready for data transmission
Ok_Info_DataClosed	Socket disconnected & destroyed - bearer context still active
Error_Info_GprsActivation	Failed to activate GPRS bearer context
Error_Info_GsmActivation	Failed to activate GSM bearer context
Error_Info_SockCreateFail	Socket creation failed
Error_Info_InvalidParameter	Invalid parameter in TCP or UDP settings (some settings are not set)
Error_Info_AddressInUse	The address or port is already in use
Error_Info_OutOfMemory	There is not enough memory to fulfill the request
Error_Info_NotSupported	The socket is not of a type that can support this operation
Error_Info_Unreachable	The specified host cannot be reached
Error_Info_ConnRefused	The connection to the specified address was refused by the remote host
Error_Info_ConnTimeout	The connection attempt timed out without establishing a connection
Error_Info_AlreadyConnected	The request could not be fulfilled because the socket is already connected
Error_Info_HostNotFound	The specified host could not be found in the DNS
Error_Info_TempDNSError	A temporary DNS error has occurred. Retrying the query may be successful
Error_Info_PermDNSError	A permanent DNS error has occurred
Error_Info_NoIPAddress	The specified name has been found in the DNS, but no IP address is available
Error_Info_MsgTooBig	The size of the data buffer is too large for a TCP or UDP socket
Error_Info_ConnReset	The connection has been reset by the remote peer
Error_Info_ConnAborted	The connection was aborted due to timeout or some other error condition
Error_Info_NoBufSpace	Sending failed temporarily because the space to buffer the message was exhausted
Error_Info_NetworkLost	The operation failed because TCP/IP and UDP bearer connection has been disconnected. As an asynchronous event code: The bearer connection has been closed.
Error_Info_InProgress	The operation failed because a similar operation is already in progress
Error_Info_AsyncError	Network event: an asynchronous error has occurred
Ok_Info_GsmActivation	GSM bearer context is activated

Display Message	Description
Ok_Info_GsmDeactivation	GSM bearer context is de-activated
Error_Info_OperNotAllowed	Current command is not allowed (eg: $IPDATAMODE cannot be changed during a connected TCP session. Need to close TCP first)

11 SUPPORT/ CONTACT US

For distributor clients, please contact your respective distributor FAE.

For direct clients, please contact iWOW FAE (Technical Support Department) or email us at technicalsupport@iwow.com.sg.

For general enquiries please contact us at:

iWOW Connections Pte Ltd
1 Lorong 2 Toa Payoh, #04-01
Yellow Pages Building
Singapore 319637
Office: (65) 6748 8123
Fax : (65) 6748 2668
Email: sales@iwow.com.sg
Website: http://www.iWOW.com.sg